N4SID[1] : Subspace Algorithms for the Identification of Combined Deterministic-Stochastic Systems [2]

Peter Van Overschee [3]
Bart De Moor [4]

ESAT/SISTA 1992-34-I

Internal report, Accepted for publication in :
Automatica, Special Issue on Statistical Processing and Control

[1] Numerical algorithms for Subspace State Space System Identification. Read as a Californian license plate : *enforce it.*

[2] Running title : N4SID : Identification of Combined Deterministic-Stochastic Systems.

[3] ESAT, Department of Electrical Engineering, Katholieke Universiteit Leuven, Kardinaal Mercierlaan 94, 3001 Heverlee, BELGIUM. email : vanoverschee@esat.kuleuven.ac.be. Research Assistant of the Belgian National Fund for Scientific Research. Author to whom all correspondence should be addressed.

[4] Research Associate of the Belgian National Fund for Scientific Research

Abstract

Recently a great deal of attention has been given to numerical algorithms for subspace state space system identification (**N4SID**). In this paper, we derive two new **N4SID** algorithms to identify mixed deterministic-stochastic systems. Both algorithms determine state sequences through the projection of input and output data. These state sequences are shown to be outputs of non-steady state Kalman filter banks. From these it is easy to determine the state space system matrices. The **N4SID** algorithms are always convergent (non-iterative) and numerically stable since they only make use of QR and Singular Value Decompositions. Both **N4SID** algorithms are similar, but the second one trades off accuracy for simplicity. These new algorithms are compared with existing subspace algorithms in theory and in practice.

Key words : Subspace identification, non-steady state Kalman filter, Riccati difference equations, QR and Singular Value Decomposition

1 Introduction

The greater part of the systems identification literature is concerned with computing polynomial models, which are however known to typically give rise to numerically ill-conditioned mathematical problems, especially for Multi Input Multi Output systems. Numerical algorithms for subspace state space system identification (**N4SID**[1]) are then viewed as the better alternatives. This is especially true for high-order multivariable systems, for which it is not trivial to find a *useful* parameterization among all possible parametrizations. This parametrization is needed to start up the classical identification algorithms (see *e.g.* Ljung, 1987), which means that a-priori knowledge of the order and of the observability (or controllability) indices is required.

With **N4SID** algorithms, most of this a-priori parametrization can be avoided. Only the order of the system is needed and it can be determined through inspection of the dominant singular values of a matrix that is calculated during the identification. The state space matrices are not calculated in their canonical forms (with a minimal number of parameters), but as full state space matrices in a certain, almost optimally conditioned basis (this basis is uniquely determined, so that there is no problem of identifiability). This implies that the observability (or controllability) indices do not have to be known in advance.

Another major advantage is that **N4SID** algorithms are non-iterative, with no non-linear optimization part involved. This is why they do not suffer from the typical disadvantages of iterative algorithms, *e.g.* no guaranteed convergence, local minima of the objective criterion and sensitivity to initial estimates.

For classical identification, an extra parametrization of the initial state is needed when estimating a state space system from data measured on a plant with an non-zero initial condition. A final advantage of the **N4SID** algorithms, is that there is no difference between zero and non-zero initial states.

Most commonly known subspace methods are realization algorithms of *e.g.* Kung (1978), where a discrete-time state space model is computed from a block Hankel matrix with Markov parameters. It is unfortunate that the theory here relies on Markov parameters as a starting point, something rather difficult to measure or compute in practice (*e.g.* think of unstable systems).

An alternative direct identification scheme for purely deterministic systems is described by *e.g.* Moonen *et al.* (1989), Moonen & Ramos (1991), where a state space model is computed directly from a block Hankel matrix constructed from the input-output data. In a first

[1] Read as a Californian license plate : *enforce it*

step, a state vector sequence is computed as an interface between a 'past' and a 'future'. Once the state vector sequence is known, the system matrices are computed from a set of linear equations.

Similar data-driven identification schemes for purely stochastic identification are well known, (see *e.g.* Arun & Kung (1990) and the references therein). Less well known is that these algorithms can compute extremely biased results. This problem was studied and solved by Van Overschee & De Moor (1991a,1991b).

The problem addressed in this paper is that of identifying a general state space model for combined deterministic-stochastic systems directly from the input-output data. Some papers in the past have already treated this problem but from a different viewpoint. In Larimore (1990) for instance, the problem is treated from a purely statistical point of view. There is no proof of correctness (in a sense of the algorithms being asymptotically unbiased) whatsoever. In De Moor *et al.* (1991), Verhaegen (1991) the problem is split up into two subproblems : deterministic identification followed by a stochastic realization of the residuals. In Moonen *et al.* (1992) the problem is solved for double infinite block Hankel matrices, which implies practical computational problems.

In this paper, we will derive two **N4SID** algorithms that determine the deterministic and stochastic system at the same time. The connection with classical system theory (Kalman filter) will be used to prove the exactness (unbiasedness for an infinite number of measurements) of algorithm 1, or the degree of approximation (calculation of the bias for an infinite number of measurements) of algorithm 2.

The approach adopted here is similar to the identification schemes of Moonen *et al.* (1989) for the purely deterministic case and Van Overschee & De Moor (1991a,1991b) for the stochastic case. First a state sequence is determined from the projection of input-output data. This projection retains all the information (deterministic and stochastic) in the past that is useful to predict the future. Then, the state space matrices are determined from this state sequence. Fig. 1 shows how these **N4SID** algorithms differ from the classical identification schemes.

The connection of the two new **N4SID** algorithms with the existing algorithms described above will also be indicated.

This paper is organized as follows : The problem description and the mathematical tools can be found in section 2. In section 3 the main projection is defined. Section 4 introduces a closed form formula for the non-steady state Kalman filter estimation problem. This result is related to the results of section 3 to find the interpretation of the main projection as a sequence of outputs of a non-steady state Kalman filter bank. Section 5 introduces a first **N4SID** algorithm that identifies the system matrices exactly. In section 6 accuracy is traded off for simplicity in a second approximate **N4SID** algorithm. Section 7 shows how these **N4SID** algorithms can be implemented in a numerically reliable way, using the QR and the Singular Value Decomposition (SVD). Section 8 investigates the connection with other existing algorithms. Finally section 9 will treat some comparative examples. The conclusions can be found in section 10.

2 Preliminaries

In this section, we describe the linear time invariant system we want to identify. We also introduce the input and output block Hankel matrices, the past and future horizon as well as the input-output equations.

2.1 System description

Consider the following combined deterministic-stochastic model to be identified :

$$x_{k+1} = Ax_k + Bu_k + w_k \tag{1}$$

$$y_k = Cx_k + Du_k + v_k \tag{2}$$

with

$$\mathbf{E}[\begin{pmatrix} w_k \\ v_k \end{pmatrix}\begin{pmatrix} w_l^t & v_l^t \end{pmatrix}] = \begin{pmatrix} Q^s & S^s \\ (S^s)^t & R^s \end{pmatrix} \delta_{kl} \quad \geq 0^2 \tag{3}$$

and $A, Q^s \in \mathbb{R}^{n \times n}, B \in \mathbb{R}^{n \times m}, C \in \mathbb{R}^{l \times n}, D \in \mathbb{R}^{l \times m}, S^s \in \mathbb{R}^{n \times l}$ and $R^s \in \mathbb{R}^{l \times l}$. The input vectors $u_k \in \mathbb{R}^{m \times 1}$ and output vectors $y_k \in \mathbb{R}^{l \times 1}$ are measured. $v_k \in \mathbb{R}^{l \times 1}$ and $w_k \in \mathbb{R}^{n \times 1}$ on the other hand are unmeasurable, Gaussian distributed, zero mean, white noise vector sequences. $\{A, C\}$ is assumed to be observable, while $\{A, \begin{pmatrix} B & (Q^s)^{1/2} \end{pmatrix}\}$ is assumed to be controllable.

This system (1)-(2) is split up in a deterministic and a stochastic subsystem, by splitting up the state (x_k) and output (y_k) in a deterministic ($.^d$) and stochastic ($.^s$) component : $x_k = x_k^d + x_k^s$, $y_k = y_k^d + y_k^s$. The deterministic state (x_k^d) and output (y_k^d) follow from the deterministic subsystem, which describes the influence of the deterministic input (u_k) on the deterministic output :

$$x_{k+1}^d = Ax_k^d + Bu_k \tag{4}$$

$$y_k^d = Cx_k^d + Du_k \tag{5}$$

[2] **E** denotes the expected value operator and δ_{kl} the Kronecker index.

The controllable modes of $\{A, B\}$ can be either stable or unstable. The stochastic state (x^s_k) and output (y^s_k) follow from the stochastic subsystem, which describes the influence of the noise sequences (w_k and v_k) on the stochastic output :

$$\begin{aligned} x^s_{k+1} &= Ax^s_k + w_k & (6) \\ y^s_k &= Cx^s_k + v_k & (7) \end{aligned}$$

The controllable modes of $\{A, (Q^s)^{1/2}\}$ are assumed to be stable.
The deterministic inputs (u_k) and states (x^d_k) and the stochastic states (x^s_k) and outputs (y^s_k) are assumed to be quasi-stationary (as defined in Ljung, 1987, pp. 27). Note that even though the deterministic subsystem can have unstable modes, the excitation (u_k) has to be chosen in such a way that the deterministic states and output are finite for all time. Also note that since the systems $\{A, B\}$ and $\{A, (Q^s)^{1/2}\}$ are not assumed to be controllable, the deterministic and stochastic subsystem may have common as well as completely decoupled input-output dynamics.

The main problem of this paper can now be stated : Given input and output measurements $u_1, \ldots, u_N$ and $y_1, \ldots, y_N$ ($N \to \infty$), and the fact that these two sequences are generated by an unknown combined deterministic-stochastic model of the form described above, find $A, B, C, D, Q^s, R^s, S^s$ (up to within a similarity transformation).

In the next two sections, we will define some more useful properties and notations for the deterministic and the stochastic subsystem.

2.1.1 The deterministic subsystem

Associated with the deterministic subsystem (4)-(5), we define the following matrices :

- The extended ($i > n$) observability matrix Γ_i (where the subscript i denotes the number of block rows) :

$$\Gamma_i \stackrel{\text{def}}{=} \begin{pmatrix} C \\ CA \\ CA^2 \\ \ldots \\ CA^{i-1} \end{pmatrix}$$

- The reversed extended controllability matrix Δ^d_i (where the subscript i denotes the number of block columns) :

$$\Delta^d_i \stackrel{\text{def}}{=} \begin{pmatrix} A^{i-1}B & A^{i-2}B & \ldots & AB & B \end{pmatrix}$$

- The lower block triangular Toeplitz matrix H_i^d :

$$H_i^d \overset{\text{def}}{=} \begin{pmatrix} D & 0 & 0 & \dots & 0 \\ CB & D & 0 & \dots & 0 \\ CAB & CB & D & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots \\ CA^{i-2}B & CA^{i-3}B & CA^{i-4}B & \dots & D \end{pmatrix}$$

2.1.2 The stochastic subsystem

For the stochastic subsystem (6)-(7) we define :

$$\begin{aligned} P^s &\overset{\text{def}}{=} \mathbf{E}[x_k^s(x_k^s)^t] \\ G &\overset{\text{def}}{=} \mathbf{E}[x_k^s(y_k^s)^t] \\ \Lambda_0 &\overset{\text{def}}{=} \mathbf{E}[y_k^s(y_k^s)^t] \end{aligned}$$

With equations (3),(6),(7) and through stability of the controllable modes of the system $\{A, (Q^s)^{1/2}\}$, we find easily that the following equations are satisfied :

$$\begin{aligned} P^s &= AP^sA^t + Q^s \\ G &= AP^sC^t + S^s \\ \Lambda_0 &= CP^sC^t + R^s \end{aligned} \qquad (8)$$

This set of equations describes the set of all possible stochastic realizations that have the same second order statistics as a given stochastic sequence y_k^s. We call them the positive real equations. More details can be found in Faure (1976).

It is also easy to derive that :

$$\Lambda_i \overset{\text{def}}{=} \mathbf{E}[y_{k+i}^s(y_k^s)^t] = \begin{cases} CA^{i-1}G & i > 0 \\ \Lambda_0 & i = 0 \\ G^t(A^t)^{-i-1}C^t & i < 0 \end{cases}$$

Associated with the stochastic subsystem, we define the following matrices :

- The matrix Δ_i^s :

$$\Delta_i^s \overset{\text{def}}{=} \begin{pmatrix} A^{i-1}G & A^{i-2}G & \dots & AG & G \end{pmatrix}$$

- The block Toeplitz covariance matrix L_i^s :

$$L_i^s \overset{\text{def}}{=} \begin{pmatrix} \Lambda_0 & \Lambda_{-1} & \Lambda_{-2} & \dots & \Lambda_{1-i} \\ \Lambda_1 & \Lambda_0 & \Lambda_{-1} & \dots & \Lambda_{2-i} \\ \dots & \dots & \dots & \dots & \dots \\ \Lambda_{i-1} & \Lambda_{i-2} & \Lambda_{i-3} & \dots & \Lambda_0 \end{pmatrix}$$

- The block Toeplitz cross covariance matrix H_i^s :

$$H_i^s \stackrel{\text{def}}{=} \begin{pmatrix} \Lambda_i & \Lambda_{i-1} & \Lambda_{i-2} & \dots & \Lambda_1 \\ \Lambda_{i+1} & \Lambda_i & \Lambda_{i-1} & \dots & \Lambda_2 \\ \dots & \dots & \dots & \dots & \dots \\ \Lambda_{2i-1} & \Lambda_{2i-2} & \Lambda_{2i-3} & \dots & \Lambda_i \end{pmatrix} = \Gamma_i \Delta_i^s$$

2.2 Block Hankel matrices and input-output equations

Input and output block Hankel matrices are defined as :

$$U_{0|i-1} \stackrel{\text{def}}{=} \begin{pmatrix} u_0 & u_1 & u_2 & \dots & u_{j-1} \\ u_1 & u_2 & u_3 & \dots & u_j \\ \dots & \dots & \dots & \dots & \dots \\ u_{i-1} & u_i & u_{i+1} & \dots & u_{i+j-2} \end{pmatrix} \qquad Y_{0|i-1} \stackrel{\text{def}}{=} \begin{pmatrix} y_0 & y_1 & y_2 & \dots & y_{j-1} \\ y_1 & y_2 & y_3 & \dots & y_j \\ \dots & \dots & \dots & \dots & \dots \\ y_{i-1} & y_i & y_{i+1} & \dots & y_{i+j-2} \end{pmatrix}$$

where we presume that $j \to \infty$ throughout the paper. The subscripts of U and Y denote the subscript of the first and last element of the first column. The block Hankel matrices formed with the output y_k^s of the stochastic subsystem are defined as $Y_{0|i-1}^s$ in the same way.

Somewhat loosely we denote the "past" inputs with $U_{0|i-1}$ or $U_{0|i}$ and the "future" inputs with $U_{i|2i-1}$ or $U_{i+1|2i-1}$. A similar notation applies for the past and future outputs. This notational convention is useful when explaining concepts.

The deterministic and stochastic state matrices are defined as :

$$X_i^d \stackrel{\text{def}}{=} \begin{pmatrix} x_i^d & x_{i+1}^d & x_{i+2}^d & \dots & x_{i+j-1}^d \end{pmatrix} \qquad X_i^s \stackrel{\text{def}}{=} \begin{pmatrix} x_i^s & x_{i+1}^s & x_{i+2}^s & \dots & x_{i+j-1}^s \end{pmatrix}$$

For the deterministic subsystem we define :

$$\lim_{j\to\infty} \frac{1}{j} \left(\begin{array}{c} U_{0|i-1} \\ U_{i|2i-1} \\ \hline X_0^d \end{array} \right) \left(\begin{array}{cc|c} U_{0|i-1}^t & U_{i|2i-1}^t & (X_0^d)^t \end{array} \right) \stackrel{\text{def}}{=} \left(\begin{array}{cc|c} R_{11} & R_{12} & S_1^t \\ R_{12}^t & R_{22} & S_2^t \\ \hline S_1 & S_2 & P^d \end{array} \right) = \left(\begin{array}{c|c} R & S^t \\ \hline S & P^d \end{array} \right)$$

where we use the assumption that the limit exists (quasi-stationarity of u_k and x_k^d).

For the stochastic subsystem we find that, due to stationarity of y_k^s, the following equalities hold true :

$$\lim_{j\to\infty} \frac{1}{j} \begin{pmatrix} Y_{0|i-1}^s \\ Y_{i|2i-1}^s \end{pmatrix} \begin{pmatrix} (Y_{0|i-1}^s)^t & (Y_{i|2i-1}^s)^t \end{pmatrix} = \begin{pmatrix} L_i^s & (H_i^s)^t \\ H_i^s & L_i^s \end{pmatrix} \tag{9}$$

The **Matrix input-output equations** are defined in the following Theorem (De Moor, 1988) :

Theorem 1

$$\begin{aligned} Y_{0|i-1} &= \Gamma_i X_0^d + H_i^d U_{0|i-1} + Y_{0|i-1}^s && (10)\\ Y_{i|2i-1} &= \Gamma_i X_i^d + H_i^d U_{i|2i-1} + Y_{i|2i-1}^s && (11)\\ X_i^d &= A^i X_0^d + \Delta_i^d U_{0|i-1} && (12)\end{aligned}$$

The Theorem is easy to prove by recursive substitution into the state space equations.

3 The Main Projection

In this section, we introduce the projection of the future outputs onto the past and future inputs and the past outputs. The results can be described as a function of the system matrices and the input-output block Hankel matrices.

We define the matrices Z_i and Z_{i+1} as :

$$\begin{aligned} Z_i &= Y_{i|2i-1}/\begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix} && (13)\\ Z_{i+1} &= Y_{i+1|2i-1}/\begin{pmatrix} U_{0|2i-1} \\ Y_{0|i} \end{pmatrix} && (14)\end{aligned}$$

where $A/B = AB^t(BB^t)^{-1}B$. The row space of A/B is equal to the projection of the row space of A onto the row space of B.

Formula (13) corresponds to the optimal prediction of $Y_{i|2i-1}$ given $U_{0|2i-1}$ and $Y_{0|i-1}$ in a sense that :

$$||Y_{i|2i-1} - Z_i||_F^2$$

is minimized constrained to

$$Z_i \in \text{Row Space} \begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix}$$

So, intuitively, the k^{th} row of Z_i would correspond to a k step ahead prediction of the output. This intuition become clearer in section 4.

These projections (Z_i and Z_{i+1}) are useful in determining the combined system, since (as we will show in Theorem 2) the linear combinations to be made of the input-output block Hankel matrices to generate the matrices Z_i and Z_{i+1} are functions of the system matrices $(A, B, C, D, Q^s, S^s, R^s)$. Moreover, the system matrices can be retrieved from these linear combinations, as will be explained in section 5.

It is tedious though straightforward to prove the following Theorem which delivers formulas for the linear combinations to be made of the rows of the input-output block Hankel matrices to generate the matrices Z_i and Z_{i+1} :

Theorem 2 Main projection

- *If the deterministic input u_k and state x_k^d are uncorrelated with the stochastic output y_k^s :*

$$\lim_{j\to\infty}\frac{1}{j}Y^s_{0|i-1}U^t_{.} = 0 \qquad \lim_{j\to\infty}\frac{1}{j}Y^s_{0|i-1}(X^d_{.})^t = 0$$
$$\lim_{j\to\infty}\frac{1}{j}Y^s_{i|2i-1}U^t_{.} = 0 \qquad \lim_{j\to\infty}\frac{1}{j}Y^s_{i|2i-1}(X^d_{.})^t = 0$$

  where the subscript . denotes past or future

- *and if the input is "persistently exciting of order 2i (Ljung, 1987, pp. 363)" : rank $U_{0|2i-1} = 2mi$*

- *and if the stochastic subsystem is not identically zero (the purely deterministic case will be treated in section 6.1.3)*

then (for $j \to \infty$):

$$Z_i = \Gamma_i\hat{X}_i + H_i^d U_{i|2i-1} \qquad (15)$$
$$Z_{i+1} = \Gamma_{i-1}\hat{X}_{i+1} + H_{i-1}^d U_{i+1|2i-1} \qquad (16)$$

with :

$$\hat{X}_i = \left(\begin{array}{c|c|c} A^i - Q_i\Gamma_i & \Delta_i^d - Q_iH_i^d & Q_i \end{array}\right)\left(\begin{array}{c} SR^{-1}U_{0|2i-1} \\ \hline U_{0|i-1} \\ \hline Y_{0|i-1} \end{array}\right) \qquad (17)$$

$$\hat{X}_{i+1} = \left(\begin{array}{c|c|c} A^{i+1} - Q_{i+1}\Gamma_{i+1} & \Delta_{i+1}^d - Q_{i+1}H_{i+1}^d & Q_{i+1} \end{array}\right)\left(\begin{array}{c} SR^{-1}U_{0|2i-1} \\ \hline U_{0|i} \\ \hline Y_{0|i} \end{array}\right) \qquad (18)$$

and

$$Q_i = \chi_i\psi_i^{-1}$$
$$\chi_i = A^i(P^d - SR^{-1}S^t)\Gamma_i^t + \Delta_i^s \qquad (19)$$
$$\psi_i = \Gamma_i(P^d - SR^{-1}S^t)\Gamma_i^t + L_i^s \qquad (20)$$

A proof can be found in Appendix A. In the next section, we give an interpretation of these projections.

4 A Bank of Kalman Filters

In this section, we show how the sequences $\hat{X}_i$ and $\hat{X}_{i+1}$ can be interpreted in terms of states of a bank of j non-steady state Kalman filters, applied in parallel to the data. This interpretation will lead to a formula that will prove to be extremely useful when determining the system matrices from the data.

As stated before, it may come as no surprise that there is a connection between the states $\hat{X}_i$ defined by the projection Z_i and some optimal prediction of the outputs $Y_{i|2i-1}$.

To establish this connection, we need one more Theorem that states how the non steady state Kalman filter state estimate $\hat{x}_k$ can be written as a linear combination of $u_0, \dots, u_{k-1}$, $y_0, \dots, y_{k-1}$ and the initial state estimate $\hat{x}_0$.

Theorem 3 Kalman Filter

Given $\hat{x}_0, P_0, u_0, \dots, u_{k-1}, y_0, \dots, y_{k-1}$ and all the system matrices $(A, B, C, D, Q^s, S^s, R^s)$, then the non steady state Kalman filter state $\hat{x}_k$ defined by the following recursive formulas :

$$\begin{aligned}
\hat{x}_k &= A\hat{x}_{k-1} + Bu_{k-1} + K_{k-1}(y_{k-1} - C\hat{x}_{k-1} - Du_{k-1}) && (21)\\
K_{k-1} &= (AP_{k-1}C^t + G)(\Lambda_0 + CP_{k-1}C^t)^{-1} && (22)\\
P_k &= AP_{k-1}A^t - (AP_{k-1}C^t + G)(\Lambda_0 + CP_{k-1}C^t)^{-1}(AP_{k-1}C^t + G)^t && (23)
\end{aligned}$$

can be written as :

$$\hat{x}_k = \left(\begin{array}{c|c|c} A^k - Q_k\Gamma_k & \Delta_k^d - Q_k H_k^d & Q_k \end{array} \right) \left(\begin{array}{c} \hat{x}_0 \\ \hline u_0 \\ \dots \\ u_{k-1} \\ \hline y_0 \\ \dots \\ y_{k-1} \end{array} \right) \qquad (24)$$

where :

$$\begin{aligned}
Q_k &= \chi_k \psi_k^{-1} && (25)\\
\chi_k &= A^k P_0 \Gamma_k^t + \Delta_k^s && (26)\\
\psi_k &= \Gamma_k P_0 \Gamma_k^t + L_k^s && (27)
\end{aligned}$$

The proof of this Theorem and some details concerning the special form of the Kalman filter equations (21)-(23) can be found in Appendix B and Appendix C. Let us just indicate

that the error covariance matrix $\tilde{P}_k \overset{\text{def}}{=} \mathbf{E}[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^t]$ is given by $P^s + P_k$, with P^s the state covariance matrix from Lyapunov equation (8).
Note that the limiting solution ($k \to \infty$) of (23) is $-P_\infty$, where P_∞ is the state covariance matrix of the forward innovation model (Faure, 1976). Hence the limiting error covariance matrix is $\tilde{P}_\infty = P^s - P_\infty$, which is the smallest state error covariance matrix we can obtain (in the sense of nonnegative definiteness).
Also note that the expressions for ψ_k and χ_k (26)-(27) are equal to the expressions of ψ_i and χ_i (19)-(20) with $P^d - SR^{-1}S^t$ substituted by P_0.

If we now combine the results of Theorem 2 and 3, we find an interpretation of the sequences $\hat{X}_i$ and $\hat{X}_{i+1}$ in terms of states of a bank of non-steady state Kalman filters, applied in parallel to the data. More specifically, compare formulas (17), (18) and (24) :

1. The j columns of $\hat{X}_i$ are equal to the outputs of a bank of j non-steady state Kalman filters in parallel. The $(p+1)^{th}$ column of $\hat{X}_i$ for instance, is equal to the non-steady state Kalman filter state $\hat{x}_{i+p}$ of the Kalman filter (21), (22), (23), with initial error covariance matrix at starting time p :

$$\begin{aligned} \tilde{P}_p = P_p + P^s &= P^d - SR^{-1}S^t + P^s \\ \hat{x}_p &= SR^{-1} \begin{pmatrix} u_p \\ \dots \\ u_{p+2i-1} \end{pmatrix} \end{aligned}$$

   Notice that $\tilde{P}_p$ is independent of the column index, so it is denoted with $\tilde{P}^0$.

   In this way, all the columns can be interpreted as Kalman filter states. The initial states of the j filters together can be written as :

$$\hat{X}^0 = SR^{-1}U_{0|2i-1}$$

   All this is clarified in Fig. 2.

   The expressions for $\tilde{P}^0$ and $\hat{X}^0$ can be interpreted (somewhat loosely) as follows : If we had no information at all about the initial state, then the initial state estimate would be $\hat{X}^0 = 0$ and the initial error covariance would be equal to the expected variance of the state : $\tilde{P}^0 = \mathbf{E}[x_k x_k^t] = P^d + P^s$. Now, since the inputs are possibly correlated, we can derive information about $\hat{X}^0$ out of the inputs $U_{0|2i-1}$. This is done by projecting the (unknown) exact initial state sequence $X_0^d + X_0^s$ onto the row space of the inputs $U_{0|2i-1}$:

$$\hat{X}^0 = (X_0^d + X_0^s)/U_{0|2i-1} = SR^{-1}U_{0|2i-1}$$

This extra information on the initial state of the Kalman filter also implies that the error covariance matrix reduces from $P^d + P^s$ to :

$$\tilde{P}^0 = P^d + P^s - \lim_{j\to\infty} \frac{1}{j} \hat{X}^0 (\hat{X}^0)^t = P^d + P^s - SR^{-1}S^t$$

These are exactly the same expressions for $\hat{X}^0$ and $\tilde{P}^0$ as we found above.

It can also be seen that when the inputs are uncorrelated (white noise), the projection of $X_0^d + X_0^s$ onto the inputs $U_{0|2i-1}$ is zero, which implies that there is no information about the initial state $\hat{X}^0$ contained in the inputs $U_{0|2i-1}$.

The state sequence $\hat{X}_{i+1}$ has a similar interpretation. The p^{th} column of $\hat{X}_{i+1}$ is equal to the non-steady state Kalman filter state estimate of the same (in a sense of the same initial conditions) non-steady state Kalman filter bank as discussed above, but now the filter has iterated one step beyond the estimate of the p^{th} column of $\hat{X}_i$. This is valid for all columns $p = 1, \ldots, j$.

2. We define the residuals $\mathcal{R}_i$ of the projection as :

$$\mathcal{R}_i = Y_{i|2i-1} - Z_i = Y_{i|2i-1} - \Gamma_i \hat{X}_i - H_i^d U_{i|2i-1} \tag{28}$$

Since Z_i is the result of the projection of $Y_{i|2i-1}$ on the row space of $U_{0|2i-1}$ and $Y_{0|i-1}$, the residuals of this projection ($\mathcal{R}_i$) will always satisfy : $\mathcal{R}_i U_{0|2i-1}^t = 0$, $\mathcal{R}_i Y_{0|i-1}^t = 0$ and $\mathcal{R}_i Z_i^t = 0$. Also, since $\hat{X}_i$ can be written as a linear combination of $U_{0|2i-1}$ and $Y_{0|i-1}$ (see formula (17)), we find : $\mathcal{R}_i \hat{X}_i^t = 0$.

3. Since the corresponding columns of $\hat{X}_i$ and $\hat{X}_{i+1}$ are state estimates of the same (in a sense of the same initial conditions) non-steady state Kalman filter at two consecutive time instants, we can write (see formula (21)) :

$$\hat{X}_{i+1} = A\hat{X}_i + BU_{i|i} + K_i(Y_{i|i} - C\hat{X}_i - DU_{i|i}) \tag{29}$$

It is also trivial that :

$$Y_{i|i} = C\hat{X}_i + DU_{i|i} + (Y_{i|i} - C\hat{X}_i - DU_{i|i}) \tag{30}$$

If we inspect the formula for $\mathcal{R}_i$ a little bit closer (28), we see that its first row is equal to $Y_{i|i} - C\hat{X}_i - DU_{i|i}$. And since we know that the row space of $\mathcal{R}_i$ (and thus also the first l rows of $\mathcal{R}_i$) is perpendicular to $U_{0|2i-1}, Y_{0|i-1}$ and $\hat{X}_i$, we find (together with (29) and (30)) :

$$\hat{X}_{i+1} = A\hat{X}_i + BU_{i|i} + \begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \\ \hat{X}_i \end{pmatrix}^{\perp} \tag{31}$$

$$Y_{i|i} = C\hat{X}_i + DU_{i|i} + \begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \\ \hat{X}_i \end{pmatrix}^{\perp} \tag{32}$$

where $(.)^{\perp}$ indicates a matrix whose row space is perpendicular to the row space of $(.)$. These formulas will prove to be extremely useful in the next section where we determine the system matrices from Z_i and Z_{i+1}.

This summarizes the whole interpretation as a bank of non-steady state Kalman filters.

5 Identification Scheme

In this section, we derive an **N4SID** *algorithm to identify exactly (unbiased for $j \to \infty$) the deterministic subsystem, directly from the given inputs u_k and outputs y_k. The stochastic subsystem can be determined in an approximate sense.*

5.1 The projections

First, the projections Z_i and Z_{i+1} (13)-(14) have to be calculated. In section 7 we will describe a numerically stable way to do this.
For convenience, we rewrite these projections as follows :

$$Z_i = \left(\begin{array}{c|c|c} \underbrace{L_i^1}_{li\times mi} & \underbrace{L_i^2}_{li\times mi} & \underbrace{L_i^3}_{li\times li} \end{array}\right) \left(\begin{array}{c} U_{0|i-1} \\ \hline U_{i|2i-1} \\ \hline Y_{0|i-1} \end{array}\right) \tag{33}$$

$$Z_{i+1} = \left(\begin{array}{c|c|c} \underbrace{L_{i+1}^1}_{l(i-1)\times m(i+1)} & \underbrace{L_{i+1}^2}_{l(i-1)\times m(i-1)} & \underbrace{L_{i+1}^3}_{l(i-1)\times l(i+1)} \end{array}\right) \left(\begin{array}{c} U_{0|i} \\ \hline U_{i+1|2i-1} \\ \hline Y_{0|i} \end{array}\right) \tag{34}$$

with, from (15)-(18) :

$$L_i^1 = \Gamma_i \Big(\; [A^i - Q_i\Gamma_i]S(R^{-1})_{1|mi} + \Delta_i^d - Q_i H_i^d \; \Big) \tag{35}$$

$$L_i^2 = H_i^d + \Gamma_i[A^i - Q_i\Gamma_i]S(R^{-1})_{mi+1|2mi} \tag{36}$$

$$L_i^3 = \Gamma_i Q_i \tag{37}$$

with $(R^{-1})_{1|mi}$ denoting the submatrix from column 1 to column mi.
The expressions for L^1_{i+1}, L^2_{i+1} and L^3_{i+1} are similar, but with shifted indices.

5.2 Determination of Γ_i and n.

An important observation is that the column space of the matrices L^1_i and L^3_i coincides with the column space of Γ_i. This implies that Γ_i and the order of the system n can be determined from the column space of one of these matrices. The basis for this column space actually determines the basis for the states of the final (identified) state space description. Let us mention two other possible matrices that have the same column space as Γ_i :

$$L^1_i + L^3_i L^2_i \tag{38}$$

$$\left(\begin{array}{c|c} L^1_i & L^3_i \end{array} \right) \left(\frac{U_{0|i-1}}{Y_{0|i-1}} \right) \tag{39}$$

It should be mentioned that, for $i \to \infty$ the first one (38) will lead to a deterministic subsystem that is balanced (See also Moonen & Ramos, 1991) while the second one (39) leads to a deterministic system that is frequency weighted (with the input spectrum) balanced (Enns, 1984) together with a stochastic subsystem of which the forward innovation model is balanced in a deterministic sense. We will not expand any more on this, but keep this for future work.
We can now determine Γ_i, Γ_{i-1} and the order n as follows : Let T be any rank deficient matrix whose column space coincides with that of Γ_i.

- Calculate the Singular Value Decomposition :

$$T = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{pmatrix} V^t$$

- Since T is of rank n, the number of singular values different from zero will be equal to the order of the system.

- The column spaces of Γ_i and $U_1\Sigma_1^{1/2}$ coincide [3]. So, Γ_i can be put equal to $U_1\Sigma_1^{1/2}$.

- With $\underline{\Gamma_i}$ defined as Γ_i without the last l rows (l is the number of outputs), we get :

$$\Gamma_{i-1} = \underline{\Gamma_i}$$

In the following, we will take T equal to the expression in formula (39), but one can replace this with any other matrix of which the column space coincides with Γ_i.

[3] The factor $\Sigma_1^{1/2}$ is introduced for symmetry reasons.

5.3 Determination of the system matrices

We now assume that Γ_i, Γ_{i-1} and n are determined as described in the previous section, and are thus known. From (15)-(16) it follows that[4] :

$$\begin{aligned} \hat{X}_i &= \Gamma_i^\dagger (Z_i - H_i^d U_{i|2i-1}) && (40) \\ \hat{X}_{i+1} &= \Gamma_{i-1}^\dagger (Z_{i+1} - H_{i-1}^d U_{i+1|2i-1}) && (41) \end{aligned}$$

In these formulas, the only unknowns on the right hand side are the matrices H_i^d and H_{i-1}^d. From (31) and (32) we also know that :

$$\left(\frac{\hat{X}_{i+1}}{Y_{i|i}} \right) = \left(\frac{A}{C} \right) \hat{X}_i + \left(\frac{B}{D} \right) U_{i|i} + \begin{pmatrix} U_{0|2i-1} \\ Z_i \\ \hat{X}_i \end{pmatrix}^\perp \qquad (42)$$

If we now substitute the expressions for $\hat{X}_i$ and $\hat{X}_{i+1}$ (40)-(41) in this formula, we get :

$$\left(\frac{\Gamma_{i-1}^\dagger Z_{i+1}}{Y_{i|i}} \right) = \underbrace{\left(\frac{A}{C} \right)}_{\text{term 1}} \Gamma_i^\dagger Z_i + \underbrace{\left(\frac{\mathcal{K}_{12}}{\mathcal{K}_{22}} \right)}_{\text{term 2}} U_{i|2i-1} + \underbrace{\begin{pmatrix} U_{0|2i-1} \\ Z_i \\ \hat{X}_i \end{pmatrix}^\perp}_{\text{term 3}} \qquad (43)$$

where we define :

$$\left(\frac{\mathcal{K}_{12}}{\mathcal{K}_{22}} \right) \overset{\text{def}}{=} \left(\begin{array}{cc} B - A\Gamma_i^\dagger \begin{pmatrix} D \\ \Gamma_{i-1} B \end{pmatrix} & \Gamma_{i-1}^\dagger H_{i-1}^d - A\Gamma_i^\dagger \begin{pmatrix} 0 \\ H_{i-1}^d \end{pmatrix} \\ \hline D - C\Gamma_i^\dagger \begin{pmatrix} D \\ \Gamma_{i-1} B \end{pmatrix} & -C\Gamma_i^\dagger \begin{pmatrix} 0 \\ H_{i-1}^d \end{pmatrix} \end{array} \right) \qquad (44)$$

Observe that the matrices B and D appear linearly in the matrices $\mathcal{K}_{12}$ and $\mathcal{K}_{22}$.
Let Π be a matrix whose row space coincides with that of

$$\begin{pmatrix} \Gamma_i^\dagger Z_i \\ U_{i|2i-1} \end{pmatrix}$$

then (from (43)) :

$$\left(\frac{\Gamma_{i-1}^\dagger Z_{i+1}}{Y_{i|i}} \right) / \Pi = \left(\begin{array}{c|c} A & \mathcal{K}_{12} \\ C & \mathcal{K}_{22} \end{array} \right) \left(\frac{\Gamma_i^\dagger Z_i}{U_{i|2i-1}} \right) / \Pi$$

[4] $A^\dagger$ denotes the Moore-Penrose pseudo-inverse

Obviously, this is a set of linear equations in the unknowns $A, C, \mathcal{K}_{12}, \mathcal{K}_{22}$.
Another point of view is that one could solve the least squares problem :

$$\min_{A,C,\mathcal{K}_{12},\mathcal{K}_{22}} \| \left(\frac{\Gamma_{i-1}^{\dagger} Z_{i+1}}{Y_{i|i}} \right) - \left(\begin{array}{c|c} A & \mathcal{K}_{12} \\ \hline C & \mathcal{K}_{22} \end{array} \right) \left(\frac{\Gamma_i^{\dagger} Z_i}{U_{i|2i-1}} \right) \|_F^2 \tag{45}$$

Either way, from (43) we find (term by term) :

term 1. A and C exactly.

term 2. $\mathcal{K}_{12}$ and $\mathcal{K}_{22}$ from which B and D can be unraveled by solving a set of linear equations, analogous to the one described in De Moor (1988). Note that in (44), B and D appear linearly. Hence if $A, C, \Gamma_i, \Gamma_{i-1}, \mathcal{K}_{12}$ and $\mathcal{K}_{22}$ are known, solving for B and D is equivalent with solving a set of linear equations.

term 3. The residuals of the least squares solution (43) can be written as :

$$\rho = \begin{pmatrix} U_{0|2i-1} \\ Z_i \\ \hat{X}_i \end{pmatrix}^{\perp} = \left(\frac{W_{i|i}}{V_{i|i}} \right)$$

where $W_{.|.}$ and $V_{.|.}$ are block Hankel matrices[5] with as entries w_k and v_k : the process and measurement noise. This is clearly indicated by equation (42).

The system matrices R^s, S^s and Q^s are determined approximately from ρ as follows :

$$\frac{1}{j}(\rho \rho^t) \simeq \begin{pmatrix} Q^s & S^s \\ (S^s)^t & R^s \end{pmatrix}$$

The approximation is due to the fact that the bank of Kalman filters for finite i is not in steady state (see for instance the Riccati difference equation (23)). As i grows larger, the approximation error grows smaller. For infinite i the stochastic subsystem is determined exactly (unbiased). More details on this purely stochastic aspect can be found in Van Overschee & De Moor (1991a,1991b).

5.4 N4SID Algorithm 1

In this section, we summarize the first **N4SID** algorithm step by step, not yet paying attention to the fine numerical details, which will be treated in section 7.

[5] The block Hankel matrices of the residual ρ have only one block row. The notation is introduced to be consistent with previous notations.

1. Determine the projections :

$$
\begin{aligned}
Z_i &= Y_{i|2i-1} / \left(\begin{array}{c} U_{0|i-1} \\ \hline U_{i|2i-1} \\ \hline Y_{0|i-1} \end{array} \right) \\
&= \left(\begin{array}{c|c|c} \underbrace{L_i^1}_{li \times mi} & \underbrace{L_i^2}_{li \times mi} & \underbrace{L_i^3}_{li \times li} \end{array} \right) \left(\begin{array}{c} U_{0|i-1} \\ \hline U_{i|2i-1} \\ \hline Y_{0|i-1} \end{array} \right) \\
Z_{i+1} &= Y_{i+1|2i-1} / \left(\begin{array}{c} U_{0|i} \\ \hline U_{i+1|2i-1} \\ \hline Y_{0|i} \end{array} \right)
\end{aligned}
$$

2. Determine the Singular Value Decomposition

$$
\left(\begin{array}{c|c} L_i^1 & L_i^3 \end{array} \right) \left(\begin{array}{c} U_{0|i-1} \\ \hline Y_{0|i-1} \end{array} \right) = \left(\begin{array}{cc} U_1 & U_2 \end{array} \right) \left(\begin{array}{cc} \Sigma_1 & 0 \\ 0 & 0 \end{array} \right) V^t
$$

The order is equal to the number of non-zero singular values.

$$
\Gamma_i = U_1 \Sigma_1^{1/2} \qquad \text{and} \qquad \Gamma_{i-1} = \underline{U_1} \Sigma_1^{1/2}
$$

3. Determine the least squares solution (ρ_1 and ρ_2 are residuals) :

$$
\begin{array}{cc} & j \\ n & \\ & \Gamma_{i-1}^{\dagger} Z_{i+1} \\ & \hline \\ & Y_{i|i} \\ l & \end{array}
\quad
\begin{array}{c} \\ \\ \left(\begin{array}{c} \Gamma_{i-1}^{\dagger} Z_{i+1} \\ \hline Y_{i|i} \end{array} \right) \end{array}
$$

$$
\begin{array}{c|c} & j \\ \hline n & \Gamma_{i-1}^{\dagger} Z_{i+1} \\ l & Y_{i|i} \end{array}
=
\begin{array}{c|cc} & n & mi \\ \hline n & \mathcal{K}_{11} & \mathcal{K}_{12} \\ l & \mathcal{K}_{21} & \mathcal{K}_{22} \end{array}
\cdot
\begin{array}{c|c} & j \\ \hline n & \Gamma_i^{\dagger} Z_i \\ mi & U_{i|2i-1} \end{array}
+
\begin{array}{c|c} & j \\ \hline n & \rho_1 \\ l & \rho_2 \end{array}
$$

4. The system matrices are determined as follows :

4.1
$$
\begin{aligned}
A &\leftarrow \mathcal{K}_{11} \\
C &\leftarrow \mathcal{K}_{21}
\end{aligned}
$$

4.2 B, D follow from A, C and $\mathcal{K}_{12}, \mathcal{K}_{22}$ through a set of linear equations.

4.3
$$
\left(\begin{array}{c|c} Q^s & S^s \\ \hline (S^s)^t & R^s \end{array} \right) \leftarrow \frac{1}{j} \left(\begin{array}{c|c} \rho_1 \rho_1^t & \rho_1 \rho_2^t \\ \hline \rho_2 \rho_1^t & \rho_2 \rho_2^t \end{array} \right)
$$

The deterministic subsystem will be identified exactly (as $j \rightarrow \infty$, independent of i). The approximation of the stochastic subsystem is still dependent on i and converges as $i \rightarrow \infty$.

6 A Simple Approximate Solution

In this section, we introduce an **N4SID** *algorithm that is very similar to the "exact" (as $j \to \infty$) algorithm of the previous section (algorithm 1). The algorithm we present now finds a good approximation to the state $\hat{X}_i$, and to the system matrices, without having to go through the complicated step 4 for the determination of B and D. This results in a simple and elegant algorithm with a slightly lower computational complexity as compared to algorithm 1. Another advantage of this simplified* **N4SID** *algorithm is that it is very closely related to existing algorithms (Larimore, 1990). This means that the analysis of this simplified algorithm can also be applied to the other algorithms, and can thus contribute to a better understanding of the mechanism of these algorithms. A disadvantage is that the results are not exact (unbiased) for finite i (except for special cases), but an estimate for the bias on the solutions can be calculated.*

If we could determine the state sequences $\hat{X}_i$ and $\hat{X}_{i+1}$ directly from the data, the matrices A, B, C and D could be found as the least squares solution of (31)-(32).

Unfortunately, it is not possible to separate the effect of the input $H_i^d U_{i|2i-1}$ from the effect of the state $\Gamma_i \hat{X}_i$ in formula (15), by just dropping the term with the linear combinations of $U_{i|2i-1}$ (L_i^2) in the expression for Z_i. We would automatically drop a part of the initial state if we did this (see for instance (36)). So, it is not possible to obtain an explicit expression for $\Gamma_i \hat{X}_i$ and $\Gamma_{i-1} \hat{X}_{i+1}$, without knowledge of H_i^d, which would require knowledge of the system matrices.

It is however easy to find a good approximation of the state sequences, directly from the data. If we use this approximation in (31)-(32), we obtain a second very elegant and simple **N4SID** algorithm that calculates approximations of the system matrices.

6.1 The Approximate States

In this section, we derive an approximate expression for the states $\hat{X}_i$ and $\hat{X}_{i+1}$. This approximation can be calculated directly from input-output data.

The approximate state sequences are calculated by dropping the linear combinations of $U_{i|2i-1}$ out of Z_i, and the linear combinations of $U_{i+1|2i-1}$ out of Z_{i+1}. In this way, we obtain Kalman filter states of a different Kalman filter, compared to the one that produces $\hat{X}_i$ (in a sense of different initial conditions). We call the resulting matrices $\Gamma_i \tilde{X}_i$ and $\Gamma_{i-1} \tilde{X}_{i+1}$:

$$\begin{aligned} \Gamma_i \tilde{X}_i &= Z_i - L_i^2 U_{i|2i-1} && (46) \\ \Gamma_{i-1} \tilde{X}_{i+1} &= Z_{i+1} - L_{i+1}^2 U_{i+1|2i-1} && (47) \end{aligned}$$

From (33) and (34) we derive :

$$\begin{aligned} \tilde{X}_i &= \left(\; [A^i - Q_i\Gamma_i]S(R^{-1})_{1|mi} + \Delta_i^d - Q_i H_i^d \;\middle|\; Q_i \; \right) \left(\frac{U_{0|i-1}}{Y_{0|i-1}} \right) && (48) \\ \tilde{X}_{i+1} &= \left(\; [A^{i+1} - Q_{i+1}\Gamma_{i+1}]S(R^{-1})_{1|m(i+1)} + \Delta_{i+1}^d - Q_{i+1} H_{i+1}^d \;\middle|\; Q_{i+1} \; \right) \left(\frac{U_{0|i}}{Y_{0|i}} \right) \end{aligned}$$

The matrices $\Gamma_i\tilde{X}_i$ and $\Gamma_{i-1}\tilde{X}_{i+1}$ can be obtained directly from the data without any knowledge of the system matrices, and can be interpreted as oblique projections as described in Appendix D.

The state sequence $\tilde{X}_i$ is generated by a bank of non-steady state Kalman filters, with : $\tilde{P}^0 = P^d - SR^{-1}S^t + P^s$ and $\tilde{X}^0 = S(R^{-1})_{1|mi}U_{0|i-1}$. The state sequence $\tilde{X}_{i+1}$ on the other hand, is generated by a bank of non-steady state Kalman filters, with : $\tilde{P}^0 = P^d - SR^{-1}S^t + P^s$ and $\tilde{X}^0 = S(R^{-1})_{1|m(i+1)}U_{0|i}$. So clearly, both sequences do not belong to the same bank of Kalman filters, and the useful formulas (31)-(32) are not valid for these 2 sequences.

Still, we will see below, that $\tilde{X}_i$ and $\tilde{X}_{i+1}$ are very close to $\hat{X}_i$ and $\hat{X}_{i+1}$, and that $\hat{X}_i = \tilde{X}_i$, $\hat{X}_{i+1} = \tilde{X}_{i+1}$ if at least one of the following conditions is satisfied :

- $i \rightarrow \infty$
- The deterministic input u_k of the combined deterministic-stochastic system is white noise
- The system is purely deterministic

For these three special cases, we will analyze the difference between $\hat{X}_i$ (17) and $\tilde{X}_i$ (48), defined as δX_i :

$$\delta X_i \overset{\text{def}}{=} \hat{X}_i - \tilde{X}_i = [A^i - Q_i\Gamma_i]S(R^{-1})_{mi+1|2mi}U_{i|2i-1} \qquad (49)$$

6.1.1 $i \rightarrow \infty$

Define the term between square brackets in (49) as $\mathcal{P}_i \overset{\text{def}}{=} A^i - Q_i\Gamma_i$. Now, it is easy to prove (Appendix E) that :

$$\mathcal{P}_i = \prod_{k=0}^{i-1}(A - K_k C) \qquad (50)$$

Since the non-steady state Kalman filter converges to a stable closed loop system (Anderson & Moore, 1979), we find that $\mathcal{P}_i$ grows smaller when i grows larger. It is also clear that :

$\lim_{i\to\infty} \mathcal{P}_i = 0$. In the limit, equation (49) thus becomes (with $S(R^{-1})_{mi+1|2mi}U_{i|2i-1}$ finite) :

$$\lim_{i\to\infty} \delta X_i = \lim_{i\to\infty} \mathcal{P}_i S(R^{-1})_{mi+1|2mi}U_{i|2i-1} = 0$$

The same holds for δX_{i+1}. So, we can conclude that, for $i \to \infty$, there is no difference between the state sequences $\hat{X}_i$ and $\tilde{X}_i$. Actually, when $i \to \infty$, the non-steady state Kalman filter bank converges to a steady state Kalman filter bank, i.e. the Kalman filters converge (the Riccati difference equation (23) converges to an algebraic Riccati equation). To obtain the effect of $i \to \infty$ on a computer, in theory we would need a test like :

$$\frac{\|\delta X_i\|_F}{\sqrt{n.j}} = \frac{\|\mathcal{P}_i S(R^{-1})_{mi+1|2mi}U_{i|2i-1}\|_F}{\sqrt{n.j}} < \epsilon_{mach}$$

where ϵ_{mach} is the machine precision. This is a condition that can only be checked, after the identification is done. Appendix G indicates how this quantity can be calculated.
In practice, it turns out that i does not have to be that large. The identification results are already very good for reasonably small i ($\simeq 10$). This will become apparent in the examples of section 9.

6.1.2 u_k white noise

With the deterministic input u_k white noise, we find : $S = 0$ and $R = I_{2mi}$ with I_{2mi} the $2mi \times 2mi$ identity matrix. So, for a white noise input, we find for any i (see (49)) : $\hat{X}_i = \tilde{X}_i$

6.1.3 Purely deterministic system

From Moonen *et al.* (1989), we know that generically for deterministic systems we have (if there is no 'rank-cancellation', see De Moor (1988)) :

$$\text{rank}\begin{pmatrix} U_{0|i-1} \\ Y_{0|i-1} \end{pmatrix} = mi + n \quad \text{and} \quad \text{rank}\begin{pmatrix} U_{i|2i-1} \\ Y_{i|2i-1} \end{pmatrix} = mi + n \tag{51}$$

which implies that

$$\text{rank}\begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix} = 2mi + n$$

This rank deficiency implies that for purely deterministic systems, the proof of Theorem 2 breaks down ($\mathcal{B}^{-1}$ can not be calculated). The following Theorem is an alternative for Theorem 2 for purely deterministic systems :

Theorem 4 Purely deterministic systems
If the input is persistently exciting and the stochastic subsystem is zero we have :

$$\Gamma_i \tilde{X}_i = \left(\; \Gamma_i [\Delta_i^d - A^i \Gamma_i^\dagger H_i^d] \; \middle| \; \Gamma_i A^i \Gamma_i^\dagger \; \right) \left(\frac{U_{0|i-1}}{Y_{0|i-1}} \right) = \Gamma_i X_i = \Gamma_i \hat{X}_i$$

A proof can be found in Appendix F. This implies that for deterministic systems, δX_i is also equal to zero.

6.2 The Algorithm

We know from (31) and (32) that the least squares solution $\mathcal{L}$ of :

$$\min_{\mathcal{L}} \| \begin{pmatrix} \hat{X}_{i+1} \\ Y_{i|i} \end{pmatrix} - \mathcal{L} \begin{pmatrix} \hat{X}_i \\ U_{i|i} \end{pmatrix} \|_F^2$$

is equal to :

$$\mathcal{L} = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$$

We also know that from the residuals of this least squares problem, we can approximately calculate the stochastic subsystem (exactly if $i \to \infty$). Unfortunately, it is impossible to calculate the states $\hat{X}_i$ and $\hat{X}_{i+1}$ directly from the data, without any knowledge of the system matrices. In the previous section, we have seen that for some special cases $\hat{X}_i = \tilde{X}_i$. In the general case, we have $\hat{X}_i \simeq \tilde{X}_i$ if i is reasonably large. This is because (section 6.1.1)

$$\delta X_i = \prod_{k=0}^{i-1} (A - K_k C) S (R^{-1})_{mi+1|2mi} U_{i|2i-1}$$

Consequently, for the least squares solution $\tilde{\mathcal{L}}$ of :

$$\min_{\tilde{\mathcal{L}}} \| \begin{pmatrix} \tilde{X}_{i+1} \\ Y_{i|i} \end{pmatrix} - \tilde{\mathcal{L}} \begin{pmatrix} \tilde{X}_i \\ U_{i|i} \end{pmatrix} \|_F^2 \qquad (52)$$

we have :

$$\tilde{\mathcal{L}} \simeq \begin{pmatrix} A & B \\ C & D \end{pmatrix}$$

Contrary to $\hat{X}_i$, $\tilde{X}_i$ can be calculated directly from the data, without any knowledge of the system matrices. The solution is exact ($\tilde{\mathcal{L}} = \mathcal{L}$) for the cases of section 6.1.1 ($i \to \infty$), section 6.1.2 (input is white noise) and section 6.1.3 (purely deterministic), since then $\hat{X}_i = \tilde{X}_i$. In Appendix G, an approximate expression for $(\mathcal{L} - \tilde{\mathcal{L}})$ is derived. This is the bias on the solution.

6.3 N4SID Algorithm 2

In this section, we summarize the second **N4SID** algorithm step by step, not yet paying attention to the fine numerical details, which will be treated in section 7.

1. Determine the projections :

$$\begin{aligned}
Z_i &= Y_{i|2i-1}/\left(\begin{array}{c} U_{0|i-1} \\ \hline U_{i|2i-1} \\ \hline Y_{0|i-1} \end{array}\right) \\
&= \left(\underbrace{L_i^1}_{li\times mi} \;\middle|\; \underbrace{L_i^2}_{li\times mi} \;\middle|\; \underbrace{L_i^3}_{li\times li} \right) \left(\begin{array}{c} U_{0|i-1} \\ \hline U_{i|2i-1} \\ \hline Y_{0|i-1} \end{array}\right) \\
Z_{i+1} &= Y_{i+1|2i-1}/\left(\begin{array}{c} U_{0|i} \\ \hline U_{i+1|2i-1} \\ \hline Y_{0|i} \end{array}\right) \\
&= \left(\underbrace{L_{i+1}^1}_{l(i-1)\times m(i+1)} \;\middle|\; \underbrace{L_{i+1}^2}_{l(i-1)\times m(i-1)} \;\middle|\; \underbrace{L_{i+1}^3}_{l(i-1)\times l(i+1)} \right) \left(\begin{array}{c} U_{0|i} \\ \hline U_{i+1|2i-1} \\ \hline Y_{0|i} \end{array}\right)
\end{aligned}$$

2. Determine the Singular Value Decomposition

$$\left(L_i^1 \;\middle|\; L_i^3 \right) \left(\begin{array}{c} U_{0|i-1} \\ \hline Y_{0|i-1} \end{array}\right) = \left(U_1 \quad U_2 \right) \begin{pmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{pmatrix} V^t$$

The order is equal to the number of non-zero singular values.

$$\Gamma_i = U_1 \Sigma_1^{1/2} \qquad \text{and} \qquad \Gamma_{i-1} = \underline{U_1} \Sigma_1^{1/2}$$

3. Determine the states $\tilde{X}_i$ and $\tilde{X}_{i+1}$:

$$\begin{aligned}
\tilde{X}_i &= \Gamma_i^\dagger \left(L_i^1 \;\middle|\; L_i^3 \right) \left(\begin{array}{c} U_{0|i-1} \\ \hline Y_{0|i-1} \end{array}\right) \\
\tilde{X}_{i+1} &= \Gamma_{i-1}^\dagger \left(L_{i+1}^1 \;\middle|\; L_{i+1}^3 \right) \left(\begin{array}{c} U_{0|i} \\ \hline Y_{0|i} \end{array}\right)
\end{aligned}$$

4. Determine the least squares solution :

$$\begin{array}{cc} & j \\ n & \\ l & \end{array}\!\!\left(\begin{array}{c} \tilde{X}_{i+1} \\ \hline Y_{i|i} \end{array}\right) = \begin{array}{c} \\ n \\ l \end{array}\!\!\begin{array}{c} \begin{array}{cc} n & m \end{array} \\ \left(\begin{array}{cc} \tilde{\mathcal{L}}_{11} & \tilde{\mathcal{L}}_{12} \\ \hline \tilde{\mathcal{L}}_{21} & \tilde{\mathcal{L}}_{22} \end{array}\right) \end{array} \cdot \begin{array}{c} \\ n \\ m \end{array}\!\!\begin{array}{c} j \\ \left(\begin{array}{c} \tilde{X}_i \\ \hline U_{i|i} \end{array}\right) \end{array} + \begin{array}{c} \\ n \\ l \end{array}\!\!\begin{array}{c} j \\ \left(\begin{array}{c} \rho_1 \\ \hline \rho_2 \end{array}\right) \end{array}$$

5. The system matrices are (approximately) determined as follows :

$$5.1 \qquad \begin{pmatrix} A & B \\ C & D \end{pmatrix} \quad \leftarrow \quad \begin{pmatrix} \tilde{\mathcal{L}}_{11} & \tilde{\mathcal{L}}_{12} \\ \tilde{\mathcal{L}}_{21} & \tilde{\mathcal{L}}_{22} \end{pmatrix}$$

$$5.2 \qquad \left(\begin{array}{c|c} Q^s & S^s \\ \hline (S^s)^t & R^s \end{array}\right) \quad \leftarrow \quad \frac{1}{j}\left(\begin{array}{c|c} \rho_1\rho_1^t & \rho_1\rho_2^t \\ \hline \rho_2\rho_1^t & \rho_2\rho_2^t \end{array}\right)$$

where in the general case the approximation of the deterministic and stochastic subsystem depends on i (even when $j \to \infty$) and converges as $i, j \to \infty$.

7 A Numerically Stable and Efficient Implementation

In this section, we describe how **N4SID** *algorithms 1 and 2 can be implemented in a numerically stable and efficient way. We make extensive use of the QR decomposition and SVD.*

In section 5.4 and 6.3 we described step by step, two **N4SID** algorithms that determine the system matrices from given input-output data. In this section, we will show how these two algorithms can be implemented in a numerically stable and efficient way.

Steps 1 through 4 are common to both algorithms.

1. Construct the block Hankel matrix $\mathcal{H}$ [6]

$$\mathcal{H} = \underbrace{\begin{pmatrix} U_{0|2i-1} \\ Y_{0|2i-1} \end{pmatrix} / \sqrt{j}}_{2(m+l)i \times j}$$

2. Calculate the R factor of the RQ factorization of $\mathcal{H}$

$$\mathcal{H} = \underbrace{R}_{2(m+l)i \times 2(m+l)i} \quad . \quad \underbrace{Q^t}_{2(m+l)i \times j}$$

with $Q^tQ = I$ and R lower triangular. It is important to note that in the final calculations, only the R factor is needed, which lowers the computational complexity significantly.

[6] The scalar $\sqrt{j}$ is used to be conform with the definition of $\mathbf{E}$.

Partition this factorization in the following way :

$$\begin{array}{l} \\ mi \\ m \\ m(i-1) \\ li \\ l \\ l(i-1) \end{array}
\begin{array}{c} j \\ \left(\begin{array}{c} U_{0|i-1} \\ U_{i|i} \\ U_{i+1|2i-1} \\ Y_{0|i-1} \\ Y_{i|i} \\ Y_{i+1|2i-1} \end{array}\right) \end{array}
=
\begin{array}{l} \\ mi \\ m \\ m(i-1) \\ li \\ l \\ l(i-1) \end{array}
\begin{array}{c} \begin{array}{cccccc} mi & m & m(i-1) & li & l & l(i-1) \end{array} \\ \left(\begin{array}{cccccc} R_{11} & 0 & 0 & 0 & 0 & 0 \\ R_{21} & R_{22} & 0 & 0 & 0 & 0 \\ R_{31} & R_{32} & R_{33} & 0 & 0 & 0 \\ R_{41} & R_{42} & R_{43} & R_{44} & 0 & 0 \\ R_{51} & R_{52} & R_{53} & R_{54} & R_{55} & 0 \\ R_{61} & R_{62} & R_{63} & R_{64} & R_{65} & R_{66} \end{array}\right) \end{array}
\begin{array}{c} j \\ \left(\begin{array}{c} Q_1^t \\ Q_2^t \\ Q_3^t \\ Q_4^t \\ Q_5^t \\ Q_6^t \end{array}\right) \end{array}$$

where we use the shorthand notation $R_{4:6,1:3}$ for the submatrix of R consisting of block rows 4 to 6 and block columns 1 to 3.

3. Calculate the projections (see also the note at the end of this section).

$$\begin{aligned} Z_i &= R_{5:6,1:4}\, R_{1:4,1:4}^{-1} \left(\begin{array}{c} U_{0|2i-1} \\ Y_{0|i-1} \end{array}\right) = \left(\begin{array}{c|c|c} L_i^1 & L_i^2 & L_i^3 \end{array}\right) \left(\begin{array}{c} U_{0|i-1} \\ \hline U_{i|2i-1} \\ \hline Y_{0|i-1} \end{array}\right) \\ Z_{i+1} &= R_{6:6,1:5}\, R_{1:5,1:5}^{-1} \left(\begin{array}{c} U_{0|2i-1} \\ Y_{0|i} \end{array}\right) = \left(\begin{array}{c|c|c} L_{i+1}^1 & L_{i+1}^2 & L_{i+1}^3 \end{array}\right) \left(\begin{array}{c} U_{0|i} \\ \hline U_{i+1|2i-1} \\ \hline Y_{0|i} \end{array}\right) \end{aligned}$$

4. Determine Γ_i and n through the SVD of $\Gamma_i \tilde{X}_i$:

$$\left(\begin{array}{ccc} L_i^1 & 0 & L_i^3 \end{array}\right) R_{1:4,1:4} Q_{1:4}^t = \left(\begin{array}{cc} U_1 & U_2 \end{array}\right) \left(\begin{array}{cc} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{array}\right) (Q_{1:4}\, V)^t$$

Note that this SVD can be calculated through the SVD of $(\ L_i^1\ 0\ L_i^3\ ) R_{1:4,1:4}$, since $Q_{1:4}$ is an orthonormal matrix. The rank is determined from the dominant singular values of this decomposition (Σ_1), and Γ_i can be chosen as :

$$\Gamma_i = U_1 \Sigma_1^{1/2} \quad , \quad \Gamma_{i-1} = \underline{\Gamma_i}$$

where the underbar means deleting the last l rows (l is the number of outputs). The two algorithms described in section 5 and 6 now differ in step 5 and 6.

N4SID Algorithm 1 :

5. We find for the left hand side and right hand side of equation (43), written as a function of the original Q matrix :

$$\begin{pmatrix} \Gamma_i^\dagger Z_i \\ U_{i|2i-1} \end{pmatrix} = \begin{pmatrix} \Sigma_1^{-1/2} U_1^t R_{5:6,1:4} \\ R_{2:3,1:4} \end{pmatrix} Q_{1:4}^t$$

$$\begin{pmatrix} \Gamma_{i-1}^\dagger Z_{i+1} \\ Y_{i|i} \end{pmatrix} = \begin{pmatrix} \Sigma_1^{-1/2} (\underline{U_1})^\dagger R_{6:6,1:5} \\ R_{5:5,1:5} \end{pmatrix} Q_{1:5}^t$$

6. Now the least squares problem (45) can be rewritten as

$$\min_{\mathcal{K}} \| \begin{pmatrix} \Sigma_1^{-1/2} (\underline{U_1})^\dagger R_{6:6,1:4} \\ R_{5:5,1:4} \end{pmatrix} - \mathcal{K} \begin{pmatrix} \Sigma_1^{-1/2} U_1^t R_{5:6,1:4} \\ R_{2:3,1:4} \end{pmatrix} \|_F^2$$

and solved in a least squares sense for $\mathcal{K}$. Note that the Q matrix has completely disappeared from these final formulas. The first n columns of $\mathcal{K}$ are A and C stacked on top of each other. The next columns determine B and D as described in section 5. The residuals of this solution determine the stochastic system as described in section 5.

N4SID Algorithm 2 :

5. We have :

$$\begin{pmatrix} \tilde{X}_i \\ U_{i|i} \end{pmatrix} = \begin{pmatrix} \Sigma_1^{-1/2} U_1^t \begin{pmatrix} L_i^1 & 0 & L_i^3 \end{pmatrix} R_{1:4,1:4} \\ R_{2:2,1:4} \end{pmatrix} Q_{1:4}^t$$

$$\begin{pmatrix} \tilde{X}_{i+1} \\ Y_{i|i} \end{pmatrix} = \begin{pmatrix} \Sigma_1^{-1/2} (\underline{U_1})^\dagger \begin{pmatrix} L_{i+1}^1 & 0 & L_{i+1}^3 \end{pmatrix} R_{1:5,1:5} \\ R_{5:5,1:5} \end{pmatrix} Q_{1:5}^t$$

6. Now the least squares problem (52) can be rewritten as

$$\min_{\tilde{\mathcal{L}}} \| \begin{pmatrix} \Sigma_1^{-1/2} (\underline{U_1})^\dagger \begin{pmatrix} L_{i+1}^1 & 0 & L_{i+1}^3 \end{pmatrix} R_{1:5,1:4} \\ R_{5:5,1:4} \end{pmatrix} - \tilde{\mathcal{L}} \begin{pmatrix} \Sigma_1^{-1/2} U_1^t \begin{pmatrix} L_i^1 & 0 & L_i^3 \end{pmatrix} R_{1:4,1:4} \\ R_{2:2,1:4} \end{pmatrix} \|_F^2$$

Once again, the Q matrix has completely disappeared from these final formulas. From $\tilde{\mathcal{L}}$ and the residuals we can find the system matrices as described in section 6.

Note :

- The projection of step 3 is written as a function of U and Y, and not as a function of Q, because we need to be able to drop the linear combinations (L_i^2) of the rows of $U_{i|2i-1}$ in Z_i to find the sequence $\Gamma_i\tilde{X}_i$ and the linear combination (L_{i+1}^2) of the rows of $U_{i+1|2i-1}$ in Z_{i+1} to find the sequence $\Gamma_{i-1}\tilde{X}_{i+1}$ (see formulas (46)-(47)).

- The possible rank deficiency of the row space we are projecting on (in step 3) has to be taken into account (see section 6.1.3). This occurs when $R_{1:4,1:4}$ and $R_{1:5,1:5}$ are rank deficient. It can be checked by inspection of the singular values of $R_{1:4,1:4}$. If one of them turns out to be zero (purely deterministic case), a basis Π for the space we are projecting on has to be selected. This basis should contain the row vectors of $U_{i|2i-1}$ explicitly. This makes it easy to drop the linear combinations of $U_{i|2i-1}$ (L_i^2) in Z_i to obtain the sequence $\Gamma_i\tilde{X}_i$.

  With the singular value decomposition :

$$\begin{pmatrix} U_{0|i-1} \\ Y_{0|i-1} \end{pmatrix} = \begin{pmatrix} R_{1:1,1:4} \\ R_{4:4,1:4} \end{pmatrix} Q_{1:4}^t = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_1^t \\ V_2^t \end{pmatrix} Q_{1:4}^t$$

  we find as an appropriate basis :

$$\Pi = \begin{pmatrix} V_1^t \\ R_{2:3,1:4} \end{pmatrix} Q_{1:4}^t$$

  The rest of the algorithm is very similar to the general case.

  We should note that in the generic real life case, the problem of rank deficiency is a pure academic one.

8 Connection with existing algorithms

8.1 Instrumental variable method

This method was described by De Moor *et al.* (1991), Verhaegen (1991). The basic idea will be shortly repeated here. We start from the input-output equation (11) : $Y_{i|2i-1} = \Gamma_i X_i^d + H_i^d U_{i|2i-1} + Y_{i|2i-1}^s$. Projection of this equation on the row space perpendicular to that of $U_{i|2i-1}$, gives :

$$Y_{i|2i-1}/U_{i|2i-1}^\perp = \Gamma_i X_i^d/U_{i|2i-1}^\perp + Y_{i|2i-1}^s \tag{53}$$

Note that $Y_{i|2i-1}^s/U_{i|2i-1}^\perp = Y_{i|2i-1}^s$ since $Y_{i|2i-1}^s/U_{i|2i-1} = 0$ as $j \to \infty$. Projection of (53) on $U_{0|i-1}$ gives : $[Y_{i|2i-1}/U_{i|2i-1}^\perp]/U_{0|i-1} = \Gamma_i[X_i^d/U_{i|2i-1}^\perp]/U_{0|i-1}$ since $Y_{i|2i-1}^s/U_{0|i-1} = 0$. It

can be seen from this equation that the column space of $[Y_{i|2i-1}/U^{\perp}_{i|2i-1}]/U_{0|i-1}$ is equal to that of Γ_i^d (only the deterministic controllable modes are observed). So, from the column space of Γ_i^d, A and C can be derived. B and D are then found from the fact that : $(\Gamma_i^d)^{\perp}[Y_{i|2i-1}/U_{i|2i-1}]U^{\dagger}_{i|2i-1} = (\Gamma_i^d)^{\perp}H_i^d$ which leads to a set of linear equations in B and D. Finally, the stochastic subsystem is identified from the difference between the measured output and a simulation of the identified deterministic subsystem with the given input u_k. This difference is almost equal to the output of the stochastic subsystem (y_k^s). The stochastic subsystem can then be identified with for instance one of the algorithms described by Arun & Kung (1990), Van Overschee & De Moor (1991a,1991b).
A clear disadvantage of this algorithm is the fact that the deterministic and stochastic subsystem are identified separately. This involves more calculations. Two different A matrices will also be identified (one for the deterministic and one for the stochastic system), even though a lot of the dynamics could be shared between the two subsystems.

8.2 Intersection Algorithms

In the literature, a couple of algorithms, which we call 'intersection algorithms' have been published by Moonen *et al.* (1989), Moonen *et al.* (1992). It is interesting to note the analogy between these algorithms, and algorithm 2 described in section 6. We show that the row space of $\tilde{X}_i$ can also be found as the intersection between the row spaces of two Hankel matrices $\mathcal{H}_1$ and $\mathcal{H}_2$. Define $\mathcal{I}$ = row space $\mathcal{H}_1 \cap$ row space $\mathcal{H}_2$ with :

$$\mathcal{H}_1 = \begin{pmatrix} Y_{0|i-1} \\ U_{0|i-1} \end{pmatrix} \quad \text{and} \quad \mathcal{H}_2 = \begin{pmatrix} Y_{i|2i-1}/\begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix} \\ U_{i|2i-1} \end{pmatrix}$$

Facts :

- The subspace $\mathcal{I}$ is n dimensional.

  Proof : We have : rank $\mathcal{H}_1 = \kappa_1$, with $\kappa_1 = (m+l)i$ in the generic combined deterministic-stochastic case. From equation (15), we find that : rank $\mathcal{H}_2 = mi + n$. Finally, if the input is persistently exciting : rank $\begin{pmatrix} \mathcal{H}_1^t & \mathcal{H}_2^t \end{pmatrix}^t = \kappa_1 + mi$. With Grassmann's Theorem, we find : dim $[\mathcal{H}_1 \cap \mathcal{H}_2]$ = rank $\mathcal{H}_1$ + rank $\mathcal{H}_2$ − rank $\begin{pmatrix} \mathcal{H}_1^t & \mathcal{H}_2^t \end{pmatrix}^t = (\kappa_1) + (mi + n) - (\kappa_1 + mi) = n$

- The intersection subspace $\mathcal{I}$ is equal to the row space of $\tilde{X}_i$. **Proof :**

  – $\tilde{X}_i$ lies in the row space of $\mathcal{H}_1$, since it is written as a linear combination of $U_{0|i-1}$ and $Y_{0|i-1}$ (see Formula (48)).

- $\tilde{X}_i$ lies also in the row space of $\mathcal{H}_2$ since we know from (46) that :

$$\tilde{X}_i = \Gamma_i^{\dagger}[Y_{i|2i-1}/ \begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix} - L_i^2 U_{i|2i-1}]$$

So, $\tilde{X}_i$ lies in the intersection of the row spaces of $\mathcal{H}_1$ and $\mathcal{H}_2$. Since both the intersection subspace $\mathcal{I}$ and the row space of $\tilde{X}_i$ are n dimensional, they must coincide.

This derivation indicates strong similarities between **N4SID** algorithm 2 and the existing intersection algorithm mentioned above. When $j \to \infty$, both algorithms will calculate the same solution (if the same state space basis is used). In practice however (finite j), they calculate slightly different models. In the references given above, only purely deterministic systems and the asymptotic case $i \to \infty$ are treated. The interpretation of Kalman filter states is totally absent, as is the effect of finite i.
The similarities indicate an alternative way to calculate the states $\tilde{X}_i$ and $\tilde{X}_{i+1}$ as intersections (see Moonen, 1992 for example).

8.3 Interpretation of general projection methods

Larimore (1990) shows that viewed from a statistical point of view, the state (memory) M_i is the solution to :

$$p[Y_{i|2i-1}| \begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix}] = p[Y_{i|2i-1}| \begin{pmatrix} U_{i|2i-1} \\ M_i \end{pmatrix}]$$

where $p[A|B]$ is the probability distribution of A given B.
For Gaussian processes, these distributions are characterized by the first two moments, and we can replace $p[A|B]$ by the expectation operation $\mathbf{E}[A|B]$. This expectation operator can in its turn be replaced by the projection operator (Papoulis, 1984). We then get :

$$Y_{i|2i-1}/ \begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix} = Y_{i|2i-1}/ \begin{pmatrix} U_{i|2i-1} \\ M_i \end{pmatrix} \qquad (54)$$

By substitution it is easy to verify that all of the following 3 alternatives : $M_i = \hat{X}_i$, $M_i = \tilde{X}_i$ and $M_i = \hat{X}_i/U^{\perp}_{i|2i-1}$ satisfy equation (54). Actually, every matrix M_i satisfying $M_i = Z_i - \Omega U_{i|2i-1}$ and rank $M_i = n$ with Ω an arbitrary $li \times mi$ matrix, will also satisfy equation (54) and give rise to an n dimensional memory. This ambiguity is due to the fact that the exact influence of the input variable $U_{i|2i-1}$ on the output is not known. In the frame work of linear theory, the most natural choice for M_i is though the one for which :

$$Y_{i|2i-1} / \begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix} = \Gamma_i M_i + H_i^d U_{i|2i-1}$$

is satisfied ($\Omega = H_i^d$). This is because this choice corresponds to the linear state space equations :

$$\begin{aligned} M_{i+1} &= AM_i + BU_{i|i} \\ Y_{i|i} / \begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix} &= CM_i + DU_{i|i} \end{aligned}$$

This leads to the choice $M_i = \hat{X}_i$.

9 Examples

9.1 A simple example

This simple simulation example illustrates the concepts explained in this paper. We consider the single input single output single state system in forward innovation form (Faure, 1976) :

$$\begin{aligned} x_{k+1} &= 0.9490 x_k + 1.8805 u_k - 0.1502 e_k \\ y_k &= 0.8725 x_k - 2.0895 u_k + 2.5894 e_k \end{aligned}$$

With e_k a unit energy ($\sigma = 1$), zero mean, Gaussian distributed stochastic sequence.
Effect of i :
We first investigate the effect of the number of block rows (i) of the block Hankel matrices. The input u_k to the system is the same for all experiments and is equal to a filtered unit energy ($\sigma = 1$), zero mean, Gaussian distributed stochastic sequence added to a Gaussian white noise sequence ($\sigma = 0.1$). The filter is a second order Butterworth filter, with cut-off frequency equal to 0.025 (sampling time 1). The number of data used for identification is fixed at 1000. 100 different disturbance sequences e_k were generated. For each of these sequences and for each i in the interval $[2, 15]$, two models were identified using **N4SID** algorithm 1 (section 5) and 2 (section 6). Also for each of these models the bias was calculated using the expression in appendix G. Then the mean of all these quantities over the 100 different disturbance sequences was calculated (Monte Carlo experiment).

Fig. 4 shows the results as a function of i for both **N4SID** algorithms. The results for algorithm 1 are represented with a dotted line (:) and circles (o). The results for algorithm 2 are represented with a dotted-dashed (-.) line and the stars (*). The exact values are indicated with a dashed line.

Fig. a/ shows the eigenvalue of the system as a function of i. Clearly the estimates are accurate, and there is hardly any difference between the two algorithms. Fig. b/ shows the deterministic zero of the system $(A - BD^{-1}C)$ as a function of i. The difference between the two **N4SID** algorithms is clearly visible. Algorithm 1 estimates a zero that is close to the exact deterministic zero, independently of i. Algorithm 2 on the other hand is clearly biased for small i. Fig. c/ shows this bias (dashed-dotted line) and the estimated bias (dotted line) using the expressions of appendix G as a function of i. As expected, the bias grows smaller as i grows larger. Fig. d/ shows the calculated and estimated bias on the identified D matrix. This matrix seems to be a lot more sensitive to i than A.

Finally Fig. e/ shows the estimated stochastic zero as a function of i (the eigenvalues of $A - KC$, with K the steady state Kalman gain). The convergence is very slow (the exact zero is 0.9996). More details can be found in Van Overschee & De Moor (1991a,1991b).

For this example, we can conclude that both algorithms do a good job of estimating A and C. B and D are estimated accurately with **N4SID** algorithm 1, but not with algorithm 2 (for small i). The bias can be calculated though. The accuracy of the stochastic subsystem is strongly dependent on i for both algorithms.

Effect of j :

Throughout the paper, we assumed that $j \to \infty$. The effect of finite j is shown in Fig. 4 f/. This Figure shows the standard deviation (stars) of the estimate of A as a function of j $(i = 5)$. For each j, 100 Monte Carlo experiments were done. The standard deviation is proportional to $1/\sqrt{j}$ (dashed line). So, the accuracy of the results depend on j as $1/\sqrt{j}$.

9.2 A glass oven

The glass oven has 3 inputs (2 burners and 1 ventilator) and 6 outputs (temperature). The data have been pre-processed : detrending, peak shaving, delay estimates, normalization (Backx, 1987).

Using 700 data points, five different models are identified (the 300 following points are used as validation data) :

1. A state space model with **N4SID** algorithm 1 of section 5
2. A state space model with **N4SID** algorithm 2 of section 6

3. A state space model with the algorithm of section 8.1
4. An ARX model (Ljung, 1991) followed by a balanced truncation
5. A prediction error model (Ljung, 1991)

The results are summarized in Table 1
The first row indicates the chosen order. Fig. 5 shows the singular values that led to the system order 5 for **N4SID** algorithm 1 and 2. The gap in the singular values is clearly visible. For the ARX model, the parameters 'na/nb' indicate that 'na' was a 6×6 matrix with as entries 4, and 'nb' a 6×3 matrix with as entries 4 (see also Ljung, 1991). In this way, the resulting state space model was of order 33. It was reduced to a fourth order model using balanced truncation of the deterministic subsystem. This is basically the same philosophy as for the **N4SID** algorithms, but the last ones do not calculate the intermediate model explicitly. For the prediction error model, we used the identified model of algorithm 1 in the controllability canonical form (controllability indices : $2, 2, 1$. Number of parameters : 93) as an initial value (the initial values obtained from 'canform' or 'canstart' of Ljung (1991) did not converge). We restricted the number of iterations to 3 (no improvement with more iterations).
The third row shows the number of floating point operations. Algorithm 2 needs about 5% less computation (simpler algorithm). The calculation of the ARX model takes 3 times as much computation. The calculation of the prediction error model (only 3 iterations) takes about 25 times as much computation ! It should be mentioned that none of the algorithms was optimized in the number of operations (nor ours, nor the ones in the Matlab toolbox), but these figures still give an idea of the order of magnitude of the number of computations.
The fourth row is the error (in percent) between the measured validation output and the simulated output using only the deterministic subsystem. The fifth row shows the error between measured and Kalman filter one step ahead prediction.
With $[y_k^v]_i$ the i^{th} validation output channel and $[y_k^s]_i$ the i^{th} simulated output channel, the error is defined as ($N_v = 300$) :

$$\epsilon = 100 \; [\; \frac{1}{6} \sum_{i=1}^{6} \sqrt{\frac{\sum_{k=1}^{N_v}([y_k^v]_i - [y_k^s]_i)^2}{\sum_{k=1}^{N_v}([y_k^v]_i)^2}} \;] \; \%$$

The best deterministic prediction is obtained with the ARX model, which is logical since it came about by balanced truncation of the *deterministic* subsystem. The other models perform a little worse (but all four about the same).

The Kalman filter error is the smallest though for the first two **N4SID** algorithms. The projection retains the deterministic *and* stochastic information in the past useful to predict the future. This implies that the resulting models will perform best when used with a Kalman filter. The prediction error model does not improve this Kalman filter prediction error at all. There is even a slight decline which is probably due to numerical errors since during the iteration procedure a warning for a badly conditioned matrix was given. This means that, even though there is no optimization involved, the results obtained with the **N4SID** algorithms, for this industrial MIMO process are close to optimal.
For this example, the **N4SID** algorithms thus calculate a state space model without any a-priori fixed parametrization, and this in a fast and numerically reliable way.
Finally, we should mention that, as we found out in discussion with Lennart Ljung, the prediction error method can be made to work better with some extra manipulations. The main problem with the prediction error method is that the outputs are pairwise collinear. This results in a very hard, ill-conditioned optimization problem, for every possible parametrization. This problem can be solved as follows : first estimate an initial model. Now, in the optimization step, only the parameters of the C and D matrix corresponding to three outputs can vary (output 1, 4, and 6 for this example). The optimization is better conditioned now and the resulting model has a similar performance as the subspace models for this example. The number of floating point operations stays extremely high though and this method is rather complicated and requires quite some insight from the identifiers.

10 Conclusions

In this paper two new **N4SID** algorithms to identify combined deterministic-stochastic linear systems have been derived. The first one calculates unbiased results, the second one is a simpler biased approximation. The connection between these **N4SID** algorithms and the classical Kalman filter estimation theory has been indicated. This allows interpretations and proofs of correctness of the algorithms.

In future work, we will discuss other connections with linear system theory : model reduction properties, frequency interpretations and connections with robust control.
The open problems that remain to be solved are the connection with maximum likelihood algorithms (Ljung, 1987) and the problem of finding good asymptotic statistical error estimates.

11 Acknowledgments

We would like to thank Lennart Ljung and Mats Viberg for their help and useful comments.

References

Anderson B.D.O., Moore J.B. (1979). *Optimal Filtering.* Prentice Hall.

Arun K.S., Kung S.Y. (1990). *Balanced Approximation of Stochastic Systems.* SIAM Matrix Analysis and Applications, **11**, 42-68.

Astrom K., Wittenmark B. (1984). *Computer Controlled Systems : Theory and Design.* Prentice Hall.

Backx T. (1987). *Identification of an Industrial Process : A Markov Parameter Approach.* Doctoral Dissertation, Technical University Eindhoven, The Netherlands.

Baksalary J.K., Kala R. (1979). *Two relations Between Oblique and Λ-Orthogonal Projectors.* Linear Algebra and its Applications **24**, 99-103.

De Moor B. (1988). *Mathematical concepts and techniques for modeling of static and dynamic systems.* Doctoral Dissertation, Department of Electrical Engineering, Kath. Universiteit Leuven, Dept. E.E. Belgium.

De Moor B., Van Overschee P., Suykens J. (1991). *Subspace algorithms system identification and stochastic realization.* Proceedings MTNS, Kobe, Japan, part 1, 589-594.

Enns D. (1984). *Model reduction for control system design.* Ph.D.. dissertation, Dep. Aeronaut. Astronaut., Stanford University, Stanford CA.

Faure P. (1976). *Stochastic realization algorithms.* System identification : Advances and case studies. Eds. Mehra R. and Lainiotis D., Academic Press.

Kailath T. (1980). *Linear Systems.* Prentice Hall.

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

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

Ljung L. (1987). *System Identification : theory for the user.* Prentice Hall Information and System Sciences Series.

Ljung L. (1991). *System Identification Toolbox.* The Mathworks, Inc.

Moonen M., De Moor B., Vandenberghe L., Vandewalle J. (1989). *On- and off-line identification of linear state space models.* International Journal of Control, **49**, 219-232.

Moonen M., Ramos J. (1991). *A subspace algorithm for balanced state space system identification.* ESAT/SISTA report 1991-07, Kath. Universiteit Leuven, Dept. E.E., Belgium. Accepted for publication in IEEE Automatic Control.

Moonen M., Van Overschee P., De Moor B. (1992). *A Subspace Algorithm for Combined Stochastic-Deterministic State Space Identification.* ESAT/SISTA report 1992-10, Kath. Universiteit Leuven, Dept. E.E. , Belgium.

Papoulis A. (1984). *Probability, Random Variables, and Stochastic Processes.* McGraw-Hill International Book Company.

Van Overschee P., De Moor B. (1991a). *Subspace Algorithms for the Stochastic Identification Problem.* 30th IEEE Conference On Decision and Control, Brighton, England, 1321-1326.

Van Overschee P., De Moor B. (1991b). *Subspace Algorithms for the Stochastic Identification Problem.* ESAT/SISTA report 1991-26, Kath. Universiteit Leuven, Dept. E.E. , Belgium. Accepted for publication in Automatica.

Verhaegen M. (1991). *A novel non-iterative MIMO state space model identification technique.* Preprints 9th IFAC/IFORS symposium on Identification and System parameter estimation, Budapest, Hungary, 1453-1458.

Appendices :

A Proof of the projection theorem

Proof of formula (33) :

Note : Due to the complexity of the formulas, we use the subscripts p (past) and f (future) throughout this appendix as follows : For U, Y^d and Y^s, these subscripts denote respectively the subscript $0|i-1$ and $i|2i-1$. For X^d and X^s they denote the subscript 0 and i. We also make use of (9) - (12) without explicitly mentioning it.

Define

$$\mathcal{A} \overset{\text{def}}{=} \lim_{j\to\infty} \frac{1}{j}(Y_f \left(\begin{array}{c|c|c} U_p^t & U_f^t & Y_p^t \end{array} \right)) = \left(\begin{array}{c|c|c} \mathcal{A}_1 & \mathcal{A}_2 & \mathcal{A}_3 \end{array} \right)$$

We have :

$$\begin{aligned}
\mathcal{A}_1 &= \lim_{j\to\infty} \frac{1}{j} Y_f U_p^t \\
&= \lim_{j\to\infty} \frac{1}{j}(\Gamma_i A^i X_p^d + \Gamma_i \Delta_i^d U_p + H_i^d U_f + Y_f^s) U_p^t \\
&= \Gamma_i A^i S_1 + \Gamma_i \Delta_i^d R_{11} + H_i^d R_{12}^t
\end{aligned}$$

$$\begin{aligned}
\mathcal{A}_2 &= \lim_{j\to\infty} \frac{1}{j} Y_f U_f^t \\
&= \lim_{j\to\infty} \frac{1}{j}(\Gamma_i A^i X_p^d + \Gamma_i \Delta_i^d U_p + H_i^d U_f + Y_f^s) U_f^t \\
&= \Gamma_i A^i S_2 + \Gamma_i \Delta_i^d R_{12} + H_i^d R_{22}
\end{aligned}$$

$$\begin{aligned}
\mathcal{A}_3 &= \lim_{j\to\infty} \frac{1}{j} Y_f Y_p^t \\
&= \lim_{j\to\infty} \frac{1}{j}(\Gamma_i A^i X_p^d + \Gamma_i \Delta_i^d U_p + H_i^d U_f + Y_f^s)((X_p^d)^t \Gamma_i^t + U_p^t (H_i^d)^t + (Y_p^s)^t) \\
&= \Gamma_i A^i P^d \Gamma_i^t + \Gamma_i A^i S_1 (H_i^d)^t + \Gamma_i \Delta_i^d S_1^t \Gamma_i^t + \Gamma_i \Delta_i^d R_{11} (H_i^d)^t + H_i^d S_2^t \Gamma_i^t + H_i^d R_{12}^t (H_i^d)^t + H_i^s \\
&= \Gamma_i A^i P^d \Gamma_i^t + \Gamma_i \Delta_i^d R_{11} (H_i^d)^t + \Gamma_i A^i S_1 (H_i^d)^t + \Gamma_i \Delta_i^d S_1^t \Gamma_i^t + H_i^d R_{12}^t (H_i^d)^t + H_i^d S_2^t \Gamma_i^t + \Gamma_i \Delta_i^s
\end{aligned}$$

Thus, we find :

$$\begin{aligned}
\mathcal{A} &= \left(\begin{array}{cc|c} \underbrace{\mathcal{A}_1}_{li\times mi} & \underbrace{\mathcal{A}_2}_{li\times mi} & \underbrace{\mathcal{A}_3}_{li\times li} \end{array} \right) \\
&= \left[\begin{array}{c|c} \Gamma_i A^i S + \Gamma_i \Delta_i^d (\; R_{11} \; R_{12} \;) & \Gamma_i A^i P^d \Gamma_i^t + \Gamma_i \Delta_i^s + \Gamma_i \Delta_i^d R_{11} (H_i^d)^t + \Gamma_i A^i S_1 (H_i^d)^t \\ + H_i^d (\; R_{12}^t \; R_{22} \;) & + \Gamma_i \Delta_i^d S_1^t \Gamma_i^t + H_i^d R_{12}^t (H_i^d)^t + H_i^d S_2^t \Gamma_i^t \end{array} \right]
\end{aligned} \quad (55)$$

The second part we need to express in terms of the system matrices is :

$$\mathcal{B} \overset{\text{def}}{=} \lim_{j\to\infty} \frac{1}{j} \left(\begin{array}{c} U_p \\ U_f \\ \hline Y_p \end{array} \right) \left(\begin{array}{cc|c} U_p^t & U_f^t & Y_p^t \end{array} \right) = \left(\begin{array}{c|c} \mathcal{B}_{11} & \mathcal{B}_{21}^t \\ \hline \mathcal{B}_{21} & \mathcal{B}_{22} \end{array} \right)$$

with :

$$\begin{aligned}
\mathcal{B}_{21} &= \lim_{j\to\infty}\frac{1}{j}Y_p[\,U_p^t\;U_f^t\,] \\
&= \lim_{j\to\infty}\frac{1}{j}(\Gamma_i X_p^d + H_i^d U_p + Y_p^s)[\,U_p^t\;U_f^t\,] \\
&= \Gamma_i S + H_i^d[\,R_{11}\;R_{12}\,]
\end{aligned}$$

$$\begin{aligned}
\mathcal{B}_{22} &= \lim_{j\to\infty}\frac{1}{j}Y_p Y_p^t \\
&= \lim_{j\to\infty}\frac{1}{j}(\Gamma_i X_p^d + H_i^d U_p + Y_p^s)((X_p^d)^t\Gamma_i^t + U_p^t(H_i^d)^t + (Y_p^s)^t) \\
&= \Gamma_i P^d \Gamma_i^t + \Gamma_i S_1 (H_i^d)^t + H_i^d S_1^t \Gamma_i^t + H_i^d R_{11}(H_i^d)^t + L_i^s
\end{aligned}$$

and $\mathcal{B}_{11} = R$.
Note that $\mathcal{B}$ is guaranteed to be of full rank $((2m+l)i)$ due to the persistently exciting input and the non-zero stochastic subsystem. To compute $\mathcal{B}^{-1}$, we use the formula for the inverse of a block matrix (Kailath, 1980) :

$$\left(\begin{array}{c|c} \mathcal{B}_{11} & \mathcal{B}_{21}^t \\ \hline \mathcal{B}_{21} & \mathcal{B}_{22} \end{array}\right)^{-1} = \left(\begin{array}{c|c} \mathcal{B}_{11}^{-1} + \mathcal{B}_{11}^{-1}\mathcal{B}_{21}^t\psi^{-1}\mathcal{B}_{21}\mathcal{B}_{11}^{-1} & -\mathcal{B}_{11}^{-1}\mathcal{B}_{21}^t\psi^{-1} \\ \hline -\psi^{-1}\mathcal{B}_{21}\mathcal{B}_{11}^{-1} & \psi^{-1} \end{array}\right) \tag{56}$$

with $\psi = \mathcal{B}_{22} - \mathcal{B}_{21}\mathcal{B}_{11}^{-1}\mathcal{B}_{21}^t$.
So, in our case, this becomes :

$$\mathcal{B}^{-1} = \left(\begin{array}{c|c} R^{-1} + (\begin{pmatrix} I \\ 0 \end{pmatrix}(H_i^d)^t + R^{-1}S^t\Gamma_i^t)\psi_i^{-1}(H_i^d\begin{pmatrix} I & 0 \end{pmatrix} + \Gamma_i S R^{-1}) & -(\begin{pmatrix} I \\ 0 \end{pmatrix}(H_i^d)^t + R^{-1}S^t\Gamma_i^t)\psi_i^{-1} \\ \hline -\psi_i^{-1}(H_i^d\begin{pmatrix} I & 0 \end{pmatrix} + \Gamma_i S R^{-1}) & \psi_i^{-1} \end{array}\right) \tag{57}$$

with

$$\begin{aligned}
\psi_i &= \Gamma_i P^d \Gamma_i^t + L_i^s + \Gamma_i S_1 (H_i^d)^t + H_i^d S_1^t \Gamma_i^t + H_i^d R_{11}(H_i^d)^t - (\Gamma_i S + H_i^d[\,R_{11}\;R_{12}\,])R^{-1}(S^t\Gamma_i^t + \begin{bmatrix} R_{11} \\ R_{12}^t \end{bmatrix}(H_i^d)^t) \\
&= \Gamma_i(P^d - SR^{-1}S^t)\Gamma_i^t + L_i^s
\end{aligned}$$

From formulas (13) and (33) and the definitions of $\mathcal{A}$ and $\mathcal{B}$, we find (with $j \to \infty$):

$$\begin{aligned}
\begin{pmatrix} L_i^1 & L_i^2 & L_i^3 \end{pmatrix} &= \frac{1}{j}Y_{i|2i-1}\begin{pmatrix} U_{0|2i-1}^t & Y_{0|i-1}^t \end{pmatrix}[\frac{1}{j}\begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix}\begin{pmatrix} U_{0|2i-1}^t & Y_{0|i-1}^t \end{pmatrix}]^{-1} \\
&= \mathcal{A}\,\mathcal{B}^{-1}
\end{aligned}$$

Using (55) and (57), we thus find :

$$\begin{aligned}
\begin{pmatrix} L_i^1 & L_i^2 \end{pmatrix} &= \Gamma_i A^i S R^{-1} + \Gamma_i \Delta_i^d \begin{pmatrix} I & 0 \end{pmatrix} + H_i^d \begin{pmatrix} 0 & I \end{pmatrix} \\
&\quad + [\Gamma_i A^i S_1 (H_i^d)^t + \Gamma_i A^i S R^{-1} S^t \Gamma_i^t + \Gamma_i \Delta_i^d R_{11} (H_i^d)^t + \Gamma_i \Delta_i^d S_1^t \Gamma_i^t \\
&\quad + H_i^d R_{12}^t (H_i^d)^t + H_i^d S_2^t \Gamma_i^t - \Gamma_i A^i P^d \Gamma_i^t - \Gamma_i \Delta_i^s - \Gamma_i \Delta_i^d R_{11} (H_i^d)^t \\
&\quad - \Gamma_i A^i S_1 (H_i^d)^t - \Gamma_i \Delta_i^d S_1^t \Gamma_i^t - H_i^d R_{12}^t (H_i^d)^t - H_i^d S_2^t \Gamma_i^t] \psi_i^{-1} (H_i^d \begin{pmatrix} I & 0 \end{pmatrix} + \Gamma_i S R^{-1}) \\
&= \begin{pmatrix} \Gamma_i \Delta_i^d & H_i^d \end{pmatrix} + \Gamma_i A^i S R^{-1} + \Gamma_i (A^i (S R^{-1} S^t - P^d) \Gamma_i^t - \Delta_i^s) \psi_i^{-1} (H_i^d \begin{pmatrix} I & 0 \end{pmatrix} + \Gamma_i S R^{-1}) \\
&= \begin{pmatrix} \Gamma_i \Delta_i^d - \Gamma_i \chi_i \psi_i^{-1} H_i^d & H_i^d \end{pmatrix} + \Gamma_i A^i S R^{-1} - \Gamma_i \chi_i \psi_i^{-1} \Gamma_i S R^{-1} \\
&= \begin{pmatrix} \Gamma_i [\Delta_i^d - Q_i H_i^d] & H_i^d \end{pmatrix} + \Gamma_i [A^i - Q_i \Gamma_i] S R^{-1}
\end{aligned}$$

which is exactly the same as the first part of (33).
Once again, using (55) and (57), we find for L_i^3 :

$$\begin{aligned}
L_i^3 &= [-\Gamma_i A^i S_1 (H_i^d)^t - \Gamma_i \Delta_i^d R_{11} (H_i^d)^t - H_i^d R_{12}^t (H_i^d)^t - \Gamma_i A^i S R^{-1} S^t \Gamma_i^t \\
&\quad - \Gamma_i \Delta_i^d \begin{pmatrix} I & 0 \end{pmatrix} S^t \Gamma_i^t - H_i^d \begin{pmatrix} 0 & I \end{pmatrix} S^t \Gamma_i^t + \Gamma_i A^i P^d \Gamma_i^t + \Gamma_i \Delta_i^s \\
&\quad + \Gamma_i \Delta_i^d R_{11} (H_i^d)^t + \Gamma_i A^i S_1 (H_i^d)^t + \Gamma_i \Delta_i^d S_1^t \Gamma_i^t + H_i^d R_{12}^t (H_i^d)^t + H_i^d S_2^t \Gamma_i^t] \psi_i^{-1} \\
&= \Gamma_i (A^i (P^d - S R^{-1} S^t) \Gamma_i^t + \Delta_i^s) \psi_i^{-1} \\
&= \Gamma_i \chi_i \psi_i^{-1} \\
&= \Gamma_i Q_i
\end{aligned}$$

which is exactly the same as the second part of (33).

□

The proof of formula (34) is completely analogous.

B Notes on the special form of the Kalman filter

In this Appendix, we give two different forms of the recursive Kalman filter. The first one is the classical form that can be found in for instance Astrom & Wittenmark, 1984. This form is transformed into the form of (21)-(23), which is more useful for this paper.
Consider the system (1)-(2) :

$$\begin{aligned}
x_{k+1} &= A x_k + B u_k + w_k \\
y_k &= C x_k + D u_k + v_k
\end{aligned}$$

where, for the time being, we assume that A, B, C, D as well as

$$\mathbf{E}[\begin{pmatrix} w_k \\ v_k \end{pmatrix} \begin{pmatrix} w_l^t & v_l^t \end{pmatrix}] = \begin{pmatrix} Q^s & S^s \\ (S^s)^t & R^s \end{pmatrix} \delta_{kl}$$

are known.
Given $\hat{x}_0$, $\tilde{P}_0$ and $u_0, \ldots, u_{k-1}$, $y_0, \ldots, y_{k-1}$, the non-steady state Kalman filter state estimate $\hat{x}_k$ is then given by the following set of recursive formulas (Astrom & Wittenmark, 1984) :

$$\hat{x}_k = A\hat{x}_{k-1} + Bu_{k-1} + K_{k-1}(y_{k-1} - C\hat{x}_{k-1} - Du_{k-1}) \tag{58}$$

with :

$$\begin{aligned} K_{k-1} &= (A\tilde{P}_{k-1}C^t + S^s)(C\tilde{P}_{k-1}C^t + R^s)^{-1} && (59) \\ \tilde{P}_k &= A\tilde{P}_{k-1}A^t + Q^s - (A\tilde{P}_{k-1}C^t + S^s)(C\tilde{P}_{k-1}C^t + R^s)^{-1}(A\tilde{P}_{k-1}C^t + S^s)^t && (60) \end{aligned}$$

and $\tilde{P}_k$ the error covariance matrix :
For our purpose, a different form of these general Kalman filter equations is more useful. With P^s, G, Λ_0 the solution of the set of positive real equations :

$$\begin{aligned} P^s = AP^sA^t + Q^s &\quad\Rightarrow\quad P^s \\ G = AP^sC^t + S^s &\quad\Rightarrow\quad G \\ \Lambda_0 = CP^sC^t + R^s &\quad\Rightarrow\quad \Lambda_0 \end{aligned}$$

and the transformation :

$$\tilde{P}_k = P^s + P_k$$

we get (from (59)) :

$$\begin{aligned} K_{k-1} &= ((AP^sC^t + S^s) + AP_{k-1}C^t)((CP^sC^t + R^s) + CP_{k-1}C^t)^{-1} \\ &= (AP_{k-1}C^t + G)(\Lambda_0 + CP_{k-1}C^t)^{-1} \end{aligned}$$

For the Riccati equation (from (60)) we have :

$$\begin{aligned} P^s + P_k &= AP^sA^t + AP_{k-1}A^t + (P^s - AP^sA^t) \\ &\quad -((AP^sC^t + S^s) + AP_{k-1}C^t)((CP^sC^t + R^s) + CP_{k-1}C^t)^{-1}((AP^sC^t + S^s) + AP_{k-1}C^t)^t \\ P_k &= AP_{k-1}A^t - (AP_{k-1}C^t + G)(\Lambda_0 + CP_{k-1}C^t)^{-1}(AP_{k-1}C^t + G)^t \end{aligned}$$

This means that the Kalman filter (21)-(23) calculates the same state estimate $\hat{x}_k$ as the original Kalman filter (58)-(60) with $\tilde{P}_0 = P^s + P_0$.

C Proof of the Kalman filter linear combinations

In this Appendix we prove Theorem 3.

First we show that :

$$P_k = A^k P_0 (A^t)^k - \chi_k \psi_k^{-1} \chi_k^t \tag{61}$$

We prove (61) by induction :

Proof 1 :

<u>$k=1$</u> :

From the Kalman filter formula (23) , we find :

$$P_1 = AP_0A^t - (AP_0C^t + G)(CP_0C^t + \Lambda_0)^{-1}(AP_0C^t + G)^t$$

which is exactly the same as (61) for $k=1$.

<u>$k=i \;\Rightarrow\; k=i+1$</u>

We have to prove that

$$P_{i+1} = A^{i+1} P_0 (A^t)^{i+1} - \chi_{i+1} \psi_{i+1}^{-1} \chi_{i+1}^t \tag{62}$$

given that :

$$P_i = A^i P_0 (A^t)^i - \chi_i \psi_i^{-1} \chi_i^t$$

Hereto, we write $\chi_{i+1}\psi_{i+1}^{-1}\chi_{i+1}^t$ as a function of χ_i and ψ_i.

$$\begin{aligned} \chi_{i+1}\psi_{i+1}^{-1}\chi_{i+1}^t &= \left(\begin{array}{c|c} A\chi_i & G + A^{i+1}P_0(A^t)^iC^t \end{array} \right) \left(\begin{array}{c|c} \psi_i & \chi_i^tC^t \\ \hline C\chi_i & \Lambda_0 + CA^iP_0(A^t)^iC^t \end{array} \right)^{-1} \\ & \left(\begin{array}{c} \chi_i^tA^t \\ \hline (G + A^{i+1}P_0(A^t)^iC^t)^t \end{array} \right) \end{aligned} \tag{63}$$

To calculate the inverse of the middle block matrix in (63), we use formula (56). (To avoid confusion we use ψ_2 in stead of ψ).

$$\begin{aligned} \psi_2 &= \Lambda_0 + C(A^iP_0(A^t)^i - \chi_i\psi_i^{-1}\chi_i^t)C^t \\ &= \Lambda_0 + CP_iC^t \end{aligned}$$

Finally, if we put $\phi = A^{i+1}P_0(A^t)^iC^t + G$, we find for (63) :

$$\begin{aligned} \chi_{i+1}\psi_{i+1}^{-1}\chi_{i+1}^t &= \left(\begin{array}{c|c} A\chi_i & \phi \end{array} \right) \left(\begin{array}{c|c} \psi_i^{-1} + \psi_i^{-1}\chi_i^tC^t\psi_2^{-1}C\chi_i\psi_i^{-1} & -\psi_i^{-1}\chi_i^tC^t\psi_2^{-1} \\ \hline -\psi_2^{-1}C\chi_i\psi_i^{-1} & \psi_2^{-1} \end{array} \right) \left(\begin{array}{c} \chi_i^tA^t \\ \hline \phi^t \end{array} \right) \\ &= A\chi_i\psi_i^{-1}\chi_i^tA^t + A\chi_i\psi_i^{-1}\chi_i^tC^t\psi_2^{-1}C\chi_i\psi_i^{-1}\chi_i^tA^t \\ & \quad -A\chi_i\psi_i^{-1}\chi_i^tC^t\psi_2^{-1}\phi^t - \phi\psi_2^{-1}C\chi_i\psi_i^{-1}\chi_i^tA^t + \phi\psi_2^{-1}\phi^t \end{aligned}$$

So, from (62) :

$$\begin{aligned}
P_{i+1} &= A^{i+1}P_0(A^t)^{i+1} - \chi_{i+1}\psi_{i+1}^{-1}\chi_{i+1}^t \\
&= A[A^iP_0(A^t)^i - \chi_i\psi_i^{-1}\chi_i^t]A^t - (A\chi_i\psi_i^{-1}\chi_i^tC^t - \phi)\psi_2^{-1}(C\chi_i\psi_i^{-1}\chi_i^tA^t - \phi^t) \\
&= AP_iA^t - [A(\chi_i\psi_i^{-1}\chi_i^t - A^iP_0(A^t)^i)C^t - G]\psi_2^{-1}[A(\chi_i\psi_i^{-1}\chi_i^t - A^iP_0(A^t)^i)C^t - G]^t \\
&= AP_iA^t - (AP_iC^t + G)(\Lambda_0 + CP_iC^t)^{-1}(AP_iC^t + G)^t
\end{aligned}$$

Which is the equation of the Kalman filter (23). This implies that P_{i+1} as written in (62) will be the state covariance at time instant $i+1$.

End of Proof 1

□

The rest of the proof is also a proof by induction.
We prove (24) :

Proof 2 :

$\underline{k = 1}$

- The Kalman filter gain K_0 is given by (22) :

  $$K_0 = (AP_0C^t + G)(\Lambda_0 + CP_0C^t)^{-1}$$

  and the state by (21) :

  $$\begin{aligned}
  \hat{x}_1 &= A\hat{x}_0 + Bu_0 + K_0(y_0 - C\hat{x}_0 - Du_0) \\
  &= (A - K_0C)\hat{x}_0 + (B - K_0D)u_0 + K_0y_0
  \end{aligned}$$

- The linear combinations (24) are for $k = 1$:

  $$\left(\; A - \chi_1\psi_1^{-1}C \;\middle|\; B - \chi_1\psi_1^{-1}D \;\middle|\; \chi_1\psi_1^{-1} \; \right)$$

  But we also know that for $k = 1$ (27) and (26) :

  $$\chi_1\psi_1^{-1} = (AP_0C^t + G)(CP_0C^t + \Lambda_0)^{-1}$$

  So, these linear combinations give us the Kalman filter state estimate, for a Kalman filter with initial state covariance matrix P_0, and initial state estimate $\hat{x}_0$.

$\underline{k = i \;\Rightarrow\; k = i+1}$

We know that (24) is valid for $k = i$:

$$\hat{x}_i = (A^i - Q_i\Gamma_i)\hat{x}_0 + (\Delta_i^d - Q_iH_i^d)\begin{pmatrix} u_0 \\ \ldots \\ u_{i-1} \end{pmatrix} + Q_i\begin{pmatrix} y_0 \\ \ldots \\ y_{i-1} \end{pmatrix}$$

with $Q_i = \chi_i \psi_i^{-1}$.
So, from this we find the state estimate at time $i+1$ through the Kalman filter equation (21)

$$\hat{x}_{i+1} = \left(\begin{array}{c|c|c|c|c} (A-K_iC)(A^i - Q_i\Gamma_i) & (A-K_iC)(\Delta_i^d - Q_iH_i^d) & (B-K_iD) & (A-K_iC)Q_i & K_i \end{array} \right) \left(\begin{array}{c} \hat{x}_0 \\ \hline u_0 \\ \dots \\ u_{i-1} \\ \hline u_i \\ \hline y_0 \\ \dots \\ y_{i-1} \\ \hline y_i \end{array} \right) \tag{64}$$

Now we have to prove that the linear combinations in (64) are the same as those of (24) for $k = i+1$.

Proof 3 :

Hereto, we have to prove that :

$$Q_{i+1} = \left(\begin{array}{cc} (A-K_iC)Q_i & K_i \end{array} \right) \tag{65}$$

From the same formulas as in Proof 1, we find :

$$\begin{array}{rcl} \chi_{i+1}\psi_{i+1}^{-1} & = & \left(\begin{array}{c|c} A\chi_i\psi_i^{-1} + A\chi_i\psi_i^{-1}\chi_i^tC^t\psi_2^{-1}C\chi_i\psi_i^{-1} - \phi\psi_2^{-1}C\chi_i\psi_i^{-1} & -A\chi_i\psi_i^{-1}\chi_i^tC^t\psi_2^{-1} + \phi\psi_2^{-1} \end{array} \right) \\ & = & \left(\begin{array}{c|c} AQ_i - (\phi - AQ_i\chi_i^tC^t)\psi_2^{-1}CQ_i & (\phi - AQ_i\chi_i^tC^t)\psi_2^{-1} \end{array} \right) \\ & = & \left(\begin{array}{c|c} AQ_i - (AP_iC^t + G)(\Lambda_0 + CP_iC^t)^{-1}CQ_i & (AP_iC^t + G)(\Lambda_0 + CP_iC^t)^{-1} \end{array} \right) \\ & = & \left(\begin{array}{c|c} (A-K_iC)Q_i & K_i \end{array} \right) \end{array}$$

End of Proof 3

□

If we now look at (24) for $k = i+1$, we get :

$$\begin{array}{rl} & \left(A^{i+1} - (A-K_iC)Q_i\Gamma_i - K_iCA^i \;\middle|\; A\Delta_i^d - (A-K_iC)Q_iH_i^d - K_iC\Delta_i^d \;\middle|\right. \\ & \quad \left. B - K_iD \;\middle|\; (A-K_iC)Q_i \;\middle|\; K_i \right) \\ = & \left((A-K_iC)(A^i - Q_i\Gamma_i) \;\middle|\; (A-K_iC)(\Delta_i^d - Q_iH_i^d) \;\middle|\; B - K_iD \;\middle|\; (A-K_iC)Q_i \;\middle|\; K_i \right) \end{array}$$

These last linear combinations are exactly the same as (64). This concludes the proof of the Kalman filter states.

End of Proof 2

□

D Oblique Projections

In this section, we introduce oblique projections. It is shown that $\Gamma_i\tilde{X}_i$ and $\Gamma_{i-1}\tilde{X}_{i+1}$ can be interpreted as oblique projections. A geometrical interpretation will also be provided.

D.1 Oblique projections

Definition 1 Oblique Projection
Given two subspaces $\mathcal{B}$,$\mathcal{C}$ of a j-dimensional ambient space and two matrices $B \in \mathbb{R}^{q\times j}, C \in \mathbb{R}^{r\times j}$ the rows of which generate a basis for these spaces , then the oblique projection along $\mathcal{C}$ on $\mathcal{B}$ denoted with $\mathcal{P}_{\mathcal{B}|\mathcal{C}}$ is uniquely defined by (Baksalary & Kala, 1979) :

$$\begin{aligned} B\mathcal{P}_{\mathcal{B}|\mathcal{C}} &= B \\ C\mathcal{P}_{\mathcal{B}|\mathcal{C}} &= 0 \end{aligned}$$

The following Theorem is easy to derive from this definition :

Theorem 5 Oblique Projection
Given two matrices $B \in \mathbb{R}^{q\times j}$ with row space $\mathcal{B}$ and $C \in \mathbb{R}^{r\times j}$ with row space $\mathcal{C}$, then the oblique projection along $\mathcal{C}$ onto $\mathcal{B}$ can be written as :

$$\mathcal{P}_{\mathcal{B}|\mathcal{C}} = \begin{pmatrix} B^t & C^t \end{pmatrix} (\begin{pmatrix} BB^t & BC^t \\ CB^t & CC^t \end{pmatrix}^{-1})_{1|q} B$$

with $(.)^{-1}_{1|q}$ the first q columns of $(.)^{-1}$.

Proof :

$$\begin{aligned} B\mathcal{P}_{\mathcal{B}|\mathcal{C}} &= B \begin{pmatrix} B^t & C^t \end{pmatrix} \begin{pmatrix} BB^t & BC^t \\ CB^t & CC^t \end{pmatrix}^{-1}_{1|q} B \\ &= \begin{pmatrix} BB^t & BC^t \end{pmatrix} \begin{pmatrix} BB^t & BC^t \\ CB^t & CC^t \end{pmatrix}^{-1}_{1|q} B \\ &= \begin{pmatrix} I_q & 0 \end{pmatrix}_{1|q} B \\ &= B \end{aligned}$$

$$\begin{aligned} C\mathcal{P}_{\mathcal{B}|\mathcal{C}} &= C \begin{pmatrix} B^t & C^t \end{pmatrix} \begin{pmatrix} BB^t & BC^t \\ CB^t & CC^t \end{pmatrix}^{-1}_{1|q} B \\ &= \begin{pmatrix} CB^t & CC^t \end{pmatrix} \begin{pmatrix} BB^t & BC^t \\ CB^t & CC^t \end{pmatrix}^{-1}_{1|q} B \\ &= \begin{pmatrix} 0 & I_r \end{pmatrix}_{1|q} B \\ &= 0 \end{aligned}$$

□

Theorem 6
With $A \in \mathbb{R}^{p \times j}$ and

$$A \begin{pmatrix} B^t & C^t \end{pmatrix} \begin{pmatrix} BB^t & BC^t \\ CB^t & CC^t \end{pmatrix}^{-1} = \begin{pmatrix} M & N \end{pmatrix}$$

we have :

$$A\mathcal{P}_{\mathcal{B}|\mathcal{C}} = MB = A/\begin{pmatrix} B \\ C \end{pmatrix} - NC$$

The Theorem is easy to prove starting from Theorem 5.
As a last fact on oblique projections, we find from Baksalary & Kala (1979) that the oblique projection on $\mathcal{B}$ along $\mathcal{C}$, can also be written as a Λ orthogonal projection on $\mathcal{B}$:

$$\mathcal{P}_{\mathcal{B},\Lambda} = \Lambda B^t (B \Lambda B^t)^{-1} B$$

with

$$\Lambda = \mathcal{P}_{\mathcal{C}^\perp}$$

and $\mathcal{P}_{\mathcal{C}^\perp}$ denoting the orthogonal projection operator on the row space orthogonal to $\mathcal{C}$.

D.2 Interpretation of the states

From Theorem 6 and the definition of $\tilde{X}_i$ and $\tilde{X}_{i+1}$ (46)-(47), it can clearly be seen that $\Gamma_i \tilde{X}_i$ and $\Gamma_{i-1}\tilde{X}_{i+1}$ are oblique projections :

$$\begin{aligned}
\Gamma_i \tilde{X}_i &= Y_{i|2i-1} \mathcal{P}_{\begin{pmatrix} U_{0|i-1} \\ Y_{0|i-1} \end{pmatrix} | \mathrm{U}_{i|2i-1}} \\
&= \begin{pmatrix} L_i^1 \mid L_i^3 \end{pmatrix} \left(\frac{U_{0|i-1}}{Y_{0|i-1}} \right) \\
\Gamma_{i-1} \tilde{X}_{i+1} &= Y_{i+1|2i-1} \mathcal{P}_{\begin{pmatrix} U_{0|i} \\ Y_{0|i} \end{pmatrix} | \mathrm{U}_{i+1|2i-1}} \\
&= \begin{pmatrix} L_{i+1}^1 \mid L_{i+1}^3 \end{pmatrix} \left(\frac{U_{0|i}}{Y_{0|i}} \right)
\end{aligned}$$

Fig. 3 shows how the oblique projection of the row space of $Y_{i|2i-1}$ on the row spaces of $\begin{pmatrix} U_{0|i-1}^t & Y_{0|i-1}^t \end{pmatrix}^t$ along the row space of $U_{i|2i-1}$ can be interpreted as the following sequence of operations :

- an orthogonal projection to determine Z_i

- followed by a decomposition of Z_i into directions along the row spaces of $(U^t_{0|i-1} \; Y^t_{0|i-1})^t$ and $U_{i|2i-1}$. This gives respectively $\Gamma_i \tilde{X}_i$ and $L^2_i U_{i|2i-1}$.

The decomposition of Z_i into $\Gamma_i \hat{X}_i$ and $H^d_i U_{i|2i-1}$ is also indicated.
This is just a nice geometrical interpretation of the kalman filter states, and has no further consequences for the derivation of the identification algorithm.

E Additional Proof

In this Appendix, we prove that :

$$\mathcal{P}_i = \prod_{k=0}^{i-1} (A - K_k C)$$

Proof :
We have (with (65)) :

$$\begin{aligned}
\mathcal{P}_i &= A^i - Q_i \Gamma_i \\
&= A^i - (A - K_{i-1}C) Q_{i-1} \Gamma_{i-1} - K_{i-1} C A^{i-1} \\
&= A^i - A Q_{i-1} \Gamma_{i-1} + K_{i-1} C Q_{i-1} \Gamma_{i-1} - K_{i-1} C A^{i-1} \\
&= A(A^{i-1} - Q_{i-1}\Gamma_{i-1}) - K_{i-1} C (A^{i-1} - Q_{i-1} \Gamma_{i-1}) \\
&= (A - K_{i-1}C)(A^{i-1} - Q_{i-1}\Gamma_{i-1}) \\
&= (A - K_{i-1}C) \mathcal{P}_{i-1}
\end{aligned}$$

From this, we find easily :

$$\mathcal{P}_i = \prod_{k=0}^{i-1} (A - K_k C)$$

□

F Purely Deterministic Systems

In this appendix, we prove that for purely deterministic systems, the algorithms still work.
From the deterministic input-output equations (10)-(12) (Moonen *et al.* , 1989) we find :

$$Y_{i|2i-1} = \Gamma_i [\Delta^d_i - A^i \Gamma^\dagger_i H^d_i] U_{0|i-1} + H^d_i U_{i|2i-1} + \Gamma_i A^i \Gamma^\dagger_i Y_{0|i-1}$$

So, from this and (51) and the fact that the input is persistently exciting, we find :

$$Y_{i|2i-1} = \left(\begin{array}{c|c|c} T_1 & T_2 & T_3 \end{array} \right) \left(\begin{array}{c} U_{0|i-1} \\ \hline U_{i|2i-1} \\ \hline Y_{0|i-1} \end{array} \right)$$

with :

$$\begin{aligned} T_1 &= \Gamma_i[\Delta_i^d - A^i \Gamma_i^\dagger H_i^d] + T_1^h \\ T_2 &= H_i^d \\ T_3 &= \Gamma_i A^i \Gamma_i^\dagger + T_3^h \end{aligned}$$

and the homogeneous solution satisfying :

$$\left(\begin{array}{c|c|c} T_1^h & 0 & T_3^h \end{array} \right) \left(\begin{array}{c} U_{0|i-1} \\ \hline U_{i|2i-1} \\ \hline Y_{0|i-1} \end{array} \right) = 0$$

So, we find that :

$$\begin{aligned} Z_i &= Y_{i|2i-1} / \left(\begin{array}{c} U_{0|2i-1} \\ Y_{0|i-1} \end{array} \right) \\ &= \left(\begin{array}{c|c|c} T_1 & T_2 & T_3 \end{array} \right) \left(\begin{array}{c} U_{0|i-1} \\ \hline U_{i|2i-1} \\ \hline Y_{0|i-1} \end{array} \right) \\ &= \left(\begin{array}{c|c|c} \Gamma_i[\Delta_i^d - A^i \Gamma_i^\dagger H_i^d] & H_i^d & \Gamma_i A^i \Gamma_i^\dagger \end{array} \right) \left(\begin{array}{c} U_{0|i-1} \\ \hline Y_{0|i-1} \\ \hline Y_{i|2i-1} \end{array} \right) \\ &= Y_{i|2i-1} \end{aligned}$$

This implies that :

$$\begin{aligned} \Gamma_i \tilde{X}_i &= Z_i - L_i^2 U_{i|2i-1} \\ &= Y_{i|2i-1} - H_i^d U_{i|2i-1} \\ &= \Gamma_i X_i \\ &= \Gamma_i \hat{X}_i \end{aligned}$$

□

G Calculation of the bias

For the calculation of the bias $\mathcal{L}-\tilde{\mathcal{L}}$ the error on the states $\tilde{X}_i$ and $\tilde{X}_{i+1}$ has to be calculated. From (49)-(50), we have :

$$\delta\tilde{X}_i = \prod_{k=0}^{i-1}(A - K_k C)S(R^{-1})_{mi+1|2mi}U_{i|2i-1}$$

This quantity can only be calculated after the system is identified. In the following it is thus assumed that the system matrices $A, B, C, D, Q^s, S^s, R^s$ are known. (R and Q are the submatrices of the RQ decomposition as defined in section 7).

- SR^{-1} can be calculated as :

$$\begin{aligned} SR^{-1} &= X_0^d U_{0|2i-1}^t (U_{0|2i-1}U_{0|2i-1}^t)^{-1} \\ &= \Gamma_i^\dagger [Y_{0|i-1}^d - H_i^d U_{0|i-1}] U_{0|2i-1}^t (U_{0|2i-1}U_{0|2i-1}^t)^{-1} \\ &= \Gamma_i^\dagger [Y_{0|i-1} U_{0|2i-1}^t - H_i^d U_{0|i-1} U_{0|2i-1}^t] (U_{0|2i-1}U_{0|2i-1}^t)^{-1} \\ &= \Gamma_i^\dagger [R_{4:4,1:3} - H_i^d R_{1:1,1:3}] R_{1:3,1:3}^{-1} \end{aligned}$$

- For the calculation of the Kalman filter gains K_k, we also need the initial covariance estimate :

$$\tilde{P}_0 = P^d + P^s - SR^{-1}S^t$$

  - We have approximately :

$$\lim_{j\to\infty} \frac{1}{j}\tilde{X}_i\tilde{X}_i^t \simeq P^d + P^s$$

    And from the description of the algorithm (see section 7) we have :

$$\frac{\tilde{X}_i\tilde{X}_i^t}{j} = \Sigma_1$$

    Thus we have :

$$P^d + P^s \simeq \Sigma_1$$

  - $SR^{-1}S^t$ can be easily found as :

$$\begin{aligned} SR^{-1}S^t &= SR^{-1}(R_{1:3,1:3}R_{1:3,1:3}^t)(SR^{-1})^t \\ &= \Gamma_i^\dagger [R_{4:4,1:3} - H_i^d R_{1:1,1:3}][R_{4:4,1:3}^t - R_{1:1,1:3}^t (H_i^d)^t](\Gamma_i^\dagger)^t \end{aligned}$$

algorithm	1	2	3	4	5
order	5	5	3/3	4/4	5
i	5	5	5	-	-
flops	14355148	13590364	24056676	41744504	330529255
pred. determ.	47.41	47.95	48.56	45.72	47.34
pred. Kalman	10.76	10.76	14.17	28.98	10.87

Table 1: Comparison of 5 identification algorithms.

Care should be taken when subtracting both quantities to obtain $\tilde{P}_0$. It is possible that due to the approximations, $\tilde{P}_0$ becomes negative definite. If this is the case, it should be put equal to zero.

- Now the Kalman gains K_k can be calculated using formulas (22)-(23).

The errors $\delta\tilde{X}_i$ and $\delta\tilde{X}_{i+1}$ can be easily calculated as :

$$\begin{aligned}
\delta\tilde{X}_i &\simeq \prod_{k=0}^{i-1}(A - K_k C) S (R^{-1})_{mi+1|2mi} R_{2:3,1:3} Q_{1:3} \\
\delta\tilde{X}_{i+1} &\simeq \prod_{k=0}^{i}(A - K_k C) S (R^{-1})_{m(i+1)+1|2mi} R_{3:3,1:3} Q_{1:3}
\end{aligned}$$

The least squares solution of :

$$\min_{\check{\mathcal{L}}} \| \begin{pmatrix} \tilde{X}_{i+1} + \delta\tilde{X}_{i+1} \\ Y_{i|i} \end{pmatrix} - \check{\mathcal{L}} \begin{pmatrix} \tilde{X}_i + \delta\tilde{X}_i \\ U_{i|i} \end{pmatrix} \|_F^2$$

will thus be a better approximation for $\mathcal{L}$ as $\tilde{\mathcal{L}}$ is. The bias can then be calculated as :

$$\mathcal{L} - \tilde{\mathcal{L}} \simeq \check{\mathcal{L}} - \tilde{\mathcal{L}}$$

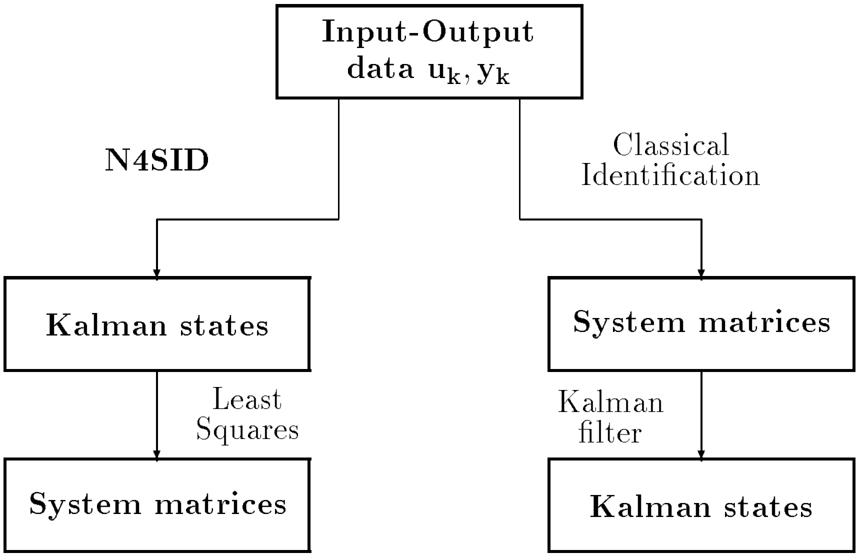

Figure 1: The left hand side shows the **N4SID** approach : first the (Kalman) states, then the system matrices. The right hand side is the classical approach : first the system, and then an estimate of the states.

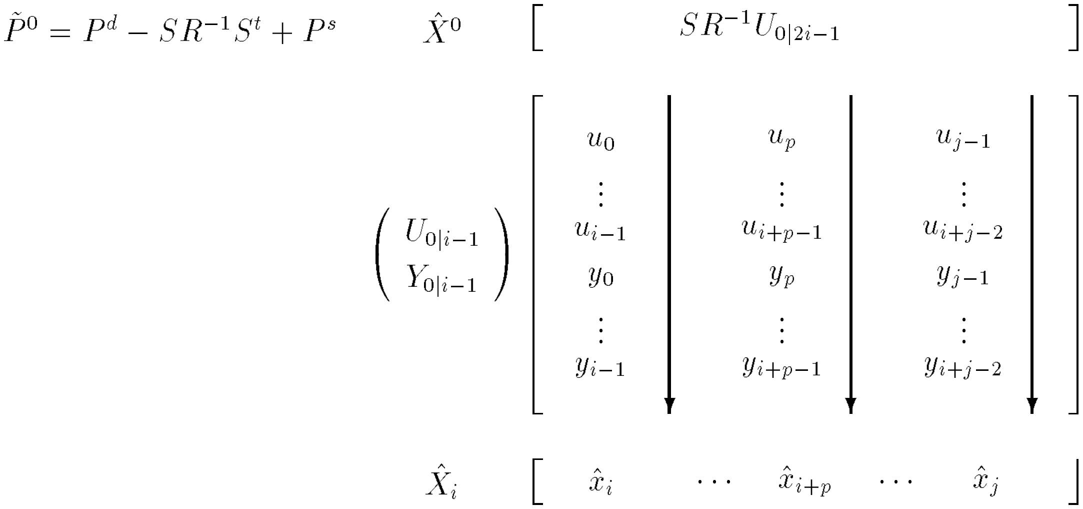

Figure 2: Interpretation of the sequence $\hat{X}_i$ as a sequence of non-steady state Kalman filter state estimates based upon i measurements of u_k and y_k.

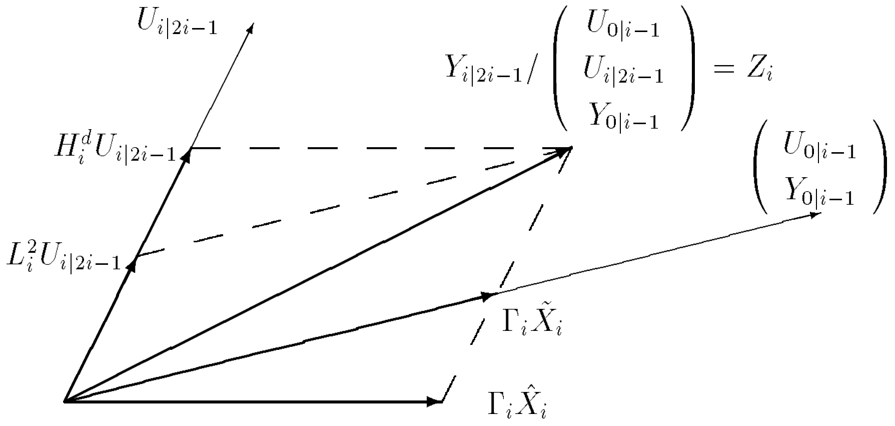
$U_{i|2i-1}$
$Y_{i|2i-1}/\begin{pmatrix} U_{0|i-1} \\ U_{i|2i-1} \\ Y_{0|i-1} \end{pmatrix} = Z_i$
$H_i^d U_{i|2i-1}$
$\begin{pmatrix} U_{0|i-1} \\ Y_{0|i-1} \end{pmatrix}$
$L_i^2 U_{i|2i-1}$
$\Gamma_i \tilde{X}_i$
$\Gamma_i \hat{X}_i$

Figure 3: Interpretation of $\Gamma_i \tilde{X}_i$ as an oblique projection.

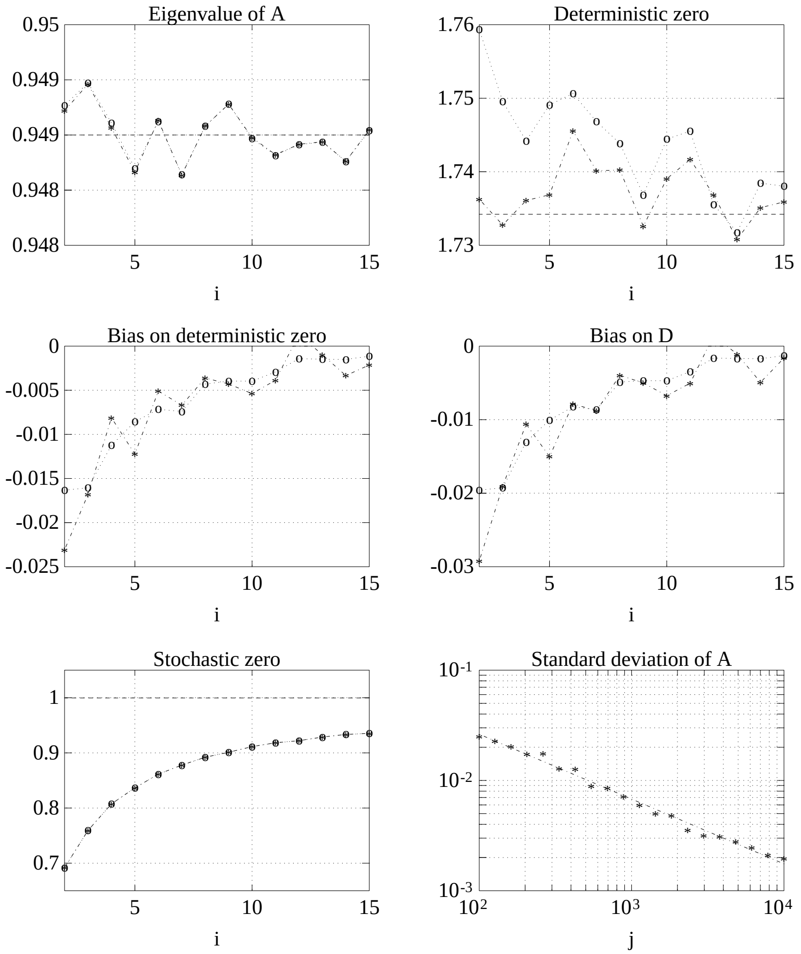

Figure 4: a/ Eigenvalue of A as a function of i (**N4SID** algorithm 1 (*) and 2 (o)). b/ Deterministic zero as a function of i (**N4SID** algorithm 1 (*) and 2 (o)). c/ Calculated (*) and estimated (o) bias on the deterministic zero as a function of i (**N4SID** algorithm 2). d/ Calculated (*) and estimated (o) bias on D as a function of i (**N4SID** algorithm 2). e/ Stochastic zero as a function of i (**N4SID** algorithm 1 (*) and 2 (o)). f/ Standard deviation of A as a function of j (**N4SID** algorithm 1).

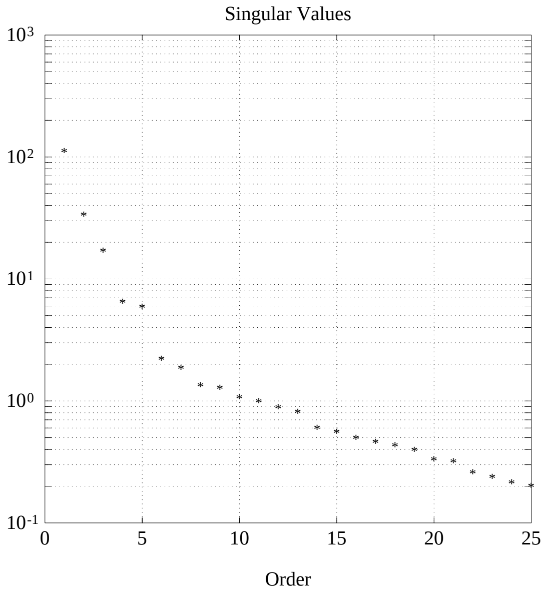

Figure 5: Order decision based on the dominant singular values.