Czech Technical University in Prague
Faculty of Electrical Engineering
Department of Control Engineering

Subspace Identification Methods

Technical Report

Ing. Pavel Trnka
Supervisor: Prof. Ing. Vladimír Havlena, CSc.

Prague, September 2005

Acknowledgement

First, I would like to thank to my supervisor Vladimír Havlena for his very good guidance, his support and for sharing his ideas with me.

I would like to thank to my friends for letting me spend pleasant time with them far away from the school and science, giving me new energy for my further work. And finally I have to thank to my parents for their continuous support in anything I do.

Abstract

This technical report shows the recent advances in Subspace Identification Methods (4SID) and describes the reasons making these methods attractive to the industry. The 4SID methods are used for the identification of the linear time-invariant models in a state space form from the input/output data. On the contrary to the other methods, they are based on the different principles, like the geometrical projections and numerical linear algebra. Therefore the stress is laid on the interpretations of these methods in the well known frameworks of Kalman filter, Prediction error minimization and Instrumental variable methods. Recently proposed 4SID algorithms, improvements and open problems are discussed and finally a new available research direction in subspace identification is presented. The new idea would allow to incorporate prior information into 4SID while recursifying the algorithm.

Contents

Chapter 1

Introduction

> *"The results of realization theory and the application of subspace identification algorithms form an example of what has been called the* 'unreasonable effectiveness' *of mathematics."*
>
> Jan H. van Schuppen, Vrije University, Amsterdam

Our research deals with relatively new methods in linear system identification, which are generally entitled Subspace Identification Methods or more accurately 4SID methods (Subspace State Space Systems IDentification). They are used for identification of LTI state space models directly from the input/output data. 4SID methods are an alternative to the regression methods like ARX or ARMAX. However, they are based on a very different approach of *the geometric projections* and *linear algebra*.

The basic ideas of these methods were developed about 20 years ago and they are quite well accepted in the control engineering community, however the applications of these methods in the systems identification are still rather exceptional, which is mainly due to their complex theoretical backgrounds, until recently problematic recursification and the problems with the closed-loop identification.

Up-to-date algorithms of 4SID methods are competing Prediction Error Methods (PEM) in their performance and have shown several good properties, which make them favorable candidate for the industrial applications.

1.1 Properties of 4SID methods

The following list shows the most important positive and negative properties of 4SID methods.

Positive properties

MIMO systems identification. The complexity of the identification for large MIMO (Multiple Inputs Multiple Outputs) systems is the same as for SISO

(Single Input Single Output) systems. There is no need for an extensive MIMO structure parametrization, which is a very appealing property.

Numerical robustness. 4SID methods can be implemented with QR and SVD factorizations, which have well known properties and very good numerical robustness.

Few user parameters. In fact there is only one parameter and that is the system order. There is no need for the complex parametrization even for MIMO systems, because 4SID methods are identifying a state space model. Therefore 4SID methods are suitable for automatic multi-variable system identification.

Model order reduction. The algorithms of 4SID incorporates implicit model order reduction, which is useful especially for the real-world data, where noises and disturbances play important role and increase the order of estimated model.

Negative properties

Need a large set of input/output data. The statistical properties of geometrical methods used in 4SID are the reason for the fact, that they need large amount of input/output data samples. That limits the application in some areas, where data are rather rare, such as economic modelling.

Theoretically complex background. The algorithms are built on the geometrical projections in the high dimensional row or column spaces of certain matrices. This makes them uneasy to understand, however, as will be shown (Chapter 4), they can be interpreted in the other well known frameworks.

Difficult recursification. The basic algorithms were developed to identify the system parameters from off-line data, i.e. identification from given complete sequence of input/output data. However, the industrial applications need the recursive algorithm to identify in the real-time from on-line data. This is still rather problem for 4SID methods, but recent articles have shown promising results.

Prior knowledge can not be easily incorporated into 4SID methods. These methods have black-box approach to the identified system, however there is often a priori information, which should be exploited to increase the quality and the robustness of the identification.

Recent articles showed, that subspace methods can be modified to perform well even in the closed-loop identification [11, 12] and the new effective algorithms for an on-line recursive identification [9, 10] were also proposed.

The usual description of a deterministic identification, which is rather academical, will be skipped in this work and the stochastic systems will be treated right from the beginning.

This work is organized as follows. First an overview of the geometric tools, used in 4SID, is presented in chapter 2. Chapter 3 shows the principles of 4SID methods and describes the mostly used algorithms and their properties. Chapter 4 interprets 4SID methods in the well known frameworks of least-squares, Kalman filter and instrumental variable methods. Chapter 5 summarizes the recent advances in the field of 4SID, followed by chapter 6 proposing new research direction for our further work, dealing with the prior information incorporation and 4SID recursification.

Several chapters are supplied with a section 'Notes', usually giving the less formal comments about the principles and the inner nature of the described methods.

Chapter 2

Geometric Tools

This chapter shows an overview of main geometric tools used in 4SID methods and some necessary algebraic background. The main tools are the orthogonal and the oblique projections.

2.1 Matrices with Hankel structure

The matrices with Hankel structure play important role in 4SID methods, because the signals (the input/output data and the noises) appear in the algorithms in the form of Hankel matrices. The first step of each algorithm is therefore to arrange the available data into Hankel matrices.

A square or non-square matrix $A \in \mathbb{R}^{m\times n}$ with Hankel structure is a matrix with constant skew diagonals (anti-diagonals). In other words, in Hankel matrix, the value of (i,j)th entry depends only on the sum $i+j$. The matrix A with Hankel structure can be created from a vector $\mathbf{a} = \begin{pmatrix} a_1 & \dots & a_{m+n-1} \end{pmatrix}$ with $m+n-1$ elements

$$A = \begin{pmatrix} a_1 & a_2 & a_3 & \cdots & a_n \\ a_2 & a_3 & a_4 & \cdots & a_{n+1} \\ a_3 & a_4 & a_5 & \cdots & a_{n+2} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ a_m & a_{m+1} & a_{m+2} & \cdots & a_{m+n-1} \end{pmatrix}.$$

Each entry of Hankel matrix can be also a matrix. This composition is than called Block Hankel matrix.

The matrices with Hankel structure are usually denoted shortly as Hankel matrices, as it will be also used in this work. However, that can be confusing, because Hankel matrix is also a special square matrix $H \in \mathbb{R}^{n\times n}$ defined as

$$h_{i,j} = \begin{cases} 0 & \text{if } i+j-1 > n \\ i+j-1 & \text{otherwise} \end{cases}$$

2.2 Matrix Row/Column Space

The row space of an $m \times n$ matrix A, denoted by $row(A)$ is the set of all linear combinations of the row vectors of A.

Similarly *the column space* of an $m \times n$ matrix A, denoted by $col(A)$ is the set of all linear combinations of the column vectors of A.

2.3 Projections

In this section the definitions of the orthogonal and the oblique projection will be shown. General matrices $A \in \mathrm{R}^{p\times j}$, $B \in \mathrm{R}^{q\times j}$ and $C \in \mathrm{R}^{r\times j}$ will be considered.

Orthogonal Projection

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

$$A/B = AB^T \left(BB^T\right)^\dagger B = A\Pi_B, \tag{2.1}$$

where $\bullet^\dagger$ is Moore-Penrose pseudo-inverse [14] of the matrix $\bullet$. Similarly the projection of the row space A on the orthogonal space to the row space of B is defined as

$$A/B^\perp = A\left(I - \Pi_B\right) = A\left(I - B^T\left(BB^T\right)^\dagger B\right) = A\Pi_B^\perp, \tag{2.2}$$

The projections A/B and $A/B^\perp$ decompose the matrix A into two matrices, whose row spaces are mutually orthogonal (Figure 2.1)

$$A = A/B + A/B^\perp.$$

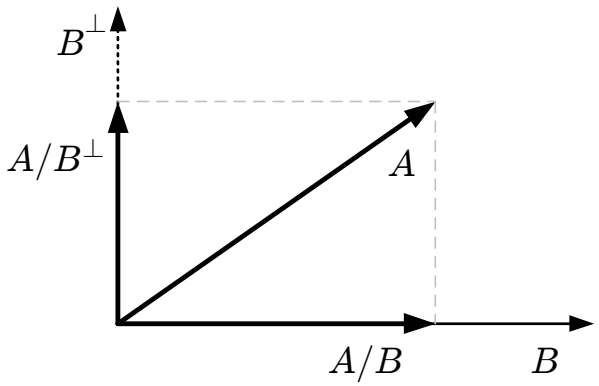

Figure 2.1: Orthogonal projection.

A numerically efficient and robust computation of the orthogonal projection can be done by LQ decomposition[1]

$$\begin{pmatrix} B \\ A \end{pmatrix} = LQ = \begin{pmatrix} L_{11} & 0 \\ L_{21} & L_{22} \end{pmatrix} \begin{pmatrix} Q_1 \\ Q_2 \end{pmatrix}.$$

Then, the the orthogonal projections can be written as

$$A/B = L_{21}Q_1, \tag{2.3}$$
$$A/B^{\perp} = L_{22}Q_2. \tag{2.4}$$

Oblique Projection

The oblique projection of the row space A along the row space of B on the row space of C is defined as

$$A \underset{B}{/} C = A \begin{pmatrix} C^T & B^T \end{pmatrix} \left[\left(\begin{pmatrix} C \\ B \end{pmatrix} \begin{pmatrix} C^T & B^T \end{pmatrix} \right)^{\dagger} \right]_{\substack{\text{first r} \\ \text{columns}}} C. \tag{2.5}$$

This result can be derived from an orthogonal projection of the row space A to the row space of $\begin{pmatrix} B \\ C \end{pmatrix}$. A more effective way is to use LQ decomposition again

$$\begin{pmatrix} B \\ C \\ A \end{pmatrix} = \begin{pmatrix} L_{11} & 0 & 0 \\ L_{21} & L_{22} & 0 \\ L_{31} & L_{32} & L_{33} \end{pmatrix} \begin{pmatrix} Q_1 \\ Q_2 \\ Q_3 \end{pmatrix}.$$

Then, the the orthogonal projections can be written as

$$A \underset{B}{/} C = L_{32}L_{22}^{-1}C = L_{32}L_{22}^{-1} \begin{pmatrix} L_{21} & L_{22} \end{pmatrix} \begin{pmatrix} Q_1 \\ Q_2 \end{pmatrix}. \tag{2.6}$$

The oblique projection decomposes the matrix A into three matrices (Figure 2.2)

$$A = A \underset{B}{/} C + A \underset{C}{/} B + A/\begin{pmatrix} B \\ C \end{pmatrix}.$$

Statistical Properties of Projections

Assume two general sequences $u_k \in \mathbb{R}^{n_u}$ and $e_k \in \mathbb{R}^{n_e}$, $k = 1, 2, \ldots, j$, where e_k is zero mean and *independent* of y_k

$$\mathbf{E}(e_k) = 0,$$
$$\mathbf{E}\left(u_k e_k^T\right) = 0. \tag{2.7}$$

[1]The LQ factorization of A is essentially the same as the QR factorization of A^T. That is $A = (A^T)^T = (QR)^T = LQ^T$

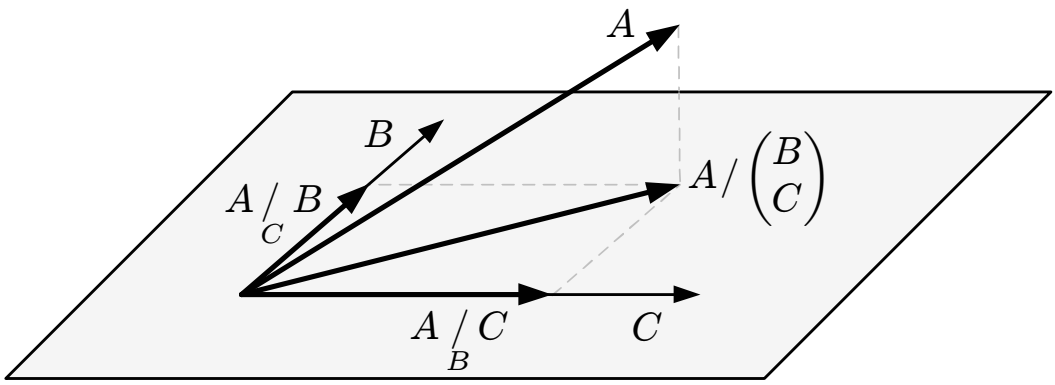

Figure 2.2: Oblique projection.

For the long series of data ($j \to \infty$) usual in 4SID and assuming ergodicity, the expectation operator $\mathbf{E}$ can be replaced by an average over one, infinitely long experiment

$$\mathbf{E}\left(u_k e_k^T\right) = \lim_{j\to\infty} \frac{1}{j} \sum_{i=1}^{j} u_i e_i^T. \tag{2.8}$$

Putting the sequences into the row matrices

$$\begin{aligned} u &= \begin{pmatrix} u_1 & u_2 & \dots & u_j \end{pmatrix}, \\ e &= \begin{pmatrix} e_1 & e_2 & \dots & e_j \end{pmatrix}, \end{aligned}$$

the sum in (2.8) can be rewritten to

$$\mathbf{E}\left(u_k e_k^T\right) = \lim_{j\to\infty} \frac{1}{j} u e^T.$$

Considering independency (2.7) we find that

$$u e^T = 0,$$

which implies that the signal u is perpendicular to the signal e.

Assume that y is a vector of the inputs and e is a vector of the additive disturbances. In the geometrical sense and for $j \to \infty$, the row vectors of the disturbances are perpendicular to the row vectors of the inputs. Using an orthogonal projection of the disturbance on the input, the noise is asymptotically eliminated

$$e/u = 0, \qquad (\text{for } \; j \to \infty).$$

This property is used in 4SID *to eliminate the noise influence* and is among basic concepts of 4SID methods.

2.4 Notes

The following notes may become more clear after reading the next chapter.

- An effect of the projections can be simplified into an ability to separate the subspaces, which were joined together by a linear combination. The projections are able to eliminate the independent stochastic inputs, and to eliminate the influence of measurable deterministic inputs.
- In the next chapter the deterministic inputs, disturbances and response from the states will be shown to generate certain high-dimensional subspaces (spanned by the rows of appropriate Hankel matrices). The output of LTI system will be shown to generate similar subspace as *a linear combination of these subspaces*. The power of projections is in their *ability to separate* this linear combination from the knowledge of the inputs and outputs.
- *The orthogonal projection* is used to asymptotically eliminate the influence of the disturbances from the joint space. This can be viewed as a separation of an effective signal from the noise.
- The more capable *oblique projection* is also able to simultaneously eliminate the influence of the disturbances and moreover to eliminate the influence of the deterministic measurable signal (the input). This is mainly used in 4SID methods to obtain the subspace generated only by the state sequence, which is later used to obtain the model parameters.

Chapter 3

Subspace Identification Methods

Subspace identification methods are used to identify the parameters (matrices) of LTI state space model from the input/output data. In this chapter the basics of 4SID methods will be given and three main algorithms (N4SID, MEOSP and CVA) will be described.

The name 'subspace' denotes the fact, that the model parameters of identified linear system are obtained from the row or the column subspace of a certain matrix, which is formed from the input/output data. Typically the column space is used to extract information about the model and the row space is used to get Kalman filter states sequence for the model.

3.1 State Space Model

In this work we will consider a state space model of combined deterministic-stochastic system (Figure 3.1) in an innovation form [7]

$$x_{k+1} = Ax_k + Bu_k + Ke_k, \tag{3.1}$$
$$y_k = Cx_k + Du_k + e_k, \tag{3.2}$$

where $u_k \in \mathbb{R}^m$ is the m-dimensional input, $x_k \in \mathbb{R}^n$ is the n-dimensional state, $y_k \in \mathbb{R}^l$ is the l-dimensional output, K is the steady state Kalman gain and $e_k \in \mathbb{R}^l$ is an unknown innovation with covariance matrix $E\left[e_k e_k^T\right] = R_e$. This model has close relation with widely used stochastic state space model

$$x_{k+1} = Ax_k + Bu_k + v_k, \tag{3.3}$$
$$y_k = Cx_k + Du_k + w_k, \tag{3.4}$$

where $v_k \in \mathbb{R}^m$ and $w_k \in \mathbb{R}^l$ are the process and the measurement noise with covariance matrices $E\left[v_k v_k^T\right] = Q$, $E\left[w_k w_k^T\right] = R$ and $E\left[v_k w_k^T\right] = S$. The process noise represents the disturbances entering the system and the measurement noise represent the uncertainty in the system observations.

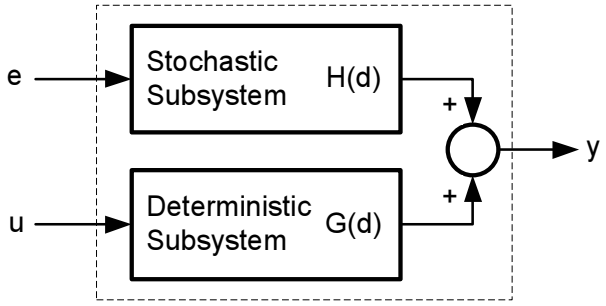

Figure 3.1: Combined deterministic-stochastic system.

Now there is a question: "Why to use an innovation form?". The model (3.3–3.4) has good physical interpretation, however the innovation model is more suitable for 4SID methods, because it has a noise model with less degrees freedom, which can be more appropriately identified from the given input/output data. Both models can be shown to be equivalent from the input/output point of view up to the second order statistics (means and covariances).

From algebraic Riccati equation [13] the Kalman gain K and the innovations covariance R_e can be computed so that the stochastic properties of the output are same for both models. The output spectral density of stochastic state space model (3.1–3.2)

$$S_{yy} = C(zI - A)^{-1}Q(z^{-1}I - A)^{-T}C^T + R$$

and the output spectral density of innovation form (3.1), (3.2)

$$S'_{yy} = [C(zI - A)^{-1}K + I)]R_e[C(z^{-1}I - A)^{-1}K + I]^T$$

are equal for

$$\begin{aligned} K &= APC^T(CPC^T + R)^{-1}, \\ R_e &= CPC^T + R, \end{aligned}$$

which is the solution of algebraic Riccati equation for steady state Kalman filter and where

$$P = APA^T - K(C^TPC + R)K^T + Q.$$

As a consequence the uncertainty contained in the noises v_k and w_k ($m + l$ dimensions) can be for the outputs and the states described by the innovations e_k (only l dimensions).

3.2 Notations and Definitions

This section introduces further used notation. This makes it easy to find, but difficult to understand before the rest is read.

One of the principaly new ideas in 4SID methods, is to combine the recursive state space model into *single linear matrix equation, relating the signal matrices with the parameters matrices* (Section 3.5). Prior to do this some definitions are necessary.

Signal Related Matrices

For the use in 4SID algorithms, all signals (inputs, outputs and noises) are arranged into the Hankel matrices. Assume known set of input/output data samples u_k,y_k for $k \in \langle 0, 1, \ldots, i+h+j-2 \rangle$. These samples can be arranged into Hankel matrices with i and h block rows and j columns as follows

$$\left(\frac{U_p}{U_f}\right) = \left(\begin{array}{cccc} u_0 & u_1 & \ldots & u_{j-1} \\ u_1 & u_2 & \ldots & u_j \\ \vdots & \vdots & \ddots & \vdots \\ u_{i-1} & u_i & \ldots & u_{i+j-2} \\ \hline u_i & u_{i+1} & \ldots & u_{i+j-1} \\ u_{i+1} & u_{i+2} & \ldots & u_{i+j} \\ \vdots & \vdots & \ddots & \vdots \\ u_{i+h-1} & u_{i+h} & \ldots & u_{i+h+j-2} \end{array}\right) = \left(\frac{U_p^+}{U_f^-}\right) = \left(\begin{array}{cccc} u_0 & u_1 & \ldots & u_{j-1} \\ u_1 & u_2 & \ldots & u_j \\ \vdots & \vdots & \ddots & \vdots \\ u_{i-1} & u_i & \ldots & u_{i+j-2} \\ u_i & u_{i+1} & \ldots & u_{i+j-1} \\ \hline u_{i+1} & u_{i+2} & \ldots & u_{i+j} \\ \vdots & \vdots & \ddots & \vdots \\ u_{i+h-1} & u_{i+h} & \ldots & u_{i+h+j-2} \end{array}\right),$$

where U_p is the matrix of past inputs and U_f is the matrix of future inputs. Although most data samples can be found in both matrices, the notation past/feature is appropriate, because corresponding columns of U_p and U_f are subsequent without any common data samples and therefore have the meaning of the past and the future. The distinction between the past and the future is important for Kalman filter and Instrumental variables concepts used in 4SID.

Notice that the entries of Hankel matrices can be the vectors $u_k \in \mathbb{R}^m$, therefore they are called block Hankel matrices with the dimensions $U_p \in \mathbb{R}^{im \times j}$, $U_f \in \mathbb{R}^{hm \times j}$, $U_p^+ \in \mathbb{R}^{(i+1)m \times j}$, $U_f^- \in \mathbb{R}^{(h-1)m \times j}$. The parameters i and h allow for the different number of block rows for past U_p and U_f future. This is different to some sources, where both parameters are assumed equal.

The values of the coefficients i and h are usually selected slightly larger then the upper bound of expected system order and the coefficient j is approximately equal to the number of measured data at disposal ($j \gg i$, $j \gg h$). From i and h to j ratio it is obvious that Hankel matrices U_p and U_f have a structure with long rows.

Matrices U_p^+ and U_f^- are created from U_p and U_f by moving the first block row from U_f to the end of U_p. This variation is later used to retrieve the system matrices.

For the outputs y_k and the noises e_k similar Hankel matrices Y_p, Y_f and E_p, E_f can be constructed. A combination of U_p and Y_p denoted as W_p is used as a regressor

$$W_p = \left(\begin{array}{c} U_p \\ Y_p \end{array}\right)$$

System state sequence is also used in a matrix form with the following structure

$$X_p = (x_0\ x_1\ \dots\ x_{j-1}), \quad X_f = (x_i\ x_{i+1}\ \dots\ x_{i+j-1}).$$

Parameters Related Matrices

The extended observability matrix Γ_k is an extension of observability matrix for a number of block rows higher then the system order $k \geq n$

$$\Gamma_k = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{k-1} \end{pmatrix} \in \mathbb{R}^{kl \times n}.$$

Similarly *the reversed extended controllability matrices* Δ_i^d and Δ_i^s corresponding to the deterministic and stochastic parts respectively are defined as

$$\begin{aligned} \Delta_k^d &= \begin{pmatrix} A^{k-1}B & A^{k-2}B & \dots & B \end{pmatrix} \in \mathbb{R}^{n \times km}, \\ \Delta_k^s &= \begin{pmatrix} A^{k-1}K & A^{k-2}K & \dots & K \end{pmatrix} \in \mathbb{R}^{n \times km}, \end{aligned}$$

where K is the stationary Kalman gain.

The last two matrices H_k^d and H_k^s are Toeplitz matrices composed from the impulse responses (Markov parameters) of deterministic and stochastic subsystems respectively

$$H_k^d = \begin{pmatrix} D & 0 & 0 & \dots & 0 \\ CB & D & 0 & \dots & 0 \\ CAB & CB & D & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{k-2}B & CA^{k-3}B & CA^{k-4}B & \dots & D \end{pmatrix} \in \mathbb{R}^{kl \times km}, \tag{3.5}$$

$$H_k^s = \begin{pmatrix} I & 0 & 0 & \dots & 0 \\ CK & I & 0 & \dots & 0 \\ CAK & CK & I & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{k-2}K & CA^{k-3}K & CA^{k-4}K & \dots & I \end{pmatrix} \in \mathbb{R}^{kl \times kl}.$$

3.3 Problem formulation

The basic solved problem can be simply stated as follows

Given
s samples of the input sequence $\{u(0), \ldots, u(s-1)\}$ and the output sequence $\{y(0), \ldots, y(s-1)\}$

Estimate
the parameters of the combined deterministic-stochastic model in the innovation form (3.1–3.2). It means to estimate the system order n and to obtain system matrices A, B, C, D, K and covariance matrix R_e of the noise e_k.

3.4 Basic Idea

The very basic and important idea in 4SID is that to identify the parameters of a state space model (3.1–3.2), it is sufficient to known either

column space of Extended observability matrix

$$\Gamma_h = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{h-1} \end{pmatrix}$$

or

row space of State sequence matrix

$$X_f = \begin{pmatrix} | & | & & | \\ \overline{x}_i & \overline{x}_{i+1} & \ldots & \overline{x}_{i+j-1} \\ | & | & & | \end{pmatrix}.$$

An important fact is that we do not need exact numerical values, instead it is sufficient to know only the respective subspaces spanned by these matrices. That is because the state space model is not unique due to arbitrary state space basis selection, but the subspaces spanned by the columns of Γ_h and by the rows of X_f are invariant. Any basis of these subspaces formed into matrix is then valid Extended observability matrix or State sequence matrix (basis selection must be jointed).

Subspaces sufficiency

The subspaces are sufficient, because for example the simplest way to get the model parameters from the state sequence matrix row space is to take its any basis which will be a valid state sequence. Then the inputs, the outputs and the states are known and they are related by the state space model, which can be written into a set of linear equation

$$\begin{pmatrix} x_{i+1} & x_{i+2} & \ldots & x_{i+j-1} \\ y_i & y_{i+1} & \ldots & y_{i+j-2} \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} x_i & x_{i+1} & \ldots & x_{i+j-2} \\ u_i & u_{i+1} & \ldots & u_{i+j-2} \end{pmatrix}$$

and solved by the least squares (LS) or total least squares (TLS) to get the model parameters A, B, C, D.

3.5 Single equation formulation of SS model

As already mentioned before, the starting point of 4SID methods is a combination of the recursive state space innovation model (3.1–3.2) into one single linear matrix equation. This can be done with Hankel matrices by recursively substituting (3.1) into (3.2)

$$\begin{aligned} Y_p &= \Gamma_i X_p + H_i^d U_p + H_i^s E_p, && (3.6)\\ Y_f &= \Gamma_h X_f + H_h^d U_f + H_h^s E_f, && (3.7)\\ X_f &= A^i X_p + \Delta_i^d U_p + \Delta_i^s E_p. && (3.8) \end{aligned}$$

The equations (3.6) and (3.7) are similarly defining outputs as a linear combination of previous states by the extended observability matrix $\Gamma_\bullet$ (response from the states) and a linear combination of previous inputs and noises by their respective impulse responses $H_\bullet^d$ and $H_\bullet^s$. The equation (3.8) is relating the future and the past states under the influence of the inputs and the noises.

3.6 Estimating subspaces

Assuming the sufficiency of the subspaces for SS model identification, now the problem of estimating invariant subspaces of the extended observability matrix Γ_h and the state sequence matrix X_f from the input/output data will be treated. This estimation is also related to the determination of the system order.

The important observation is that to obtain these subspaces, only the term $\Gamma_h X_f$ is needed and its estimate can be obtained from data by the projections. This term is usually denoted as a matrix $\mathcal{O}_h$

$$\mathcal{O}_h = \Gamma_h X_f.$$

and it can be split into the required subspaces by singular value decomposition (SVD).

Description of the matrix $\mathcal{O}_h$ content is usually avoided, but each column can be seen as a response of the system to the nonzero initial state from an appropriate column of the matrix X_f, without any deterministic or stochastic inputs.

The corollary is that $\mathcal{O}_h$ is *numerically* invariant to the changes of the state space basis, although matrices Γ_h and X_f have invariant only the respective column and row space.

Deterministic $\mathcal{O}_h$ splitting

For the pure deterministic system it is a simple task, because the matrix $\mathcal{O}_h$ has the rank equal to the system order with the following SVD factorization

$$\mathcal{O}_h = USV^T = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix},$$

where S_1 is $n \times n$ sub-matrix of S containing nonzero singular values of $\mathcal{O}_h$ and U_1 and V_1^T are the appropriate parts of the matrices U and V^T. The number of nonzero singular values is equal to the order of the system and the required subspaces can be obtained as

$$\Gamma_h = U_1 S_1^{1/2}, \qquad X_f = S_1^{1/2} V_1^T,$$

where a square root of S_1 is simple, because S_1 is a diagonal matrix. Notice that multiplication by $S_1^{1/2}$ can be omitted, because we are interested only in the spanned spaces, but it is present for the equality $\mathcal{O}_h = \Gamma_h X_f$ to hold.

Stochastic $\mathcal{O}_h$ splitting

In the presence of the noise, the matrix $\mathcal{O}_h$ will be of the full rank. Thus all singular values of $\mathcal{O}_h$ will be nonzero, i.e. the diagonal of the matrix S will have nonzero entries in nonincreasing order. The rank of the identified system has to be chosen from the number of significant singular values. This can be tricky task, for which few theoretical guidelines are available. Assuming the system order to be determined, the SVD matrices are partitioned into the 'signal' and 'noise' parts

$$\mathcal{O}_h = \begin{pmatrix} U_s & U_n \end{pmatrix} \begin{pmatrix} S_s & 0 \\ 0 & S_n \end{pmatrix} \begin{pmatrix} V_s^T \\ V_n^T \end{pmatrix},$$

where U_s and V_s^T contain n principal singular vectors, whose corresponding singular values are collected in the $n \times n$ diagonal matrix S_s. The 'cleaned' estimates of the extended observability matrix and state sequence matrix are then

$$\Gamma_h = U_s S_s^{1/2}, \qquad X_f = S_s^{1/2} V_s^T. \tag{3.9}$$

Estimating $\mathcal{O}_h$ from the I/O data

In this section the term $\Gamma_h \hat{X}_f$ will be obtained from the I/O data in the signal matrices by an oblique projection. It will be obtained from the equation (3.7), where the last two terms on the right side will be eliminated by a projection and later it will be used as an approximation of $\mathcal{O}_h \doteq \Gamma_h \hat{X}_f$, where $\hat{X}_f$ is the Kalman filter estimate of X_f from the available past data W_p.

Let's try an orthogonal projection of the future output Y_f onto the subspace of past data W_p and future inputs U_f

$$Y_f \,/ \begin{pmatrix} W_p \\ U_f \end{pmatrix} = \Gamma_h \, \hat{X}_f \; + \; H_h^d \, U_f.$$

The orthogonal projection eliminated the noise term, because the estimated state sequence $\hat{X}_f$ and the future inputs U_f lies in the joint row space of W_p and U_f, but the future noise E_f is perpendicular to this subspace for the number of samples going to infinity (Figure 3.2).

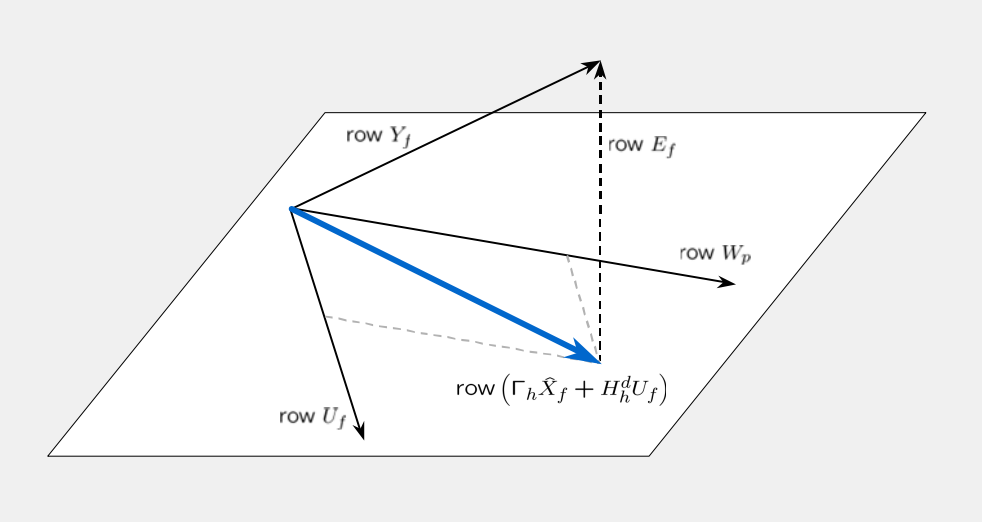

Figure 3.2: Orthogonal projection of input/output data.

Proof of $\hat{X}_f$ laying in the joint row space of W_p and U_f: From the equation (3.6), the past states X_p can be expressed as

$$X_p = \Gamma_i^\dagger Y_p - \Gamma_i^\dagger H_i^d U_p - \Gamma_i^\dagger H_i^s E_p = \begin{pmatrix} \Gamma_i^\dagger & -\Gamma_i^\dagger H_i^d & -\Gamma_i^\dagger & H_i^s \end{pmatrix} \begin{pmatrix} Y_p \\ U_p \\ E_p \end{pmatrix}.$$

Substituting this expression into (3.8) will yield for the future states X_f

$$\begin{aligned} X_f &= A^i X_p + \Delta_i^d U_p + \Delta_i^s E_p = A^i \Gamma_i^\dagger Y_p - A^i \Gamma_i^\dagger H_i^d U_p - A^i \Gamma_i^\dagger H_i^s E_p + \Delta_i^d U_p + \Delta_i^s E_p \\ &= \begin{pmatrix} A^i \Gamma_i^\dagger & (\Delta_i^d - A^i \Gamma_i^\dagger H_i^d) & (\Delta_i^s - A^i \Gamma_i^\dagger H_i^s) \end{pmatrix} \begin{pmatrix} Y_p \\ U_p \\ E_p \end{pmatrix}. \end{aligned}$$

These two last equations clearly indicate, that both past states X_p and future states X_f can be obtained as a linear combination of past data Y_p, U_p, E_p. In other words, they lie in their joint row space.

To compute an estimate of X_f the noise term can be replaced by its mean value

$$\hat{X}_f = \begin{pmatrix} A^i \Gamma_i^\dagger & (\Delta_i^d - A^i \Gamma_i^\dagger H_i^d) \end{pmatrix} \begin{pmatrix} Y_p \\ U_p \end{pmatrix} = L_w W_p,$$

showing $\hat{X}_f$ to lie in the row space of W_p.

□

Orthogonal projection helps us to get rid of the noise term, but we need to eliminate also the influence of the future inputs U_f and that is where *an oblique projection* is the right tool

$$Y_f \underset{U_f}{/} W_p = \Gamma_h \hat{X}_f. \tag{3.10}$$

Making the oblique projection of the future outputs Y_f onto the row space of past data W_p along the row space of future inputs U_f will give exactly the term we are looking for - $\Gamma_i \hat{X}_f$ (Figure 3.3).

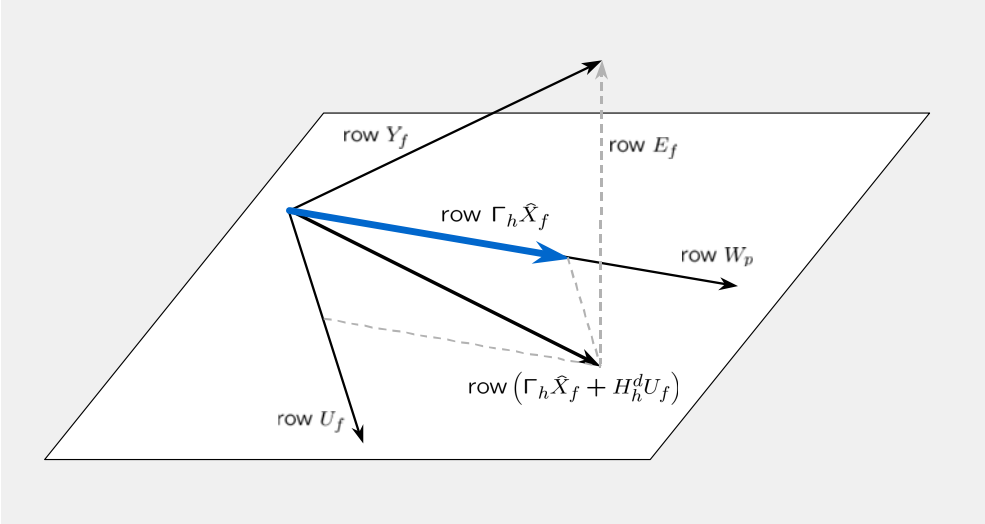

Figure 3.3: Oblique projection of input/output data.

Proof: Firstly the oblique projection of every term in (3.7) will be considered

$$Y_f \underset{U_f}{/} W_p = \left(\Gamma_h X_f + H_h^d U_f + H_h^s E_f\right) \underset{U_f}{/} W_p = \Gamma_h X_f \underset{U_f}{/} W_p + H_h^d U_f \underset{U_f}{/} W_p + H_h^s E_f \underset{U_f}{/} W_p.$$

Taking each term separately and considering the assumptions from section 3.8:

- For the number of samples and number of past data (rows) going to infinity $(i, j \to \infty)$

$$\lim_{i,j\to\infty} X_f \underset{U_f}{/} W_p = \lim_{i,j\to\infty} X_f / W_p = \hat{X}_f,$$

  because as has been shown, $\hat{X}_f$ is X_f estimate lying in W_p.

- From the oblique projection definition it is obvious that

$$U_f \underset{U_f}{/} W_p = 0.$$

- For the number of samples going to infinity and from the noises independency

$$\lim_{j\to\infty} E_f \underset{U_f}{/} W_p = 0.$$

In the summary

$$\lim_{i,j\to\infty} Y_f \underset{U_f}{/} W_p = \Gamma_h \hat{X}_f$$

□

All these projection equalities are exact for the number of data approaching infinity ($i, j \to \infty$). To summarize the effect of the projections

$$\begin{aligned} Y_f &= \Gamma_h\, X_f \;+\; H_h^d\, U_f \;+\; H_h^s\, E_f, \\ Y_f \,/ \begin{pmatrix} W_p \\ U_f \end{pmatrix} &= \Gamma_h\, \hat{X}_f \;+\; H_h^d\, U_f, \\ Y_f \underset{U_f}{/} W_p &= \Gamma_h\, \hat{X}_f. \end{aligned}$$

3.7 Estimating SS model parameters

Assume that the matrix $\mathcal{O}_h$ is already known. Now it can be used to get the state space model parameters by splitting it into the required subspaces by SVD as was shown in the previous section. The parameters can be basically obtained from the extended observability column space $\mathrm{col}(\Gamma_h)$ or the state sequence matrix row space $\mathrm{row}(\hat{X}_f)$.

Estimating parameters from row$(\hat{X}_f)$

Assume the inputs, the outputs and the state sequences are now available (the basis for the state sequence row space was chosen)

$$\begin{aligned} \hat{X}_i &= \begin{pmatrix} \hat{x}_i, & \ldots, & \hat{x}_{i+j-1} \end{pmatrix}, \quad \hat{X}_{i+1} = \begin{pmatrix} \hat{x}_{i+1}, & \ldots, & \hat{x}_{i+j} \end{pmatrix}, \\ U_i &= \begin{pmatrix} u_i, & \ldots, & u_{i+j-1} \end{pmatrix}, \quad Y_i = \begin{pmatrix} y_i, & \ldots, & y_{i+j-1} \end{pmatrix}. \end{aligned}$$

Then in the presence of no feedback the parameters of the innovation model can be consistently estimated from the following matrix equation relating all data by the state space model

$$\begin{pmatrix} \hat{X}_{i+1} \\ Y_i \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} \hat{X}_i \\ U_i \end{pmatrix} + \varepsilon.$$

The solution can be obtained by least squares or total least squares. Denoting

$$\Theta = \begin{pmatrix} A & B \\ C & D \end{pmatrix}, \quad \mathcal{X} = \begin{pmatrix} \hat{X}_i \\ U_i \end{pmatrix}, \quad \mathcal{Y} = \begin{pmatrix} \hat{X}_{i+1} \\ Y_i \end{pmatrix},$$

the least squares solution can be obtained as

$$\Theta = \mathcal{Y}\mathcal{X}^\dagger = \mathcal{Y}\mathcal{X}^T \left(\mathcal{X}\mathcal{X}^T\right)^{-1},$$

and the stochastic parameters as

$$R_e = \Sigma_{22}, \quad K = \Sigma_{12}\Sigma_{22}^{-1},$$

where

$$\Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix} = \frac{1}{j - (n+m)} \left(\mathcal{Y} - \Theta\mathcal{X}\right) \left(\mathcal{Y} - \Theta\mathcal{X}\right)^T.$$

The consistency of solution requires $j \to \infty$.

Estimating parameters from col(Γ_h)

First the extended observability matrix Γ_h is obtained as a basis of col(Γ_h) or it is obtained directly from SVD of $\Gamma_h \hat{X}_f$ (3.9). Then the matrices A, B, C and D are determined in two steps.

Determination of A and C

The matrix C can be read directly from the first block row of Γ_h. The matrix A is determined from the shift structure of Γ_h. Denoting

$$\underline{\Gamma_h} = \begin{pmatrix} C \\ \vdots \\ CA^{k-2} \end{pmatrix} \in \mathbb{R}^{l(h-1)\times n}, \qquad \overline{\Gamma_h} = \begin{pmatrix} CA \\ \vdots \\ CA^{k-1} \end{pmatrix} \in \mathbb{R}^{l(h-1)\times n},$$

where $\underline{\Gamma_h}$ is Γ_h without the last block row and $\overline{\Gamma_h}$ is Γ_h without the first block row, the shift structure implies that

$$\underline{\Gamma_h} A = \overline{\Gamma_h}.$$

This equation is linear in A and can be solved by LS or TLS.

Determination of B and D

Obtaining B and D is more laborious. Multiplying the I/O equation (3.7) from the left by $\Gamma_h^\perp$ and from the right by $U_f^\dagger$ yields to

$$\Gamma_h^\perp Y_f U_f^\dagger = \Gamma_h^\perp \Gamma_h X_f U_f^\dagger + \Gamma_h^\perp H_h^d U_f U_f^\dagger + \Gamma_h^\perp H_h^s E_f U_f^\dagger,$$

where $\Gamma_h^\perp \in \mathbb{R}^{(lh-n)\times lh}$ is a full row rank matrix satisfying $\Gamma_h^\perp \Gamma_h = 0$. The equation can be simplified to

$$\Gamma_h^\perp Y_f U_f^\dagger = \Gamma_h^\perp \, . \, H_h^d.$$

Denote i-th column of LHS as $\mathcal{M}_i$ and i-th column of $\Gamma_h^\perp$ as $\mathcal{L}_i$, then

$$\begin{pmatrix} \mathcal{M}_1 & \mathcal{M}_2 & \dots & \mathcal{M}_h \end{pmatrix} = \begin{pmatrix} \mathcal{L}_1 & \mathcal{L}_2 & \dots & \mathcal{L}_h \end{pmatrix} \begin{pmatrix} D & 0 & 0 & \cdots & 0 \\ CB & D & 0 & \cdots & 0 \\ CAB & CB & D & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ CA^{h-2}B & CA^{h-3}B & CA^{h-4}B & \cdots & D \end{pmatrix}.$$

As shown in [1], this can be rewritten to

$$\begin{pmatrix} \mathcal{M}_1 \\ \mathcal{M}_2 \\ \vdots \\ \mathcal{M}_h \end{pmatrix} = \begin{pmatrix} \mathcal{L}_1 & \mathcal{L}_2 & \cdots & \mathcal{L}_{h-1} & \mathcal{L}_h \\ \mathcal{L}_2 & \mathcal{L}_3 & \cdots & \mathcal{L}_h & 0 \\ \mathcal{L}_3 & \mathcal{L}_4 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \mathcal{L}_h & 0 & 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} I_l & 0 \\ 0 & \underline{\Gamma_h} \end{pmatrix} \begin{pmatrix} D \\ B \end{pmatrix},$$

which is an overdetermined set of linear equations in the unknowns B and D and it can be solved by LS or TLS.

3.8 N4SID algorithm summary

Here the complete 4SID algorithm [1] will be shown

Assumptions

1. The process noise w_k and measurement noise v_k are not correlated with the input u_k (open-loop identification).

2. The input $u(t)$ is assumed to be persistently exciting [7] of the order $i+h$. It means that the spectrum $\Theta_u(\omega)$ is different from zero in at least $i+h$ points on the interval $-\pi < \omega < +\pi$ or that for multiple-input system the input covariance matrix
$$R^{uu} = \begin{pmatrix} U_p \\ U_f \end{pmatrix} \begin{pmatrix} U_p \\ U_f \end{pmatrix}^T$$
has full rank which is $m(i+h)$.

3. The number of measurements goes to infinity $j \to \infty$.

4. The user-defined weighting matrices $W_1 \in \mathbb{R}^{lh \times lh}$ and $W_2 \in \mathbb{R}^{j \times j}$ are such that W_1 is of full rank and W_2 obeys: $\text{rank}(W_p) = \text{rank}(W_p W_2)$.

Algorithm

1. Firstly arrange the I/O data into Hankel signal matrices U_p, U_f, Y_p, Y_f and their respective +/- modifications.

2. Calculate the oblique projections
$$\begin{aligned} \mathcal{O}_h &= Y_f \underset{U_f}{/} W_p, \\ \mathcal{O}_{h+1} &= Y_f^- \underset{U_f^-}{/} W_p^+. \end{aligned}$$

3. Compute SVD of the weighted oblique projection
$$W_1 \mathcal{O}_h W_2 = USV^T$$

4. Determine the system order by inspecting the singular values of S and partition the SVD accordingly to obtain U_1 and S_1
$$\mathcal{O}_h = USV^T = \begin{pmatrix} U_1 & U_2 \end{pmatrix} \begin{pmatrix} S_1 & 0 \\ 0 & S_2 \end{pmatrix} \begin{pmatrix} V_1^T \\ V_2^T \end{pmatrix}.$$

5. Determine Γ_h and $\underline{\Gamma_h}$ as

$$\begin{aligned} \Gamma_h &= W_1^{-1} U_1 S_1^{1/2}, \\ \underline{\Gamma_h} &= \Gamma_h(1:h-1,:). \qquad \text{(Matlab like notation)} \end{aligned}$$

6. Determine the state sequences

$$\begin{aligned} \hat{X}_i &= \Gamma_h^{\dagger} \mathcal{O}_h, \\ \hat{X}_{i+1} &= \underline{\Gamma_h}^{\dagger} \mathcal{O}_{h+1}. \end{aligned}$$

7. Estimate the parameters A, B, C and D form a set of linear equations

$$\begin{pmatrix} \hat{X}_{i+1} \\ Y_i \end{pmatrix} = \begin{pmatrix} A & B \\ C & D \end{pmatrix} \begin{pmatrix} \hat{X}_i \\ U_i \end{pmatrix} + \epsilon.$$

8. Estimate the stochastic parameters R_e and K from the covariance estimate of the residuals as

$$\begin{aligned} \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix} &= \frac{1}{j-(n+m)} \varepsilon \varepsilon^T \\ R_e &= \Sigma_{22} \\ K &= \Sigma_{12} \Sigma_{22}^{-1} \end{aligned}$$

There are several more sophisticated modification improving robustness and eliminating bias.

Weighting matrices

The weighting matrices W_1 and W_2 were introduced to integrate several formerly separate 4SID algorithms, which will be described later, into this single one [6].

	W_1	W_2
N4SID	I_{li}	I_j
MOESP	I_{li}	$\Pi_{U_f}^{\perp}$
CVA	$\left((Y_f/U_f^{\perp})(Y_f/U_f^{\perp})^T\right)^{-1/2}$	$\Pi_{U_f}^{\perp}$

However there are some other interpretations of weighting matrices. 4SID methods determine Γ_h and $\hat{X}_f$ up to within a non-singular similarity transformation $T \in R^{n \times n}$

$$\begin{aligned} \Gamma_h' &= \Gamma_h T \\ \hat{X}_f' &= T^{-1} \hat{X}_f. \end{aligned}$$

This ambiguity raises the following question: In which state space basis are Γ_h and $\hat{X}_f$ determined when a subspace method is used to estimate them? And that is where the weighting matrices play the key role. The matrices W_1 and W_2 determines the basis of estimates. By a proper choice of weighting matrices, the identification result can be also influenced in a frequency specific way [7].

3.9 Other 4SID algorithms

The algorithm described in a previous section is called N4SID (Numerical algorithm for 4SID) [1]. There are several others (PI-MOESP, PO-MOESP, CVA), which will be described in this section. It is good to be aware, that all these advanced 4SID algorithms share several fundamental steps. Early 4SID algorithms will be omitted, because they work well for deterministic data, but struggle under the noise burden. The idea is to correlate both sides of the basic matrix equation(3.11) with quantities that eliminate the term with U_f and make the noise influence term with E_f disappear asymptotically. In more details:

1. They all start with a single equation matrix formulation of SS model

$$Y_f = \Gamma_h X_f + \underbrace{H_h^d U_f}_{\text{input term}} + \underbrace{H_h^s E_f}_{\text{noise term}} \tag{3.11}$$

   and try to use different projections to estimate the term $\Gamma_h X_f$ out of Y_f or at least to get the column space of the extended observability matrix Γ_h. This incorporates the elimination of the last two terms - the effect of the input and the effect of the noise.

2. **Removing the input term.** The straightforward idea is to orthogonally project (3.11) onto the row space orthogonal to U_f. The projection matrix

$$\Pi_{U_f}^{\perp} = I - U_f^T \left(U_f U_f^T\right)^{\dagger} U_f,$$

   will eliminate the input influence

$$U_f \Pi_{U_f}^{\perp} = U_f - U_f U_f^T \left(U_f U_f^T\right)^{\dagger} U_f = 0,$$

   but the noise term E_f will be left intact, because it is uncorrelated with the deterministic input.

$$Y_f \Pi_{U_f}^{\perp} = \Gamma_h X_f \Pi_{U_f}^{\perp} + H_i^s E_f \Pi_{U_f}^{\perp}. \tag{3.12}$$

   The oblique projection used in N4SID combines this and the next step.

3. **Removing the noise term.** To eliminate the effect of the noise, the algorithms use instrumental variable methods (IVM) with different instruments selection. In some methods the IV principles are not obvious at the first sight and will be discussed in Section 4.3.

   Shortly it consists of finding good instruments $\xi(t)$, which are uncorrelated with the noise, sufficiently correlated with the state and which are orthogonal to the inputs U_f. These instruments are usually created from the past inputs and outputs U_p, Y_p to decorrelate the noise term from (3.12):

$$Y_f \Pi_{U_f}^{\perp} \xi^T = \Gamma_h X_f \Pi_{U_f}^{\perp} \xi^T,$$

$$H_i^s E_f \Pi_{U_f}^{\perp} \xi^T \to 0 \quad \text{as} \quad j \to \infty.$$

4. Finally they use the estimates of extended observable matrix Γ_h or state sequence matrix X_f or extended controllability matrix Δ_h or their combination to compute SS model parameters.

It is obvious, that there are several design variables in general 4SID algorithm, which differentiate the particular algorithms. The reason for many 4SID algorithms is the different selection of these design variables, because **at present it is not fully understood how to choose them optimally** [7].

MOESP

The acronym MOESP [8, 16] stands for "Multivariable Output Error State sPace". These methods are based on the combination of the projections and explicit instrumental variable methods. SS model parameters are recovered from the estimated extended observability matrix.

For noise free data, the subspace identification reduces to the orthogonal projection of the equation (3.11) onto the row space of $U_f^{\perp}$, resulting in numerically exact term $\Gamma_h X_f$. However in the presence of noise, the geometrical properties of (3.11) are lost. Therefore the IVs are used as the instruments for removing the effect of the noise term. The informative part of the signal term must, however, be left intact. Let $\xi(t)$ denote the vector of instruments. The instrumental variables must satisfy the following requirements

1. IVs $\xi(t)$ must be uncorrelated with the noise

$$E\left[e(t)\xi(t)^T\right] = 0.$$

2. IVs $\xi(t)$ must be sufficiently correlated with the state to allow recovery of the estimated Γ_h column space

$$\operatorname{rank}\left(E\left[x(t)\xi(t)^T\right]\right) = n. \tag{3.13}$$

3. IVs $\xi(t)$ must be orthogonal to the input U_f (to leave the informative part undisrupted).

The first two requirements (usual in IVM) suggest the input signal as a candidate instrument. Clearly, this is incompatible with the third requirement. A solution is the very partitioning of data into the past and the future parts. Afterwards the feasible instrument is the past input U_p.

For simultaneously removing the U_f term and decorrelating the noise, Verhaegen [16] proposed to consider the following quantity (the equation 3.11 multiplied by $\Pi_{U_f}^{\perp} U_p^T$ from the right side)

$$Y_f \Pi_{U_f}^{\perp} U_p^T = \Gamma_h X_f \Pi_{U_f}^{\perp} U_p^T + H_h^s E_f \Pi_{U_f}^{\perp} U_p^T. \tag{3.14}$$

It is not difficult to see that the input term was eliminated $H_i^d U_f \Pi_{U_f}^{\perp} U_p^T = 0$ and the noise term is asymptotically disappearing with the number of samples going to infinity

$$\lim_{j\to\infty} E_f \Pi_{U_f}^{\perp} U_p^T = 0.$$

The rank condition (3.13) of the matrix

$$X_f \Pi_{U_f}^{\perp} U_p^T$$

is shown in [16] to be equal to the number of purely deterministic states. In other words, with only past inputs as IVs, only the deterministic subsystem can be identified. Any additional dynamics due to the colored disturbances are lost in the IVs correlation. The algorithm based on this choice of IVs is called **PI-MOESP** (Past Inputs MOESP) and it consists of applying SVD to LHS of (3.14) and taking its left singular vectors as an estimate of the extended observability matrix, which is later used for retrieving the system parameters.

If a complete state space model is desired, incorporating both the deterministic and the stochastic subsystem, then IVs must be extended with the past outputs Y_p as an additional instrument. The algorithm is then called **PO-MOESP** (Past Outputs MOESP). Similarly to (3.14) of PI-MOESP, with extended IVs we get

$$Y_f \Pi_{U_f}^{\perp} \begin{pmatrix} U_p^T \\ Y_p^T \end{pmatrix} = \Gamma_h X_f \Pi_{U_f}^{\perp} \begin{pmatrix} U_p^T \\ Y_p^T \end{pmatrix} + H_h^s E_f \Pi_{U_f}^{\perp} \begin{pmatrix} U_p^T \\ Y_p^T \end{pmatrix}. \tag{3.15}$$

The matrix on the left side can be used as in PI-MOESP, but PO-MOESP usually works differently. Firstly computing the following QR factorization

$$\begin{pmatrix} U_f \\ U_p \\ Y_p \\ Y_f \end{pmatrix} = \begin{pmatrix} L_{11} & 0 & 0 & 0 \\ L_{21} & L_{22} & 0 & 0 \\ L_{31} & L_{32} & L_{33} & 0 \\ L_{41} & L_{42} & L_{43} & L_{44} \end{pmatrix} Q,$$

where the L_{ii} are lower triangular. The next step is to compute truncated SVD

$$\begin{pmatrix} R_{42} & R_{43} \end{pmatrix} = \hat{U}_s \hat{S}_s \hat{V}_s^T + \hat{U}_n \hat{S}_n \hat{V}_n^T.$$

The extended observability matrix is then estimated by $\hat{U}_s$, and A and C are extracted in the usual way. For finding B and D, Verhaegen argues [16] that the least-squares solution to the overdetermined system of equations

$$\begin{pmatrix} R_{31} & R_{42} \end{pmatrix} \approx H_h^d \begin{pmatrix} R_{11} & R_{22} \end{pmatrix}$$

provides a consistent estimate of H_h^d, from which B and D are easily calculated.

CVA

CVA [22] is a class of algorithms, which stands for "Canonical Variate Analysis". They are based on the concept of principal angles between subspaces.

The considered subspaces are the row space of conditional past data $W_p/U_f^\perp$ and the row space of conditional future outputs $Y_f/U_f^\perp$. In [6] it is shown that the number of principal angels between those subspaces $\left[W_p/U_f^\perp \triangleleft Y_f/U_f^\perp\right]$ different from $\pi/2$ is equal to the model order. Actually selecting non-perpendicular angles for the noisy data is the same problem as selecting non-zero singular numbers of $\mathcal{O}_h$ SVD in N4SID algorithm.

The fundamental idea of CVA is that the estimate of the future states $\hat{X}_f$ can be found from the principal directions between past and future data $\left[\mathbf{W_p}/\mathbf{U_f^\perp} \triangleleft Y_f/U_f^\perp\right]$. More precisely not the states, but their projection

$$\left[\mathbf{W_p}/\mathbf{U_f^\perp} \triangleleft Y_f/U_f^\perp\right] = \hat{X}_f/U_f^\perp.$$

These methods, although they are based on the principal angles and directions, can be also implemented by N4SID with appropriate weighting matrices (Section 3.8). Larimore claims [22] that the weighting used for this method is optimal for the state order determination from finite samples sizes. This has been shown by example, but has never been proven. A last remark concerns the sensitivity to scaling. While the algorithms N4SID and MOESP are sensitive to scaling of the inputs and/or outputs, the CVA algorithm is insensitive. This is because only angles and normalized directions are considered in the CVA algorithm.

3.10 Historical note

The subspace methods have their origin in state-space realization theory as developed in the 1960s. A classical contribution is by Ho and Kalman (1966), where a scheme for recovering the system matrices from the impulse response is outlined.

A problem with realization theory is the difficulty of obtaining a reliable non-parametric estimate of the impulse response. A number of algorithms require special inputs, such as impulse or white noise signals. An alternative approach is to extract the desired information directly from the input/output data, without explicitly forming impulse response and that is the case of 4SID methods.

Realization theory

Realization theory allows to get the system matrices of state space model from a discrete impulse response of a system. Assume a state space description of a deterministic system

$$x(t+1) = Ax(t) + Bu(t), \tag{3.16}$$

$$y(t) = Cx(t) + Du(t). \tag{3.17}$$

This description is not unique due to arbitrary state space basis selection. For any nonsingular matrix $T \in \mathcal{R}^{n\times n}$ the state $x(t)$ can be transformed to the new state $x(t) = T\bar{x}(t)$ giving the following model

$$\begin{aligned} \bar{x}(t+1) &= T^{-1}AT\bar{x}(t) + T^{-1}Bu(t), \\ y(t) &= CT\bar{x}(t) + Du(t), \end{aligned}$$

which is equivalent with (3.16),(3.17). However the impulse responses are the same in any state space basis

$$\bar{H}(z) = CT\left(zI - T^{-1}AT\right)^{-1}T^{-1}B + D = C\left(zI - A\right)^{-1}B + D = H(z)$$

The impulse sequence of the deterministic model can be easily computed from (3.16–3.17) as

$$h(t) = \begin{cases} 0 & t < 0 \\ D & t = 0 \\ CA^{t-1}B & t > 0 \end{cases} \tag{3.18}$$

Consider the finite impulse sequence is known. The matrix D can be read directly from $h(0)$. To obtain other matrices the elements of impulse response are arranged into Hankel matrix as follows

$$H = \begin{pmatrix} h(1) & h(2) & h(3) & \dots & h(n+1) \\ h(2) & h(3) & h(4) & \dots & h(n+2) \\ h(3) & h(4) & h(5) & \dots & h(n+2) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ h(n+1) & h(n+2) & h(n+3) & \dots & h(2n+1) \end{pmatrix}.$$

Using (3.18) it is straightforward to verify that H can be factorized as

$$H = \Gamma_{n+1}\Delta_{n+1},$$

where

$$\Delta_i = \begin{pmatrix} B & AB & \dots & A^{i-1}B \end{pmatrix}$$

is the extended controllability matrix. For a minimal realization, the matrices Γ_{n+1} and Δ_{n+1} have full rank, and hence H has rank n, equal to the system order. The algorithm of Ho and Kalman is based on the above observations.

In the first step the matrix H is factorized by SVD to get Γ_{n+1} and Δ_{n+1}. In the next step, the matrices B and C can be read directly from the first m columns of Δ_{n+1} and l rows of Γ_{n+1} respectively. The remaining matrix A is computed using shift invariance structure of Γ_i:

$$\overline{\Gamma_{n+1}} = \underline{\Gamma_{n+1}}A.$$

This property of extended observability matrix can be exploited by the use of least squares or total least squares to get the system matrix A.

The disadvantage of the realization methods is their need for the impulse response. Several non-parametric methods can be used to estimate it, e.g. Correlation analysis or Inverse discrete Fourier transformation, but the results are not satisfactory. 4SID methods are in principal similar, but they do not need the impulse response of the system.

The similarity is in the matrix $\mathcal{O}_h$, which can be seen as 'an impulse response from the states'. And it is shown to be equal to the multiplication of the extended observability matrix and the state sequence matrix, just like $H = \Gamma_{n+1}\Delta_{n+1}$. The advantage is that $\mathcal{O}_h$ can be obtained directly from the I/O data.

3.11 Notes

- One has to keep in mind that 4SID consider signal matrices (U_p, U_f, Y_p, Y_f, E_p, E_f) as a definition of subspaces spanned by their rows. In other words the exact numerical values of the row vectors are of lower importance.

- The very simple observation allowing us to work with the signal matrices row spaces, is that in the single matrix SS equations (3.6-3.8), they are always multiplied by the parameters matrices from the left. This means, that the multiplication result and the result on the RHS is only a linear combination of the signal matrix rows.

- The basic reasons allowing 4SID methods to work are simply non-uniqueness of the state space model description and asymptotic properties of the projections allowing to eliminate the noise and to project out the known input signal.

- Several proposed 4SID algorithms are basically same and they differ only in the projection and instrumental variable selection and in the way of obtaining the model parameters from the estimated subspaces of Γ_h or X_f. The N4SID algorithm with the weighting matrices unifies the others, because its middle product of the estimation, the matrix $\mathcal{O}_h$, is independent of the state basis selection.

Chapter 4

Interpretations and Relations

In this chapter, several interpretations of 4SID methods in the well known frameworks will be shown in one integral form. The connections and the similarities with other methods will be emphasized.

4.1 Multi-step Predictions Optimization

There is a natural question to ask about 4SID methods. Is there any criterion that is minimized in the process of identification as it is usual for the other methods (ARX, ARMAX)? Even the very recent articles [15] state, that there is no such suitable minimization criterion:

> "The available methods (4SID methods) are algebraic in nature as they are not based on a suitable identification criterion minimization."

However we have shown [21] that it is possible to find such criterion and by its minimization, the pivotal oblique projection of N4SID can be obtained. In the following, it will be shown, that 4SID methods are minimizing multi-step predictions error of the model.

Multi-step Predictor Form

Assume the state space model (3.1–3.2) is known, and at time k, the system state x_k and the sequence of future inputs $\{u_k, u_{k+1}, \ldots, u_{k+h-1}\}$ are also known. Then from the state equations (3.1–3.2) we can compute the predictions of the outputs from 0 to $h-1$ steps ahead (unknown innovations are replaced by their mean value $e_k = 0$)

$$\hat{y}_{k+m} = \begin{cases} Cx_k + Du_k & m = 0, \\ CA^m x_k + Du_{k+m} + \sum_{n=0}^{m-1} CA^{m-n-1} Bu_{k+n} & m \in \langle 1, h-1 \rangle . \end{cases}$$

This expression can be rewritten utilizing matrices defined in Section 3.2 as

$$\begin{pmatrix} \hat{y}_k \\ \hat{y}_{k+1} \\ \vdots \\ \hat{y}_{k+h-1} \end{pmatrix} = \Gamma_h x_k + H_h^d \begin{pmatrix} u_k \\ u_{k+1} \\ \vdots \\ u_{k+h-1} \end{pmatrix}. \tag{4.1}$$

In this way we get the sequence of output predictions from time k to $k+h-1$ for one initial time k. Additionally, the output predictions for j subsequent initial time instants can be written in a compact form using Hankel matrix notation

$$\begin{pmatrix} \hat{y}_i & \cdots & \hat{y}_{i+j-1} \\ \hat{y}_{i+1} & \cdots & \hat{y}_{i+j} \\ \vdots & \ddots & \vdots \\ \hat{y}_{i+h-1} & \cdots & \hat{y}_{i+h+j-2} \end{pmatrix} = \Gamma_h \begin{pmatrix} x_i & \ldots & x_{i+j-1} \end{pmatrix} + H_h^d \begin{pmatrix} u_i & \cdots & u_{i+j-1} \\ u_{i+1} & \cdots & u_{i+j} \\ \vdots & \ddots & \vdots \\ u_{i+h-1} & \cdots & u_{i+h+j-2} \end{pmatrix},$$

which can be shortened to

$$\hat{Y}_f = \Gamma_h X_f + H_h^d U_f. \tag{4.2}$$

Note that every column of $\hat{Y}_f$ represents the sequence of linear output predictions (for the initial time determined by the index of the first element in the column) based on the initial state x and the sequence of successive inputs u from the corresponding columns.

An important property of the equation (4.2) is that for $i, j \to \infty$ it can be rewritten as

$$\hat{Y}_f = L_w W_p + H_h^d U_f,$$

where the future states X_f were replaced by a linear combination of past inputs and outputs $W_p = \begin{pmatrix} Y_p \\ U_p \end{pmatrix}$ (as it was shown in Section 3.6). It correspond to the fact, that the system states can be obtained from the Kalman filter running over infinite set of historical input/output data W_p. System states are thus representing system history.

In the real case, the condition $i, j \to \infty$ is not satisfied (only a finite data set is available) and thus the Kalman filter will give only the best possible linear state estimate $\hat{X}_f$ based on available data. Then similarly like before it can be shown that the following replacement can be made

$$\widetilde{Y}_f = \Gamma_i \hat{X}_f + H_h^d U_f \quad \sim \quad \widetilde{Y}_f = L_w W_p + H_h^d U_f, \tag{4.3}$$

where $\widetilde{Y}_f$ is the best output linear estimate using the limited available data set.

Estimated system states $\hat{X}_f$ are the same as we can obtain from the bank of parallel non-stationary Kalman filters running over the columns of the matrix W_p with the following initial conditions [1]

$$\begin{aligned} \hat{X}_0 &= X_p \underset{U_f}{/} U_p, \\ P_0 &= \operatorname{cov}(X_p) = \frac{1}{j} X_p X_p^T. \end{aligned}$$

Deriving 4SID oblique projection from multi-step predictions optimization

We will try to find a linear state space model of a certain order, that will most accurately predict the behavior of the real process according to the measured data. Firstly we will show how to derive the oblique projection (3.10) from the multi-step predictor optimization and then we will reveal the intuition behind.

In other words we would like to find matrices L_w and H_i such that the predictions according to the equation (4.3) will correspond with the measured data. The quality of the predictions will be measured by Frobenius matrix norm

$$\min_{L_w,H_i} \left\| Y_f - \widetilde{Y}_f \right\|_F^2 = \min_{L_w,H_i} \left\| Y_f - \begin{pmatrix} L_w & H_i \end{pmatrix} \begin{pmatrix} W_p \\ U_f \end{pmatrix} \right\|_F^2 . \tag{4.4}$$

Minimizing (4.4) means finding the best linear predictor in the sense of the least squares. The optimal values of L_w and H_i can be found from matrix pseudo-inverse

$$\begin{pmatrix} L_w & H_i \end{pmatrix} = Y_f \begin{pmatrix} W_p \\ U_f \end{pmatrix}^{\dagger} .$$

Denoting $\mathcal{D} = \begin{pmatrix} W_p \\ U_f \end{pmatrix}$, the pseudo-inversion can be written as

$$\begin{pmatrix} L_w & H_i \end{pmatrix} = Y_f \mathcal{D}^T \left(\mathcal{D}\mathcal{D}^T \right)^{-1}, \tag{4.5}$$

multiplying both sides by the matrix $\mathcal{D}$ from the right yields to

$$\underbrace{\begin{pmatrix} L_w & H_i \end{pmatrix} \mathcal{D}}_{\hat{Y}_f} = Y_f \underbrace{\mathcal{D}^T \left(\mathcal{D}\mathcal{D}^T \right)^{-1} \mathcal{D}}_{\Pi_{\mathcal{D}}} . \tag{4.6}$$

We obtained an expression

$$\hat{Y}_f = Y_f \Big/ \begin{pmatrix} W_p \\ U_f \end{pmatrix},$$

which represents the best linear prediction of Y_f based on the available data. From $\hat{Y}_f$ we need to get *only the part coming from the term* $L_w W_p$, because it is equal to $\Gamma_i \hat{X}_i$, which is an estimate of the matrix $\mathcal{O}_i$ necessary for further identification using SVD (Section 3.6). To get it separately from (4.6) it is sufficient to use the right side of (4.5) and take only the first $2i$ columns and multiply them by the matrix W_p alone

$$L_w W_p = Y_f \mathcal{D}^T \Big[\left(\mathcal{D}\mathcal{D}^T \right)^{-1} \Big]_{\substack{\text{first 2i} \\ \text{columns}}} W_p .$$

On the right side we obtained an expression for the oblique projection (2.5)

$$L_w W_p = Y_f \underset{U_f}{/} W_p,$$

which can be rewritten using (4.3) as

$$Y_f \underset{U_f}{/} W_p = \Gamma_i \hat{X}_f \qquad (= \mathcal{O}_i),$$

yielding to the fundamental equation of subspace identification algorithm (3.10). If the following conditions are satisfied

- the deterministic inputs u_k are not correlated with the stochastic inputs e_k (open-loop identification),
- the inputs u_k and e_k are persistently exciting (i.e. all modes of the system are excited either by the inputs or by the noises),
- the number of measurements goes to infinity $j \to \infty$,

then the identification will be consistent.

ARMAX vs. 4SID

The interpretation showed in the previous section revealed that 4SID and ARMAX have closer relation than it may seems on the first sight. Both can be interpreted to identify the models, which are the optimal predictors, but with the different optimality criterion.

Regression methods give a model, which is *a single-step optimal predictor* and 4SID methods give a model which is *a multi-step optimal predictor*.

ARMAX	4SID
$\min_{\theta} \|y - Z\theta\|_F$	$\min \left\|Y_f - \hat{Y}_f\right\|_F = \min_{L_w, H_i} \left\|Y_f - \begin{pmatrix} L_w & H_i \end{pmatrix} \begin{pmatrix} W_p \\ U_f \end{pmatrix}\right\|_F$

Consequences

- ARMAX is a single-step optimal predictor and it can be shown that by cyclic application of this single-step optimal predictor we can get multi-step optimal predictor.
- However, this is true, but only for the data generated by ideal linear system. For the real-world data the multi-step predictions will not be optimal. And that is where 4SIM comes. Their models give multi-step optimal predictions even for the real-world data. Especially for the models with a reduced order and for the colored noises.

Simulations

This section shows a comparison of the prediction abilities of ARMAX and 4SID for non-ideal data and different model order reductions. The test input/output sequences were generated by the 4th order state space model

$$
\begin{aligned}
x_{t+1} &= \begin{pmatrix} 0.603 & 0.603 & 0 & 0 \\ -0.603 & 0.603 & 0 & 0 \\ 0 & 0 & -0.603 & -0.603 \\ 0 & 0 & 0.603 & -0.603 \end{pmatrix} x_t + \begin{pmatrix} 0.924 \\ 2.755 \\ 4.317 \\ -2.644 \end{pmatrix} u_t + \begin{pmatrix} -0.021 \\ 0.11 \\ 0.071 \\ -0.103 \end{pmatrix} e_t \\
y_t &= \begin{pmatrix} -0.575 & 1.075 & -0.523 & 0.183 \end{pmatrix} x_t + (-0.714)\, u_t + e_t
\end{aligned}
$$

The quality of the models is estimated by the ability to predict, which is quite different from the ability to accurately simulate the identified system, because for predicting we need also a good description of the stochastic subsystem. Simulations were done with 100 Monte-Carlo iterations using the following inputs:

Deterministic inputs: 1000 samples of Pseudo-Random Binary Signal (PRBS).

Stochastic inputs: white noise filtered with a low pass filter.

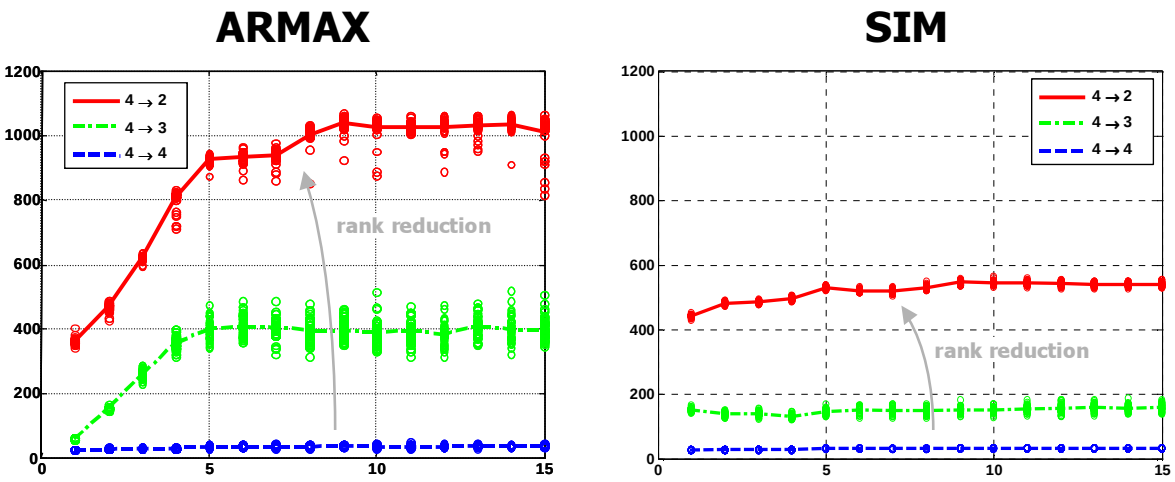

Figure 4.1: Prediction Error vs. Model Order.

Figure 4.1 shows two graphs relating the prediction error sum of squares to the number of predictions steps for ARMAX and 4SID identification. Different line colors represent different model order reductions and correspond for both graphs.

The blue lines represent identification without order reduction (not considerring stochastic subsystem) and both methods give similar results. The green lines are one order reduction from the 4th order to the 3rd order, where the difference is

becoming obvious and the red lines are two orders reduction from the 4th order to the 2nd order showing significantly lower error for 4SID methods.

Notice that 4SID methods give flat errors, showing almost equal weight on the prediction error steps, while regression methods clearly give stress to the single step-error. This can be more clearly seen on Figure 4.2 emphasizing two rank reduction comparison. This figure exactly follows the theoretical expectations showing lower

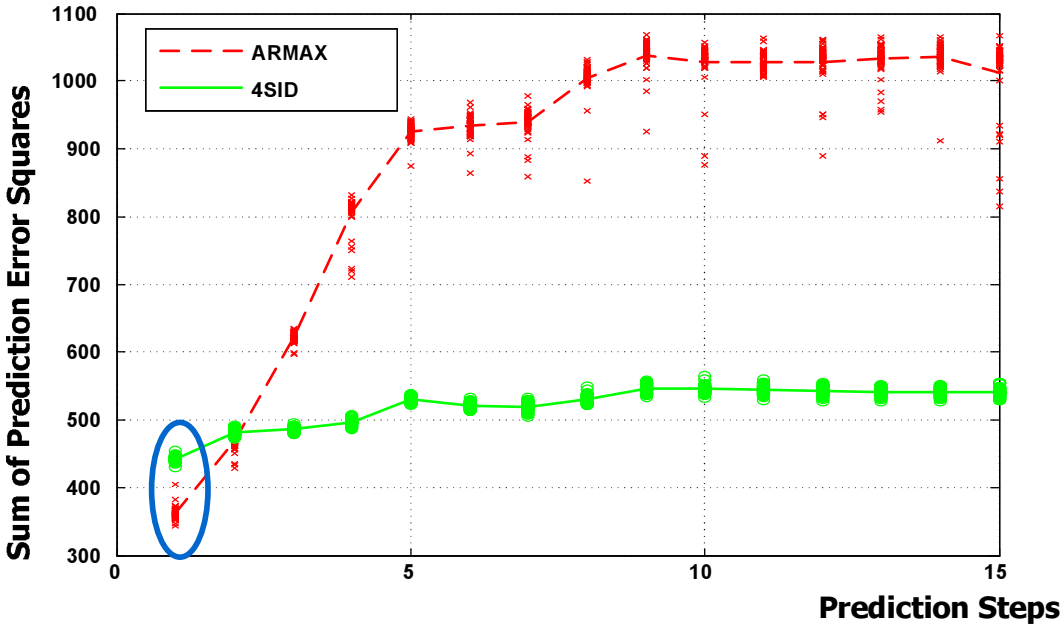

Figure 4.2: Prediction Error for reduced order model 4 → 2. In the blue oval notice that ARMAX is better in single-step predictions, but significantly worse for multi-step predictions.

prediction error of ARMAX (red line) for single-step prediction, but significantly lower overall prediction error of 4SID (green line) on more prediction steps.

4.2 Kalman Filter Relation

The 4SID methods have one very interesting property in relation with Kalman filter (KF). The 4SID algorithm can be interpreted as KF, estimating the system states *without prior knowledge of the system parameters (matrices).*

This is possible due to an arbitrary state space basis choice of SS model. The 4SID methods estimate only the row space of state sequence matrix X_f, which is independent of the basis selection and consequently use the fact that any of its basis is a valid state sequence.

Firstly it will be shown that Kalman filter recursive formulas can be rewritten into one single matrix linear equation.

Non-steady state Kalman filter

Given:

- the initial state estimate $\hat{x}_0$,
- the initial estimate of the matrix P_0,
- the input and output measurements $u_0, y_0, \ldots, u_{k-1}, y_{k-1}$,

then the non-steady state Kalman filter state estimate $\hat{x}_k$ is defined by the following recursive formulas:

$$\begin{aligned} \hat{x}_k &= A\hat{x}_{k+1} + Bu_k + K_{k-1}\left(y_{k-1} - C\hat{x}_{k-1} - Du_{k-1}\right), && (4.7)\\ K_{k-1} &= \left(G - AP_{k-1}C^T\right)\left(\Lambda_0 - CP_{k-1}C^T\right)^{-1}, && (4.8)\\ P_k &= AP_{k-1}A^T + \left(G - AP_{k-1}C^T\right)\left(\Lambda_0 - CP_{k-1}C^T\right)^{-1}\left(G - AP_{k-1}C^T\right)^T, && (4.9) \end{aligned}$$

where G is state to output covariance matrix and Λ_0 is the output covariance matrix, defined as

$$G = A\Sigma C^T + S, \qquad \Lambda_0 = C\Sigma C^T + R,$$

where Σ is the state covariance matrix $E\left[x_k x_k^T\right] = \Sigma$ given by the solution of the Lyapunov equation

$$\Sigma = A\Sigma A^T + Q.$$

In [1] it is shown and proved that the recursive KF formula (4.7 - 4.9) can be rewritten into one single linear matrix equation as

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

where

$$\Omega_k = \left(\Delta_k^c - A^k P_0 \Gamma_k^T\right)\left(L_k - \Gamma_k P_0 \Gamma_k^T\right)^{-1}.$$

The significance of KF matrix form is that it indicates how the Kalman filter state estimate $\hat{x}_k$ can be written as a linear combination of the past input and output measurements $u_0, y_0, \ldots, u_{k-1}, y_{k-1}$.

This allows the definition of the state sequence that is recovered by 4SID methods with $\hat{X}_0$ as initial states

$$\begin{aligned}\hat{X}_f &= \begin{pmatrix} \hat{x}_i & \hat{x}_{i+1} & \dots & \hat{x}_{i+j-1} \end{pmatrix} \\ &= \left(A^k - \Omega_i\Gamma_i \;\middle|\; \Delta_i^d - \Omega_i H_i^d \;\middle|\; \Omega_i \right) \left(\begin{array}{c} \hat{X}_0 \\ \hline U_p \\ \hline Y_p \end{array}\right) \\ &= \left(A^k - \Omega_i\Gamma_i \;\middle|\; \begin{pmatrix} \Delta_i^d - \Omega_i H_i^d & \Omega_i \end{pmatrix} \right) \begin{pmatrix} \hat{X}_0 \\ W_p \end{pmatrix}. \end{aligned} \tag{4.10}$$

This state sequence is generated by *a bank of non-steady state Kalman filters*, working in parallel on each of the columns of the block Hankel matrix of past inputs and outputs W_p. As can be seen, each state is filtered only from the limited length of past input/output information (i samples), which is contained in one column of W_p. Therefore the initial state estimate

$$\hat{X}_0 = \begin{pmatrix} \hat{x}_0^0 & \hat{x}_1^0 & \hat{x}_2^0 & \dots & \hat{x}_{j-1}^0 \end{pmatrix}$$

plays important role and advanced 4SID methods are shown to use nonzero initial estimates. For example the described N4SID method uses

$$\hat{X}_0 = X_p^d \underset{U_f}{/} U_p, \tag{4.11}$$

where X_p^d is the deterministic component of the state sequence matrix X_p.

One has to keep in mind, that this is only an interpretation of 4SID methods from the Kalman filter point of view. For example the prior estimate (4.11) is never computed in 4SID algorithm, but it can be shown that a bank of non-steady state KF with this prior estimate and working with the input/output data from W_p columns will give the same state sequence estimate as N4SID algorithm. The prior estimate (4.11) can not be even explicitly computed, because for example X_p^d is of course a priori unknown.

The major observation in the subspace algorithms is that the system matrices A, B, C, D, Q, R and S do not have to be known to determine the state sequence $\hat{X}_f$. It can be determined directly from the input/output data.

4.3 Instrumental Variable Methods Relation

Some 4SID methods can be interpreted to directly use the tools of Instrumental variable methods (PI-MOESP, PO-MOESP), but in the others (N4SID, CVA), the principles of IVMs are hidden. In this section, the methods will be compared from IVMs point of view.

IVMs are traditionally used and well understood in PEM framework, the extension of IVMs to state space models is less obvious. It was shown in [8], that

subspace-based methods can be viewed as one way of generalizing IVMs in a numerically reliable way for SS model identification.

The IVM approach in MOESP methods was described in Section 3.9. All methods are summarized in the following table:

Method	Instrumental Variable	$\mathcal{O}_h$ estimate
PI-MOESP	U_p	$Y_f \Pi_{U_f}^{\perp}$
PO-MOESP	$\begin{pmatrix} U_p \\ Y_p \end{pmatrix}$	$(Y_f / W_p)\, \Pi_{U_f}^{\perp}$
N4SID	$\begin{pmatrix} U_p \\ Y_p \end{pmatrix}$	$Y_f \underset{U_f}{/} W_p$

PO-MOESP and N4SID methods use identical instruments, but different weighting matrices [8]. The resulting subspace estimates should therefore have very similar properties. However, the schemes for unravelling the system matrices in MOESP and 4SID are quite different.

Writing both estimate of $\mathcal{O}_h$ in the similar notation

$$\begin{aligned} \mathcal{O}_{\text{N4SID}} &= Y_f \Pi_{U_f}^{\perp} W_p^T (W_p \Pi_{U_f}^{\perp} W_p^T)^{-1} W_p, \\ \mathcal{O}_{\text{PO-MOESP}} &= Y_f \Pi_{U_f}^{\perp} W_p^T (W_p \Pi_{U_f}^{\perp} W_p^T)^{-1} W_p \Pi_{U_f}^{\perp}, \end{aligned}$$

the difference is clear. It is only an extra projection $\Pi_{U_f}^{\perp}$ in PO-MOESP estimate.

4.4 Notes

As it has been illustrated in this chapter, 4SID methods are not so distant from the traditional approaches, as it may seem at the first sight. They should be viewed from two different view points:

1. As *implicit implementation of Kalman filter*. More precisely as a bank of non stationary Kalman filters, estimating each state separately from i (chosen parameter) historical I/O data and certain prior estimate. The important property is that the system parameters (A,B,C,D,Q,R,S) are a priori unknown and it is still possible to estimate systems states, but without particular basis selection. This property of KF is less known.

2. Second view point is that by using 4SID methods for identification, we will get the model with the optimal predictions for 1 to h (chosen parameter) steps. More precisely the sum of squares for the sum of 1 to h predictions steps will be minimal among all linear models of the same dimension. In other words, 4SID methods can be seen as a very elegant way how to specify and solve a problem of identifying a model, which is optimal for multi-step predictions.

Chapter 5

New Trends and Open Problems

In this chapter some recent advances will be shown to complete the state-of-the-art of 4SID. Nowadays, there are several open or only partially solved problems:

- Efficient recursive 4SID algorithm,
- Closed-loop identification,
- Prior knowledge incorporation,
- Optimal choice of the design variables for the general algorithm (weight matrices W_1, W_2),
- Consistency, convergence and error analysis.

Here we will treat the first two problems, because their solution is in our industrially motivated interest and several notable advances in these topics has been recently made. The third problem of prior knowledge incorporation seems also to be very interesting and in the next chapter, we will share some ideas, we have about our further research in this direction.

5.1 Recursive 4SID

The main 4SID methods (N4SID, PI/PO-MOESP, CVA) are dedicated for identification from off-line data. Their good properties are appreciated, but for the real industrial applications, especially in the control engineering, the efficient on-line recursive identification algorithm is necessary. Great effort was recently devoted to this field with many interesting results, but it seems that the goal is still far from reached.

The off-line 4SID algorithms are directly unusable for on-line identification due to the computationally burdensome steps such as singular value decomposition of large matrices, which is the main bottleneck in recursification. Therefore it was necessary to find SVD alternative algorithms in order to apply the subspace concept in a recursive framework.

The first attempts [18, 19], similarly compresses given I/O data recursively into a matrix with a fixed size. An estimate of the extended observability matrix is not updated directly, but in order to obtain the estimate, it is necessary to perform SVD on the data-compressed matrix at every update step.

The recent advances [10, 9] took inspiration mainly from the methods used in the field of *Array Signal Processing*, where a similar problem of Recursive Subspace Tracking can be found and it is solved by *Propagator method*.

Recursive Subspace Tracking

A similar problem from the sensor array signal processing is following:

Assume we have an array with multiple antennas in configuration, where each antenna has different directional sensitivity. The array is receiving several signals as planar waveforms coming from the different directions (the distance between antennas can be neglected). The considered problem is to recursively determine:

1. the number of incoming signals,
2. the direction of arrival for each signal,
3. reconstruct each waveform.

Mathematically the output from the general sensor array can be described as follows

$$z(t) = \Gamma(\theta)s(t) + b(t), \tag{5.1}$$

where z is the output of n_z sensors (antennas), s is the vector of n_s signal waveforms an b is the additive noise. The matrix $\Gamma(\theta)$ is the *array propagation matrix*. Each column $\Gamma(\theta) = \begin{pmatrix} a(\theta_1) & \dots & a(\theta_{n_s}) \end{pmatrix}$ represents the sensitivity of each sensor to the respective waveform and it is a function of the direction of waveform arrival.

We can see that the problem 1 corresponds to the model order determination, problem 2 is related to system modelling and problem 3 is Kalman filtering.

When studying the main techniques developed in both fields, it is interesting to notice that the mathematical problem is the same: to track some eigencomponents of particular matrices by adapting specific subspaces with the last observation. In array signal processing, notable algorithms have been developed to avoid the use of eigenvalue decomposition. Thus, it seems to be interesting to adapt some of these SVD alternatives to recursive subspace identification.

The first step is to rewrite a state space model (3.1–3.2) into an equivalent form to (5.1). Before it is necessary to introduce a temporal window

$$y_f^+(t) = \begin{pmatrix} y^T(t) \dots y^T(t+f-1) \end{pmatrix}^T \in \mathbb{R}^{lf \times 1}.$$

Then the state space model can be rewritten as

$$y_f^+(t) = \Gamma_f x(t) + H_f^d u_f^+(t) + \underbrace{H_f^s e_f^+(t)}_{=b_f^+(t)}.$$

The connection between subspace identification and array signal processing becomes apparent by writing

$$z_f^+(t) = y_f^+(t) - H_f^d u_f^+(t) = \Gamma_f x(t) + b_f^+(t).$$

The recursive algorithm is then made of two stages:

1. The update of the "observation vector" z_f^+ from the noisy input/output measurements
$$z_f^+(t) = y_f^+(t) - H_f^d u_f^+(t).$$
The matrix H_f^d is unknown at time t since it is constructed from SS model parameters. However, the observation vector $z_f^+(t)$ can be obtained by the projections.

2. The estimation of a basis of Γ_f from the observation vector
$$z_f^+(t) = \Gamma_f x(t) + b_f^+(t). \tag{5.2}$$
This step is carried out by *Propagator method*.

Propagator method

Assume that the system is observable. Then the extended observability matrix $\Gamma_f' \in \mathbb{R}^{lf \times n}$ with $lf > n$, can be reorganized to Γ_f containing n linearly independent vectors in the first rows, forming a submatrix Γ_{f_1}. Then, the complement Γ_{f_2} of Γ_{f_1} can be epxressed as a linear combination of this submatrix. So, there is a unique linear operator $P_f \in \mathbb{R}^{n \times (fl-n)}$ named *propagator* defined as

$$\Gamma_{f_2} = P_f^T \Gamma_{f_2}.$$

It can be verified that

$$\Gamma_f = \begin{pmatrix} \Gamma_{f_1} \\ \Gamma_{f_2} \end{pmatrix} = \begin{pmatrix} I_n \\ P_f^T \end{pmatrix} \Gamma_{f_1}.$$

Since rank $\left(\Gamma_{i_f}\right) = n$

$$\operatorname{col}\left(\Gamma_f\right) = \operatorname{col}\begin{pmatrix} I_n \\ P_f^T \end{pmatrix}.$$

This last expression implies that it is possible to get an expression of the observability matrix in a particular basis by determining the propagator. This operator can be estimated from the equation (5.2). After having applied a data reorganization so

that the first n rows of Γ_f are linearly independent, the following partition can be introduced

$$z_f^+(t) = \begin{pmatrix} z_{f_1}^+(t) \\ z_{f_2}^+(t) \end{pmatrix} = \begin{pmatrix} I_n \\ P_f^T \end{pmatrix} \Gamma_{f_1} x(t) + \begin{pmatrix} b_{f_1}^+ \\ b_{f_2}^+ \end{pmatrix}.$$

In the noise free case it is easy to show that

$$z_{f_2}^+ = P_f^T z_{f_1}^+,$$

but in the presence of noise, the equality is lost. However, an estimate of P_f^T can be obtained by minimizing the following cost function

$$J(P_f^T) = E \left\| z_{f_2}^+ - P_f^T z_{f_1}^+ \right\|^2,$$

which leads to LS solution. Unfortunately this solution is biased even if the noise is spatially and temporally white. Better solutions, circumventing this difficulty, based on the correlations are proposed in [10].

Subspace prediction for time-varying systems

A new interesting 4SID algorithm for *predicting linear time-varying systems* has been recently proposed in [9]. It uses a concept of *principal angles* [14] between subspaces, spanned by the past and current extended observability matrix.

The future subspace is then predicted by rotating the current subspace in the geometrical sense by the angles obtained from an extrapolation of the principal angles development trend.

The proposed algorithm is recursive and it is suitable for identifying slowly time-varying linear systems. Most existing systems show time-varying and/or nonlinear behavior, and the nonlinear systems are sometimes treated as high order linear time-varying systems from the practical point of view. Therefore this algorithm could be used with an advantage for adaptive controllers or MPC.

5.2 Closed-loop Identification

The closed-loop identification is of special interest for a large number of engineering applications. For safety reasons or quality restrictions, it is desirable that the identification experiments are carried out under the closed-loop conditions. The fundamental problem with closed-loop data is the correlation between the unmeasurable disturbances and the input caused by an effort of the controller to eliminate the influence of disturbances [7].

It is well known, that the most traditional 4SID methods (N4SID, MOESP, CVA) are not applicable to the systems operating under the closed-loop conditions without special treatments [11]. Most of them can not provide consistent estimates from the closed-loop I/O data.

However, the recent advances have shown that the modified 4SID algorithms can work even under the closed-loop conditions [11, 12, 20]. The names of interested people, such as Lennart Ljung show, that this research direction is promising. There are recently two main approaches, which will be only sketched here:

Innovation Estimation Method (IEM) [20, 11] This method pre-estimates the innovations $e(t)$ and then use them as known, to decouple the noise and the controlled input. The pre-estimation is usually done by a high order ARX or by decomposing matrix equation (3.11) into i block rows, and use the estimated innovations from the previous block rows to further estimate model parameters of the next block row sequentially.

The basic idea is that under certain assumptions, the consistency estimation with 4SID methods can be achieved if the innovation sequence is already known.

Whitening Filter Approach (WFA) This approach claims not to require any extra data and not to be sensible to the instability in the deterministic subsystem [12]. It is based on the idea of looking at the inverse system generating the innovations $e(t)$ from the joint process $y(t)$ and $u(t)$, namely the "whitening filter realization"

$$\begin{aligned} x(t+1) &= \bar{A}x(t) + Bu(t) + Ky(t), \\ e(t) &= -Cx(t) - Du(y) + y(t), \end{aligned}$$

where $\bar{A} = A - KC$ and $\bar{B} = B - KD$. The state $x(t)$ of this model is the same as for the innovation model (3.1–3.2), therefore both models have the same state space row(X_f). The whitening filter can be rewritten into single equation matrix form

$$Y_f = \bar{\Gamma}_f X_k + \bar{H}_f U_f + \bar{G}_f Y_f + E_f.$$

The next difference to the standard algorithms is that the projections are carried out on the model row by row (multi-stage projections). No pre-estimation is involved, but the projections have to be done block-row wise to decouple noise from control input.

Chapter 6

Our Contribution and Future Research Direction

The reason of our interest in 4SID methods is their ability to identify large MIMO systems without structural parametrization, their good numerical properties, and because they also proved [23] to be beneficial in the industrially successful model predictive control (MPC).

6.1 Our Contribution

So far, our main contribution to the field of Subspace identification was that 4SID methods can be interpreted to identify the model as an optimal multi-step predictor [21]. Our contribution is that we have mathematically derived that oblique projections used in Subspace identification can be obtained by minimizing the multi-step prediction error criterium (4.4). Some results were shown in section 4.1. This also showed 4SID methods in the well known framework of the least-squares and proved, that even 4SID methods, with their geometrical identification approach, hide inside a criterion, which is minimized during the identification process. As we have shown this criterion was not expected to exist, as it was stated in some recent articles.

We are also very interested in the recursification, because it is an essential assumption for the identification methods to be used in on-line industrial application. Several algorithms have been proposed. The best one seems to be the subspace tracking by propagator methods. Recusrification of 4SID would allow to create adaptive MPC.

6.2 Prior knowledge incorporation

Our derivation of 4SID methods in the least-squares framework, showed new opportunity to incorporate prior knowledge, which is a persistent problem of 4SID methods and as far as we know, *no solution has been proposed yet.*

To incorporate prior knowledge into the algorithm of 4SID identification would be very beneficial. So far, 4SID methods have black-box approach to the identified system. This is good for 'automatic identification', where no additional information about the identified system is available, but the I/O data may be corrupted and the pure black-box approach may lead to a model, which is significantly deviated from the real system.

However, there is often a priori information, which should be exploited to increase the quality of the identification and to ensure that the identified model is close to some expectations (lies within some boundaries). The following list shows some a priori information, which would be useful to incorporated:

- known steady gain,
- known time constants range of step response,
- enforcing impulse (step) response smoothness - by specifying prior values of impulse response second differences (useful to keep the model from fitting on the high-frequency noise in data),
- Limit the step response to a funnel shaped region (Figure 6.1). This would allow to constrain the time constants, model gain, over-shots and under-shots, which would be very useful and versatile.

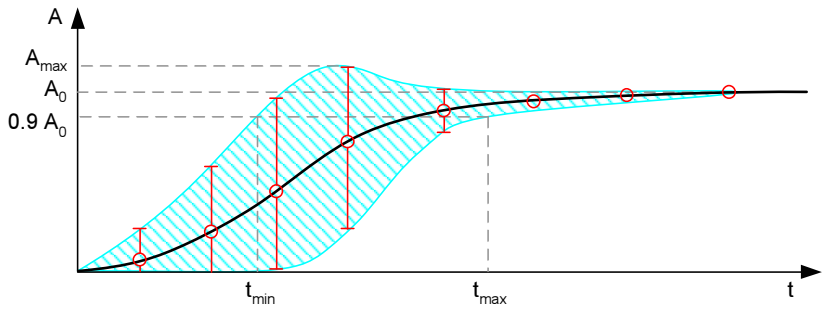

Figure 6.1: A priory information as a funnel limiting model step response.

To incorporate prior information, we expect, that we will be able to minimize the multi-step criterion

$$\min_{L_w, H_i^d} \left\| Y_f - \widetilde{Y}_f \right\|_F^2 = \min_{L_w, H_i^d} \left\| Y_f - \begin{pmatrix} L_w & H_i^d \end{pmatrix} \begin{pmatrix} W_p \\ U_f \end{pmatrix} \right\|_F^2, \tag{6.1}$$

while considering prior knowledge. This will involve to combine information from the I/O measurements and a priori information, which can be done by covariance matrix shaping and recursive least squares.

The expected main problems will be

1. The reformulation of some types of a priori information described above to the prior information on the parameters, which are optimized in the multi-step criterion (6.1), i.e. L_w and H_i^d or maybe only on the matrix $\mathcal{O}_h$. We hope to take advantage of the fact, that these parameters are not dependent on the state space basis selection, because they are related to the impulse responses from the states and the inputs.

2. The recursification - we expect that the least squares approach would allow straightforward recursification of 4SID algorithm. The stress would be on the combination of the information from the input/output data and from a priori information on the model which have to be combined in each step.

The preliminary idea of the complete recursive algorithm is following:

1. Transformation of a priori information about the identified system to the covariance matrices of L_w and H_i^d, which are minimized in the criterion (and maybe to their prior mean values).

2. Recursive least squares minimization of (6.1) while combining the information from the previous step with the new information from the measured data and a priori information. Some type of forgetting should be incorporated, but we have to assure that a priori information about the system is not completely forgotten. Usual recursive minimization is possible, because the equation

$$Y_f = \begin{pmatrix} L_w & H_i^d \end{pmatrix} \begin{pmatrix} W_p \\ U_f \end{pmatrix} + \epsilon,$$

   is linear in the parameters L_w and H_i^d, and the sizes of L_w and H_i^d are independent on the amount of measured data.

3. Finally L_w can be used to estimate the system order and to estimate system matrices A and C, because it has the same column space as extended observability matrix Γ_i, since

$$L_w W_p = \Gamma_i X_f.$$

   This fact should be exploited by SVD or Propagator method. Matrices B and D can be for example obtained from H_i^d.

Note that the matrix H_i^d, with the deterministic impulse response in the Toeplitz form, has lower triangular form (3.5). Obtaining H_i^d with large nonzero entries above the main diagonal means badly conditioned I/O data.

Abbreviations

4SID	Subspace State Space System Identification
ARX	Auto-Regressive with eXogenous inputs
ARMAX	Auto-Regressive Moving Average with eXogenous inputs
CVA	Canonical Variate Analysis
IV	Instrumental Variable
LTI	Linear Time Invariant
LS	Least Squares
IEM	Innovation Estimation Method
IVM	Instrumental Variable Methods
KF	Kalman Filter
LHS	Left Hand Side
MOESP	Multivariable Output Error State sPace
MPC	Model Predictive Control
N4SID	Numerical Subspace State Space System Identification
PEM	Prediction Error Methods
PI-MOESP	Past Inputs Multivariable Output Error State sPace
PO-MOESP	Past Outputs Multivariable Output Error State sPace
RHS	Right Hand Side
SS	State Space
TLS	Total Least Squares
WFA	Whitening Filter Approach

Bibliography

[1] Peter Van Overschee and Bart De Moor. *Subspace Identification for Linear Systems: Theory-Implementation-Applications*. Kluwer Academic Publishers, 1996.

[2] Hiroshi Oku and Hidenori Kimura. Recursive 4SID algorithms using gradient type subspace tracking. *Automatica*, 38:1035–1043, June 2002.

[3] Wouter Favoreel, Bart De Moor, and Peter Van Overschee. Subspace state space system identification for industrial processes. *Journal of Process Control*, 10:149–155, 2000.

[4] Anders Lindquist and Giorgio Picci. Geometric Methods for State Space Identification (unpublished).

[5] W. Favoreel, B. De Moor, M. Gevers, and P. Van Overschee. Closed loop model-free subspace-based LQG-design (unpublished). 1998.

[6] Peter Van Overschee and Bart De Moor. A Unifying Theorem for Three Subspace System Identification. *Automatica, Special Issue on Trends in System Identification*, 31(12):1853–1864, 1995.

[7] Lennart Ljung. *System Identification: Theory for the User (2nd Edition)*. Prentice Hall PTR, 1998.

[8] Mats Viberg. Subspace-based Methods for the Identification of Linear Time-invariant Systems. *Automatica*, 31(12):1835–1851, 1995.

[9] Kentaro Kameyama and Akira Ohsumi. Recursive Subspace Prediction of Linear Time-Vayring Stochasti Systems. In *Proceedings of the 16th IFAC World Congress*. Kidlington, Oxford, GB, Elsevier, 2005. ISBN 0-08-144130-1.

[10] Guillaume Mercère, Stéphane Lecæuche, and Christian Vasseur. Sequential Correlation Based Propagator Algorithm for Recursive Subspace Identification. In *Proceedings of the 16th IFAC World Congress*. Kidlington, Oxford, GB, Elsevier, 2005. ISBN 0-08-144130-1.

[11] Weilu Lin, S. Joe Qin, and Lennart Ljung. Comparison of Subspace Identification Methods for Systems Operating on Closed-loop. In *Proceedings of the 16th IFAC World Congress*. Kidlington, Oxford, GB, Elsevier, 2005. ISBN 0-08-144130-1.

[12] Alessandro Chiuso and Giorgio Picci. Prediction Error vs. Subspace Methods in Closed Loop Identification. In *Proceedings of the 16th IFAC World Congress*. Kidlington, Oxford, GB, Elsevier, 2005. ISBN 0-08-144130-1.

[13] Mohinder S. Grewal and Angus P. Andrews. *Kalman Filtering: Theory and Practise, Second Edition*. John Wiley & Sons, 2001. ISBN 0-471-39254-5.

[14] David A. Harville. *Matrix Algebra From a Statistician's Perspective*. Springer, 1997. ISBN 038794978X.

[15] Pouliquen Mathieu and M'Saad Mohammed. An Interpretation of Subspace Identification Methods. In *Proceedings of the 16th IFAC World Congress*. Kidlington, Oxford, GB, Elsevier, 2005. ISBN 0-08-144130-1.

[16] M. Verhaegen. Identification of the deterministic part of MIMO state space models given in innovations form from input-output data. *Automatica*, 30:61–74, 1994.

[17] Peter Van Overschee and Bart DeMoor. N4SID: Subspace Algorithms for the Identification of Combined Deterministic-Stochasti Systems. *Automatica*, 30:75–93, 1994.

[18] Hiroshi Oku and Hidenori Kimura. A Recursive 4SID from the Input-Output Point of View. *Asian Journal of Control*, 1(4):258–269, 1999.

[19] Y. M. Cho, G. Xu, and T. Kailath. Fast Recursive Identification of State Space Models via Exploitation of Dispalcement Structure. *Automatica*, 30(1):45–59, 1994.

[20] Weilu Lin, S. Joe Qin, and Lennart Ljung. A Framework for Closed-loop Subspace Identification with Innovation Estimation. Technical Report 2004-07, Texas-Wisconsin Modeling and Control Consorcium, February 2004.

[21] Pavel Trnka and Valdimír Havlena. Subspace Identification as Multi-step Predictions Optimization. In *Proceedings of the Fifth IASTED International Conference on Modelling, Simulation and Optimization*, pages 223–228. Anaheim: ACTA Press, 2005. ISBN: 0-88986-524-8.

[22] W. E. Larimore. Canonical variate analysis in identification, filtering and adaptive control. In *Proceedings of 29th Conference on Decision and Control, Hawai, USA*, pages 596–604, 1990.

[23] Ramesh Kadali, Biao Huang, and Anthony Rossiter. A Data Driven Subspace Approach to Predictive Controller Design. *Control Engineering Practise*, 11:261–278, 2002.

My Publications

[1] Pavel Trnka and Petr Smolík. Profibus DP Master for PC. In *Proceedings of 8th International Student Conference on Electrical Engineering POSTER 2004 [CD-ROM]*. Prague: CTU FEE Prague, 2004.

[2] Jiří Roubal, Vladimír Havlena, Jaroslav Neuhauser, and Pavel Trnka. MPC Controller for Two-Dimensional Systems. In *Proceedings of the 15th International Conference on Systems Science*, pages 132–140. Wroclaw: Oficyna Wydawnicza Politechniki Wroclawskiej, 2004. ISBN 83-7085-804-X.

[3] Jiří Roubal, Vladimír Havlena, and Pavel Trnka. Simulation of Two-Dimensional Distributed Parameters Systems. In *Proceedings of Matlab 2004 Conference*, volume 1-2, pages 500–504. Prague: Institute of Chemical Technology, 2004. ISBN 80-7080-550-1.

[4] Pavel Trnka and Hana Pytelová. Czech Economy Model Estimation by the Bootstrap Filter. In *Proceedings of 9th International Student Conference on Electrical Engineering POSTER 2005 [CD-ROM]*. Prague: CTU FEE Prague, 2005.

[5] Pavel Trnka, Vladimír Havlena, and Jan Štecha. Bootstrap Filtering for Czech Macro-economic Model Estimation. In *Proceedings of 15th International Conference on Process Control 05 [CD-ROM]*. Bratislava: Slovak University of Technology, 2005. ISBN 80-227-2235-9.

[6] Pavel Trnka and Valdimír Havlena. Subspace Identification as Multi-step Predictions Optimization. In *Proceedings of the Fifth IASTED International Conference on Modelling, Simulation and Optimization*, pages 223–228. Anaheim: ACTA Press, 2005. ISBN: 0-88986-524-8.