Mededelingen van de Koninklijke
Academie voor Wetenschappen, Letteren
en Schone Kunsten van België

Overdruk uit

Academiae Analecta

AWLSK

Klasse der Wetenschappen, Jaargang 55, 1993, Nr. 5

Paleis der Academiën, Brussel

NUMERICAL ALGORITHMS FOR STATE SPACE SUBSPACE SYSTEM IDENTIFICATION

BY

Bart DE MOOR

ESAT — Department of Electrical Engineering, Katholieke Universiteit Leuven, Kardinaal Mercierlaan 94, B-3001 Leuven (Heverlee), Belgium, tel : 32/16/220931 — fax : 32/16/221855, email : demoor@esat.kuleuven.ac.be.

ABSTRACT

We present a survey of new dynamical system identification algorithms. In this new framework, the concept of a *state* is emphasized. Moreover, the numerical implementation is based on robust algorithms from numerical linear algebra. An extensive comparison with 'classical' identification approaches is discussed.

1. Introduction

In this work, we will present a new set of algorithms for dynamical system identification, which is the field of modeling dynamical systems from experimental data. The novelty of our approach lies in the fact that the importance of *the state of the dynamical system* is emphasized in the context of system identification, whereas 'classical' approaches are based on an input-output framework. This relatively recent introduction of the state into the identification area may come as a surprise since in control theory and the analysis of dynamical systems, the importance of the concept of state has been appreciated for quite some time now. A second aspect we would like to emphasize is the fact that our system identification approach makes full use of the by now well developed body of concepts and algorithms from numerical linear algebra. While classical methods are basically inspired by least squares, our methods use 'modern' algorithms such as the QR-decomposition, the singular value decomposition and its generalizations, and angles between subspaces. A third contribution lies in the fact that our approach provides a geometric framework, in which seemingly different models are treated in a unified manner. This will be explained in detail below. We think that the conceptual and algorithmic simplicity of our algorithms should be confronted with and compared to the sometimes extremely complicated and cumbersome arguments and approaches that are often found in present day system identification literature.

The basic structure of a subspace algorithm for a linear time invariant finite dimensional system can be summarized in the following table, which in some sense also summarizes the present work on the linear system, geometric and algorithmic level.

System	Geometry	Algorithm
Determination of the order and the state	Projection (orthogonal or oblique)	QR-decomposition
	Determine finite dimensional subspace	(Generalized) singular value decomposition
System matrices	Linear relations	Least squares

This paper is organized as follows :

- For readers not quite familiar with the field of system identification, a brief introduction is provided in Section 2 in which we confront the approach of mathematical modeling based on physical insight with the system identification approach.
- In Section 3, we describe the class of systems that will be considered in this work.
- In Section 4, the subspace structure of these systems and associated data is explored in some detail.
- A first geometrical result is provided in Section 5, in which we show how the projection of future outputs to the inputs and past outputs, will allow us to determine the Kalman filter states of the model.
- This idea is pursued in Section 6, in which we give a prototype identification algorithm (there are many variations possible, for which we will refer to the literature).
- In Section 7, we briefly describe the numerical issues, which will lead to an extremely elegant yet robust numerical implementation.
- Some historical review (which is far from complete) and special cases are treated in Section 8.
- In Section 9, we give an argumentation why subspace algorithms could be more robust and preferable over other identification algorithms.
- The conclusions can be found in Section 10, which also contains some suggestions for future work.

2. Why system identification ?

2.1. *Models of systems*

A dynamic system can conceptually be described as in Figure 1, which covers almost all physical, economical, biological, industrial, etc ... systems. One could distinguish between mental, intuitive or verbal models, or graphically oriented approaches such as graphs and tables, but we will mainly be interested

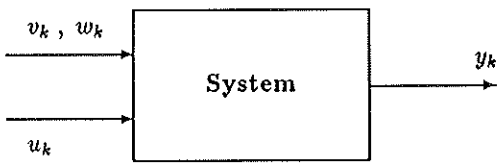

Fig. 1. — A dynamic system with inputs u_k, outputs y_k and disturbances v_k and w_k (see below). All arrows represent vector signals and k is the discrete time index. The user can control u_k but not v_k or w_k. In some applications, either u_k, v_k or w_k can be missing. The output signals provide useful information about the system.

in mathematical models. Such models are described as differential (continuous time) or difference (discrete time) equations. They describe the dynamic behavior of a system as a function of time. Models in general are used for simulation, prediction, fault detection, operator training, state and parameter estimation and system analysis and most typically in those situations in which experimenting with the real system is too expensive, too dangerous, too difficult or even impossible. Last but not least, mathematical models are used for *control* and *feedback*, which, by the way, is one of the major engineering inventions. Our every day actions are (or should be) based on a careful assessment of the parameters of a given situation (which is constructing a model) followed by a certain control action or decision, which will have its impact on the given circumstances. The idea of *model-based control system design* is completely similar. Based on a mathematical model of a the system, a certain mathematically calculated control strategy is deduced, which is then applied to the real system. This control strategy takes into account predefined engineering specifications, such as robustness criteria (e.g. conserving stability in the face of uncertainties) and performance objectives (tracking, disturbance rejection, ...). Ever since the path breaking work of Kalman in the beginning of the sixties (on least squares estimation of the state and least squares control), the central paradigm in control system design is that a controller that is more ambitious than the traditional trial-and-error tuning, should be model-based (or at least contain a certain model implicitly). While this is straightforward to see as far as *feed-forward* control strategies are concerned (prediction based), it is far less obvious that also for *feedback* strategies models are indispensable.

Basically, there are two ways of constructing a mathematical model. One is the analytic approach, which employs the laws of physics to describe the dynamic behavior of a phenomenon or process. Here one heavily relies on a priori information [1]. The second approach is *system identification*. This is the field of modeling dynamical systems from experimental data : Experiments are performed on a system, a certain parameterized model class is predefined by the user and suitable numerical values are assigned to the parameters so as to fit as good as possible the recorded data. In this sense, system identification is the dynamic extension of curve fitting.

A comparison can be made between the two basic approaches described above. In many cases, the physical processes are so complex that it is not possible to obtain a reasonable model from first principles only. As an example, consider the modeling of an industrial process such as a glass furnace that

[1] As an example, Halley conducted a prediction exercise in 1704, when, realizing that reports of comets in 1531, 1607, and 1682 were related to one single object, he calculated the parameters of its orbit and predicted its return for 1758 (Unfortunately, he did not live to see his prediction fulfilled as he died in 1742). His a priori information consisted of his knowledge of Newtonian dynamics and gravitational theory.

produces glass tubes for light bulbs. While the physical laws governing this process are perfectly well known (heat, convection-diffusion equations), a complete mathematical description of the process is extremely complicated (non-linear time varying partial differential equations, with complicated boundary conditions) so that only a numerical simulation may produce good simulations (even then, there are many unknown parameters such as diffusion and heat conduction coefficients which are hard to estimate). Even if such a model would provide good simulations and predictions, the design of a control strategy based on this model would be a mathematically and numerically intractable problem.

System identification however provides a meaningful alternative. Models are prespecified by the user, and then fitted to the recorded data by identification methods. Compared to models obtained from physics, system identification models have a limited validity and working range and in some cases have no direct physical meaning. But, they are relatively easy to obtain and use and even more importantly, these models are simple enough to make model-based control system design mathematically (and also practically) tractable. Of course, there are still problems such as the choice of an appropriate model structure, the fact that many systems are time-varying and often underestimated measurement problems. Let us return to our glass furnace. One can consider the fuel consumption of the heaters and the ventilation flow of a cooling channel as inputs, and the temperature distribution in a cross-section of the furnace as an output. This distribution is measured by putting a certain number of temperature sensors (which are very expensive to withstand the high temperatures of melting glass) in this cross-section, in this way effectively discretizing the continuous distributed parameter medium. A control specification here is to design a controller that regulates the inputs in such a way that the temperature in the cross-section is as homogeneous as possible (the homogeneity of the temperature will influence the quality of the glass). By *approximating* the dynamic behavior between the inputs and the outputs by a *lumped finite-dimensional, linear, time-invariant* model, one can not only obtain extremely reproducible simulations and predictions, but also design a quite attractive, robust and implementable minimum variance controller, which on the average, effectively minimizes the differences in temperature between the several output sensors and in this way guarantees a uniform temperature pattern. Obviously, such a task is impossible when starting from the partial differential equation model (which when discretized e.g. via finite elements can lead to reliable simulations, at a prize of having to solve sets of extremely large sets of linear equations).

Let us conclude this section by saying that system identification, being a typical engineering discipline, borrows many of its concepts and techniques from other mathematical and engineering fields, such as optimization, numerical

analysis, linear algebra, complex function theory, statistics, sensors and physical devices, experimental design, software engineering, etc ... and therefore is in many respects an interdisciplinary activity.

2.2. *System identification experiments*

The first issue in setting up a system identification experiment is the determination of relevant inputs and outputs. For some systems, such as industrial plants, this causality issue is not a real problem. Here, it is more of a problem to determine which signals are relevant, i.e. influential as possible control or disturbance inputs or relevant from the output point of view. For other systems, such as for instance ecological or economical ones, the problem of causality is much more subtle and involved. Nevertheless, we will assume from now on that this non-trivial causality problem has been solved for our application at hand and we will no longer deal with it here.

In an industrial environment, a system identification experiment is typically performed by exciting the system via its inputs by actuators. The necessity of excitation is quite intuitive : If we want to model the dynamics of a certain phenomenon, they need to be excited and observable in the system's outputs. If for instance we want to model the springs of a car from measurements on the velocity and acceleration, it does not suffice to drive straight on at a constant speed on a very flat road. Instead, one has to make sharp turns or well planned accelerations and decelerations in order to excite the dynamic modes of the spring system, so that these will be picked up in the observations from which they can be identified using an identification method. Hence we will impose the requirement that the inputs applied to the system should be *persistently exciting*, i.e. they need to excite all the dynamic modes that we want to model and later on control. For this purpose, it suffices to apply sufficiently 'wild' inputs at the system. Examples of such inputs are white noise, pseudo-random binary noise, etc ... However, in an industrial environment, this idea is often found repulsive because it implies loss of production, unallowable deviations from quality tolerances or even additional safety requirements. In some circumstances, it's merely impossible to perform experiments (such as with power plants for instance) so that one has to resort to data obtained under normal operating conditions.

Let us summarize by saying that *good models cannot be obtained from bad data !*

Besides the sufficiently wild activation of a system at its inputs, a careful recording of the outputs via sensor measurements is also required. The practical difficulties associated with measuring signals are often understimated. In an industrial environment, the measurement noise may be excessive. As a matter of fact, there is a well known trade-off between costs and accuracy of sensors, a relationship which is often exponential. Secondly, not all the signals that

influence the system's output are defined by the user or can be measured. As an example, we think of a wind disturbance acting on a airplane, which will reveal itself via a deviation of the planned trajectory. These disturbances require the presence in the model and the identification methods of so-called disturbance models, which are often stochastic.

In general, data acquisition is not an easy undertaking. One has to make decisions (based on lots of experience) concerning choices of filters, data storage, number and quality of the sensors required, etc ... As an example, let us consider the choice of the sampling time. It is well known from the Nyquist-Shannon theorem, that information in continuous time variables is lost when they are sampled at too low a rate. We must sample at $2f$ if we want to preserve a component at frequency f. However, due to the imperfections of practical filters, a realistic sampling rate is often taken to be $10 f_{cut}$, where f_{cut} is the cut-off frequency of the anti-aliasing low-pass filters. To rapid sampling however may cause problems too, as continuous time minimum-phase systems (all zeros in the left-half complex plane) may be turned into discrete time non-minimum-phase systems (zeros outside the unit circle) which can then impose lower bounds on certain performance criteria that are requested from the controller. Another difficult issue is the preprocessing of the data, which often contain outliers, peaks, drifts, etc ... Very often, industrial processes have substantial dead times (or delays) which have to be known or estimated too.

When all of these difficult decisions have been made, one can finally try to obtain a mathematical model of the measured data records using one's favorite identification technique. If such a model is obtained, one has to check its validity in a phase of the modeling process, which is called *model validation*. The quality of the model depends of course on what we intend to do with it. For control system design, surprisingly often, it is not required to have a detailed high-quality model. Especially since the advent of *robust controller* design approaches, one can now permit a trade-off between the accuracy of the model, and the robustness of the control strategy that takes into account deviations of the model from the 'real' system.

Because of the complicated interaction between experiment set up, data generation and acquisition, model selection, system identification, model validation, most often, there is a need for an iterative global identification scheme, in which all of these actions are repeatedly performed with other settings and user choices.

A good survey of the state of the art of system identification up to 1971 is provided in [6]. By now, there are some books that treat the classical identification approaches in some detail. One of the earliest is the book [10], which first appeared in 1970 and played a major emancipating and stimulating role in the development of the field. Recent books include [33] [43] [55] [57]. The main emphasis in these books is on prediction error methods. An impressive reference work on stochastic systems is [13].

3. Linear time invariant models

It goes without saying that there is an infinite collection of mathematical models. They can be classified into classes characterized as systems with lumped or distributed parameters, time invariant or time varying, continuous time or discrete time, linear versus non-linear, parametric or non-parametric, etc ...

In this work, we will restrict ourselves to a certain model class : We will identify for a given data set consisting of input-output pairs (u_k, y_k), a so-called discrete time, linear, time-invariant, state space model, which is described by the following set of equations :

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

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

with

$$\mathbf{E}\left[\begin{pmatrix} w_p \\ v_p \end{pmatrix}(w_q^t \; v_q^t)\right] = \begin{pmatrix} Q_s & S_s \\ S_s^t & R_s \end{pmatrix}\delta_{pq} \geq 0^2 \tag{3}$$

and A, $Q_s \in \mathbb{R}^{n\times n}$, $B \in \mathbb{R}^{n\times m}$, $C \in \mathbb{R}^{l\times n}$, $D \in \mathbb{R}^{l\times m}$, $S_s \in \mathbb{R}^{n\times l}$ and $R_s \in \mathbb{R}^{l\times l}$. The vectors $u_k \in \mathbb{R}^{m\times 1}$ and $y_k \in \mathbb{R}^{l\times 1}$ are the measurements of respectively input and output, $v_k \in \mathbb{R}^{l\times 1}$ and $w_k \in \mathbb{R}^{n\times 1}$ are unmeasurable vector signals. We assume that they are normally distributed, zero mean, white noise vector sequences. The matrix pair $\{A, C\}$ is assumed to be observable. The system $\{A, (B\ Q_s^c)\}$ is assumed to be controllable. The controllable modes of the system $\{A, Q_s^c)\}$ are assumed to be stable [3].

Let us comment in some detail why it is often a good idea to try to fit experimental (industrial) process data to the model just described. First of all, for multiple-input, multiple output systems, the state space representation is the only model that is convenient to work with in *computer aided control system design* (CACSD). Most optimal controllers can be effectively computed in terms of the state space model, while for other system representations (such as e.g. matrix fractional forms) the calculations are not so elegant. The matrix B represents the linear transformation by which the deterministic inputs influence the next state. The matrix A describes the dynamics of the systems (as completely characterized by its eigenvalues). The matrix C describes how the internal state is transferred to the outside world in the measurements y_k while the term with the matrix D is called the direct feedthrough term. In

[2] E denotes the expected value operator and δ_{pq} the Kronecker delta.

[3] This implies the stationarity of the stochastic model to be defined below. However, it can be shown that subspace algorithms will also work for marginally stable stochastic processes (that have poles on the unit circle).

continuous time systems this term is most often 0, which is not the case in discrete time systems due to the sampling. As we will see below, in our framework the states x_k don't have a direct physical interpretation but they have a conceptual relevance. Of course, if the system states would have some physical meaning, one could always find a similarity transformation of the state space model to convert the states to physically meaningful ones. Observe that we have collected *all* dynamis in one matrix A, that is to say that the eigenvalues of the matrix A will describe all the dynamical modes that have been measured, whether they come from the real system, from stochastic dynamic disturbances, from measurement sensors or the dynamics of the input actuators. This is quite unusual as compared to approaches that are described in the literature, in which one always distinguishes carefully between e.g. deterministic models (such as models for the 'real' system and sensor and actuator dynamics) and noise models for stochastic disturbances (as is for instance the case in the Box-Jenkins approach [10]). The point here is that more often than not, we don't care about the precise origin of the dynamic modes, since, if they are important, they will certainly influence the controller action, independent of their origin. There is a modern trend in CACSD to define what is called a *standard plant* (see e.g. [11]), which contains the model of all disturbances, all sensors and the system model in one general state space description, which exactly correspond to the model we will use.

The assumption that the noise sequences v_k and w_k are Gaussian is quite natural for many applications, due to the central limit theorem (which here acts as an important engineering simplification). Of course, it can not always be made uncriticaly. Nevertheless, the approximation is often very satisfactory. Returning to our glass furnace, one can for instance model the dynamic impact of the inhomogeneities in the melting glass, as white noise being sent through a linear system. The corresponding output would then be a stochastic signal which is colored (and therefore will influence the control action). Another example are stochastic models of wind disturbances, which have a very specific frequency spectrum, hence can be stochastically modeled by finite dimensional linear systems driven by white noise. Measurement noise (which is ubiquitous in industrial environments) is included in the stochastic white noise sequence v_k, while it is assumed that the input is applied to the system without distortion (in other words, we do not assume that the sequence u_k is corrupted by noise).

A crucial question is of course why linearity would apply to everyday processes, since we all know that most phenomena are intrinsically non-linear. One reason is the experience that many industrial processes are really well approximated by linear finite dimensional systems and that sometimes, complex behavior can be captured by choosing the order n high enough. In order to cope with non-linearities, two measures are possible : Either the non-linearity is dealt with by identifying a time-varying system using a recursive updating of the model. This corresponds to a local linearization of the nonlinear system.

A second possibility is provided by the observation that (mild) nonlinearities do not matter as they can be incorporated in the control design (robustness for dynamic uncertainties). Moreover, it is well known that a controller effectively linearizes the behavior of a system around a working point. Finally, we recall that the design of a controller is relatively easy for linear finite dimensional systems. As a matter of fact, this is the only class of systems for which CACSD is actually tractable in full generality and for which there is a complete rigorous theory available.

We are now ready to state the main mathematical problem of this paper : Given input and output measurements $u_1, \ldots, u_N$ and $y_1, \ldots, y_N$, $(N \to \infty)$, find the matrices, $A, B, C, D, Q_s, R_s, S_s$.

4. Subspace structure of linear systems

Before we tackle the question of subspace system identification from the given data u_k, y_k, we will first analyze in some detail the structure of the problem. This will result in a careful enumeration of properties that are crucial for our algorithms to be described below. We will decompose the states and outputs of the system into a deterministic part and a stochastic part as $x_k = x_k^d + x_k^s$ and $y_k = y_k^d + y_k^s$.

4.1. *The deterministic subsystem*

We define the deterministic subsystem as [4]

$$x_{k+1}^d = Ax_k^d + Bu_k$$
$$y_k^d = Cx_k^d + Du_k$$

This subsystem describes the influence of the deterministic input u_k on the deterministic output y_k^d. We call x_k^d the deterministic state. Associated with the deterministic subsystem, we define the following matrices. The extended observability matrix Γ_i (where the subscript i denotes the number of block rows) and the reversed extended controllability matrix Δ_i^d (where the subscript i denotes the number of block columns) :

$$\Gamma_i = \begin{pmatrix} C \\ CA \\ CA^2 \\ \ldots \\ CA^{i-1} \end{pmatrix} \qquad \Delta_i^d = (A^{i-1}B \quad A^{i-2}B \ldots AB \quad B)$$

[4] Note that this is just a name, since we have repeatedly emphasized that A also may contain dynamics of the stochastic disturbances. This implies that all modes of A are not necessarily controllable by B.

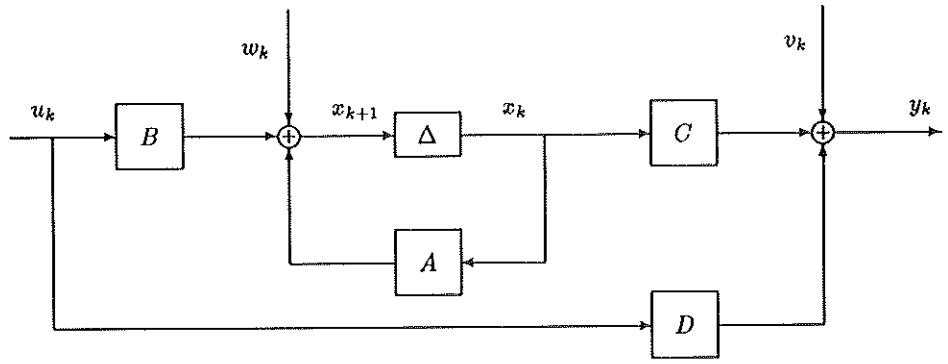

Fig. 2. — This picture is the same as the one in Figure 1. But here, we have restriced ourselves to finite dimensional linear time invariant systems to be identified. The vector signals u_k and y_k are available (measured) while v_k, w_k are unknown disturbances. The symbol Δ represents a delay. Note the inherent feedback via the matrix A (which represents the dynamics). Sensor or actuator dynamics are completely contained in A too. It is assumed that u_k is available without measurement noise.

The lower block triangular Toeplitz matrix H_i^d is defined as :

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

4.2. *The stochastic subsystem*

Together with the covariance structure (3), the stochastic subsystem is defined as :

$$x_{k+1}^s = Ax_k^s + w_k \tag{4}$$

$$y_k^s = Cx_k^s + v_k \tag{5}$$

This subsystem describes the influence of the noise sequences w_k and v_k on the stochastic output. We call x_k^s the stochastic state. We also define $P^s = \mathbf{E}[x_k^s (x_k^s)^t]$, $G = \mathbf{E}[x_k^s (y_k^s)^t]$, $\Lambda_0 = \mathbf{E}[y_k^s (y_k^s)^t]$. With equations (3), (4), (5) and through stationarity of the controllable modes of the system $\{A, Q_s^{1/2}\}$, we find easily that these matrices are related by

$$P^s = AP^sA^t + Q_s \tag{6}$$

(which is a Lyapunov equation), $G = A P^s C^t + S_s$ and $\Lambda_0 = C P^s C^t + R_s$. This is the set of positive real equations, which, in the context of stochastic systems, was introduced in [25]. It is also easy to derive that :

$$\Lambda_i \overset{\text{def}}{=} \mathbf{E}\,[y^s_{k+i}\,(y^s_k)^t] = \begin{cases} C A^{i-1} G & i > 0 \\ \Lambda_0 & i = 0 \\ G^t\,(A^t)^{-i-1}\,C^t & i < 0 \end{cases} \tag{7}$$

We also define the matrix $\Delta^s_i = (A^{i-1} G\ A^{i-2} G \ldots A G\ G)$ and the block Toeplitz covariance matrix L^s_i and the block Toeplitz cross covariance matrix H^s_i:

$$L^s_i = \begin{pmatrix} \Lambda_0 & \Lambda_{-1} & \Lambda_{-2} & \ldots & \Lambda_{1-i} \\ \Lambda_1 & \Lambda_0 & \Lambda_{-1} & \ldots & \Lambda_{2-i} \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \Lambda_{i-1} & \Lambda_{i-2} & \Lambda_{i-3} & \ldots & \Lambda_0 \end{pmatrix} \quad H^s_i = \begin{pmatrix} \Lambda_i & \Lambda_{i-1} & \Lambda_{i-2} & \ldots & \Lambda_1 \\ \Lambda_{i+1} & \Lambda_i & \Lambda_{i-1} & \ldots & \Lambda_2 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ \Lambda_{2i-1} & \Lambda_{2i-2} & \Lambda_{2i-3} & \ldots & \Lambda_i \end{pmatrix} = \Gamma_i\,\Delta^s_i$$

4.3. *Block Hankel matrices and input-output equations*

Input and output block Hankel matrices are defined as :

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

where we assume that $j \to \infty$. This implies that we have a sufficiently large amount of data. The reason to do so is that equalities in this work only hold asymptotically as $j \to \infty$ when stochastic disturbances are involved. In order to emphasize this, we could introduce a new symbol which indicates this dependence (as we have done explicitly in e.g. [64].) [5]. We will also assume that $i \geq n$. The subscripts of U and Y denote the subscript of the first and last element of the first column. For convenience and short hand notation, we will use $U_p = U_{0|i-1}$, $Y_p = Y_{0|i-1}$, $U_f = U_{i|2i-1}$ and $Y_f = Y_{i|2i-1}$ where a subscript 'p' refers to past and 'f' to future. We define the deterministic state matrix X^d_i as :

$$X^d_i = (x^d_i\, x^d_{i+1}\, x^d_{i+2} \ldots x^d_{i+j-1})$$

[5] We will not undertake here a detailed explanation of ergodicity which allows us to replace sample averages by time averages and for which we also need that $j \to \infty$. If only (part of) one realization of a process is ever observed, it is costless to assume that this process is ergodic, for that merely says that we hold as fixed these aspects of the stochastic structure that are fixed over the one realization we shall ever see. In most practical applications, we only have one data record anyway, but we might make it as long as required by taking enough measurements.

The past and future deterministic states are then defined as $X_p^d = X_0^d$ and $X_f^d = X_i^d$. The block Hankel matrices formed with the output y_k^s and state x_k^s of the stochastic subsystem are defined as respectively $Y_{0|i-1}^s$ and X_i^s in the same way. Once again, we define for short $Y_p^s = Y_{0|i-1}^s$, $X_f^s = Y_{i|2i-1}^s$. Due to stationarity, the following limits are well defined :

$$\lim_{j\to\infty} \frac{1}{j} Y_p^s (Y_p^s)^t = L_i^s \tag{8}$$

$$\lim_{j\to\infty} \frac{1}{j} Y_f^s (Y_f^s)^t = L_i^s \tag{9}$$

$$\lim_{j\to\infty} \frac{1}{j} Y_f^s (Y_p^s)^t = H_i^s \tag{10}$$

All of these matrices, can be related via the following

Theorem 1 Matrix input-output equations

$$Y_p = \Gamma_i X_p^d + H_i^d U_p + Y_p^s \tag{11}$$

$$Y_f = \Gamma_i X_f^d + H_i^d U_f + Y_f^s \tag{12}$$

$$X_p^d = A^i X_p^d + \Delta_i^d U_p \tag{13}$$

This Theorem is easy to prove by recursive substitution into the state space equations. Earlier references (where the deterministic part is derived as $Y_p^d = \Gamma_i X_p^d + H_i^d U_p$) include [30] [12] [42] [14] [15]. It can also be found in [38]. This equation has played a 'historical' role in the (slow) development of subspace algorithms for system identification. In the derivation and proof of the results to be presented below, it plays a crucial role.

5. The main projection and its interpretation via the Kalman filter

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

$$Z_i = Y_{i|2i-1} / \begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix} \qquad Z_{i+1} = Y_{i+1|2i-1} / \begin{pmatrix} U_{0|2i-1} \\ Y_{0|i} \end{pmatrix} \tag{14}$$

where A/B denotes the projection of the row space of A onto the row space of B (If B is not rank deficient, then $A/B = AB^t (BB^t)^{-1} B$). We will now derive an expression for Z_i and Z_{i+1} in function of the system matrices and the input-output block Hankel matrices. For this we need some extra definitions :

$$\left(\begin{array}{c|c} R & S^t \\ \hline S & P_d \end{array}\right) = \left(\begin{array}{cc|c} R_{11} & R_{12} & S_1^t \\ R_{11}^t & R_{22} & S_2^t \\ \hline S_1 & S_2 & P_d \end{array}\right) = \lim_{j\to\infty} \frac{1}{j} \left(\begin{array}{c} U_p \\ U_f \\ \hline X_p^d \end{array}\right) (U_p^t \; U_f^t \mid (X_p^d)^t)$$

$$= \lim_{j\to\infty} \frac{1}{j} \left(\begin{array}{cc|c} U_p U_p^t & U_p U_f^t & U_p (X_p^d)^t \\ U_f U_p^t & U_f U_f^t & U_f (X_p^d)^t \\ \hline X_p^d U_p^t & X_p^d U_f^t & X_p^d (X_p^d)^t \end{array}\right)$$

under the assumption that the limits exist (which requires quasi-stationarity for the deterministic signal u_k). It is tedious though straightforward to prove the following Theorem which delivers formulas for the linear combinations to be made of the rows of the input-output block Hankel matrices to generate the matrices Z_i and Z_{i+1}. A proof can be found in [65].

Theorem 2 Main projection

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

$$\lim_{j\to\infty} \frac{1}{j} Y_{0|i-1}^s U_.^t = 0 \qquad \lim_{j\to\infty} \frac{1}{j} Y_{0|i-1}^s (X_.^d)^t = 0$$

$$\lim_{j\to\infty} \frac{1}{j} Y_{i|2i-1}^s U_.^t = 0 \qquad \lim_{j\to\infty} \frac{1}{j} Y_{i|2i-1}^s (X_.^d)^t = 0$$

where the subscripts . denotes past or future and if the input satisfies rank $(U_{0|2i-1}) = 2mi$ *and if the stochastic subsystem is not identically zero (the purely deterministic case will be treated in Section 8), then, for $j \to \infty$:*

$$Z_i = \Gamma_i \hat{X}_i + H_i^d U_{i|2i-1} \tag{15}$$

$$Z_{i+1} = \Gamma_{i-1} \hat{X}_{i+1} + H_{i-1}^d U_{i+1|2i-1} \tag{16}$$

with :

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

$$\hat{X}_{i+1} = (A^{i+1} - Q_{i+1}\Gamma_{i+1} \mid \Delta^d_{i+1} - Q_{i+1}\mathrm{H}^{\mathrm{d}}_{i+1} \mid Q_{i+1}) \left(\begin{array}{c} SR^{-1}U_{0|2i-1} \\ \hline U_{0|i} \\ \hline Y_{0|i} \end{array}\right) \tag{18}$$

and $Q_i = \chi_i \psi_i^{-1}$ *in which*

$$\chi_i = A^i (P^d - SR^{-1}S^t)\,\Gamma_i^t + \Delta_i^s \tag{19}$$

$$\psi_i = \Gamma_i (P^d - SR^{-1}S^t)\,\Gamma_i^t + L_i^s \tag{20}$$

Let us give some comments on this Theorem :

— The rank condition on the deterministic input sequence is just the algebraic translation of the fact that the input sequence should be sufficiently wild, i.e. *persistently exciting*. It might possibly be relaxed but in most applications represents no major difficulties. If for instance $U_{0|2i-1}$ would be rank deficient, this would imply that the input is generated by an automous linear time invariant system (as directly follows from realization theory).

— Equations (15) and (16) should be compared to the deterministic input-output equation we have been referring to after Theorem 1. Apart from the fact that Z_i nor Z_{i+1} will be block Hankel, they look exactly as the input-output matrix equations.

— Equations (17) and (18) write the matrices $\hat{X}_i$ and $\hat{X}_{i+1}$ as linear combinations of all inputs and the past outputs. The interpretation of the sequences in these matrices is quite natural : They are the Kalman filter states, i.e. the least squares estimates of the 'real' states, that would be provided by a Kalman filter if the complete system were known. Let us analyze in some more detail the exact interpretation.

It may come as no surprise that there is a connection between the states $\hat{X}_i$ defined by the projection Z_i and some optimal prediction of the ouputs $Y_{i|2i-1}$. To establish this connection, we need one more Theorem that states how the non steady state Kalman filter state estimate $\hat{x}_k$ can be written as a linear combination of $u_0, \ldots, u_{k-1}, y_0, \ldots, y_{k-1}$ and the initial state estimate $\hat{x}_0$.

Theorem 3 Kalman Filter

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

$$\hat{x}_k = A\hat{x}_{k-1} + Bu_{k-1} + K_{k-1}(y_{k-1} - C\hat{x}_{k-1} - Du_{k-1}) \tag{21}$$

$$K_{k-1} = (A P_{k-1} C^t + G)(\Lambda_0 + C P_{k-1} C^t)^t \tag{22}$$

$$P_k = (A P_{k-1} A^t - (A P_{k-1} C^t + G)(\Lambda_0 + C P_{k-1} C^t)^t (A P_{k-1} C^t + G)^t \tag{23}$$

can be written as :

$$\hat{x_k} = (A^k - Q_k \Gamma_k \mid \Delta_k^d - Q_k H_k^d \mid Q_k) \left(\begin{array}{c} \hat{x}_0 \\ \hline u_0 \\ \cdots \\ u_{k-1} \\ \hline y_0 \\ \cdots \\ y_{k-1} \end{array} \right) \tag{24}$$

where :

$$Q_k = \chi_k \psi_\kappa^{-1} \tag{24}$$

$$\chi_k = A^k P_0 \Gamma_k^t + \Delta_k^s \tag{26}$$

$$\psi_k = \Gamma_k P_0 \Gamma_k^t + L_k^s \tag{27}$$

The proof of this Theorem and some details concerning the special form of the Kalman filter equations (21)-(23) can be found in [65]. Let us just indicate that the error covariance matrix $\tilde{P}_k \overset{\text{def}}{=} \mathbf{E}[(x_k - \hat{x}_k)(x_k - \hat{x}_k)^t]$ is given by $P^s + P_k$, with P^s the state covariance matrix (6).

Note that the limiting solution ($k \to \infty$) of (23) is $-P\infty$, where $P\infty$ is the state covariance matrix of the forward innovation model (Faure, 1976). Hence the limiting error covariance is $\tilde{P}\infty = P^s - P\infty$, which is the smallest state error covariance matrix we can obtain (in the sense of nonnegative definiteness). Also note that the expressions for ψ_k and χ_k (26)-(27) are equal to the expressions of ψ_i and χ_i (19)-(20) with $P^d - SR^{-1}S^t$ substituted by P_0.

If we now combine the results of Theorem 2 and 3, we find an interpretation of the sequences $\hat{X}_i$ and $\hat{X}_{i+1}$ in terms of states of a bank of non-steady state Kalman filters, applied in parallel to the data. More specifically, compare formulas (17), (18) and (24) : It can be seen that the j columns of $\hat{X}_i$ are equal to the outputs of a bank of j non-steady state Kalman filters in parallel. The $(p+1)^{th}$ column of $\hat{X}_i$ for instance, is equal to the non-steady state Kalman filter state $\hat{x}_{i+p}$ of the Kalman filter (21), (22), (23), with initial error covariance matrix at starting time p

$$\tilde{P}_p = P_p + P^s = P^d - SR^{-1}S^t + P^s \text{ and } \hat{x}_p = SR^{-1} \left(\begin{array}{c} u_p \\ \cdots \\ u_{p+2i-1} \end{array} \right).$$

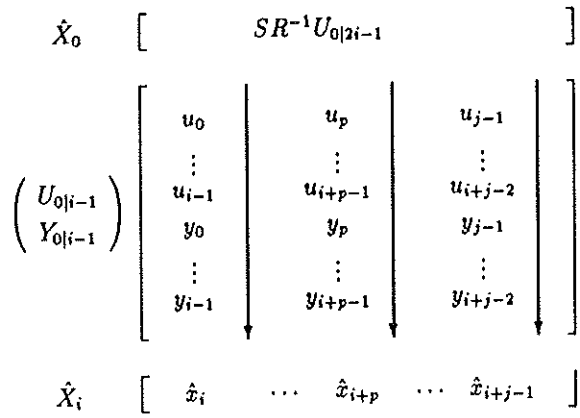

Fig. 3. — Interpretation of the sequence $\hat{X}_i$ as a sequence of non-steady state Kalman filter state estimates based upon i measurements of u_k and y_k.

Notice that $\tilde{P}_p$ is independent of the column index, so it is denoted with $\tilde{P}^0$. In this way, all the columns can be interpreted as Kalman filter states. The initial states of the j filters together can be written as $\hat{X}^0 = SR^{-1}U_{0|2i-1}$. All this is clarified in Figure 3. The expressions for $\tilde{P}^0$ and $\hat{X}^0$ can be interpreted (somewhat loosely) as follows : If we had no information at all about the initial state, then the initial state estimate would be $\hat{X}^0 = 0$ and the initial error covariance would be equal to the expected variance of the state $\tilde{P}^0 = \mathbf{E}[x_k\, x_k^t] = P^d + P^s$. Now, since the inputs are possibly correlated, we can derive information about $\hat{X}^0$ out of the inputs $U_{0|2i-1}$. This is done by projecting the (unknown) exact initial state sequence $X_0^d + X_0^s$ onto the row space of the inputs $U_{0|2i-1}$ as $\hat{X}^0 = (X_0^d + X_0^s)/U_{0|2i-1} = SR^{-1}U_{0|2i-1}$. This extra information on the initial state of the Kalman filter also implies that the error covariance matrix reduces from $P^d + P^s$ to $\tilde{P}^0 = P^d + P^s - \lim_{j\to\infty} \frac{1}{j}\hat{X}^0(\hat{X}^0)^t = P^d + P^s - SR^{-1}S^t$. These are exactly the same expressions for $\hat{X}^0$ and $\tilde{P}^0$ as we found above.

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

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

We define the residuals R_i of the projection as :

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

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

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

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

It is also trivial that :

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

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

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

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

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

6. Identification Scheme

First, the projections Z_i and Z_{i+1} (14) have to be calculated. In Section 7 we will describe a numerically stable way to do this. Let us now assume for the moment that we have computed these matrices Z_i and Z_{i+1}. For convenience, we rewrite these projections as linear combinations of the matrices on which row space we have been projecting on :

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

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

with, from (15)-(18) :

$$L^1_i = \Gamma_i ([A^i - Q_i \Gamma_i] S (R^{-1})_{1|mi} + \Delta^d_i - Q_i H^d_i) \tag{35}$$

$$L^2_i = H^d_i + \Gamma_i [A^i - Q_i \Gamma_i] S (R^{-1})_{mi+1|2mi} \tag{36}$$

$$L^3_i = \Gamma_i Q_i \tag{37}$$

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

6.1. *How to determine* Γ_i *and* n.

An import observation is that the column space of the matrices L^1_i and L^3_i coincides with the column space of Γ_i. The implies that Γ_i and the order of the system n can be determined from the column space of one of these matrices. The basis for this column space actually determines the basis for the states of the final (identified) state space description [6] [7]. We could now determine Γ_i, Γ_{i-1} and the order n by just applying Kung's (approximate) realization idea [39] to the matrices L^1_i or L^3_i. Let us for the moment just assume that we have Γ_i (and hence also Γ_{i-1}) and therefore also its Moore-Penrose pseudo-inverse $\Gamma^\dagger_i$.

6.2. *Determination of the system matrices*

From (15)-(16) it follows that

$$\hat{X}_i = \Gamma^\dagger_i (Z_i - H^d_i U_{i|2i-1}) \tag{38}$$

$$\hat{X}_{i+1} = \Gamma^\dagger_{i-1} (Z_{i-1} - H^d_{i-1} U_{i+1|2i-1}) \tag{39}$$

[6] Let us mention two other possible matrices that have the same column space as Γ_i, namely $L^1_i + L^3_i L^2_i$ and $(L^1_i \mid L^3_i) \begin{pmatrix} U_{0|2i-1} \\ Y_{0|2i-1} \end{pmatrix}$. It should be mentioned that, for $i \to \infty$ the first one will lead to a deterministic sybsystem that is balanced (see [49]) while the second one leads to a deterministic system that is frequency weighted (with the input spectrum) balanced [23] together with a stochastic subsystem of which the forward innovation model is balanced in a deterministic sense. We will not expand any more on this, but keep this for future work.

[7] Actually, since Γ_i is the (extended) observability matrix, one could deduce the structure for a canonical model (such as the Kronecker indices) from it by investigating the linear independence of the rows of the matrices L^1_i and/or L^3_i. In Section 9, we will however argue that this is a bad idea to apply and that it pays to use instead the robust SVD based approach to be explained below.

In these formulas, the only unknowns on the right hand side are the matrices H_i^d and H_{i-1}^d. If we now substitute the expressions for $\hat{X}_i$ and $\hat{X}_{i+1}$ (38)-(39) in the formulas (31)-(32), we find :

$$\left(\frac{\Gamma_{i-1}^{\dagger} Z_{i+1}}{Y_{i|i}}\right) = \underbrace{\left(\frac{A}{C}\right) \Gamma_i^{\dagger} Z_i}_{\text{term 1}} + \underbrace{\left(\frac{K_{12}}{K_{22}}\right) U_{i|2i-1}}_{\text{term 2}} + \underbrace{\begin{pmatrix} U_{0|2i-1} \\ Z_i \\ \hat{X}_i \end{pmatrix}^{\perp}}_{\text{term 3}} \tag{40}$$

where we define :

$$\left(\frac{K_{12}}{K_{22}}\right) \stackrel{\text{def}}{=} \left(\begin{array}{c} B - A\Gamma_i^{\dagger}\begin{pmatrix} D \\ \Gamma_{i-1} B \end{pmatrix} \Gamma_{i-1}^{\dagger} H_{i-1}^d - A\Gamma_i^{\dagger}\begin{pmatrix} 0 \\ H_{i-1}^d \end{pmatrix} \\ \hline D - C\Gamma_i^{\dagger}\begin{pmatrix} D \\ \Gamma_{i-1} B \end{pmatrix} \quad - C\Gamma_i^{\dagger}\begin{pmatrix} 0 \\ H_{i-1}^d \end{pmatrix} \end{array}\right) \tag{41}$$

Observe that the matrices B and D appear linearly in the matrices K_{12} and K_{22}. Let Π be a matrix whose row space coincides with that of $\begin{pmatrix} \Gamma_i^{\dagger} Z_i \\ U_{i|2i-1} \end{pmatrix}$ then (from (40)) :

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

Obviously, this is a set of linear equations in the unknowns A, C, K_{12}, K_{22}. Another interpretation is that one could solve the least squares problem :

$$\min_{A,\, C,\, K_{12},\, K_{22}} \left\| \left(\frac{\Gamma_{i-1}^{\dagger} Z_{i+1}}{Y_{i|i}}\right) - \left(\begin{array}{c|c} A & K_{12} \\ \hline C & K_{22} \end{array}\right) \left(\frac{\Gamma_i^{\dagger} Z_i}{U_{i|2i-1}}\right) \right\|_F^2 \tag{42}$$

This is easy to see from (40), in which the third term would represent the least squares residuals.

Either way, from (40) we find (term by term) :

Term 1 : A and C exactly.

Term 2 : K_{12} and K_{22} from which B and D can be unraveled by solving a set of linear equations, analogous to the one described in [14] [15, p. 289]. Note that in (41), B and D appear linearly. Hence if A, C, Γ_i, Γ_{i-1}, K_{12} and

K_{22} are known, solving for B and D is equivalent with solving a set of linear equations.

Terms 3 : The residuals of the least squares solution (40) can be written as :

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

where $W_{|.}$ and $V_{|.}$ are block Hankel matrices (each of which has only one block row) with as entries the process noise w_k and the measurement noise v_k. The covariance matrices R_s, S_s and Q_s are determined approximately from ρ as follows :

$$\frac{1}{j}(\rho\rho^t) \simeq \begin{pmatrix} Q_s & S_s \\ S_s^t & R_s \end{pmatrix} \tag{43}$$

The approximation is due to the fact that the bank of Kalman filters for finite i is not in steady state (a phenomenon which is analyzed in detail in [64]). As i grows larger, the approximation error grows smaller. For infinite i the stochastic subsystem is determined exactly (unbiased).

6.3. *Algorithm*

In this section, we summarize the algorithm step by step, not yet paying attention to the fine numerical details, which will be treated in Section 7. The reader should now also return to the table we have presented in Section 1, in which the basic steps of our subspace approach are summarized.

Step 1 : Determine the projections :

$$Z_i = Y_{i|2i-1} / \begin{pmatrix} U_{0|i-1} \\ U_{i|2i-1} \\ Y_{0|i-1} \end{pmatrix} = \begin{pmatrix} \underbrace{L_i^1}_{li \times mi} & \underbrace{L_i^2}_{li \times mi} & \underbrace{L_i^3}_{li \times li} \end{pmatrix} \begin{pmatrix} U_{0|i-1} \\ U_{i|2i-1} \\ Y_{0|i-1} \end{pmatrix}$$

and

$$Z_{i+1} = Y_{i+1|2i-1} / \begin{pmatrix} U_{0|i} \\ U_{i+1|2i-1} \\ Y_{0|i} \end{pmatrix}$$

Step 2 : Calculate the Singular Value Decomposition

$$(L_i^1 \mid L_i^3) \begin{pmatrix} U_{0|i-1} \\ Y_{0|i-1} \end{pmatrix} = (U_1 \; U_2) \begin{pmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{pmatrix} V^t$$

The order is equal to the number of non-zero singular values. Set $\Gamma_i = U_1\Sigma_1^{1/2}$ and $\Gamma_{i-1} = \underline{U_1}\Sigma_1^{1/2}$. ($\underline{U_1}$ means omitting the last block row).

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

$$\begin{matrix} j \\ n \\ l \end{matrix}\left(\frac{\Gamma_{i-1}^{\dagger}Z_{i+1}}{Y_{i|i}}\right) = \begin{matrix} n & mi \\ l & \end{matrix}\left(\begin{array}{cc} K_{11} & K_{12} \\ \hline K_{21} & K_{22} \end{array}\right) \cdot \begin{matrix} j \\ mi \end{matrix}\left(\frac{\Gamma_i^{\dagger}Z_i}{U_{i|2i-1}}\right) + \begin{matrix} j \\ n \\ l \end{matrix}\left(\frac{\rho_1}{\rho_2}\right)$$

The system matrices are determined as follows :

— $A \leftarrow K_{11}$ and $C \leftarrow K_{21}$.

— The matrices B and D follow from A, C, and K_{11}, K_{21} via a set of linear equations.

$$- \left(\begin{array}{c|c} Q_s & S_s \\ \hline S_s^t & R_s \end{array}\right) \leftarrow \frac{1}{j}\left(\begin{array}{c|c} \rho_1\rho_1^t & \rho_1\rho_2^t \\ \hline \rho_2\rho_1^t & \rho_2\rho_2^t \end{array}\right).$$

Observe that the deterministic subsystem will be identified exactly (as $j \to \infty$, independent of i). The approximation of the stochastic subsystem is still dependent on i and converges as $i \to \infty$ (see [64] for a detailed explanation).

7. A Numerically Stable and Efficient Implementation

Let's know translate the algorithm step by step in a numerically robust procedure.

Step 1 : Projection step :

Construct the block Hankel matrix

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

Calculate the R factor of its RQ factorization

$$H = \underbrace{R}_{2(m+l)i \times 2(m+l)i} \cdot \underbrace{Q^t}_{2(m+l)i \times j} \tag{44}$$

with $Q'Q = I$ and R lower triangular. Partition this factorization as :

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

We use the shorthand notation $R_{4:6,1:3}$ for the submatrix of R consisting of block rows 4 to 6 and block columns 1 to 3. The projections are now given by

$$Z_i = R_{5:6,1:4}\, R^{-1}_{1:4,1:4} \begin{pmatrix} U_{0|2i-1} \\ Y_{0|i-1} \end{pmatrix} = (L_i^1 \mid L_i^2 \mid L_i^3) \left(\begin{array}{c} U_{0|i-1} \\ \hline U_{i|2i-1} \\ \hline Y_{0|i-1} \end{array}\right)$$

$$Z_{i+1} = R_{6:6,1:5}\, R^{-1}_{1:5,1:5} \begin{pmatrix} U_{0|2i-1} \\ Y_{0|i} \end{pmatrix} = (L_{i+1}^1 \mid L_{i+1}^2 \mid L_{i+1}^3) \left(\begin{array}{c} U_{0|i} \\ \hline U_{i+1|2i-1} \\ \hline Y_{0|i} \end{array}\right)$$

Step 2 : Determination of Γ_i and n :

Determine Γ_i and n through the SVD :

$$(L_i^1 \; 0 \; L_i^3)\, R_{1:4,1:4} Q_{1:4}' = (U_1 \; U_2) \begin{pmatrix} \Sigma_1 & 0 \\ 0 & \Sigma_2 \end{pmatrix} (Q_{1:4} V)'$$

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

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

where the underbar means deleting the last l rows (l is the number of outputs).

Step 3 : Least squares for the system matrices :

We find for the left hand side and right hand side of equation (40), written as a function of the original Q matrix :

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

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

The least squares problem (42) can be rewritten as

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

and solved in a least squares sense for K, which contains A, C and K_{12} and K_{22} as shown before.

Remarks :

- It can be verified that we do not need the factor Q explicity. We have included it in the description of the algorithm for clarity, but it can be verified that it is not required in the SVD nor in the least squares problem. This implies a substantial reduction in memory requirements and computational complexity. Moreover, it opens the door for recursive updating since we only need to update the R-factor.
- Efficient and robust algorithms for the QR-decomposition and the SVD are described in detail in [29] and are readily available in many software packages for numerical linear algebra such as LAPACK and MATLAB.

8. Special cases, historical review and relation with other approaches

In this section, we discuss several special cases of the above framework, without going too much into mathematical detail (for which we refer to the bibliography). Basically, the subspace approach we have been describing, originates in 4 historical traces : Deterministic realization theory, deterministic identification and stochastic realization. The 4-th path comes from numerical linear algebra, which has delivered us tools such as (generalized) singular value decompositions, the QR-decomposition and angles between subspaces.

While there is an almost infinite number of publications on system identification, we only discuss here the work that has been directly relevant for our subspace approach.

8.1. *Deterministic realization* : $u_k = 0$, $v_k = 0$, $w_k = 0$

If all the inputs (deterministic as well as stochastic) to the system are zero, the state space equations reduce to $x_{k+1} = Ax_k$ and $y_k = Cx_k$ which can be solved as $y_k = CA^{k-1}x_0$. The problem of identifying the matrices A and C and the initial state x_0 is closely related to the so-called (deterministic) realization problem, which is the problem of finding the state space matrices from a given multivariable impulse response $H_0 = D$, $H_k = CA^{k-1}B$, $k \geq 1$. It can now be seen that the identification of A, C and x_0 corresponds to a realization problem in which we consider the outputs y_k to be the impulse response of a system with l outputs and 1 input, with state space matrices A, x_0 and C. Realization theory was initiated in the sixties by the investigations in [34] [58] [21] (among others) and has certainly been one of the driving forces to put more emphasis on state space models. The SVD as a numerical tool to solve the realization problem is introduced in [39] and [73]. These papers have generated a whole industry on model reduction techniques. Among all these techniques however, the balanced realization [53] occupies a privileged position. Not only does it provide a useful model reduction technique based on singular values, but it is also essential in optimally Hankel norm based model reduction [28] and in obtaining realizations of digital filters that are least sensitive to round off errors [54]. As we have observed elsewhere in this paper, our subspace algorithms deliver state space models that are balanced.

8.2. *Deterministic identification* : $v_k = 0$, $w_k = 0$

For the identification of deterministic systems, an early reference is the work by Gopinath [30] in which he describes a direct deterministic identification method that starts from input-output block Hankel matrices and also exploits the input-output matrix equation from Theorem 1. Although it provided some inspiration for the present work, the approach described there does not use numerically reliable techniques, but instead applies the concept of a *selector* matrix which is very much reminiscent of determining the structure of canonically parameterized models (and hence ill-conditioned). Related results are reported in [12]. This paper also mentions the fact that the rank of the block Hankel matrix with inputs and outputs is equal to the sum of the rank of the block Hankel with inputs and the order of the system to be identified :

$$\text{rank} \begin{pmatrix} Y^d \\ U^d \end{pmatrix} = \text{rank } (U^d) + n \tag{45}$$

The input-output matrix equation is also the starting point in [42]. This work then proceeds by finding a minimal dimension state space realization from

input-output block Hankel matrices and it is observed that one can identify systems that are observable but not necessarily controllable (contrary to transfer matrix based approaches and impulse response realization). Again (sequential) selector matrices are used.

An important step forward came with the realization in [15] that the rank deficiency result (45) can be used to show, via Grassman's famous dimension theorem, that there must exist a non-trivial intersection between the row spaces of the past and the future. Indeed, one has ('dim' stands for 'dimension of the row space of', $\cap$ means 'intersection between row spaces') :

$$\dim\left[\begin{pmatrix} Y_p^d \\ U_p^d \end{pmatrix} \cap \begin{pmatrix} Y_f^d \\ U_f^d \end{pmatrix}\right] = \operatorname{rank}\begin{pmatrix} Y_p^d \\ U_p^d \end{pmatrix} + \operatorname{rank}\begin{pmatrix} Y_f^d \\ U_f^d \end{pmatrix} - \operatorname{rank}\begin{pmatrix} Y_p^d \\ U_p^d \\ Y_f^d \\ U_f^d \end{pmatrix}$$

$$= (mi + n) + (mi + n) - (2mi + n) = n$$

in which we have assumed for simplicity that the input sequence is persistent excitant. Not only is the intersection n-dimensional, but the intersection itself provides a valid state sequence, which can be obtained by choosing any basis for the intersection. From the states, the inputs and the outputs it is straightforward to calculate the system matrices. A similar result was obtained in [72], where the dimension of the intersection is called the *relative row rank*.

Situating this result in our general framework described above, we can state that the orthogonal projection of the first step and the order decision of the second step becomes here the calculation of the basis of an intersection of the row spaces. Algorithms are described in [15] [46].

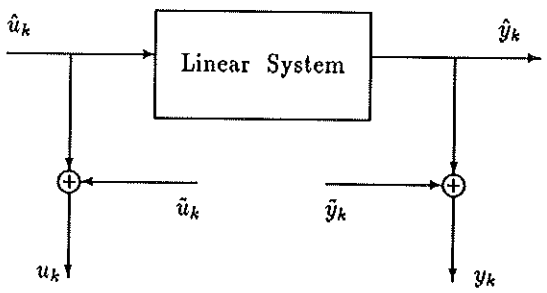

Fig. 4. — Errors-in-variables configuration : The vector signals $\hat{u}_k$ and $\hat{y}_k$ are the exact but unknown inputs and outputs. The measurements are u_k and y_k, which are corrupted version of the exact signals. The measurement noises $\tilde{u}_k$ and $\tilde{y}_k$ are assumed to be additive with known covariance matrices.

8.3. *Measurement noise : $w_k = 0$ and/or noisy inputs*

Suppose we have an identification problem as in Figure 4. A nice result is that the intersection algorithms mentioned in the previous subsection will deliver consistent estimates of the system matrices if not only the outputs, but also the inputs are corrupted by white measurement noise. In this case, there is no exact intersection between past and future, but one can calculate approximate intersections, as explained in [15] [46]. These approximate intersections could be considered as minimum variance estimates of the state (for an explanation see [18]). The cases of colored noise are treated in [15] [47] [59].

8.4. *Stochastic realization : $u_k = 0$*

When there are no deterministic inputs, the problem of identifying the matrices A and C and the covariance matrices R_s, Q_s and S_s is called the stochastic realization or identification problem. The corresponding model is called ARMA (auto-regressive moving average) and is often used to model stochastic disturbances. Of all the 'subspace' problems, this one has been analyzed most frequently in the literature, starting with the work [25] [8]. An impressive survey is provided in [13]. Another reference in which ARMA models are treated is [4]. Here we will point out some references that were influential for our work, for which the specific results on ARMA processes are summarized in [64].

Early algorithms for stochastic realization reduce the problem to one of deterministic realization. How this can be done is easy to see from the formulas of the covariances matrices (7). Obviously, they can be interpreted as the Markov parameters of a linear deterministic system with system matrices (A, G, C) and hence, the sequence can be realized via deterministic realization theory. Such an approach is mainly followed in [4] to which we also refer for further references.

Faure [25] pointed out that the non-uniqueness issues involved in the state space model for a stochastic process are much more involved than in the deterministic case. Indeed, there is a whole set of stochastic realizations of a given stochastic covariance sequence. This set is described by all solutions to the so-called *positive real lemma.*

An important contribution is the work of Akaike [1] [2]. Akaike really showed that for a stochastic process the projection of the future on the past could be obtained from the technique of canonical correlations. These canonical

[8] There are many earlier references that treat ARMA models. However, the paper [25] has been quite influential because it brought things as Riccati equations and the positive real lemma in the realm of system identification.

correlations are the cosines of the principal angles between subspaces as they are called in the numerical analysis literature [29]. These were discovered already in 1875 by Jordan [36] and embedded in a statistical framework by Hotelling [35]. These angles between subspaces are the multi-dimensional generalization of the angle between two vectors. If one considers the row spaces of past and future block Hankel matrices with the outputs, there will be exactly n principal angles unequal to 90°, all other ones being orthogonal [9]. Due to stationarity, one can equally well project the past onto the future, a projection which is also provided via the principal angle technique. If both $j \to \infty$ and $i \to \infty$, the projection of the future on the past will deliver a 'forward innovation model' while the projection of the past on the future gives the 'backward innovation model'. Both can be interpreted in terms of Kalman filters, which provided us with the main inspiration to look for Kalman filters states obtained as an orthogonal projection in the general case. For each of the Kalman filters, the forward and the backward one, there is an algebraic Riccati equation. Desai and Pal [19] showed that one can choose a contragredient transformation on the state space model, which is such that the solutions to the two Riccati equations are diagonal and equal, containing on their diagonal precisely the aforementioned canonical correlations. This delivers models that are balanced in a stochastic sense.

Our recent contribution to the ARMA identification problem (the structure of which is by now very well understood) is reported in [64]. There, we present 'square root' algorithms which work with the measured data instead of having to estimate first the covariance matrices from time averages. We also discuss what happens if only the number of columns $j \to \infty$ while the number of rows i is kept constant. In this case, the orthogonal projection of the future on the past and the other way around still deliver Kalman filter states, but of filters that are not yet in steady state (which they would reach if $j \to \infty$). The two equations that describe this behavior are now backward and forward Riccati *difference* equations. Because the noise covariance matrices as estimated via (43) are guaranteed to be positive definite, we have avoided the problem of positive realness, at least heuristically. Finally, it is shown that one of the recently introduced generalizations of the singular value decomposition (the QSVD, see [17] for references) allows for a numerically robust implementation. This technique allows us also to deal with ARMA models that are singular, i.e. have poles on the unit circle which happens a lot in practical applications, when there are for instance periodic disturbances (so there is no special need to consider things like ARIMA processes as e.g. in [10]).

[9] For some processes there will be angles equal to zero (which gives an exact intersection between past and future). Their number is equal to the number of zeros on the unit circle of the determinant of the transfer matrix, see [33, p. 84].

8.5. *Related works*

There are at least two other related approaches that seem worth mentioning here. First we have the work of Larimore [40] [41]. The basic tool there is the idea of trying to find principal angles and directions between subspaces that are obtained by orthogonalizing the row space of future outputs and the row space of past outputs and inputs onto the future inputs. While his reports indicate that on practical examples, good results are obtained, there is a consistent lack of theoretical evidence in his work showing why exactly these principal directions as described, would be a good idea to rely upon. Nevertheless, Larimore's framework can be tackled with our approach (see e.g. [65]) and there are good indications that one could demonstrate rigorously that it delivers the approximate identification scheme which is derived in [65].

A second approach is the one described in [67], which is basically an instrumental variable method, which consists of projecting away the stochastic part. The idea is then to first identify the deterministic subsystem and then perform stochastic realization on the residuals. There are many objections one could formulate to this approach : In one version of it, it is required to estimate the Markov parameters from inputs-outputs, which is well known to be an ill-conditioned problem [8]. Moreover, one needs to find suitable 'instruments', which are uncorrelated with the stochastic part and then show that these are effective in removing the stochastic contributions. Because this is a multi-stage approach, the analysis is very involved and propagation of errors from one stage to another is unavoidable.

8.6. *An approximate identification scheme*

It is possible to introduce simplicifations in the previously presented algorithm in the sense that one can trade-off accuracy of the model for (still a further) reduction in the computational complexity.

Let's have a look again at the Kalman filter interpretation, more specifically at formulas (31)-(32). Unfortunately, it is not possible to separate the effect of the input $H_i^d U_{i|2i-1}$ from the effect of the state $\Gamma_i \hat{X}_i$ in formula (15), by just dropping the term with the linear combinations of $U_{i|2i-1}$ (L_i^2) in the expression for Z_i. We would automatically drop a part of the initial state if we did this (see for instance (36)). So, it is not possible to obtain an explicit expression for $\Gamma_i \hat{X}_i$ and $\Gamma_{i-1} \hat{X}_{i+1}$, without knowledge of H_i^d, which would require knowledge of the system matrices [10].

[10] This is by the way the major difficulty that the earlier versions of subspace methods were struggling with. Instrumental variables methods to circumvent the problem were suggested in [62] [67] [68], but they tend to be complicated. The complete analysis of the difficulty and the correct complete solution was presented in [65], where also the approximate solution scheme we discuss here, is treated in more detail.

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

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

$$\Gamma_i\tilde{X}_i = Z_i - L_i^2 U_{i|2i-1} \tag{46}$$

$$\Gamma_{i-1}\tilde{X}_{i+1} = Z_{i+1} - L_{i+1}^2 U_{i+1|2i-1} \tag{47}$$

From (33) and (34) we derive :

$$\tilde{X}_i = ([A^i - Q_i\Gamma_i] S (R^{-1})_{1|mi} + \triangle_i^d - Q_{i+1}H_i^d \mid Q_i) \begin{pmatrix} U_{0|i-1} \\ Y_{0|i-1} \end{pmatrix} \tag{48}$$

$$\tilde{X}_{i+1} = ([A^{i+1} - Q_{i+1}\Gamma_{i+1}] S (R^{-1})_{1|m(i+1)} + \triangle_{i+1}^d - Q_{i+1}H_{i+1}^d \mid Q_{i+1}) \begin{pmatrix} U_{0|i} \\ Y_{0|i} \end{pmatrix}$$

The matrices $\Gamma_i\tilde{X}_i$ and $\Gamma_{i-1}\tilde{X}_{i+1}$ can be obtained directly from the data without any knowledge of the system matrices, and can be interpreted as oblique projections as described in [65].

The state sequence $\tilde{X}_i$ is generated by a bank of non-steady state Kalman filters with : $\tilde{P}^0 = P^d - SR^{-1}S^t + P^s$ and $\tilde{X}_0 = S (R^{-1})_{1|mi} U_{0|i-1}$. The state sequence $\tilde{X}_{i+1}$ on the other hand, is generated by a bank of non-steady Kalman filters, with : $\tilde{P}^0 = P^d - SR^{-1}S^t + P^s$ and $\tilde{X}_0 = S (R^{-1})_{1|m(i+1)} U_{0|i}$. So clearly, both sequences do not belong to the same bank of Kalman filters, and the useful formulas (31)-(32) are not valid for these 2 sequences.

Still, we will see below, that $\tilde{X}_i$ and $\tilde{X}_{i+1}$ are very close to $\hat{X}_i$ and $\hat{X}_{i+1}$, and that $\hat{X}_i = \tilde{X}_i$, $\hat{X}_{i+1} = \tilde{X}_{i+1}$ if at least one of the following conditions is satisfied : $i \to \infty$, the deterministic input u_k of the combined deterministic-stochastic system is white noise or the system is purely deterministic. For these three special cases, we can analyze the difference between $\hat{X}_i$ (17) and $\tilde{X}_i$ (48), defined as δX_i :

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

Case 1 : $i \to \infty$:

Define the term between square brackets in (49) as $P_i \stackrel{\text{def}}{=} A^i - Q_i\Gamma_i$. Now, it is easy to prove [65] that :

$$P_i = \prod_{k=0}^{i-1} (A - K_k C) \tag{50}$$

Since the non-steady state Kalman filter converges to a stable closed loop system (Anderson & Moore, 1979), we find that P_i grows smaller when i grows larger. It is also clear that : $\lim_{i\to\infty} P_i = 0$. In the limit, equation (49) thus becomes (with $S(R^{-1})_{mi+1|2mi} U_{i|2i-1}$ finite) :

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

The same holds for δX_{i+1}. So, we can conclude that, for $i \to \infty$, there is no difference between the state sequence $\hat{X}_i$ and $\tilde{X}_i$. Actually, when $i \to \infty$, the non-steady state Kalman filter bank converges to a steady state Kalman filter bank. In practice, it turns out that i does not have to be that large. The identification results are already very good for reasonably small i ($\simeq 10$).

Case 2 : u_k white noise

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

Case 3 : Pure deterministic system ($v_k = 0$, $w_k = 0$).

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

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

which implies that

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

This rank deficieny implies that for purely deterministic systems, the proof of Theorem 2 of [65] breaks down. The following Theorem is an alternative for Theorem 2 for purely deterministic systems :

Theorem 4 Purely deterministic systems

If the input is persistently exciting and the stochastic subsystem is zero we have :

$$\Gamma_i \tilde{X}_i = (\Gamma_i [\Delta_i^d - A^i \Gamma_i^\dagger H_i^d] \mid \Gamma_i A^i \Gamma_i^\dagger) \begin{pmatrix} U_{0|i-1} \\ Y_{0|i-1} \end{pmatrix} = \Gamma_i X_i = \Gamma_i \hat{X}_i$$

A proof can be found in [65]. This implies that for deterministic systems, δX_i is also equal to zero.

The advantage of working with the approximate state sequence instead of the 'exact' one, lies in the fact that we can simplify the least squares step (42), in which the matrices K_{12} and K_{22} (linear matrix functions of B and D) can now directly be replaced by B and D themselves.

We know from (31) and (32) that the least squares solution L of :

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

is equal to :

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

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

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

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

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

we have :

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

Contrary to $\hat{X}_i$, $\tilde{X}_i$ can be calculated directly from the data, without any knowledge of the system matrices.

The solution is exact in the special cases described above. It is in general possible to determine an estimate of the bias $L - \tilde{L}$ (see [65]).

The numerical implementation stays basically the same. For details, we refer again to [65].

9. Why would subspace methods be any better than other identification algorithms ?

In this section, we compare the 'classical' system identification approaches, as described in [33] [43] [55] [57] with the subspace approach. A first noteworthy difference is illustrated in Figure 5 : Subspace methods basically first look for a state sequence and after that, determine the system matrices from a least squares problem. Classical identification methods first determine the model, after which one could set up a Kalman filter to find the state estimates. One of our contributions is to show that the Kalman filter state can be obtained explicitly and directly from the input-output data. A second important dif-

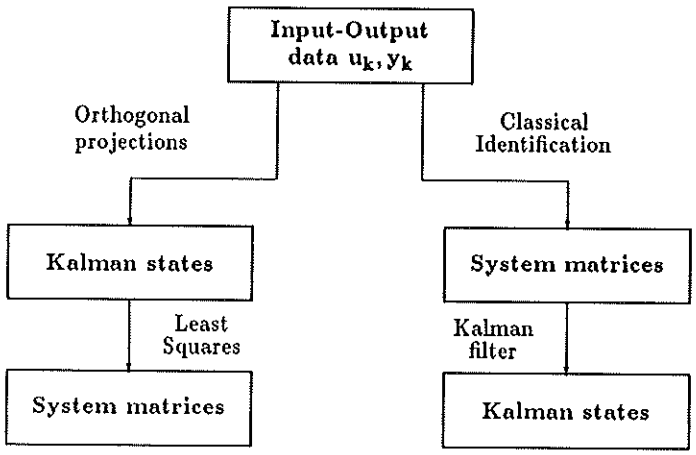

Fig. 5. — The left hand side shows the subspace approach : first the (Kalman) states, then the system matrices. The right hand side is the classical approach : first the system, and then an estimate of the states.

ference is the fact that in our approach, there is no need for parameterizations. Statisticians would argue that the number of parameters to be estimated in our 'full' state space models is extremely high (for instance, for 3 inputs, 4 outputs and 10 states, there are 338 numbers in the matrices A, B, C, D, Q_s, R_s, S_s). This 'curse of dimensionality' (as statisticians call it) is to be avoided. Therefore, the last 20 years or so quite some effort has been spent to find so-called canonical models, which have a minimal number of parameters. The analysis of parameterizations of multivariable state space models is a notoriously difficult problem in system identification. The first canonical realizations for multi-output systems were proposed in [45]. Further elaborations can be found in [22] [27] [31] [32] (see also [33] [38] [43] for more references). The root of the problem is that there is no single, smooth canonical parameterization of a multi-output system. Instead one has to work with a large number of different possible parameterizations, corresponding to different values of observability indices (which are of course unknown). There are two major difficulties here: First of all, since all data are noisy, it is very difficult to decide which rows of the observability matrix are linearly independent (generically, with noisy data, the first rows are all linearly independent, said in other words, the determination of the structure indices from noisy data is an ill-conditioned problem). Another cause of major difficulty in classical identification algorithms, is the necessity to monitor the conditioning of a certain parameterization and decide to take another one if that conditioning becomes too bad. Therefore, overlapping parameterizations have been conceived [27] [61] (see [33] for a complete survey). Subspace algorithms are in this respect much more elegant since the 'best' subspace is automatically determined by the SVD, without the need for estimating the structure indices first. It should also be observed that statistical *consistency* is independent of the parameterization, a fact that can clearly be analyzed for the subspace approach too since the variance on the results decreases proportionally to $1/j$ as $j \to \infty$. As to the statistical efficiency of the subspace method, much work remains to be done. A major source of inspiration might be recent statistical work on subspace methods for the direction of arrival problem (e.g. [56] [70]).

A third strong point of subspace methods versus classical identification techniques lies in the use of numerically robust techniques, such as the QR-decomposition, angles between subspaces and the singular value decomposition. These techniques have already a long history (the SVD was discovered by Beltrami [9], Jordan [36] and Sylvester [60], generalizations are described in [17] and references in there), angles between subspaces by Jordan [37], but it is only since the advent of numerically reliable algorithms to calculate these decompositions (see [29] for a complete survey and many references) that they have been increasingly used in scientific computing. Nevertheless, their advantages have been largely ignored in the system identification community, a mistake we hope to have corrected. Contrary to popular belief, our methods

are on the average cheaper to use (in terms of floating point operations) than e.g. the prediction error methods described in [43] for the same level of accuracy (as e.g. measured by the prediction error) (see [65] for details and numerical comparisons). In addition, since we managed to show that the Q-factor in the QR decomposition of the block Hankel matrix (44) is redundant, we only need to operate on matrices that are relatively small. Therefore, we could easily find recursive updating techniques, with potential implementation on parallel processors, as described in e.g. [48]. Moreover, studies are being undertaken to exploit the special (block) Hankel structure via the concept of displacement rank [71].

Let us also point out that some of the ideas in this work have been extended to descriptor (singular) systems (see [52]) and also direct identification of continuous time systems (see [50]).

10. Conclusions

The fact that we have surveyed in some detail the achievements of subspace methods, does not imply that there is no need for further research. Indeed, much work remains to be done to put these algorithms on a rigorous statistical footing, to analyze the properties when the data are not generated by a system that belongs to the defined model class (in which case there are two types of error, a 'deterministic' one, which is called the bias, and a 'stochastic' one, which is called the variance), to provide 'hard' bounds that could be applied in robust control system design, to analyze the behavior in closed-loop circumstances, etc ...

Nevertheless, we are convinced of the power and elegance of subspace algorithms, a conviction which is confirmed by the recent development of a System Identification Toolbox, using the CACSD software tool Xmath, which contains most of the ideas described in this work (see [3]).

Using this tool, practical applications of subspace algorithms are being tested out on a number of industrial processes (see e.g. [66]).

Let us conclude with two quotations from some recent surveys. In the recently held Workshop on Future Directions in Circuits and Systems [26], it is emphasized that ... *these matrix-based signal processing algorithms are becoming increasingly important ... and need to be blended with traditional algorithms in a compatible and complementary way.*

Ljung, one of the international experts in system identification, in his recent survey on Issues in System Identification [44], points out that it remains to be *established what these signal subspace methods have to offer and how they compare to conventional approaches* ... We hope that with this work we have bridged a little bit of this gap.

11. Acknowledgment

First of all, I would like to thank Peter Van Overschee who in the past two years has been working hard to obtain many of the results described in this work.

I would also like to mention Joos Vandewalle, who introduced me to the subject a decade ago and all the PhD and Master students that have collaborated with me on this subject. Furthermore I would like to thank Thomas Kailath and Gene Golub, both from Stanford University and Michel Gevers from the Université Catholique de Louvain, for their continuing support.

Finally, my gratitude goes to Hilde Devoghel and Thomas De Moor for numerous comments and valuable suggestions.

REFERENCES

[1] Akaike H., *Stochastic theory of minimal realization.* IEEE Transactions on Automatic Control, 19, p. 667-674, 1974.

[2] Akaike H., *Markovian representation of stochastic process by canonical variables.* Siam J. Control, Vol. 13, no. 1, p. 162-173, 1975.

[3] Aling H., Milletti U., Kosut R. L., Mesaros M. P., Van Overschee P., De Moor B., *An interactive system identification module for Xmath.* ESAT-SISTA Report 1992-38, Department of Electrical Engineering, Katholieke Universiteit Leuven.

[4] Aoki M., *State space modeling of time series.* Springer Verlag, Berlin, 1987.

[5] Arun K. S., Kung S. Y., *Balanced approximation of stochastic systems.* SIAM J. Matrix Analysis and Applications, 11, p. 42-68, 1990.

[6] Aström K., Eykhoff P., *System identification — A survey.* Automatica, Vol. 7, p. 123-167, 1971.

[7] Aström K., Wittenmark B., *Computer Controlled Systems : Theory and Design.* Prentice Hall, 1984.

[8] Audley D. R., David A. L., *Ill-posed and well-posed problems in system identification.* IEEE Transactions on Automatic Control, Vol. AC-19, no. 6, December 1974, p. 738-747.

[9] Beltrami E., *Sulle Funzioni Bilineari,* Giornale di Mathematiche, Battagline G., Fergola E. (editors), Vol. 11, p. 98-106, 1873.

[10] Box G. E., Jenkins G. M., *Time series analysis, forecasting and control.* Revised edition, Holden-Day series in time series analysis and digital processing, Holden-Day, Oackland, 1976.

[11] Boyd S., Barrat C., *Linear controller design : Limits of performance.* Prentice Hall Information and System Sciences Series, Englewood Cliffs, NJ, 1991.

[12] Budin M., *Minimal realization of discrete linear systems from input-output observations.* IEEE Transactions on Automatic Control, Vol. AC-16, no. 5, October 1971, p. 395-401.

[13] CAINES P., *Linear Stochastic Systems.* Wiley Series in Probability and Mathematical Statistics, 1988.

[14] DE MOOR B., VANDEWALLE J., *A geometrical strategy for the identification of state space models of linear multivariable systems with singular value decomposition.* Proc. of the 3rd International Symposium on Applications of Multivariable System Techniques, p. 59-69, Plymouth, UK, April 13-15 1987.

[15] DE MOOR B., *Mathematical concepts and techniques for modelling of static and dynamic systems.* PhD thesis, Department of Electrical Engineering, Katholieke Universiteit Leuven, Belgium, June 1988.

[16] DE MOOR B., VANDEWALLE J., MOONEN M., VANDENBERGHE L., VAN MIEGHEM P., *A geometrical approach for the identification of state space models with singular value decomposition.* 1988 Symposium on Identification and System Parameter Estimation, 27-31 August 1988, Beijing China, p. 700-704.

[17] DE MOOR B., ZHA H., *A tree of generalizations of the ordinary singular value decomposition.* Linear Algebra and Applications, special issue on Canonical Forms of Matrices, Vol. 147, p. 469-500, January 1991.

[18] DE MOOR B., *Long and short spaces of noisy matrices and the singular value decomposition.* Accepted for publication in the IEEE Transaction on Signal Processing, 1993.

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

[20] DESAI U. B., KIRKPATRICK R. D., PAL D., *A realization approach to stochastic model reduction.*, International Journal of Control, Vol. 42, no. 4, p. 821-838, 1985.

[21] DICKINSON B., MORF M., KAILATH T., *A minimal realization algorithm for matrix sequences.* IEEE Transactions on Automatic Control, Vol. AC-19, no. 1, February 1974, p. 31-38.

[22] DICKINSON B., KAILATH T., MORF M., *Canonical Matrix fraction and state space descriptions for deterministic and stochastic linear systems.* IEEE Transactions on Automatic Control, Vol. AC-19, p. 656-667, 1974.

[23] ENNS D., *Model reduction for control system design.* Ph. D. dissertation, Dep. Aeronaut. Astronaut., Stanford University, Stanford CA, 1984.

[24] EYKHOFF P., *System identification.* Wiley, London, 1974.

[25] FAURE P., *Stochastic realization algorithms.* in 'System Identification : Advances and case studies'. Eds. : Mehra R., Lainiotis D., Academic Press, 1976.

[26] Final report of the CAS Workshop on Future Directions in Circuits, Systems and Signal Processing, IEEE Circuits and Systems Society Newsletter, Vol. 1, No. 3, September 1990, p. 14-17.

[27] GLOVER K., WILLEMS J., *Parametrizations of linear dynamical systems : canonical forms and identifiability.* IEEE Transactions on Automatic Control, Vol. AC-19, p. 640-645.

[28] GLOVER K., *All optimal Hankel norm approximations of linear multivariable systems and their L^∞ error bounds.* International Journal of Control, 39, p. 1115-1193, 1984.

[29] GOLUB G., VAN LOAN C., *Matrix computations.*, Johns Hopkins University Press, Baltimore, Maryland, 1984.

[30] GOPINATH B., *On the identification of linear time-invariant systems from input-output data* The Bell System Technical Journal, Vol. 48, May-June 1969, no. 5, p. 1101-1113.

[31] GUIDORZI R., *Canonical structures in the identification of multivariable systems.* Automatica, 11, p. 361-374, 1975.

[32] GUIDORZI R., *Invariants and canonical forms for systems structural and parametric identification.* Automatica, 17, p. 177-133, 1981.

[33] HANNAN E. J., DEISTLER M., *The statistical theory of linear systems.* Wiley Series in Probability and Mathematical Statistics, John Wiley and Sons, New York, 1988.

[34] HO B. L., KALMAN R. E., *Efficient construction of linear state variable models from input/output functions.* Regelungstechnik, 14, p. 545-548, 1966.

[35] HOTELLING H., *Relation between two sets of variates.* Biometrika, Vol. 28, p. 321-372, 1936.

[36] JORDAN C., *Mémoire sur les formes bilinéaires.* J. Math. Pures Appl. II, Vol. 19, p. 35-54, 1874.

[37] JORDAN C., *Essai sur la géometrie à n dimensions.* Bull. Soc. Math. France, 3, 1875, p. 103-174.

[38] KAILATH T., *Linear Systems.* Prentice Hall, Englewood Cliffs, New Jersey, 1980.

[39] KUNG S. Y., *A new identification method and model reduction algorithm via singular value decomposition.* 12th Asilomar Conf. on Circuits, Systems and Comp., p. 705-714, Asilomar, CA, Nov. 1978.

[40] LARIMORE W., *System identification, reduced order filtering and modeling via canonical variate analysis.* Proc. of the American Control Conference, ACC June 1983, San Francisco.

[41] LARIMORE W., *Canonical variate analysis in identification, filtering and adaptive control.* Proc. 29th Conference on Decision and Control, Hawai, December 1990, p. 596-604.

[42] LIU R., SUEN L. C., *Minimal dimension realization and identifiability of input-output sequences.* IEEE Transactions on Automatic Control, April 1977, Vol. AC-22, p. 227-232.

[43] LJUNG L., *System identification — Theory for the User.* Prentice Hall, Englewood Cliffs, NJ, 1987.

[44] LJUNG L., *Issues in System Identification.* IEEE Control System, Vol. 11, No. 1, January 1991, p. 25-29.

[45] LUENBERGER D. G., *Canonical froms for linear multivariable systems.* IEEE Transactions on Automatic Control, Vol. AC-12 : 290, 1967.

[46] MOONEN M., DE MOOR B., VANDENBERGHE L., VANDEWALLE J., *On and off-line identification of linear state space models.* International Journal of Control, Vol. 49, No. 1, p. 219-232, 1989.

[47] MOONEN M., VANDEWALLE J., *A QSVD approach to on- and off-line state space identification.* International Journal of Control, Vol. 51, No. 5., p. 1133-1146, 1990.

[48] MOONEN M., *Jacobi-type updating algorithms for signal processing, systems identification and control.* PhD Thesis, Department of Electrical Engineering, Katholieke Universiteit Leuven, Belgium, November 1990.

[49] MOONEN M., RAMOS J., *A subspace algorithm for balanced state space system identification.* Accepted for publication in IEEE Transaction on Automatic Control, August 1993 (ESAT-SISTA Report 1991-07, Department of Electrical Engineering, Katholieke Universiteit Leuven, Belgium).

[50] MOONEN M., DE MOOR B., VANDEWALLE J., *SVD-based subspace methods for multivariable continuous time system identification.* "Identification of continuous-time systems". G. P. Rao, N. K. Sinha Eds, Kluwer Academic Publications, 1991, p. 473-488.

[51] MOONEN M., DE MOOR B., *Comments on 'State-space model identification with data correlation'.* International Journal of Contorl, Vol. 55, no. 1, p. 257-259, 1992.

[52] MOONEN M., DE MOOR B., RAMOS J., TAN S. *A subspace identification algorithm for descriptor systems.* Systems & Control Letters, Volume 19, 1992, p. 47-52.

[53] MOORE B. C., *Principal component analysis in linear systems : Controllability, Observability and Model Reduction.* IEEE Transactions on Automatic Control, Vol. AC-26, no. 1, February 1981.

[54] MULLIS C. T., ROBERTS R. A., *Synthesis of minimum round-off noise fixed point digital filters.* IEEE Transactions on Circuits and Systems, Vol. CAS-23, p. 555-562, 1976.

[55] NORTON J. P., *An introduction to identification.* Academic Press, London, 1986.

[56] OTTERSTEN B., *Parametric subspace fitting methods for array signal processing.* PhD Thesis, Information Systems Laboratory, Department of Electrical Engineering, Stanford University, CA, USA, December 1989.

[57] SÖDERSTRÖM T., STOICA P., *System Identification.* Prentice Hall International Series in Systems and Control Engineering, Prentice Hall, New York, 1989.

[58] SILVERMAN L., *Realization of linear dynamical systems.* IEEE Transaction on Automatic Control, Vol. AC-16, p. 554-567, December 1971.

[59] SWINDLEHURST A., ROY R., OTTERSTEN B., KAILATH T., *System identification via weighted subspace fitting.* Proc. of the American Control Conference, p. 2158-2163, 1992.

[60] SYLVESTER J. J., *Sur la réduction biorthogonale d'une forme linéo-lin'eaire à sa forme canonique.* Comptes Rendus, CVIII, p. 651-653, 1889.

[61] VAN OVERBEEK A. M. J., LJUNG L., *On-line structure selection for multivariable state space models.* Automatica, 18(5), p. 529-543, 1982.

[62] VAN OVERSCHEE P., DE MOOR B., SUYKENS J., *Subspace algorithms for system identification and stochastic realization.* Proc. Conf. on Mathematical Theory for Networks and Systems, MTNS, Kobe, Japan, p. 589-595, June 1991, Mita Press.

[63] VAN OVERSCHEE P., DE MOOR B., *Subspace algorithms for the stochastic identification problem.* 30th IEEE Conference on Decision and Control, Brighton, UK, p. 1321-1326, 1991.

[64] VAN OVERSCHEE P., DE MOOR B., *Subspace algorithms for the stochastic identification problem.* ESAT-SISTA Report 1992, Department of Electrical Engineering, Katholieke Universiteit Leuven, Belgium. Accepted for publication in Automatica, 1992.

[65] VAN OVERSCHEE P., DE MOOR B., *Two subspace algorithms for the identification of combined deterministic-stochastic systems.* ESAT-SISTA Report 1992-34,

Department of Electrical Engineering, Katholieke Universiteit Leuven, September 1992 (submitted for publication).

[66] VAN OVERSCHEE P., DE MOOR B., *Subspace identification of a glass tube manufacturing process.* ESAT-SISTA Report 1992-35, Department of Electrical Engineering, Katholieke Universiteit Leuven, September 1992 (submitted for publication).

[67] VERHAEGEN M., DEWILDE P., *Subspace model identification, Part I : The output-error state space model identification class of algorithms, Part II : Analysis of the elementary output-error state space identification algorithms.* Internal Report, Dept. of EE, Delft University of Technology, December 1990.

[68] VERHAEGEN M., *A novel non-iterative MIMO state space model identification technique.* Proc. 9th IFAC/IFORS Symp. on Identification and System Parameter Estimation, Budapest, Hungary, p. 1453-1457, 1991.

[69] VIBERG M., *Subspace fitting concepts in sensor array processing.* PhD Thesis, Department of Electrical Engineering, Linköping University, Sweden, 1989.

[70] VIBERG M., OTTERSTEN B., *Sensor array processing based on subspace fitting.* IEEE Transactions on Acoustics, Speech and Signal Processing, ASSP-39, May 1991.

[71] Young MAN CHO, Guanghan XU, Thomas KAILATH, *Fast Recursive Identification of State Space Models via Exploitation of Displacement Structure.* Information Systems Lab, Stanford University, 27 p., July 1992 (submitted for publication).

[72] WILLEMS J., *From time series to linear systems.* Automatica, Part I : Vol. 22, no. 5, p. 561-580, 1986, Part II : Vol. 22, no. 6, p. 675-694, 1986, Part III : vol. 23, no. 1, p. 87-115, 1987.

[73] ZEIGER H., MCEWEN A., *Approximate linear realizations of given dimension via Ho's algorithm.* IEEE Transactions on Automatic Control, 19, p. 153, 1974.