OKID as a general approach to linear and bilinear system identification

Francesco Vicario

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
in the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2014

©2014
Francesco Vicario
All rights reserved

ABSTRACT

OKID as a general approach to linear and bilinear system identification

Francesco Vicario

This work advances the understanding of the complex world of system identification, i.e. the set of techniques to find mathematical models of dynamical systems from measured input-output data, and exploits well-established approaches for linear systems to address nonlinear system identification problems.

We focus on observer/Kalman filter identification (OKID), a method for simultaneous identification of a linear state-space model and the associated Kalman filter from noisy input-output measurements. OKID, developed at NASA, resulted in a very successful algorithm known as OKID/ERA (OKID followed by eigensystem realization algorithm). We show how ERA is not the only method to complete the OKID process, developing novel algorithms based on the preliminary estimation of the Kalman filter output residuals.

The new algorithms do not only show potential for better performance, they also cast light on OKID, explicitly establishing the Kalman filter as central to linear system identification in the presence of noise, paralleling its role in signal estimation and filtering. The Kalman filter embedded in the OKID core equation is capable of converting the original problem, affected by random noise, into a purely deterministic problem. The new interpretation leads to the extension of OKID to output-only system identification, providing a new tool for applications in structural health monitoring, and raises OKID to the level of a unified approach for input-output

and output-only linear system identification. Any algorithm for linear system identification formulated in the absence of noise can now optimally handle noisy data via a preliminary step consisting in solving the OKID core equation.

The OKID framework developed for linear system identification is then extended to bilinear systems, which are of interest because several natural phenomena are inherently bilinear and also because high-order bilinear models are universal approximators for a wide class of nonlinear systems. The formulation of an optimal bilinear observer for bilinear state-space models, similar to the Kalman filter in the linear case, leads to the development of an extension of OKID to bilinear system identification. This is the first application of OKID to nonlinear problems, not only because bilinear systems are themselves nonlinear, but also because one can think of bilinear OKID as a technique to find bilinear approximations of nonlinear systems.

Furthermore, the same strategy adopted in this work could be used to extend OKID directly to other classes of nonlinear models.

Contents

List of Figures

List of Tables

Acknowledgements

This is the end of my incredible adventure as a PhD student at Columbia. I am extremely grateful and also proud to be living this moment with a big smile and a feeling of great satisfaction. This was my hope four years ago, when I started. It became inconceivable just three years ago. Many people made these years a unique, unforgettable and invaluable experience, with a lifetime impact. I would like to thank them all.

My deepest gratitude goes to Professor Richard W. Longman (my advisor), Professor Raimondo Betti and Professor Minh Q. Phan. Our weekly meeting shaped my mind, taught me how to do research and most importantly how to have fun doing it. For two and a half years I have been asking myself how they could dedicate so much time to me. I believe the reason is extraordinarily simple: they like what they do and they are friends. They enjoy working together. Well, I am extremely grateful to their friendship as it allowed me to constantly see them in action and put me in the best conditions to learn and grow, week after week. In particular, Richard provided me with invaluable technical guidance and an unachievable model of professional competence and human thickness. Raimondo is the person who most believed in me even when I did not at all; he kept pushing and motivating me to accomplish more and more. Minh has been a continuous source of inspiration; he forged me with his unique research style. I will never be able to thank them enough for the chance they gave me to live my PhD to the fullest.

My special thanks are also due to Dr. Nicolas W. Chbat. He introduced me to the world of healthcare and transferred to me the passion for physiological modeling. He also gave me the amazing opportunity to apply what I learnt during my PhD to some exciting applications at Philips Research North America. I would also like to explicitly thank the Department of Mechanical Engineering at Columbia, which played a key role, supporting my studies and offering

me a unique chance to engage in teaching.

Additionally, the completion of this work would not have been possible without the constant support and encouragement of my parents Silvana and Ezio, my beloved sisters Giulia and Margherita and my lifetime friends Paolo, Giuseppe, Ilaria, Paola, Marco and Luca. Somehow they could feel when I most needed them, no matter where they were. I will for ever thank Giorgia for pushing me to move to New York and start a PhD. I thank Huade, with whom I shared many moments in preparation for that scary qualifying examination. I thank Paolo, Fabio, Luciana, Suparno, Demi and Ali for the countless, indispensable coffee breaks at the fourth floor. I thank Te, Jianzhong, Ae and Bing for the lovely atmosphere created in our office. Finally, a huge thank you goes to Roberto, Antonio and Nikolaos. In a city where everything changes at the speed of light, where people always come and leave, they became my fixed points, the best friends I could imagine to make here. Without your energy, patience, enthusiasm and even care, I would have never made it to the finish line of this rollercoaster. And for the home stretch, we could not have found a better companion than Daisy. Thank you!

Finally, I feel I need to thank myself. For the energy I invested in this long journey, for the resilience that took me through the toughest to the best times of my life, for believing until the end in the possibility of turning the impossible into possible. When I was younger, I wanted to become a soccer player, play and win for Juventus. I did not make it. But I completed a PhD with the same determination and following the same values that led Juventus to a historical come back after unprecedented difficulties. It is just soccer, but ...

Francesco Vicario

October 2014

"I rigori li sbaglia solo chi ha il coraggio di tirarli."

"Only those who have the courage to take a penalty miss them."

Chapter 1

Introduction

1.1 System identification

System identification has attracted a lot of research interest over the last three decades in many areas of engineering and other fields, virtually in any application concerning models of dynamic systems. The basic purpose of system identification is to find from experimental data a mathematical model representing the behavior of the dynamic system of interest. In this regard, it can be seen as a powerful alternative to traditional modeling by first principles. The latter is often extremely useful in providing the structure of the dynamic equations and understanding qualitative aspects of the system behavior. However, the values of the equation coefficients can be critical to assign, for example because they depend on material properties difficult to assess or on complex geometry. Additionally, the physical parameters determining such coefficients can vary significantly from the static conditions in which they are measured to the dynamic conditions in which the system will operate. System identification methods have been developed that provide both the order of the mathematical model and its coefficients from

dynamic input-output data measured from the system of interest in a dedicated experiment or in normal operation (on line).

Among the possible mathematical representations of dynamic systems, state-space models have become very popular in particular because they lend themselves well to system analysis, numerical simulation and controller design via established and robust techniques. A discrete-time linear-time-invariant state-space model is defined as

$$\boldsymbol{x}(k+1) = \mathbf{A}\boldsymbol{x}(k) + \mathbf{B}\boldsymbol{u}(k) \tag{1.1a}$$

$$\boldsymbol{y}(k) = \mathbf{C}\boldsymbol{x}(k) + \mathbf{D}\boldsymbol{u}(k) \tag{1.1b}$$

where k is the time step (sample counter), $\boldsymbol{x} \in \mathbb{R}^{n\times 1}$ is the state vector, $\boldsymbol{u} \in \mathbb{R}^{m\times 1}$ is the input vector, $\boldsymbol{y} \in \mathbb{R}^{q\times 1}$ is the output vector, $\mathbf{A} \in \mathbb{R}^{n\times n}$ is the system matrix, $\mathbf{B} \in \mathbb{R}^{n\times m}$ is the input matrix, $\mathbf{C} \in \mathbb{R}^{q\times n}$ is the output matrix and $\mathbf{D} \in \mathbb{R}^{q\times m}$ is the direct influence matrix. Equation (1.1a) is the *state equation* and equation (1.1b) is the *observation equation*. Such a model is linear since the variables $\boldsymbol{u}$, $\boldsymbol{x}$ and $\boldsymbol{y}$ appear linearly in equation (1.1), and it is time-invariant since the matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ are constant over time. Although physical dynamical systems are usually in continuous time (and state-space models can be cast in continuous time, too), discrete-time models are of interest because they are particularly suitable for digital implementation. This is true both in applications of dynamic models, for example in controllers implemented via software, and in the identification process, where the measured data come from sampling devices.

State-space models are not the only option to represent linear dynamical systems. Although a review of all possible linear models is beyond the scope of this work, it is worth mentioning the

following finite-difference equation

$$\begin{aligned}\boldsymbol{y}(k) = \alpha_1\boldsymbol{y}(k-1) + \alpha_2\boldsymbol{y}(k-2) + \ldots + \alpha_n\boldsymbol{y}(k-n)\\ + \beta_0\boldsymbol{u}(k) + \beta_1\boldsymbol{u}(k-1) + \ldots + \beta_n\boldsymbol{u}(k-n)\end{aligned} \tag{1.2}$$

where α_j's and β_j's are constant matrices. Equation (1.2) expresses the output at the current time step k as a linear combination of past outputs and current and past inputs, looking back n time steps. n is referred to as the order of the system. Equation (1.2) is known in the time-series literature as an autoregressive model with exogenous input (ARX). Note that the models in equations (1.1) and (1.2) are equivalent in terms of input-output mapping. It is indeed possible to convert one into another by simple operations.

The model in equation (1.1) is purely deterministic since the input $\boldsymbol{u}$ and the output $\boldsymbol{y}$ are supposed to be measured and therefore completely known. Given some initial conditions, the state response over time is fully determined as well. However in real applications noise inevitably affects the system. It can be in the form of electrical noise in actuators and sensors, disturbances in the environment where the dynamic system operates or even modeling error. Every mathematical model, no matter how accurate it is, is based on simplifying assumptions that hopefully well represent the characteristics of the system that mostly affect its dynamic response. Nevertheless, unknown disturbances or unmodeled secondary phenomena, although negligible, often contribute to the response and appear in the measurements as noise.

To take into account noise, the state-space model in equation (1.1) is usually modified as

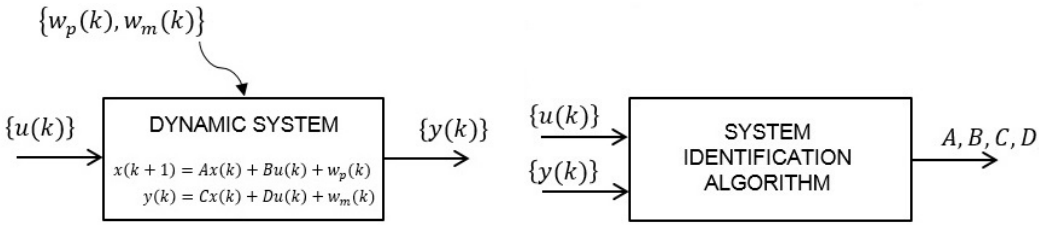
$$\boldsymbol{x}(k+1) = \mathbf{A}\boldsymbol{x}(k) + \mathbf{B}\boldsymbol{u}(k) + \boldsymbol{w}_p(k) \tag{1.3a}$$

$$\boldsymbol{y}(k) = \mathbf{C}\boldsymbol{x}(k) + \mathbf{D}\boldsymbol{u}(k) + \boldsymbol{w}_m(k) \tag{1.3b}$$

where the vectors $\boldsymbol{w}_p \in \mathbb{R}^{n\times 1}$ and $\boldsymbol{w}_m \in \mathbb{R}^{q\times 1}$ represent some process and measurement noise affecting the dynamics of the state and the measured output. A common assumption for $\boldsymbol{w}_p(k)$ and $\boldsymbol{w}_m(k)$ is that they are uncorrelated with $\boldsymbol{u}(k)$ and $\boldsymbol{y}(k)$ are zero-mean and white. Additionally, they often are considered to have constant covariance matrices $\mathbf{Q}$ and $\mathbf{R}$, respectively. Such assumptions have become standard in the literature as they are a good compromise between mathematical tractability of the model and accurate representation of reality. However it must be kept in mind that, for instance, modeling errors cannot be white by definition and disturbances, even when random, usually have a specific frequency content, violating the assumption of flat spectrum. Obviously, the closer the noise is to the stated assumptions, the better the system identification methods based on equation (1.3) perform. A good reference on state-space models, covering all the concepts useful to thoroughly understand the material presented in the next chapters is Reference 1.

1.1.1 System identification vs. parameter estimation

Figure 1.1 shows the system identification process by input-output diagrams. Two aspects of system identification are worth pointing out. First of all the order of the system is not known a priori. The dimensions of the identified matrices (in particular $\mathbf{A}$) are not pre-specified and they are somehow determined by information embedded in the measured input-output data sequences. Secondly, the identified model is necessarily linear, although it can be of any order.



(A) Data collection. (B) Identification.

FIGURE 1.1: System identification.

Typically, when a nonlinear model is desired, one has to specify the form of the nonlinearity, fixing the order of the system as well.

Those two aspects are peculiar to system identification and distinguish it from the sister field of parameter estimation. In parameter estimation the equations of the dynamic model are completely specified a priori, apart from the value of their coefficients (parameters). Hence parameter estimation techniques lend themselves to finding nonlinear models as well. On the other hand, they have no freedom in choosing the order of the system, which therefore needs to be known a priori. Parameter estimation requires knowledge of the physics of the phenomenon that one wants to model, whereas system identification is often referred to as *black-box modeling* since in principle it does not require any a priori information on the system to be identified. In summary, system identification provides the best linear approximation of the model representing the true dynamics of the system. Because many systems behave linearly, at least in a limited range of operation, and the most popular system analysis and controller design techniques are based on linear models, system identification has become a very popular modeling technique.

1.1.2 Deterministic vs. combined system identification methods

Many methods for system identification have been developed. Providing a comprehensive review is beyond the scope of this work, and the task is complicated by the large number of algorithms that researchers have devised. It is however very relevant to this work remarking the difference between deterministic and combined system identification methods, following the same classification done in Reference 2.

Deterministic methods are developed from the state-space model in equation (1.1). Since their formulation does not take into account the presence of noise, their behavior with experimental input-output data, inevitably polluted with noise, is unknown a priori. Their robustness is

entirely in the hands of the numerical techniques used in the implementation. Some deterministic algorithms have actually proven to be quite robust to noise. Nevertheless, better methods have been worked out based on the model in equation (1.3), i.e. taking explicitly into account process and measurement noise. Their formulation then directly addresses noise in the data and tries to minimize the resulting uncertainty in the identification. Such methods are referred to as *combined* since they handle both the deterministic ($\boldsymbol{u}$) and stochastic ($\boldsymbol{w}_p$ and $\boldsymbol{w}_m$) drive of the system. Another class can be defined, that of *stochastic* methods, which address the identification problem where no deterministic input is present and will be the topic of chapter 4.

1.2 Bilinear systems

Bilinear systems are a specific class of nonlinear systems and are defined in discrete-time state-space form as

$$\boldsymbol{x}(k+1) = \mathbf{A}\boldsymbol{x}(k) + \sum_{i=1}^{m} \mathbf{N}_i \boldsymbol{x}(k) u_i(k) + \mathbf{B}\boldsymbol{u}(k) + \boldsymbol{w}_p(k) \tag{1.4a}$$

$$\boldsymbol{y}(k) = \mathbf{C}\boldsymbol{x}(k) + \mathbf{D}\boldsymbol{u}(k) + \boldsymbol{w}_m(k) \tag{1.4b}$$

Bilinear systems differ from linear systems because of the presence of products between the state and the input, whose effect on the dynamics of the state is determined by matrices $\mathbf{N}_i \in \mathbb{R}^{n \times n}$. Bilinear systems are important per se since several phenomena in engineering, biology, physiology, sociology and other fields are inherently bilinear (References 3–5). Some examples of inherently bilinear systems are DC motors, automobile braking systems, processes involving the regulation of thermal energy by fluid flow, chemical reactors. Even more appealing is the fact that bilinear models can approximate more general nonlinear systems, namely input-affine dynamic systems, i.e. systems with input appearing linearly (References 6, 7). At the same

time, bilinear models have sufficient mathematical structure to aim to develop techniques for controller and observer design so that one can think of bilinear models as a unified approach to handle nonlinear control problems. Interest in bilinear systems has recently seen a spark after Carleman linearization (References 8) was found to be a technique to obtain a bilinear approximation of input-affine dynamic systems (Reference 9).

1.3 Research motivation

The main motivation for this research work is to develop methods for the identification of discrete-time bilinear state-space models, continuing the work initiated by Hizir (Reference 10). Significant work in the area of bilinear and more generally of nonlinear system identification has been recorded over the past few years, with valuable contribution in particular from Juang (References 9, 11–14). Although well-established and robust system identification techniques have been developed for linear system identification, to date the same cannot be stated for bilinear systems and more generally for nonlinear systems. The property of bilinear models of being able to approximate input-affine nonlinear system positions bilinear system identification as a general method to identify mathematical models of nonlinear systems.

The shortcomings of existing bilinear system identification methods are introduced in chapters 6, 7, 9, which describe novel bilinear identification methods to overcome them. One of the main objectives is concerned with the development of bilinear identification methods with explicit connection with state observers for bilinear systems. For this reason, the approach to bilinear system identification starts with the same technique that in the past led to a method that has not only been very successful in linear system identification (see for example References 15–19) but also has an explicit connection with state observers, as suggested by the

name *observer/Kalman filter identification* (OKID) followed by *eigensystem realization algorithm* (ERA), References 1, 20. More precisely, under the stated assumptions on the noise $\boldsymbol{w}_p$ and $\boldsymbol{w}_m$, OKID/ERA simultaneously identifies the system model and the associated optimal observer for the (unknown) covariances of the noise embedded in the measured data. With reference to the classification in section 1.1.2, OKID/ERA qualifies as a combined system identification method. State observers are an indispensable component to design state-feedback controllers, since the measured outputs are often fewer than the state variables of the dynamic system and therefore an estimate for the entire state is necessary to close the loop. For bilinear systems approximating more general nonlinear systems, the need for bilinear observers is even more important because Carleman linearization introduces additional non-physical state variables that cannot be measured.

The extension of OKID to bilinear systems turns out to require some significant modifications to the original OKID/ERA algorithm. In particular, the lack of ERA for bilinear systems motivates the development of an alternative approach to complete the OKID process. The research on bilinear system identification presented in this thesis leads then to the development of new methods for linear system identification, too. Such methods provide a novel and powerful interpretation to OKID that raises OKID to the level of a general approach to system identification and establishes the OKID core equation and the underlying Kalman filter as the bridge between deterministic (noise-free) and combined (noisy) system identification.

1.4 Thesis organization

The organization of this thesis could have followed two alternative paths. From a chronological point of view, chapter 6 should come first, followed by chapter 7. The former rigorously extends

to the bilinear case the concept of interaction matrices, which is also at the core of the development of linear OKID, resulting in new deterministic bilinear system identification algorithms that have the advantage of not imposing any restriction to the form of input used to excite the system. Chapter 7 describes an alternative method to overcome the approximate nature and the curse of dimensionality affecting the method in chapter 6, at the price of relying on structured excitation. Chapter 8 establishes the connection between the bilinear identification method in chapter 6 and the bilinear state estimation problem, proving that the above mentioned interaction matrices correspond to the gains of optimal bilinear observers, similar to OKID in the linear case. Also, chapter 8 lays the foundation for the extension of the deterministic bilinear identification method in chapter 6 to the combined case, paving the ground for bilinear OKID. For that purpose, the Kalman filter properties at the core of OKID needs to be revisited as reported in chapter 2. The development of bilinear OKID described in chapter 9 relies on a new approach to system identification, which is the most important conceptual contribution of this thesis. Before being applied to bilinear system identification, the new approach is completely developed and implemented for linear system identification leading to the new OKID-based linear system identification algorithms described in chapter 3. The sudden expansion of the family of OKID methods and the intuitive interpretation of the new methods motivate the development of a general framework for OKID, which is presented in chapter 5, and suggest the possibility of applying OKID to stochastic (or output-only) system identification (chapter 4).

The author has chosen the second path, organizing the work in accordance with the logical flow of concepts and results. The work is then split into two parts. The first one (chapters 2 - 5) presents OKID for linear systems, from its theoretical foundations, through classic and novel algorithms, up to the illustration of the general framework for system identification that OKID provides, including output-only identification. The second part is concerned with bilinear systems and

begins with the development of two new methods for their identification in the absence of noise, continues with the link between such methods and bilinear state observers (which is also exploited to develop a new method for bilinear observer design), establishing the new methods for bilinear system identification as belonging to the OKID framework, and culminates with the development of bilinear OKID in the presence of noise.

Chapter 2

Observer/Kalman filter identification

2.1 Introduction

After a clear statement of the problem that observer/Kalman filter identification (OKID) addresses, this chapter provides a comprehensive and self-contained presentation of the keystone of OKID. The equation at the core of OKID, relating measured input-output data without the state appearing explicitly, is derived following an approach slightly different from that used in the traditional OKID literature (References 1, 20). The derivation better highlights the role of the observer in OKID. Additionally, the connection between the least-squares solution to the OKID core equation and the optimal observer (Kalman filter) is presented in detail. The need to revisit how such a connection is established arises from the desire to develop an extension of OKID for bilinear systems, which highlights the necessity of working out more rigorously the proof at the core of OKID. For the purpose, the relevant properties of the steady-state Kalman

filter are proven and linked to the linear-time-invariant (LTI) observer used in the derivation of the OKID core equation. In chapter 8, the same approach is extended to the bilinear case.

Linear-time-invariant observers and in particular the Kalman filter are at the core of OKID, therefore they deserve a thorough introduction. A review of observers and state estimators is beyond the scope of this chapter, which is limited to the presentation of the concepts useful to develop and understand OKID. Although not comprehensive, such a presentation is at the same time rigorous and self-contained to the benefit of the reader with little background in the area.

2.2 Problem Statement

Consider the linear-time-invariant (LTI) dynamical system in discrete-time state-space form given in equation (1.3). A single set of length l of input-output data, measured from the system starting at some unknown initial state $\boldsymbol{x}(0)$, is given

$$\{\boldsymbol{u}(k)\} = \{\boldsymbol{u}(0),\ \boldsymbol{u}(1),\ \boldsymbol{u}(2),\ \ldots,\ \boldsymbol{u}(l-1)\} \tag{2.1a}$$

$$\{\boldsymbol{y}(k)\} = \{\boldsymbol{y}(0),\ \boldsymbol{y}(1),\ \boldsymbol{y}(2),\ \ldots\ \boldsymbol{y}(l-1)\} \tag{2.1b}$$

The objective is to identify the system of equation (1.3) from the measured input-output data provided in equation (2.1), i.e. to find the matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ given the sequences $\{\boldsymbol{u}(k)\}$ and $\{\boldsymbol{y}(k)\}$. The data is assumed to be of sufficient length and richness so that the system of equation (1.3) can be correctly identified. Neither the noise sequences $\{\boldsymbol{w}_p(k)\}$ and $\{\boldsymbol{w}_m(k)\}$ or their covariance matrices $\mathbf{Q}$ and $\mathbf{R}$ are known.

It would be ideal to extract from the measured input-output data also the optimal linear observer of the system state, to be used for example in a state-feedback control loop for the identified

system. The optimal observer is uniquely determined by the system and noise covariance matrices (see equation (2.7) or References 1, 21). Whereas the measurement noise covariance can usually be estimated via dedicated experiments on the sensors, the process noise covariance is harder to assess, making the design of the observer more difficult. It is then desired to estimate the Kalman gain $\mathbf{K}$ as well, which will be introduced in the next section.

As a final note, the assumptions made in section 1.1 on the noise affecting the dynamic system to be identified are standard in the literature of system identification and state estimation. In real applications, noise is not perfectly white, in particular the process noise, which usually is due to modeling errors or disturbances. The stated assumptions have the benefit of mathematical tractability, while generally retaining a good approximation of actual noise. Obviously, the smaller the approximation, the better the identification techniques developed from equation (1.3) and the stated assumptions will perform. Similar to how the Kalman filter is a robust state estimator even in conditions violating its assumptions, even OKID turns out to perform well in such conditions. This is supported by numerous successful applications of the traditional OKID/ERA algorithm (see for example References 1, 15, 17, 20, 22, 23) as well as by the experimental examples shown in chapters 3 and 4.

2.3 Observers and Kalman filter

A typical problem in system dynamics is the estimation of the state $\boldsymbol{x}$, given the system matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ and input-output measurements. For instance, in modern control system engineering, the knowledge of the current state is necessary to implement state-feedback control laws. However, in most cases the state is not measured and only a subset (or, with more generality, a linear combination) of the state variables is available from the output measurement $\boldsymbol{y}$. Also the input $\boldsymbol{u}$ is measured.

The state estimation problem (or observer problem) for the LTI system of equation (1.3) can be formulated as follows. Given the system model and the measured values of the input $\boldsymbol{u}$ and output $\boldsymbol{y}$ from time sample 0 to k, what is the best estimate that we can get for the state $\boldsymbol{x}$ at the next time step $k+1$? Any set of equations yielding at time k an estimate $\hat{\boldsymbol{x}}(k+1)$ for the state $\boldsymbol{x}(k+1)$ is called *state estimator* for the system in equation (1.3), *state observer* or simply *observer*. An alternative name is that of *one-step-ahead predictor* to remark the fact that an observer is asked to predict the behavior of the state at the future step knowing the input and output up to the current time step.

In general, an observer for the LTI system of equation (1.3) has then the form

$$\hat{\boldsymbol{x}}(k+1) = \mathcal{F}\left(k, \boldsymbol{u}(0), \boldsymbol{u}(1), \ldots, \boldsymbol{u}(k), \boldsymbol{y}(0), \boldsymbol{y}(1), ..., \boldsymbol{y}(k)\right) \tag{2.2}$$

which might also be written in recursive (or adaptive) form

$$\hat{\boldsymbol{x}}(k+1) = \mathcal{G}\left(k, \hat{\boldsymbol{x}}(k), \boldsymbol{u}(k), \boldsymbol{y}(k)\right) \tag{2.3}$$

The adaptive form is highly desirable for computational efficiency, but a priori it is not guaranteed that any general form can be converted into adaptive. Note that for generality, both $\mathcal{F}$ and $\mathcal{G}$ can explicitly depend on time.

A special family of state estimators is given by the linear observers, defined by equation (2.2) with linear $\mathcal{F}$ or, in recursive form, by equation (2.3) with linear $\mathcal{G}$. Note that any linear observer can be written in recursive form. The most general form for a linear observer can then

be written as

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{F}(k)\hat{\boldsymbol{x}}(k) + \mathbf{H}(k)\boldsymbol{u}(k) + \mathbf{K}(k)\boldsymbol{y}(k) \tag{2.4a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \mathbf{D}\boldsymbol{u}(k) \tag{2.4b}$$

where $\mathbf{F}(k) \in \mathbb{R}^{n\times n}$, $\mathbf{H}(k) \in \mathbb{R}^{n\times m}$ and $\mathbf{K}(k) \in \mathbb{R}^{n\times q}$. Equation (2.4b) is added to provide an estimate of the true system output as well. Such an estimate is indicated by $\hat{\boldsymbol{y}}$, which is also known as observer output. Note that equation (2.4) is the equation of a dynamic system, whose state is driven by $\boldsymbol{u}$ and $\boldsymbol{y}$, which are the input and output of the original, physical system in equation (1.3) but from the point of view of the observer are both inputs. In other words, an observer is a dynamical system itself. Whereas the original system of equation (1.3) is the mathematical representation of a physical (real) system, an observer is *just* a set of equations whose purpose is the estimation of the state and possibly the output of the original system. From a mathematical point of view, both equation (1.3) and equation (2.4) are dynamic systems (the former is linear-time-invariant, LTI, the latter is linear-time-varying, LTV). This observation is of paramount importance not only in the proof of the OKID core equation, but also to understand the intimate nature of OKID and in particular of the algorithms presented in section 3.4. The similarity of equation (2.4b) with equation (1.3b) is due to the following simple fact. Once the desired estimate of the next state is available and the next input is chosen, one can compute the corresponding estimate for the next output via equation (1.3b), which indeed describes how the output is related to the state and the input. Only the value of the measurement noise is unknown, but the best guess for it is its expected value, which is 0 at each time step by assumption ($\boldsymbol{w}_m$ is a zero-mean random process). Hence, equation (2.4b).

In his seminal paper (Reference 24), Kalman derived equation (2.4) as the observer form providing the optimal estimate of the state under some assumptions on the probability distribution

of the noises. A common distribution satisfying such assumptions is the normal or Gaussian distribution. Kalman also found the relations to obtain the value of the matrices $\mathbf{F}(k)$, $\mathbf{H}(k)$ and $\mathbf{K}(k)$. The resulting observer, known as *Kalman filter*, is the linear observer in equation (2.4) with

$$\mathbf{F}(k) = \mathbf{A} - \mathbf{K}(k)\mathbf{C} \tag{2.5a}$$

$$\mathbf{H}(k) = \mathbf{B} - \mathbf{K}(k)\mathbf{D} \tag{2.5b}$$

$$\mathbf{K}(k) = \mathbf{A}\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}}\left(\mathbf{R} + \mathbf{C}\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}}\right)^{-1} \tag{2.5c}$$

$$\mathbf{\Pi}(k+1) = \mathbf{A}\mathbf{\Pi}(k)\mathbf{A}^{\mathrm{T}} - \mathbf{A}\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}}\left(\mathbf{R} + \mathbf{C}\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}}\right)^{-1}\mathbf{C}\mathbf{\Pi}(k)\mathbf{A}^{\mathrm{T}} + \mathbf{Q} \tag{2.5d}$$

Equation (2.5d) is the famous Riccati equation and $\mathbf{\Pi}$ is the covariance matrix of the state estimation error $\boldsymbol{e}(k) = \boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k)$, i.e. $\mathbf{\Pi}(k) = \mathbb{E}\left[\boldsymbol{e}(k)\boldsymbol{e}^{\mathrm{T}}(k)\right]$.

It is now important to point out two facts. First, the Kalman filter is the optimal observer, among *all* the possible observers for the LTI system of equation (1.3), in the case of Gaussian distribution (or other particular distributions satisfying the conditions given in Reference 24) of the noise. Second, the Kalman filter is the optimal *linear* observer for any other distribution for the noise, provided the latter is a white, i.e. uncorrelated, process. As explained later, the optimality criterion behind the above fact can be defined in several equivalent ways, remarkably all leading to the same observer. For the sake of clarity, it is also worth mentioning that Kalman derived his famous equations with reference to a linear-time-varying (LTV) dynamic system, addressing a more general family of dynamic systems than the one considered in this work. In Reference 24 also the noise covariance matrices could be time-varying, i.e. $\mathbf{Q}(k)$ and $\mathbf{R}(k)$.

Under the assumption of stationary noise, i.e. noise with mean and covariance constant over time as assumed in section 2.2, the matrices of the Kalman filter approach over time constant

values. The Kalman filter in steady state then becomes a LTI observer. The most general form of LTI observer is

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{F}\hat{\boldsymbol{x}}(k) + \mathbf{H}\boldsymbol{u}(k) + \mathbf{K}\boldsymbol{y}(k) \tag{2.6a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \mathbf{D}\boldsymbol{u}(k) \tag{2.6b}$$

When $\mathbf{F}$, $\mathbf{H}$ and $\mathbf{K}$ are given by

$$\mathbf{F} = \mathbf{A} - \mathbf{K}\mathbf{C} \tag{2.7a}$$

$$\mathbf{H} = \mathbf{B} - \mathbf{K}\mathbf{D} \tag{2.7b}$$

$$\mathbf{K} = \mathbf{A}\mathbf{\Pi}\mathbf{C}^{\mathrm{T}}\left(\mathbf{R} + \mathbf{C}\mathbf{\Pi}\mathbf{C}^{\mathrm{T}}\right)^{-1} \tag{2.7c}$$

$$\mathbf{\Pi} = \mathbf{A}\mathbf{\Pi}\mathbf{A}^{\mathrm{T}} - \mathbf{A}\mathbf{\Pi}\mathbf{C}^{\mathrm{T}}\left(\mathbf{R} + \mathbf{C}\mathbf{\Pi}\mathbf{C}^{\mathrm{T}}\right)^{-1}\mathbf{C}\mathbf{\Pi}\mathbf{A}^{\mathrm{T}} + \mathbf{Q} \tag{2.7d}$$

the LTI observer in equation (2.6) is the steady-state Kalman filter. Equation (2.7d) is known as the algebraic Riccati equation. Note that equation (2.6) is itself the equation of a LTI dynamic model and that the matrices of the steady-state Kalman filter depend on the system matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ as well as the noise covariance matrices $\mathbf{Q}$, $\mathbf{R}$.

Many derivations of equation (2.7) have been worked out and are available in the literature. The Kalman filter can indeed be derived in several ways and, remarkably, from different starting points as there are a few properties that uniquely characterize it. Reference 1 defines the optimality criterion for the state estimator as the minimization of the expected value of the magnitude (squared, for convenience) of the state estimation error, $\mathbb{E}\left[\left(\boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k)\right)^{\mathrm{T}}\left(\boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k)\right)\right]$. The choice is intuitive, since the goal of the observer is the estimation of the state of the true system. Among the other properties that uniquely define the Kalman filter, we are interested in the following four, valid for all $k \geq p$, where p is a time index sufficiently large to consider

the Kalman filter in the steady state and

$$\boldsymbol{\epsilon}(k) = \boldsymbol{y}(k) - \hat{\boldsymbol{y}}(k) \tag{2.8}$$

is the observer (Kalman filter) output residual:

Property 1 $\boldsymbol{\epsilon}$ is a zero-mean white process, i.e. $\mathbb{E}\left[\boldsymbol{\epsilon}(k)\right] = \mathbf{0}$ and $\mathbb{E}\left[\boldsymbol{\epsilon}(i)\boldsymbol{\epsilon}^{\mathrm{T}}(j)\right] = \mathbf{0}$ for all $i \neq j$.

Property 2 $\mathbb{E}\left[\boldsymbol{\epsilon}^{\mathrm{T}}(k)\boldsymbol{\epsilon}(k)\right]$ is minimized.

Property 3 $\boldsymbol{\epsilon}$ is orthogonal to past output values and past and current input values, i.e. $\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{y}^{\mathrm{T}}(k-j)\right] = \mathbf{0}$ for all $j = 1, 2, ..., k$ and $\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{u}^{\mathrm{T}}(k-j)\right] = \mathbf{0}$ for all $j = 0, 1, ..., k$.

Their proofs are presented in section 2.4. Additionally, the Kalman filter is the only linear observer featuring the above properties. With more generality we can then state

Fact 0 The Kalman filter is the unique linear system in the form of equation (2.6), i.e. the unique LTI observer, having any of the properties 1 to 3.

This fact is the theoretical foundation of OKID.

2.4 Kalman Filter properties

In this section the properties of the Kalman filter which provide the theoretical foundation of OKID are derived. The presentation is specific to OKID, it is not meant to be a review of the general properties of the Kalman filter. Lemmas 8.4 and 8.5 prove property 1 above, lemma 8.8 proves property 2 and lemmas 8.6 and 8.7 prove property 3. Fact 0 directly follows from the above mentioned lemmas.

Consider the system in equation (1.3) and the LTI observer in equation (2.6). From the definition of state estimation error

$$\boldsymbol{e}(k) = \boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k) \tag{2.9}$$

and equations (1.3) and (2.6), the dynamics of the state estimation error is

$$\begin{aligned}
\boldsymbol{e}(k+1) &= \boldsymbol{x}(k+1) - \hat{\boldsymbol{x}}(k+1) \\
&= \mathbf{A}\boldsymbol{x}(k) + \mathbf{B}\boldsymbol{u}(k) + \boldsymbol{w}_p(k) - \mathbf{F}\hat{\boldsymbol{x}}(k) - \mathbf{H}\boldsymbol{u}(k) - \mathbf{K}\boldsymbol{y}(k) \\
&= (\mathbf{A} - \mathbf{KC})\,\boldsymbol{e}(k) + (\mathbf{A} - \mathbf{F} - \mathbf{KC})\,\hat{\boldsymbol{x}}(k) \\
&\quad + (\mathbf{B} - \mathbf{H} - \mathbf{KD})\,\boldsymbol{u}(k) + \boldsymbol{w}_p(k) - \mathbf{K}\boldsymbol{w}_m(k)
\end{aligned} \tag{2.10}$$

We would like to analyze the properties of the observer in (2.6) and in particular the conditions under which the observer state $\hat{\boldsymbol{x}}(k)$ is an unbiased estimate for the system state $\boldsymbol{x}$ and the estimation error is minimized.

2.4.1 Unbiased observer

Lemma 2.1. *The state $\hat{\boldsymbol{x}}(k)$ of the observer in equation (2.6) is an unbiased estimate for the state $\boldsymbol{x}(k)$ of the system in (1.3) for all $k \geq p$ and for any arbitrary input sequence if and only if the following conditions are satisfied*

$$\mathbf{F} = \mathbf{A} - \mathbf{KC} \tag{2.11a}$$

$$\mathbf{H} = \mathbf{B} - \mathbf{KD} \tag{2.11b}$$

$$(\mathbf{A} - \mathbf{KC})^p = \mathbf{0} \tag{2.11c}$$

Notice that when the condition in equation (2.11c) cannot be satisfied exactly, it can still be approximately met by a sufficiently large value of p provided $\mathbf{A} - \mathbf{KC}$ has all the eigenvalues within the unit circle. Equation (2.11c) corresponds then to the observer in (2.6) being stable and having reached its steady state.

Proof. Define for convenience

$$\bar{\mathbf{A}} = \mathbf{A} - \mathbf{KC} \tag{2.12a}$$

$$\bar{\mathbf{F}} = \mathbf{A} - \mathbf{F} - \mathbf{KC} \tag{2.12b}$$

$$\bar{\mathbf{H}} = \mathbf{B} - \mathbf{H} - \mathbf{KD} \tag{2.12c}$$

$$\tag{2.12d}$$

Equation (2.10) can be rewritten as

$$\boldsymbol{e}(k+1) = \bar{\mathbf{A}}\boldsymbol{e}(k) + \bar{\mathbf{F}}\hat{\boldsymbol{x}}(k) + \bar{\mathbf{H}}\boldsymbol{u}(k) + \boldsymbol{w}_p(k) - \mathbf{K}\boldsymbol{w}_m(k) \tag{2.13}$$

and the expected value of the state estimation error is

$$\mathbb{E}\left[\boldsymbol{e}(k)\right] = \bar{\mathbf{A}}\mathbb{E}\left[\boldsymbol{e}(k-1)\right] + \bar{\mathbf{F}}\mathbb{E}\left[\hat{\boldsymbol{x}}(k-1)\right] + \bar{\mathbf{H}}\boldsymbol{u}(k-1) \tag{2.14}$$

To obtain equation (2.14), the noise terms vanish thanks to the assumption of zero-mean noises and the input is pulled out of the expectation operator since it is a known, deterministic variable ($\boldsymbol{u}$ is measured). The expected values in equation (2.14) are ensemble averages, i.e. they can be interpreted as the expected values obtained over different realizations of the process and measurement noise while driving the system with the same input sequence. Propagating

equation (2.14) one time step backwards yields

$$\begin{aligned}\mathbb{E}\left[\boldsymbol{e}(k)\right] = \bar{\mathbf{A}}^2\mathbb{E}\left[\boldsymbol{e}(k-2)\right] + \bar{\mathbf{A}}\bar{\mathbf{F}}\mathbb{E}\left[\hat{\boldsymbol{x}}(k-2)\right] \\ + \bar{\mathbf{A}}\bar{\mathbf{H}}\boldsymbol{u}(k-2) + \bar{\mathbf{F}}\mathbb{E}\left[\hat{\boldsymbol{x}}(k-1)\right] + \bar{\mathbf{H}}\boldsymbol{u}(k-1)\end{aligned} \tag{2.15}$$

and, by propagating back to the initial time step, we can write the expected state estimation error as

$$\mathbb{E}\left[\boldsymbol{e}(k)\right] = \bar{\mathbf{A}}^k\mathbb{E}\left[\boldsymbol{e}(0)\right] + f\left(\bar{\mathbf{A}}, \bar{\mathbf{F}}, \bar{\mathbf{H}}, \hat{\boldsymbol{x}}(0), \boldsymbol{u}(0), \boldsymbol{u}(1), \ldots, \boldsymbol{u}(k-1)\right) \tag{2.16}$$

where the second term on the right-hand side of equation (2.16) represents all the terms not depending on the initial error and is identically zero for any input history and initial observer state if and only if $\bar{\mathbf{F}}$ and $\bar{\mathbf{H}}$ are null matrices, i.e. if and only if $\mathbf{F}$ and $\mathbf{H}$ in equation (2.6) are chosen in accordance with the conditions in equations (2.11a) and (2.11b). The first term is guaranteed to vanish for any choice of $\hat{\boldsymbol{x}}(0)$ and for any possible $\boldsymbol{x}(0)$ if and only if the observer is stable and k is sufficiently large for the observer to have reached its steady state ($k \geq p$). For the sake of completeness, the special case for which $\bar{\mathbf{A}}^k = \mathbf{0}$ for all $k \geq n$ corresponds to a deadbeat observer, which can be interpreted as the Kalman filter in the absence of noise. □

2.4.2 Optimal observer

Among all the possible LTI observers, we are now interested in the one minimizing the expected value of the norm squared of the state estimation error $\mathbb{E}\left[\boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k)\right]$ at all k. The criterion is equivalent to minimizing the trace of the covariance matrix of the state estimation error $\boldsymbol{\Pi}(k) = \mathbb{E}\left[\boldsymbol{e}(k)\boldsymbol{e}^{\mathrm{T}}(k)\right]$. Recall that the second moment for a random variable is greater than or equal to the corresponding central moment (variance). Equality is achieved only in the case of zero mean. It follows that in order to minimize the sum of the second moments of each

component of the estimation error $\mathbb{E}\left[\boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k)\right]$, the observer needs to be unbiased so that $\mathbb{E}\left[\boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k)\right]$ is the sum of the variances. Conditions (2.11a)-(2.11c) needs then to be satisfied and $\boldsymbol{\Pi}(k+1)$ can be expressed as

$$\boldsymbol{\Pi}(k+1) = \bar{\mathbf{A}}\boldsymbol{\Pi}(k)\bar{\mathbf{A}}^{\mathrm{T}} + \mathbf{K}\mathbf{R}\mathbf{K}^{\mathrm{T}} + \mathbf{Q} \tag{2.17}$$

Supposing the observer is stable, after the transient has vanished, the covariance of the state estimation error will reach a steady state value $\boldsymbol{\Pi}$ given by

$$\boldsymbol{\Pi} = \bar{\mathbf{A}}\boldsymbol{\Pi}\bar{\mathbf{A}}^{\mathrm{T}} + \mathbf{K}\mathbf{R}\mathbf{K}^{\mathrm{T}} + \mathbf{Q} \tag{2.18}$$

Lemma 2.2. *The observer in equation (2.6) minimizes the squared norm of the steady-state state estimation error if and only if the observer is unbiased (Lemma 8.2) and* $\mathbf{K}$ *satisfies the following condition*

$$\bar{\mathbf{A}}\boldsymbol{\Pi}\mathbf{C}^{\mathrm{T}} - \mathbf{K}\mathbf{R} = \mathbf{0} \tag{2.19}$$

Proof. The squared norm of the steady-state state estimation error can be computed as the trace of the covariance matrix $\boldsymbol{\Pi}$. We can then think of choosing $\mathbf{K}$ such that the trace of $\boldsymbol{\Pi}$ is minimized. Imposing the first-order conditions, from equation (2.18) we obtain

$$\frac{\partial \operatorname{trace} \boldsymbol{\Pi}}{\partial \mathbf{K}} = -2\left(\bar{\mathbf{A}}\boldsymbol{\Pi}\mathbf{C}^{\mathrm{T}} - \mathbf{K}\mathbf{R}\right) = \mathbf{0} \tag{2.20}$$

which leads to the optimality condition in equation (2.19). Note how the latter is valid for any kind of input $\boldsymbol{u}$. □

2.4.3 Properties of the residuals

Lemma 2.3. *The expected value of the output residuals of the observer in equation (2.6) is zero at all $k \geq p$ if and only if the observer state is an unbiased estimate of the system state (Lemma 8.2).*

Proof. From the definition of observer residual, we can write

$$\boldsymbol{\epsilon}(k) = \boldsymbol{y}(k) - \hat{\boldsymbol{y}}(k) = \mathbf{C}\boldsymbol{e}(k) + \boldsymbol{w}_m(k) \tag{2.21}$$

Taking the expectation (ensemble average) of both sides of (2.21) we obtain

$$\mathbb{E}\left[\boldsymbol{\epsilon}(k)\right] = \mathbf{C}\mathbb{E}\left[\boldsymbol{e}(k)\right] \tag{2.22}$$

and for the same conditions that guarantee that the observer in (2.6) is unbiased (Lemma 8.2), $\mathbb{E}\left[\boldsymbol{\epsilon}(k)\right] = \mathbf{0}$ at all $k \geq p$, i.e. for all k in steady state. □

Lemma 2.4. *The output residuals of the observer in equation (2.6) in steady state form a white sequence, i.e. $\mathbb{E}\left[\boldsymbol{\epsilon}(k+j)\boldsymbol{\epsilon}^{\mathrm{T}}(k)\right]$ for $k \geq p$ is zero for $j \geq 1$, if and only if the observer is optimal (Lemma 8.3).*

Proof. We prove here Lemma 8.5 for $j = 1$, the proof for other values of j follows the same lines. From equations (2.21) and (2.13), considering the unbiased observer of Lemma 8.2 we

can write

$$\begin{aligned}\mathbb{E}\left[\boldsymbol{\epsilon}(k+1)\boldsymbol{\epsilon}^{\mathrm{T}}(k)\right] &= \mathbb{E}\left[\left(\mathbf{C}\boldsymbol{e}(k+1)+\boldsymbol{w}_m(k+1)\right)\left(\mathbf{C}\boldsymbol{e}(k)+\boldsymbol{w}_m(k)\right)^{\mathrm{T}}\right]\\ &= \mathbb{E}\Big[\big(\mathbf{C}\bar{\mathbf{A}}\boldsymbol{e}(k)+\mathbf{C}\boldsymbol{w}_p(k)-\mathbf{C}\mathbf{K}\mathbf{w}_m(k)\\ &\qquad +\boldsymbol{w}_m(k+1)\big)\left(\mathbf{C}\boldsymbol{e}(k)+\boldsymbol{w}_m(k)\right)^{\mathrm{T}}\Big]\\ &= \mathbf{C}\left(\bar{\mathbf{A}}\boldsymbol{\Pi}(k)\mathbf{C}^{\mathrm{T}}-\mathbf{K}\mathbf{R}\right)+\mathbf{C}\left(\bar{\mathbf{N}}\boldsymbol{\Pi}(k)\mathbf{C}^{\mathrm{T}}-\mathbf{M}_2\mathbf{R}\right)\boldsymbol{u}(k) \qquad (2.23)\end{aligned}$$

In steady state($k \geq p$), $\boldsymbol{\Pi}(k)$ tends to $\boldsymbol{\Pi}$ given by equation (2.18), yielding

$$\mathbb{E}\left[\boldsymbol{\epsilon}(k+1)\boldsymbol{\epsilon}^{\mathrm{T}}(k)\right] = \mathbf{C}\left(\bar{\mathbf{A}}\boldsymbol{\Pi}\mathbf{C}^{\mathrm{T}}-\mathbf{K}\mathbf{R}\right) \qquad (2.24)$$

which vanishes under the optimality conditions (2.19). □

Lemma 2.5. *The expected value of the norm squared of the output residuals of the observer in equation (2.6) in steady state is minimized if and only if the observer is optimal (Lemma 8.3).*

Proof. Recalling equation (2.21), we obtain

$$\begin{aligned}\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{\epsilon}^{\mathrm{T}}(k)\right] &= \mathbb{E}\left[\left(\mathbf{C}\boldsymbol{e}(k)+\boldsymbol{w}_m(k)\right)\left(\mathbf{C}\boldsymbol{e}(k)+\boldsymbol{w}_m(k)\right)^{\mathrm{T}}\right]\\ &= \mathbf{C}\mathbb{E}\left[\boldsymbol{e}(k)\boldsymbol{e}^{\mathrm{T}}(k)\right]\mathbf{C}^{\mathrm{T}}+\mathbb{E}\left[\boldsymbol{w}_m(k)\boldsymbol{w}_m^{\mathrm{T}}(k)\right]\\ &= \mathbf{C}\boldsymbol{\Pi}\mathbf{C}^{\mathrm{T}}+\mathbf{R} \qquad (2.25)\end{aligned}$$

Note that $\mathbb{E}\left[\boldsymbol{\epsilon}^{\mathrm{T}}(k)\boldsymbol{\epsilon}(k)\right] = \operatorname{trace}\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{\epsilon}^{\mathrm{T}}(k)\right]$, therefore it is sufficient to prove that the latter is minimized. Linearity of the trace operator allows us to write

$$\begin{aligned}\operatorname{trace}\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{\epsilon}^{\mathrm{T}}(k)\right] &= \operatorname{trace}\left(\mathbf{C}\boldsymbol{\Pi}\mathbf{C}^{\mathrm{T}} + \mathbf{R}\right)\\ &= \operatorname{trace}\mathbf{C}\boldsymbol{\Pi}\mathbf{C}^{\mathrm{T}} + \operatorname{trace}\mathbf{R}\end{aligned} \tag{2.26}$$

The second term on the right hand-side of equation (2.26) is given. We then need to prove that the first term is minimum if $\operatorname{trace}\boldsymbol{\Pi}$ is minimized. The trace of a matrix is invariant to similarity transformations, therefore without loss of generality we can transform equation (2.26) into a new coordinate frame (denoted by superscript prime) where $\boldsymbol{\Pi}$ is diagonal

$$\operatorname{trace}\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{\epsilon}^{\mathrm{T}}(k)\right] = \operatorname{trace}\mathbf{C}'\boldsymbol{\Pi}\left(\mathbf{C}'\right)^{\mathrm{T}} + \operatorname{trace}\mathbf{R}' \tag{2.27}$$

where $\operatorname{trace}\mathbf{R}' = \operatorname{trace}\mathbf{R}$. The relevant term to be minimized can be rewritten as

$$\operatorname{trace}\mathbf{C}'\boldsymbol{\Pi}\left(\mathbf{C}'\right)^{\mathrm{T}} = \sum_{j=1}^{n}\left(\sum_{i=1}^{q} c_{ij}^2\pi_j\right) = \sum_{j=1}^{n}\left(\pi_j\sum_{i=1}^{q} c_{ij}^2\right) \tag{2.28}$$

where c_{ij}'s are the entries of $\mathbf{C}$ and π_j's are the diagonal entries of $\boldsymbol{\Pi}$. The former are given and $c_{ij}^2 \geq 0$ for all i, j, hence $\gamma_j = \sum_{i=1}^{q} c_{ij}^2 \geq 0$. Since $\boldsymbol{\Pi}$ is a covariance matrix and its diagonal entries represent variances, $\pi_j \geq 0$ for all j. As a result

$$\operatorname{trace}\mathbf{C}'\boldsymbol{\Pi}\left(\mathbf{C}'\right)^{\mathrm{T}} = \sum_{j=1}^{n}\pi_j\gamma_j \tag{2.29}$$

is minimized if every π_j is minimum, as guaranteed only by the optimal observer in Lemma 8.3.

□

Lemma 2.6. *The output residuals of the observer in equation (2.6) in steady state are orthogonal to the current and past input values, i.e.* $\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{u}^{\mathrm{T}}(k-j)\right]$ *for all* $k \geq p$ *is zero for* $j \geq 0$*, if and only if the observer is unbiased (Lemma 8.2).*

Proof. For $j = 0$, starting from equation (2.21), we can write

$$\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{u}^{\mathrm{T}}(k)\right] = \mathbf{C}\mathbb{E}\left[\boldsymbol{e}(k)\right]\boldsymbol{u}^{\mathrm{T}}(k) \tag{2.30}$$

which vanishes for zero-mean output residuals, i.e. if and only if the observer state is an unbiased estimate for the system state (Lemma 8.4). Similarly, for $j > 0$ we obtain

$$\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{u}^{\mathrm{T}}(k-j)\right] = \mathbf{C}\mathbb{E}\left[\boldsymbol{e}(k)\right]\boldsymbol{u}^{\mathrm{T}}(k-j) \tag{2.31}$$

with the same conclusion. □

Lemma 2.7. *The output residuals of the observer in (2.6) in steady state are orthogonal to the past output values, i.e.* $\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{y}^{\mathrm{T}}(k-j)\right]$ *for* $k \geq p$ *is zero for* $j \geq 1$*, if and only if the observer is optimal (Lemma 8.3).*

Proof. We prove Lemma 8.7 below for $j = 1$, the proof for other values of j follows the same lines, therefore it is omitted. From equation (2.21) and considering the unbiased observer of Lemma 8.2, we can write

$$\begin{aligned}
\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{y}^{\mathrm{T}}(k-1)\right] &= \mathbb{E}\left[\left(\mathbf{C}\boldsymbol{e}(k) + \boldsymbol{w}_m(k)\right)\boldsymbol{y}^{\mathrm{T}}(k-1)\right] = \mathbb{E}\left[\mathbf{C}\boldsymbol{e}(k)\boldsymbol{y}^{\mathrm{T}}(k-1)\right] \\
&= \mathbf{C}\mathbb{E}\Big[\left(\bar{\mathbf{A}}\boldsymbol{e}(k-1) + \boldsymbol{w}_p(k-1) - \mathbf{K}\boldsymbol{w}_m(k-1)\right)\left(\mathbf{C}\hat{\boldsymbol{x}}(k-1) + \mathbf{C}\boldsymbol{e}(k-1)\right. \\
&\qquad \left. + \mathbf{D}\boldsymbol{u}(k-1) + \boldsymbol{w}_m(k-1)\right)^{\mathrm{T}}\Big] \\
&= \mathbf{C}\bar{\mathbf{A}}\mathbb{E}\left[\boldsymbol{e}(k-1)\hat{\boldsymbol{x}}^{\mathrm{T}}(k-1)\right]\mathbf{C}^{\mathrm{T}} + \mathbf{C}\left(\bar{\mathbf{A}}\boldsymbol{\Pi}(k-1)\mathbf{C}^{\mathrm{T}} - \mathbf{K}\mathbf{R}\right)
\end{aligned} \tag{2.32}$$

In steady state($k \geq p$), $\mathbf{\Pi}(k)$ tends to $\mathbf{\Pi}$ given by equation (2.18) and we obtain

$$\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{y}^{\mathrm{T}}(k-1)\right] = \mathbf{C}\bar{\mathbf{A}}\mathbb{E}\left[\boldsymbol{e}(k)\hat{\boldsymbol{x}}^{\mathrm{T}}(k)\right]\mathbf{C}^{\mathrm{T}} + \mathbf{C}\left(\bar{\mathbf{A}}\mathbf{\Pi}\mathbf{C}^{\mathrm{T}} - \mathbf{K}\mathbf{R}\right) \tag{2.33}$$

which vanishes for the optimal observer of Lemma 8.3. The first term on the right-hand side of equation (2.33) is null because so is $\mathbb{E}\left[\boldsymbol{e}(k)\hat{\boldsymbol{x}}^{\mathrm{T}}(k)\right]$ by the orthogonality argument arising from the least-squares optimality criterion, the last term vanishes thanks to the optimality condition in equation (2.19). □

2.5 OKID core equation

The main difficulty in state-space model identification is the nonlinearity of the problem. Although the model in equation (1.3) is linear, from the viewpoint of system identification both the state $\boldsymbol{x}$ and the matrices $\mathbf{A}$ and $\mathbf{C}$ are unknowns. Their products in equation (1.3) make the identification problem nonlinear. OKID relies on an observer to remove such nonlinearity.

Defining

$$\mathbf{G} = \begin{bmatrix} \mathbf{H} & \mathbf{K} \end{bmatrix} \tag{2.34a}$$

$$\boldsymbol{v}_x(k) = \begin{bmatrix} \boldsymbol{u}(k) \\ \boldsymbol{y}(k) \end{bmatrix} \tag{2.34b}$$

the most general form of LTI observer, equation (2.6), becomes

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{F}\hat{\boldsymbol{x}}(k) + \mathbf{G}\boldsymbol{v}_x(k) \tag{2.35a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \mathbf{D}\boldsymbol{u}(k)) \tag{2.35b}$$

where $\boldsymbol{v}_x$ can be looked as the input to the observer state equation. Indeed, $\boldsymbol{u}$ and $\boldsymbol{y}$ are the given signals driving the observer state dynamics. Recalling the definition of observer output residual in equation (2.8), equation (2.36) can also be written as

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{F}\hat{\boldsymbol{x}}(k) + \mathbf{G}\boldsymbol{v}_x(k) \tag{2.36a}$$

$$\boldsymbol{y}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \mathbf{D}\boldsymbol{u}(k) + \boldsymbol{\epsilon}(k) \tag{2.36b}$$

Propagating equation (2.36a) forward in time by p time steps and then shifting the time index backward by $p+1$, we obtain

$$\hat{\boldsymbol{x}}(k) = \mathbf{F}^p\hat{\boldsymbol{x}}(k-p) + \mathbf{T}\boldsymbol{z}(k) \tag{2.37}$$

where

$$\boldsymbol{z}(k) = \begin{bmatrix} \boldsymbol{v}_x(k-1) \\ \boldsymbol{v}_x(k-2) \\ \vdots \\ \boldsymbol{v}_x(k-p) \end{bmatrix} \tag{2.38}$$

$$\mathbf{T} = \begin{bmatrix} \mathbf{G} & \mathbf{FG} & \dots & \mathbf{F}^{p-2}\mathbf{G} & \mathbf{F}^{p-1}\mathbf{G} \end{bmatrix} \tag{2.39}$$

As will be proven later, the stability of the observer guarantees that $\mathbf{F}^p$ becomes negligible for sufficiently large values of p ($p >> n$). Equation (2.37) yields then the following relation expressing the current state as a function of the sole past input and output values

$$\hat{\boldsymbol{x}}(k) = \mathbf{T}\boldsymbol{z}(k) \tag{2.40}$$

Plugging equation (2.40) into equation (2.36b), we obtain

$$\boldsymbol{y}(k) = \boldsymbol{\Phi}\boldsymbol{v}(k) + \boldsymbol{\epsilon}(k) \tag{2.41}$$

where $\boldsymbol{\Phi}$ and $\boldsymbol{v}(k)$ are augmented versions of $\mathbf{T}$ and $\boldsymbol{z}(k)$ to take into account the direct influence of the input on the output through the $\mathbf{D}$ matrix, i.e.

$$\boldsymbol{v}(k) = \begin{bmatrix} \boldsymbol{u}(k) \\ \boldsymbol{z}(k) \end{bmatrix} = \begin{bmatrix} \boldsymbol{u}(k) \\ \boldsymbol{u}(k-1) \\ \boldsymbol{y}(k-1) \\ \boldsymbol{u}(k-2) \\ \boldsymbol{y}(k-2) \\ \vdots \\ \boldsymbol{u}(k-p) \\ \boldsymbol{y}(k-p) \end{bmatrix} \tag{2.42}$$

$$\boldsymbol{\Phi} = \begin{bmatrix} \mathbf{D} & \mathbf{CG} & \mathbf{CFG} & \dots & \mathbf{CF}^{p-2}\mathbf{G} & \mathbf{CF}^{p-1}\mathbf{G} \end{bmatrix} \tag{2.43}$$

Equation (2.41) relates the input and output, without the state appearing explicitly. Note that $\boldsymbol{\Phi}$ contains the sequence of Markov parameters of the observer in the form of equation (2.35). The Markov parameters of a discrete-time linear model correspond to its unit pulse response, i.e. they are the response to a unit pulse in the input to the system, and they have the properties of being unique for a given system. The Markov parameters (or unit pulse response) are indeed often referred to as the signature of the model. Equation (2.41) can be written for each time step $k \geq p$ of the measured data record, to obtain the following system of equations in matrix form

$$\mathbf{Y} = \boldsymbol{\Phi}\mathbf{V} + \mathbf{E} \tag{2.44}$$

where

$$\mathbf{Y} = \begin{bmatrix} \boldsymbol{y}(p) & \boldsymbol{y}(p+1) & \dots & \boldsymbol{y}(l-1) \end{bmatrix} \tag{2.45a}$$

$$\mathbf{V} = \begin{bmatrix} \boldsymbol{v}(p) & \boldsymbol{v}(p+1) & \dots & \boldsymbol{v}(l-1) \end{bmatrix} \tag{2.45b}$$

$$\mathbf{E} = \begin{bmatrix} \boldsymbol{\epsilon}(p) & \boldsymbol{\epsilon}(p+1) & \dots & \boldsymbol{\epsilon}(l-1) \end{bmatrix} \tag{2.45c}$$

Equation (2.44) is the OKID core equation. $\mathbf{Y}$ and $\mathbf{V}$ are known (from measurements), $\mathbf{\Phi}$ and $\mathbf{E}$ are not. By having $l > p\,(m+q)+m$, the set of equations in (2.44) is overdetermined and, considering $\mathbf{E}$ as an error term, it is possible to find the least-squares (LS) solution

$$\tilde{\mathbf{\Phi}} = \mathbf{Y}\mathbf{V}^{\mathrm{T}}\left(\mathbf{V}\mathbf{V}^{\mathrm{T}}\right)^{-1} = \mathbf{Y}\mathbf{V}^{\dagger} \tag{2.46}$$

where † denotes the Moore-Penrose pseudoinverse of a matrix. Associated with the LS solution are the LS residuals given by

$$\tilde{\mathbf{E}} = \mathbf{Y} - \tilde{\mathbf{\Phi}}\mathbf{V} \tag{2.47}$$

The keystone of OKID is the relationship between the LS solution to the OKID core equation and the Kalman filter defined in equations (2.6) and (2.7), summarized in the two following facts:

Fact 1 The LS residuals in equation (2.47) are the output residuals of the Kalman filter.

Proof. Two alternative proofs are provided. The first one is a new result, whose main benefit is the simplicity of its interpretation.

Proof by Minimization

For simplicity, consider the case with $q = 1$ (single-output system). By definition, the LS

solution in equation (2.46) minimizes the norm of $\tilde{\mathbf{E}}$, i.e. the sum of the squares of the residuals

$$\tilde{\mathbf{\Phi}} = \mathbf{Y}\mathbf{V}^{\dagger} \iff \min \sum_{k=p}^{l-1} \boldsymbol{\epsilon}^2(k) \tag{2.48}$$

Since the assumptions in the problem statement make the process of equation (1.3) stationary, then by the ergodic property we can estimate the ensemble average of the residuals squared by their time average over a sufficiently long record. Assuming l is large and dividing the right-hand side of equation (2.48) by $l - p$, we recognize that for all $k = p, p+1, ..., l-1$

$$\lim_{l \to +\infty} \frac{1}{l-p} \sum_{i=p}^{l-1} \boldsymbol{\epsilon}^2(i) = \mathbb{E}\left[\boldsymbol{\epsilon}^2(k)\right] \tag{2.49}$$

Hence, in the limit of $l \to \infty$, equation (2.48) becomes

$$\tilde{\mathbf{\Phi}} = \mathbf{Y}\mathbf{V}^{\dagger} \iff \min \mathbb{E}\left[\boldsymbol{\epsilon}^2(k)\right] \tag{2.50}$$

Since the Kalman filter is the (only) linear system of equation (2.6) minimizing the expected value of the residuals squared, it is proven that the residuals $\tilde{\mathbf{E}}$ of the LS problem are the Kalman output residuals. The extension to the multi-output case is straightforward and therefore omitted.

Proof by Orthogonality

Right-multiplying equation (2.44) by $\mathbf{V}^{\mathrm{T}}$ and replacing $\mathbf{\Phi}$ with its LS estimate $\tilde{\mathbf{\Phi}}$, we obtain

$$\mathbf{Y}\mathbf{V}^{\mathrm{T}} = \mathbf{Y}\mathbf{V}^{\mathrm{T}} \left(\mathbf{V}\mathbf{V}^{\mathrm{T}}\right)^{-1} \mathbf{V}\mathbf{V}^{\mathrm{T}} + \mathbf{E}\mathbf{V}^{\mathrm{T}} = \mathbf{Y}\mathbf{V}^{\mathrm{T}} + \mathbf{E}\mathbf{V}^{\mathrm{T}} \tag{2.51}$$

which implies that $EV^{\mathrm{T}} = 0$. From the definition of $\boldsymbol{v}(k)$, we conclude that

$$\sum_{k=p}^{l-1} \boldsymbol{\epsilon}(k)\boldsymbol{u}^{\mathrm{T}}(k-j) = \mathbf{0} \qquad j = 0, 1, ..., p \tag{2.52a}$$

$$\sum_{k=p}^{l-1} \boldsymbol{\epsilon}(k)\boldsymbol{y}^{\mathrm{T}}(k-j) = \mathbf{0} \qquad j = 1, 2, ..., p \tag{2.52b}$$

Since the assumptions in the problem statement make the process of equation (1.3) stationary, then by the ergodic property we can estimate the ensemble average of each entry of the products between the current residual and the current input or past input and output by their time average over a sufficiently long record. Assuming l is large and dividing equation (2.52) by $l - p$, we recognize the left-hand side as the time average of each entry of $\boldsymbol{\epsilon}(k)\boldsymbol{u}^{\mathrm{T}}(k-j)$ and $\boldsymbol{\epsilon}(k)\boldsymbol{y}^{\mathrm{T}}(k-j)$. The ergodic property leads to the conclusion that, for all $k \geq p$,

$$\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{u}(k-j)^{\mathrm{T}}\right] = \mathbf{0} \qquad j = 0, 1, ..., p \tag{2.53a}$$

$$\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{y}(k-j)^{\mathrm{T}}\right] = \mathbf{0} \qquad j = 1, 2, ..., p \tag{2.53b}$$

The residuals $\boldsymbol{\epsilon}$ of the LS problem of equation (2.44) are then orthogonal to the current and past input values and to the past output values. Since the Kalman filter is the only linear system in the form of equation (2.6) that features such property, it is proven that the residuals of the LS problem are the Kalman output residuals. □

Fact 2 The LS solution in equation (2.46) is an estimate for sequence of Markov parameters of the Kalman filter in the form of equation (2.36), i.e. the true value of $\boldsymbol{\Phi}$ is

$$\boldsymbol{\Phi} = \begin{bmatrix} \mathbf{D} & \mathbf{C}\bar{\mathbf{B}} & \mathbf{C}\bar{\mathbf{A}}\bar{\mathbf{B}} & ... & \mathbf{C}\bar{\mathbf{A}}^{p-2}\bar{\mathbf{B}} & \mathbf{C}\bar{\mathbf{A}}^{p-1}\bar{\mathbf{B}} \end{bmatrix} \tag{2.54}$$

where $\bar{\mathbf{A}} = \mathbf{A} - \mathbf{KC}$ and $\bar{\mathbf{B}} = \begin{bmatrix} \mathbf{B} - \mathbf{KD} & \mathbf{K} \end{bmatrix}$.

Proof. Fact 2 is a corollary of Fact 1. By the same argument of uniqueness of the Kalman filter, the linear system in equation (2.6) that generated white/orthogonal residuals must be the Kalman filter and therefore $\mathbf{\Phi}$ contains the Markov parameters of such Kalman filter. □

Additionally, it is known that a LS estimator is asymptotically unbiased when the equation error is white and zero-mean. Since in the LS problem of equation (2.44) the error term is given by the Kalman residual $\boldsymbol{\epsilon}$, which is known to be a zero-mean white process, then $\tilde{\mathbf{\Phi}}$ is an asymptotically unbiased estimate for the Markov parameters of the Kalman filter in the form of equation (2.36) .

Having established that the observer used in the derivation of equation (2.44) is the Kalman filter, then its stability is guaranteed (see for example Reference 21) and the assumption of $\mathbf{F}^p \approx \mathbf{0}$ for large p to derive equation (2.40) and following is justified.

To summarize, the core of OKID consists in (i) using an observer to implicitly estimate the true state of the system to be identified, removing the initial nonlinearity of the identification problem; (ii) exploiting the LS solution to guarantee that such an observer is not a random observer with random $\mathbf{F}$, $\mathbf{H}$, $\mathbf{K}$ matrices but the Kalman filter, whose matrices satisfy equation (2.7) and therefore are closely related to the matrices of the system to be identified. Said link is crucial in OKID. Note how the use of an observer, which eventually turns out to be the Kalman filter, has been defined as *implicit* because the matrices of the observer are not even known before identification. Nevertheless, the sole structure of the LTI observer allows one to derive the OKID core equation whose LS solution has the same properties of the underlying

Kalman filter and provides some information on said filter, such as its Markov parameters and its output residuals.

Fact 1 and Fact 2 above are two sides of the same coin. Nevertheless they deserve to be mentioned separately because they lead to different OKID-based algorithms. Fact 1 is at the basis of the algorithms presented in section 3.4 and identifying the Kalman filter via its output residuals, estimated as the LS residuals of equation (2.44)

$$\tilde{\mathbf{E}} = \begin{bmatrix} \tilde{\boldsymbol{\epsilon}}(p) & \tilde{\boldsymbol{\epsilon}}(p+1) & \dots & \tilde{\boldsymbol{\epsilon}}(l-2) & \tilde{\boldsymbol{\epsilon}}(l-1) \end{bmatrix} \tag{2.55}$$

Fact 2 is at the core of the algorithms described in section 3.3, which identify the Kalman filter via the LS estimate of its Markov parameters

$$\tilde{\boldsymbol{\Phi}} = \begin{bmatrix} \tilde{\boldsymbol{\Phi}}_0 & \tilde{\boldsymbol{\Phi}}_1 & \dots & \tilde{\boldsymbol{\Phi}}_{p-1} & \tilde{\boldsymbol{\Phi}}_p \end{bmatrix} \tag{2.56}$$

where $\tilde{\boldsymbol{\Phi}}_0$ is the estimate of $\mathbf{D}$ and $\tilde{\boldsymbol{\Phi}}_j$ is the estimate of $\mathbf{C}\bar{\mathbf{A}}^{j-1}\bar{\mathbf{B}}$, $j = 1, 2, \dots, p$.

2.6 Conclusions

In this chapter, the problem of linear system identification from noisy input-output data has been stated. The concept of linear observer has been presented, amphasizing teh fact that the type of observer at the core of OKID is the optimal linear-time-invariant observer, also known as steady-state Kalman filter (although in this work it is often simply referred to as Kalman filter). The properties of the steady-state Kalman filter related to OKID have been proven rigorously and explicitly, without mixing their proofs with the ones for the time-varying Kalman filter, as done in Reference 1, which is an unnecessary source of confusion. The OKID core equation has

been derived and the key link between its least-squares (LS) solution and the Kalman filter has been proven in two different ways. Besides the traditional argument based on orthogonality of the residuals, the more intuitive proof based on the fact that the Kalman filter output residuals are indeed minimized in a LS sense is given.

Chapter 2 provides then the theoretical background for OKID and in particular the first step of it, which is common to all OKID-based algorithms.

Chapter 3

OKID algorithms

3.1 Introduction

The OKID equation presented in the previous chapter, equation (2.44), is at the core of the approach to system identification described in this work. However, its LS solution does not complete the identification process. It yields some preliminary information on the Kalman filter associated with the system to be identified and the noise embedded in the measured input-output data. A second step is necessary. For the sake of clarity, the OKID core equation provides the LS estimates for the Kalman filter output residuals, equation (2.55), and Markov parameters, equation (2.56). In the presentation of the theory behind the second step of the OKID approach, the notation will for simplicity refer to the true values of Markov parameters $\mathbf{\Phi}_j$ and output residuals $\boldsymbol{\epsilon}$. Nevertheless it must be born in mind that those are not available and the OKID algorithms described in sections 3.3 and 3.4 are implemented replacing the true values with their estimates $\tilde{\mathbf{\Phi}}_j$ and $\tilde{\boldsymbol{\epsilon}}$ obtained from the OKID core equation.

Before plunging into the details of the algorithms, two equivalent forms of the Kalman filter introduced in section 2.3 are presented as they help understand the intimate nature of the OKID approach. The rest of the chapter is divided into two main parts illustrating two different ways to complete OKID. The first part reviews the traditional OKID methods, i.e. OKID/ERA (OKID followed by eigensystem realization algorithm) and OKID/ERA-DC (OKID followed by ERA with data correlation), both based on the estimate of the Kalman filter Markov parameters coming from the OKID core equation. A new interpretation to the sequences of Markov parameters involved in OKID is also provided and is helpful in chapter 5 to better illustrate the general framework that OKID offers for system identification. Although not completely novel, the family of OKID/ERA and OKID/ERA-DC algorithms is here expanded to provide a comprehensive view of OKID. The second part is a completely new development, presented at the 24$^{\text{th}}$ AAS/AIAA Space Flight Mechanics Meeting in Santa Fe, NM, in 2014 (Reference 25), and gives rise to several algorithms based on the estimate of the Kalman filter output residuals. Two of them, OKID/DI (OKID followed by deterministic intersection algorithm) and OKID/DP (OKID followed by deterministic projection algorithm), will also be illustrated via examples. As a historical note, OKID/ERA and OKID/ERA-DC were the first OKID-based algorithms to be developed (Reference 20), with co-operation between NASA and Columbia University leading to the distribution of the algorithm in the software package known as SOCIT (Reference 26). The original goal was to identify lightly damped structures, quite typical in aerospace engineering applications, characterized by slow decay of the response to initial conditions, which leads to high computational effort with traditional methods. The use of an observer allows to modify the decay rate of the original system response, compressing the data for higher computational efficiency. The advantage is extreme in the absence of noise, where the observer turns out to be a deadbeat observer, i.e. an observer whose transient lasts exactly n time steps and becomes identically equal to zero afterwards (References 27, 28). The development of the new family of

OKID-based algorithms is more recent and responds to the need for an alternative to ERA, in an attempt to extend OKID to bilinear systems, for which ERA is not available (see chapter 9). The resulting approach based on the estimate of the Kalman filter residuals does not only provide new OKID-based algorithms for the linear case, but also has the benefit of casting new light on the underlying principles of OKID, leading to a broader and more general framework for OKID (chapter 5), with extension to output-only system identification (chapter 4), too.

3.2 Kalman filter forms

Before plunging into the details of the identification algorithms, it is worth pointing out how two different representations of the Kalman filter are used in OKID. In both of them, $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ indicate the state and the output of the Kalman filter, respectively. This section conceptually belongs to chapter 5, since the idea that all possible OKID algorithms indeed completely identify a Kalman filter is a somewhat new concept arising from the desire to generalize the formulation of OKID.

3.2.1 Kalman filter in innovation form

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{A}\hat{\boldsymbol{x}}(k) + \mathbf{B}\boldsymbol{u}(k) + \mathbf{K}\boldsymbol{\epsilon}(k) \tag{3.1a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \mathbf{D}\boldsymbol{u}(k) \tag{3.1b}$$

The innovation form expresses the Kalman filter as a state-space model with input given by $\boldsymbol{u}$ and $\boldsymbol{\epsilon}$. Whereas the former is also the input to the original system in equation (1.3), the latter is, in accordance with equation (2.8), the sequence of the output residuals of the Kalman filter, i.e. the difference between the measurement of the output of the original system and the output

predicted by equation (3.1b). Recall how the output residuals form a zero-mean white process (Lemmas 8.4 and 8.5). The output residual is the actual new piece of information that the output measurement provides to the Kalman filter to provide the optimal estimate for the true state of the system at the next time step, hence the name *innovation form*. It is worth noting also how the state space model in equation (3.1) explicitly includes the matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$ of the original system to be identified.

For the sake of clarity, the innovation form in equation (3.1) can be written in the more compact form

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{A}\hat{\boldsymbol{x}}(k) + \mathbf{B}'\boldsymbol{v}_x'(k) \tag{3.2a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \mathbf{D}\boldsymbol{u}(k) \tag{3.2b}$$

where

$$\mathbf{B}' = \begin{bmatrix} \mathbf{B} & \mathbf{K} \end{bmatrix} \tag{3.3a}$$

$$\boldsymbol{v}_x'(k) = \begin{bmatrix} \boldsymbol{u}(k) \\ \\ \boldsymbol{\epsilon}(k) \end{bmatrix} \tag{3.3b}$$

$$\tag{3.3c}$$

3.2.2 Kalman filter in bar form

From equation (3.1a), recalling the definition of $\boldsymbol{\epsilon}$ in equation (2.8) we can write

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{A}\hat{\boldsymbol{x}}(k) + \mathbf{B}\boldsymbol{u}(k) + \mathbf{K}\left(\boldsymbol{y}(k) - \hat{\boldsymbol{y}}(k)\right) \tag{3.4}$$

Plugging equation (3.1b) into (3.4), we further obtain

$$\hat{\boldsymbol{x}}(k+1) = \bar{\mathbf{A}}\hat{\boldsymbol{x}}(k) + (\mathbf{B} - \mathbf{KD})\,\boldsymbol{u}(k) + \mathbf{K}\boldsymbol{y}(k) \tag{3.5a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \mathbf{D}\boldsymbol{u}(k) \tag{3.5b}$$

The state space model in equation (3.5) is equivalent to the one in equation (3.1) in the sense that both provide the same time histories for $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$. However, they are driven by different inputs, as the model in equation (3.5) has $\boldsymbol{u}$ and $\boldsymbol{y}$ as forcing functions. Even more importantly, the dynamics of equation (3.5) is governed by

$$\bar{\mathbf{A}} = \mathbf{A} - \mathbf{KC} \tag{3.6}$$

instead of $\mathbf{A}$, hence the name *bar form*. Nevertheless, the overall effect of different input and system matrix on the Kalman filter state history is the same.

Similar to equation (3.1a), the forcing terms in equation (3.5a) can be grouped together to obtain

$$\hat{\boldsymbol{x}}(k+1) = \bar{\mathbf{A}}\hat{\boldsymbol{x}}(k) + \bar{\mathbf{B}}\boldsymbol{v}_x(k) \tag{3.7a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \mathbf{D}\boldsymbol{u}(k) \tag{3.7b}$$

where

$$\bar{\mathbf{B}} = \begin{bmatrix} \mathbf{B} - \mathbf{KD} & \mathbf{K} \end{bmatrix} \tag{3.8a}$$

$$\boldsymbol{v}_x(k) = \begin{bmatrix} \boldsymbol{u}(k) \\ \boldsymbol{y}(k) \end{bmatrix} \tag{3.8b}$$

Equation (3.7) is indeed the one used to derive the OKID core equation in section 2.5.

The key features that put the bar form at the core of the derivation of the OKID equation are the following two. First of all, both inputs are known (from measurements), as opposed to $\boldsymbol{\epsilon}$ in equation (3.1), making it possible to derive an equation involving solely measured data. Secondly, the fact that the Kalman filter is stabilizing makes $\bar{\mathbf{A}}$ in equation (3.5) more stable than $\mathbf{A}$ in equation (3.1), generally resulting in a faster convergence to zero of the state dependent term in equation (2.37). In other words, the bar form allows one to select a smaller value for p in the OKID core equation. The advantage is extreme in the case of noise-free data, where the Kalman filter becomes a deadbeat observer, i.e. $\mathbf{K}$ is such that $\bar{\mathbf{A}}^p = \mathbf{0}$ for p as low as n regardless the characteristics of the transient of the original system to be identified.

3.3 Identification via Kalman filter Markov parameters

This is the traditional approach of OKID (References 1, 20). The Kalman filter at the core of OKID can be identified from its Markov parameter sequence $\boldsymbol{\Phi}$, whose estimate $\tilde{\boldsymbol{\Phi}}$ is obtained from the LS solution of the OKID core equation. For convenience, partition $\boldsymbol{\Phi}$ as follows

$$\boldsymbol{\Phi} = \begin{bmatrix} \boldsymbol{\Phi}_0 & \boldsymbol{\Phi}_1 & \dots & \boldsymbol{\Phi}_p \end{bmatrix} \tag{3.9}$$

where $\boldsymbol{\Phi}_0 \in \mathbb{R}^{q \times q}$ and $\boldsymbol{\Phi}_j \in \mathbb{R}^{q \times m+q}$, $j = 1, 2, \dots, p$. Comparing equations (2.43) and (2.54), it is clear that

$$\boldsymbol{\Phi}_0 = \mathbf{D} \tag{3.10a}$$

$$\boldsymbol{\Phi}_j = \mathbf{C}\bar{\mathbf{A}}^{j-1}\bar{\mathbf{B}} \qquad \text{for } j = 1, 2, \dots, p \tag{3.10b}$$

As the sequence of Markov parameters of a dynamic system corresponds to its unit pulse response, one can complete the identification of the system matrices via algorithms capable of extracting the model matrices (also known as realization) from the system unit pulse response, such as ERA or ERA-DC (appendix A).

In the original OKID/ERA and OKID/ERA-DC algorithms (Reference 20), before applying ERA or ERA-DC, the following sequences have to be found

$$\mathbf{\Psi}^{(s)} = \begin{bmatrix} \mathbf{D} & \mathbf{CB} & \mathbf{CAB} & \dots & \mathbf{CA}^{N-1}\mathbf{B} \end{bmatrix} \tag{3.11a}$$

$$\mathbf{\Psi}^{(g)} = \begin{bmatrix} \mathbf{CK} & \mathbf{CAK} & \dots & \mathbf{CA}^{N-1}\mathbf{K} \end{bmatrix} \tag{3.11b}$$

that Reference 20 calls the system Markov parameters and the gain Markov parameters, respectively. Similarly to what done above for $\mathbf{\Phi}$, partition $\mathbf{\Psi}^{(s)}$ and $\mathbf{\Psi}^{(g)}$ as

$$\mathbf{\Psi}^{(s)} = \begin{bmatrix} \mathbf{\Psi}_0^{(s)} & \mathbf{\Psi}_1^{(s)} & \dots & \mathbf{\Psi}_N^{(s)} \end{bmatrix} \tag{3.12a}$$

$$\mathbf{\Psi}^{(g)} = \begin{bmatrix} \mathbf{\Psi}_1^{(g)} & \mathbf{\Psi}_2^{(g)} & \dots & \mathbf{\Psi}_N^{(g)} \end{bmatrix} \tag{3.12b}$$

so that

$$\mathbf{\Psi}_0^{(s)} = \mathbf{D} \tag{3.13a}$$

$$\mathbf{\Psi}_j^{(s)} = \mathbf{CA}^{j-1}\mathbf{B} \tag{3.13b}$$

$$\mathbf{\Psi}_j^{(g)} = \mathbf{CA}^{j-1}\mathbf{K} \qquad \text{for } j = 1, 2, \dots, N \tag{3.13c}$$

It is remarkable how, by simple algebraic operations, the system and gain Markov parameters can be recovered from the Markov parameters $\mathbf{\Phi}$ estimated from the OKID core equation. First of all, notice that the estimate of the sequence $\mathbf{\Phi}_j$ is finite ($j = 0, 1, ..., p$) only in appearance.

For $j > p$, $\mathbf{C}\bar{\mathbf{A}}^{j-1}\bar{\mathbf{B}}$ can be considered to be equal to $\mathbf{0}$ as assumed in the derivation of the OKID core equation. In other words, the sequence of estimated Markov parameters $\tilde{\mathbf{\Phi}}_j$ can be extended to an arbitrary value $j = N > p$ simply by padding $\tilde{\mathbf{\Phi}}$ in (2.56) with zeros. Defining for convenience

$$\mathbf{\Phi}_j^{(1)} = \mathbf{C}\bar{\mathbf{A}}^{j-1}\left(\mathbf{B} - \mathbf{KD}\right) \tag{3.14a}$$

$$\mathbf{\Phi}_j^{(2)} = \mathbf{C}\bar{\mathbf{A}}^{j-1}\mathbf{K} \tag{3.14b}$$

the conversion can be done as follows

$$\mathbf{\Psi}_0^{(s)} = \mathbf{\Phi}_0^{(1)} \tag{3.15a}$$

$$\mathbf{\Psi}_j^{(s)} = \mathbf{\Phi}_j^{(1)} + \sum_{h=1}^{j} \mathbf{\Phi}_h^{(2)}\mathbf{\Psi}_{j-h}^{(s)} \quad \text{for } j = 1, 2, \ldots, N \tag{3.15b}$$

$$\mathbf{\Psi}_j^{(g)} = \mathbf{\Phi}_j^{(2)} + \sum_{h=1}^{j-1} \mathbf{\Phi}_h^{(2)}\mathbf{\Psi}_{j-h}^{(g)} \quad \text{for } j = 1, 2, \ldots, N \tag{3.15c}$$

The sequences of system and gain Markov parameters can actually be put together as

$$\mathbf{\Psi} = \begin{bmatrix} \mathbf{\Psi}_0^{(s)} & \mathbf{\Psi}_1^{(s)} & \mathbf{\Psi}_1^{(g)} & \mathbf{\Psi}_2^{(s)} & \mathbf{\Psi}_2^{(g)} \ldots & \mathbf{\Psi}_N^{(s)} & \mathbf{\Psi}_N^{(g)} \end{bmatrix} \tag{3.16}$$

which can be interpreted as the sequence of Markov parameters of the Kalman filter in innovation form, equation (3.1). The operation in equation (3.15) corresponds then to the conversion of the Kalman filter Markov parameters from bar form to innovation form.

The traditional OKID/ERA and OKID/ERA-DC algorithms complete then the identification process feeding $\mathbf{\Psi}$ to ERA or ERA-DC to extract the matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{K}$. More precisely, $\mathbf{D}$ is readily available from $\mathbf{\Psi}_0^{(s)}$, whereas the other matrices result from ERA or ERA-DC being

applied to the sequence

$$\mathbf{\Psi}_j = \begin{bmatrix} \mathbf{\Psi}_j^{(s)} & \mathbf{\Psi}_j^{(g)} \end{bmatrix} = \mathbf{C}\mathbf{A}^{j-1} \begin{bmatrix} \mathbf{B} & \mathbf{K} \end{bmatrix} = \mathbf{C}\mathbf{A}^{j-1}\mathbf{B}' \qquad \text{for } j = 1, 2, \ldots, N \tag{3.17}$$

That is the algorithm presented in Reference 1, 20.

A variant consists in using ERA or ERA-DC to find the system matrices only and identify the Kalman gain in a separate step. $\mathbf{\Psi}^{(s)}$ alone allows ERA or ERA-DC to extract $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}$. The latter directly yields $\mathbf{D}$, whereas $\mathbf{A}, \mathbf{B}, \mathbf{C}$ are obtained by applying ERA or ERA-DC on the sequence $\mathbf{\Psi}_j^{(s)}$, $j = 1, 2, \ldots, N$. To find $\mathbf{K}$, the following equation, of straightforward derivation, between the observability matrix of the system and the gain Markov parameters

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A} \\ \vdots \\ \mathbf{C}\mathbf{A}^{N-1} \end{bmatrix} \mathbf{K} = \begin{bmatrix} \mathbf{\Psi}_1^{(g)} \\ \mathbf{\Psi}_2^{(g)} \\ \vdots \\ \mathbf{\Psi}_N^{(g)} \end{bmatrix} \tag{3.18}$$

is solved by LS

$$\mathbf{K} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A} \\ \vdots \\ \mathbf{C}\mathbf{A}^{N-1} \end{bmatrix}^{\dagger} \begin{bmatrix} \mathbf{\Psi}_1^{(g)} \\ \mathbf{\Psi}_2^{(g)} \\ \vdots \\ \mathbf{\Psi}_N^{(g)} \end{bmatrix} \tag{3.19}$$

The approach described so far is the one traditionally adopted, even though the variant just presented did not explicitly appear in the literature. Another alternative way to extract the desired matrices $\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}, \mathbf{K}$ is also possible. Whereas above the identification of the Kalman filter was performed working on the Markov parameters of its innovation form, it is also possible

to apply ERA or ERA-DC directly on the Markov parameters of the Kalman filter in bar form. $\mathbf{D}$ is readily available as usual, and the input to ERA or ERA-DC is then the estimate for $\mathbf{\Phi}_j$, $j = 1, 2, ..., p$, which produces as an output the matrices $\bar{\mathbf{A}}, \bar{\mathbf{B}}, \mathbf{C}$. Relabeling the matrix blocks in the definition of $\bar{\mathbf{B}}$ in equation (3.8a) as

$$\bar{\mathbf{B}} = \begin{bmatrix} \bar{\mathbf{B}}_1 & \bar{\mathbf{B}}_2 \end{bmatrix} \tag{3.20}$$

where $\bar{\mathbf{B}}_1 \in \mathbb{R}^{n \times m}$ and $\bar{\mathbf{B}}_2 \in \mathbb{R}^{n \times q}$, we can complete the identification recovering $\mathbf{K}$, $\mathbf{B}$ and $\mathbf{A}$ from equations (3.8a) and (3.6) as follows

$$\mathbf{K} = \bar{\mathbf{B}}_2 \tag{3.21a}$$

$$\mathbf{B} = \bar{\mathbf{B}}_1 + \mathbf{K}\mathbf{D} \tag{3.21b}$$

$$\mathbf{A} = \bar{\mathbf{A}} + \mathbf{K}\mathbf{C} \tag{3.21c}$$

Even for the identification of the Kalman filter in bar form, the variant based on the estimation of $\mathbf{K}$ via the observability matrix is possible. Instead of running ERA or ERA-DC on the sequence $\mathbf{\Phi}_j$, $j = 1, 2, \ldots, p$, apply it to $\mathbf{\Phi}_j^{(1)}$, $j = 1, 2, \ldots, p$ to find $\mathbf{C}$, $\bar{\mathbf{A}}$, $\bar{\mathbf{B}}^{(1)}$. Similar to equation (3.18), the following equation between the observability matrix of the system and the gain Markov parameters can be derived

$$\begin{bmatrix} \mathbf{C} \\ \mathbf{C}\bar{\mathbf{A}} \\ \vdots \\ \mathbf{C}\bar{\mathbf{A}}^{N-1} \end{bmatrix} \mathbf{K} = \begin{bmatrix} \mathbf{\Phi}_1^{(2)} \\ \mathbf{\Phi}_2^{(2)} \\ \vdots \\ \mathbf{\Phi}_N^{(2)} \end{bmatrix} \tag{3.22}$$

to find $\mathbf{K}$ by LS

$$\mathbf{K} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\bar{\mathbf{A}} \\ \vdots \\ \mathbf{C}\bar{\mathbf{A}}^{N-1} \end{bmatrix}^{\dagger} \begin{bmatrix} \mathbf{\Phi}_1^{(2)} \\ \mathbf{\Phi}_2^{(2)} \\ \vdots \\ \mathbf{\Phi}_N^{(2)} \end{bmatrix} \tag{3.23}$$

The matrices $\mathbf{B}$ and $\mathbf{A}$ are finally recovered from equations (3.21b) and (3.21c).

Note that the above algorithms based on the bar form of the Kalman filter are not found in the literature as their performance is generally poorer than those based on the innovation form. As the approach based on Markov parameters is well established, no illustrative examples are provided here. The reader is referred to, for instance, References 1, 15, 17, 20, 27, 28.

3.3.1 Algorithms based on Kalman filter Markov parameters

The detailed steps to implement the OKID methods based on the observer/Kalman filter Markov parameters are given below, in a comprehensive algorithm along which the user can choose which form of the Kalman filter to use and whether to implement ERA or ERA-DC for the identification of the observer. The input to the algorithm are the sequences $\{\boldsymbol{u}(k)\}$ and $\{\boldsymbol{y}(k)\}$ in equation (2.1). The output is the set of matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ and $\mathbf{K}$.

1. construct the matrices $\mathbf{Y}$ and $\mathbf{V}$ from equations (2.45a) and (2.45b)

2. solve equation (2.44) by LS

$$\tilde{\mathbf{\Phi}} = \mathbf{Y}\mathbf{V}^{\dagger} \tag{3.24}$$

For algorithms identifying the Kalman filter in innovation form

3. partition $\tilde{\mathbf{\Phi}}$ as

$$\tilde{\mathbf{\Phi}} = \begin{bmatrix} \tilde{\mathbf{\Phi}}_0 & \tilde{\mathbf{\Phi}}_1 & \dots & \tilde{\mathbf{\Phi}}_{p-1} & \tilde{\mathbf{\Phi}}_p \end{bmatrix} \tag{3.25}$$

   where $\tilde{\mathbf{\Phi}}_0 \in \mathbb{R}^{q \times q}$ and $\tilde{\mathbf{\Phi}}_j \in \mathbb{R}^{q \times (m+q)}$, $j = 1, 2, \dots, p$.

4. compute $\tilde{\mathbf{\Psi}}_j$ for $j = 0, 1, \dots, N$ from equation (3.15) with $\tilde{\mathbf{\Phi}}_j = \mathbf{0}$ for $j > p$, choosing N such that the sequence $\tilde{\mathbf{\Psi}}_j$ covers a significant portion of the impulse response of the system

5. identify $\mathbf{D} = \tilde{\mathbf{\Psi}}_0$

6. execute, with input $\tilde{\mathbf{\Psi}}_j$, $j = 1, 2, ..., N$, any of the following algorithms

   - ERA (section A.1) for OKID/ERAi
   - ERA-DC (section A.2) for OKID/ERA-DCi
   - any other algorithm for deterministic state-space model identification from unit pulse response (Markov parameters)

   and read the output matrices $\mathbf{A}_i$, $\mathbf{B}_i$, $\mathbf{C}_i$

7. extract the desired matrices (Matlab® notation)

$$\mathbf{A} = \mathbf{A}_i, \mathbf{C} = \mathbf{C}_i, \mathbf{B} = \mathbf{B}_i\left(:, 1 : m\right), \mathbf{K} = \mathbf{B}_i\left(:, m + 1 : m + q\right) \tag{3.26}$$

For algorithms identifying the Kalman filter in innovation form - variant

3. partition $\tilde{\mathbf{\Phi}}$ as

$$\tilde{\mathbf{\Phi}} = \begin{bmatrix} \tilde{\mathbf{\Phi}}_0 & \tilde{\mathbf{\Phi}}_1 & \dots & \tilde{\mathbf{\Phi}}_{p-1} & \tilde{\mathbf{\Phi}}_p \end{bmatrix} \tag{3.27}$$

where $\tilde{\mathbf{\Phi}}_0 \in \mathbb{R}^{q\times q}$ and $\tilde{\mathbf{\Phi}}_j \in \mathbb{R}^{q\times m+q}$, $j = 1, 2, \ldots, p$.

4. compute $\tilde{\mathbf{\Psi}}_j^{(s)}$ for $j = 0, 1, \ldots, N$ and $\tilde{\mathbf{\Psi}}_j^{(g)}$ for $j = 1, 2, \ldots, N$ from (3.15) with $\tilde{\mathbf{\Phi}}_j = \mathbf{0}$ for $j > p$, choosing N such that the sequence $\tilde{\mathbf{\Psi}}_j$ covers a significant portion of the impulse response of the system

5. identify $\mathbf{D} = \tilde{\mathbf{\Psi}}_0$

6. execute, with input $\tilde{\Psi}_j^{(s)}$, $j = 1, 2, ..., N$, any of the following algorithms

   - ERA (section A.1) for OKID/ERAvi
   - ERA-DC (section A.2) for OKID/ERA-DCvi
   - any other algorithm for deterministic state-space model identification from unit pulse response (Markov parameters)

   and read the output matrices $\mathbf{A}_i$, $\mathbf{B}_i$, $\mathbf{C}_i$

7. identify the desired matrices

$$\mathbf{A} = \mathbf{A}_i, \mathbf{C} = \mathbf{C}_i, \mathbf{B} = \mathbf{B}_i \tag{3.28}$$

8. find $\mathbf{K}$ as

$$\mathbf{K} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{N-1} \end{bmatrix}^{\dagger} \begin{bmatrix} \tilde{\mathbf{\Psi}}_1^{(g)} \\ \tilde{\mathbf{\Psi}}_2^{(g)} \\ \vdots \\ \tilde{\mathbf{\Psi}}_N^{(g)} \end{bmatrix} \tag{3.29}$$

For algorithms identifying the Kalman filter in bar form

3. partition $\tilde{\mathbf{\Phi}}$ as

$$\tilde{\mathbf{\Phi}} = \begin{bmatrix} \tilde{\mathbf{\Phi}}_0^{(1)} & \tilde{\mathbf{\Phi}}_1^{(1)} & \tilde{\mathbf{\Phi}}_1^{(2)} & \ldots & \tilde{\mathbf{\Phi}}_p^{(1)} & \tilde{\mathbf{\Phi}}_p^{(2)} \end{bmatrix} \tag{3.30}$$

where $\tilde{\mathbf{\Phi}}_0^{(1)} \in \mathbb{R}^{q\times q}$, $\tilde{\mathbf{\Phi}}_j^{(1)} \in \mathbb{R}^{q\times m+q}$ and $\tilde{\mathbf{\Phi}}_j^{(2)} \in \mathbb{R}^{q\times q}$, $j = 1, 2, \ldots, p$.

4. identify $\mathbf{D} = \tilde{\mathbf{\Phi}}_0$

5. execute, with input $\tilde{\mathbf{\Phi}}_j$, $j = 1, 2, ..., N$, any of the following algorithms

   - ERA (section A.1) for OKID/ERAb
   - ERA-DC (section A.2) for OKID/ERA-DCb
   - any other algorithm for deterministic state-space model identification from unit pulse response (Markov parameters)

   and read the output matrices $\mathbf{A}_b$, $\mathbf{B}_b$, $\mathbf{C}_b$

6. extract the desired matrices (Matlab® notation)

$$\mathbf{A} = \mathbf{A}_b, \mathbf{C} = \mathbf{C}_b, \mathbf{K} = \mathbf{B}_b\left(:, m+1 : m+q\right), \mathbf{B} = \mathbf{B}_b\left(:, 1 : m\right) + \mathbf{KD} \tag{3.31}$$

For algorithms identifying the Kalman filter in bar form - variant

3. Partition $\tilde{\mathbf{\Phi}}$ as

$$\tilde{\mathbf{\Phi}} = \begin{bmatrix} \tilde{\mathbf{\Phi}}_0^{(1)} & \tilde{\mathbf{\Phi}}_1^{(1)} & \tilde{\mathbf{\Phi}}_1^{(2)} & \ldots & \tilde{\mathbf{\Phi}}_p^{(1)} & \tilde{\mathbf{\Phi}}_p^{(2)} \end{bmatrix} \tag{3.32}$$

where $\tilde{\mathbf{\Phi}}_0^{(1)} \in \mathbb{R}^{q\times q}$, $\tilde{\mathbf{\Phi}}_j^{(1)} \in \mathbb{R}^{q\times m+q}$ and $\tilde{\mathbf{\Phi}}_j^{(2)} \in \mathbb{R}^{q\times q}$, $j = 1, 2, \ldots, p$.

4. Identify $\mathbf{D} = \tilde{\mathbf{\Phi}}_0$

5. Execute, with input $\tilde{\mathbf{\Phi}}_j^{(1)}$, $j = 1, 2, ..., N$, any of the following algorithms

   - ERA (section A.1) for OKID/ERAvb
   - ERA-DC (section A.2) for OKID/ERA-DCvb

- any other algorithm for deterministic state-space model identification from unit pulse response (Markov parameters)

and read the output matrices $\mathbf{A}_b$, $\mathbf{B}_b$, $\mathbf{C}_b$

6. Find $\mathbf{K}$ as

$$\mathbf{K} = \begin{bmatrix} \mathbf{C}_b \\ \mathbf{C}_b\mathbf{A}_b \\ \vdots \\ \mathbf{C}_b\mathbf{A}_b^{N-1} \end{bmatrix}^{\dagger} \begin{bmatrix} \tilde{\mathbf{\Phi}}_1^{(2)} \\ \tilde{\mathbf{\Phi}}_2^{(2)} \\ \vdots \\ \tilde{\mathbf{\Phi}}_N^{(2)} \end{bmatrix} \tag{3.33}$$

7. Extract the desired matrices

$$\mathbf{C} = \mathbf{C}_b, \mathbf{B} = \mathbf{B}_b + \mathbf{K}\mathbf{D}, \mathbf{A} = \mathbf{A}_b + \mathbf{K}\mathbf{C} \tag{3.34}$$

3.4 Identification via Kalman filter output residuals

As pointed out in sections 2.3 and 3.2, the Kalman filter is itself a dynamic system in state-space form. For instance, consider the innovation form of the Kalman filter, equation (3.1). Its inputs are $\boldsymbol{u}$ and $\boldsymbol{\epsilon}$ and its output is $\hat{\boldsymbol{y}}$: $\boldsymbol{u}$ is known from measurements and an estimate for $\boldsymbol{\epsilon}$ and $\hat{\boldsymbol{y}}$ is available from the LS solution to the OKID core equation. Indeed, once an estimate for the time history of the Kalman filter residual $\boldsymbol{\epsilon}$ is available, that can be used to obtain via equation (2.8) the estimate $\tilde{\mathbf{Y}}$ for the time history $\hat{\mathbf{Y}}$ of the Kalman filter output

$$\tilde{\mathbf{Y}} = \tilde{\mathbf{\Phi}}\mathbf{V} \tag{3.35a}$$

$$\hat{\mathbf{Y}} = \begin{bmatrix} \hat{\boldsymbol{y}}(p) & \hat{\boldsymbol{y}}(p+1) & \dots & \hat{\boldsymbol{y}}(l-1) \end{bmatrix} \tag{3.35b}$$

Both the input and the output sequences of the dynamic system in equation (3.1) are then known (measured or estimated). Additionally, in equation (3.1) no (unknown) noise term is present. We have just constructed a new noise-free identification problem: given the time histories of $\boldsymbol{u}, \boldsymbol{\epsilon}, \hat{\boldsymbol{y}}$, find the matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ and $\mathbf{K}$. Thanks to the absence of noise, *any* deterministic system identification method capable of identifying a state-space model from its response to an arbitrary input can be used to solve the new problem. Note that the solution to the new problem is also the solution to the original problem stated in section 2.2.

This gives rise to many OKID-based identification algorithms, as many as the deterministic identification methods from arbitrary input response that one can think of. In this work, to illustrate the effectiveness of the new approach, we demonstrate via examples two possible choices, namely the Deterministic Intersection (DI) and the Deterministic Projection (DP) method. In the literature of subspace methods, several intersection and projection algorithms have been developed (see for instance Reference 2). In the examples in this section, we refer to the DI algorithm of Reference 29 and the DP algorithm of Reference 30. The Matlab® codes of both algorithms are provided in Reference 2[1]. These two methods are also reviewed in appendix A. The DI and DP methods are considered deterministic because their formulation is based on purely deterministic state-space models, with no process or measurement noise, such as the one in equation (1.1). It is however worth noting that, although they do not qualify as combined identification algorithms, the numerical techniques used in the implementation of DI and DP (essentially singular value decomposition, SVD) make them very robust to noise and in some specific cases even unbiased (Reference 2). The resulting new OKID-based algorithms are referred to as OKID/DIi and OKID/DPi to remark that the underlying Kalman filter is identified in its innovation form, distinguishing them from their variant based on the identification of the Kalman filter in bar form outlined below.

[1] Also available at `http://homes.esat.kuleuven.be/~smc/sysid/software/`.

Similar to the innovation form, equation (3.5) represents a dynamic system whose input $\boldsymbol{u}$ and $\boldsymbol{y}$ and output $\hat{\boldsymbol{y}}$ are known. *Any* deterministic system identification method can be applied to find a realization of the matrices $\bar{\mathbf{A}}$, $\bar{\mathbf{B}}$, $\mathbf{C}$, $\mathbf{D}$. From the latter, equation (3.21) allows one to recover the desired matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{K}$, too, in the same way as in OKID/ERAb and OKID/ERA-DCb algorithms.

The new OKID-based identification algorithms presented in this section can then be formulated either via the innovation form of the Kalman filter to give OKID/DIi and OKID/DPi, or via the bar form to give OKID/DIb and OKID/DPb. Both alternatives are demonstrated in the examples. It is worth adding that other algorithms based on the new identification strategy can be devised simply by replacing DI and DP with other deterministic system identification methods. For instance, one could use the subspace Algorithm 1 or Algorithm 2 in Reference 2 or the algorithms from the superspace family (References 31–33).

As a last comment, note that the input to the observer to be identified (in either form) is different in the state and observation equations. More precisely, the state equation has an additional input ($\boldsymbol{\epsilon}$ or $\boldsymbol{y}$), which makes the form of the deterministic identification problem slightly different from the standard form usually considered in the literature, including in the DI and DP algorithms. Two ways to address the issue are possible. One consists in feeding the deterministic identification algorithms with the same additional input in the observation equation as well, relying on the associated coefficients in the corresponding extended $\mathbf{D}$ matrix being identified as $\mathbf{0}$. The other approach is to tailor the deterministic identification algorithms so that they identify the observer taking into account its peculiar form. The required modification is very

simple to apply in the case of the deterministic intersection[2] and the superspace algorithms. Numerical experiments show negligible difference between the two approaches when the innovation form is used for the Kalman filter. In the case of bar form, the tailored algorithms tend to provide better results. In the examples given in this work, for simplicity no modification is adopted.

3.4.1 Algorithms based on Kalman filter output residuals

The detailed steps to implement the OKID methods based on the observer/Kalman filter output residuals are given below, in a comprehensive algorithm along which the user can choose which form of the Kalman filter to use and whether to implement DI, DP or other deterministic methods for the identification of the observer. The input to the algorithm are the sequences $\{\boldsymbol{u}(k)\}$ and $\{\boldsymbol{y}(k)\}$ of equation (2.1). The output is the set of matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ and $\mathbf{K}$.

1. construct the matrices $\mathbf{Y}$ and $\mathbf{V}$ from equations (2.45a) and (2.45b)

2. solve equation (2.44) by LS

$$\tilde{\mathbf{\Phi}} = \mathbf{Y}\mathbf{V}^{\dagger} \tag{3.38}$$

[2] With reference to the DI algorithm described step by step in appendix A, only step 4 needs to modified to take into account the peculiar structure of the Kalman filter in equation (3.2) or in equation (3.7). The LS problem to be solved gets split into the following two LS problems

$$\mathbf{U}_q^{\mathrm{T}}\mathbf{U}_{12}^{\mathrm{T}}\mathbf{U}(m+q+1:(m+q)(i+1),1:2mi+n)\mathbf{S}_{11} = \begin{bmatrix}\mathbf{A} & \mathbf{B}\end{bmatrix} \begin{bmatrix}\mathbf{U}_q^{\mathrm{T}}\mathbf{U}_{12}^{\mathrm{T}}\mathbf{U}(1:mi+qi,1:2mi+n)\mathbf{S}_{11} \\ \mathbf{U}(mi+qi+1:mi+qi+m-q),1:2mi+n)\mathbf{S}_{11}\end{bmatrix} \tag{3.36}$$

to be solved for $\mathbf{A}$ and $\mathbf{B}$ and

$$\mathbf{U}(mi+qi+m+1:(m+q)(i+1),1:2mi+n)\mathbf{S}_{11} = \begin{bmatrix}\mathbf{C} & \mathbf{D}\end{bmatrix} \begin{bmatrix}\mathbf{U}_q^{\mathrm{T}}\mathbf{U}_{12}^{\mathrm{T}}\mathbf{U}(1:mi+qi,1:2mi+n)\mathbf{S}_{11} \\ \mathbf{U}(mi+qi+1:mi+qi+m-q),1:2mi+n)\mathbf{S}_{11}\end{bmatrix} \tag{3.37}$$

to be solved for $\mathbf{C}$ and $\mathbf{D}$. Note that $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ in this footnote are consistent with the notation used in the DI algorithm in appendix A, they do not refer to the matrices of the Kalman filter in equations (3.1) or in equation (3.5). Depending on the Kalman filter form used, the appropriate correspondence relationship between the former and the latter matrices can be found, similar to equations (3.42) and (3.44).

and compute

$$\left[\tilde{\boldsymbol{y}}(p) \quad \tilde{\boldsymbol{y}}(p+1) \quad ... \quad \tilde{\boldsymbol{y}}(l-1)\right] = \tilde{\mathbf{\Phi}}\mathbf{V} \tag{3.39}$$

For algorithms identifying the Kalman filter in innovation form

3. compute

$$\left[\tilde{\boldsymbol{\epsilon}}(p) \quad \tilde{\boldsymbol{\epsilon}}(p+1) \quad ... \quad \tilde{\boldsymbol{\epsilon}}(l-1)\right] = \mathbf{Y} - \tilde{\mathbf{\Phi}}\mathbf{V} \quad \text{(for innovation form only)} \tag{3.40}$$

4. define the following input and output sequences

$$\{\boldsymbol{u}_i\} = \left\{ \begin{bmatrix} \boldsymbol{u}(p) \\ \tilde{\boldsymbol{\epsilon}}(p) \end{bmatrix}, \begin{bmatrix} \boldsymbol{u}(p+1) \\ \tilde{\boldsymbol{\epsilon}}(p+1) \end{bmatrix}, ..., \begin{bmatrix} \boldsymbol{u}(l-1) \\ \tilde{\boldsymbol{\epsilon}}(l-1) \end{bmatrix} \right\} \tag{3.41a}$$

$$\{\boldsymbol{y}_i\} = \{\tilde{\boldsymbol{y}}(p), \tilde{\boldsymbol{y}}(p+1), ..., \tilde{\boldsymbol{y}}(l-1)\} \tag{3.41b}$$

5. execute, with input $\{\boldsymbol{u}_i\}$ and output $\{\boldsymbol{y}_i\}$,

   - the DI algorithm for OKID/DIi
   - the DP algorithm
   - any other algorithm for deterministic state-space model identification

   and read the output matrices $\mathbf{A}_i$, $\mathbf{B}_i$, $\mathbf{C}_i$, $\mathbf{D}_i$

6. extract the desired matrices (Matlab® notation)

$$\mathbf{A} = \mathbf{A}_i, \quad \mathbf{B} = \mathbf{B}_i(:, 1:m), \quad \mathbf{K} = \mathbf{B}_i(:, m+1:m+q),$$

$$\mathbf{C} = \mathbf{C}_i, \quad \mathbf{D} = \mathbf{D}_i(:, 1:m) \tag{3.42}$$

For algorithms identifying the Kalman filter in bar form

3. define the following input and output sequences

$$\{\boldsymbol{u}_b\} = \left\{ \begin{bmatrix} \boldsymbol{u}(p) \\ \\ \boldsymbol{y}(p) \end{bmatrix}, \begin{bmatrix} \boldsymbol{u}(p+1) \\ \\ \boldsymbol{y}(p+1) \end{bmatrix}, ..., \begin{bmatrix} \boldsymbol{u}(l-1) \\ \\ \boldsymbol{y}(l-1) \end{bmatrix} \right\} \tag{3.43a}$$

$$\{\boldsymbol{y}_b\} = \{\tilde{\boldsymbol{y}}(p), \tilde{\boldsymbol{y}}(p+1), ..., \tilde{\boldsymbol{y}}(l-1)\} \tag{3.43b}$$

4. execute, with input $\{\boldsymbol{u}_b\}$ and output $\{\boldsymbol{y}_b\}$,

   - the DI algorithm for OKID/DIb
   - the DP algorithm for OKID/DPb
   - any other algorithm for deterministic state-space model identification

   and read the output matrices $\mathbf{A}_b$, $\mathbf{B}_b$, $\mathbf{C}_b$, $\mathbf{D}_b$

5. extract the desired matrices (Matlab® notation)

$$\mathbf{C} = \mathbf{C}_b, \quad \mathbf{D} = \mathbf{D}_b(:, 1:m), \quad \mathbf{K} = \mathbf{B}_b(:, m+1:m+l),$$

$$\mathbf{B} = \mathbf{B}_b(:, 1:m) + \mathbf{KD}, \quad \mathbf{A} = \mathbf{A}_b + \mathbf{KC} \tag{3.44}$$

3.4.2 Demonstration and interpretation

In this section we introduce a simple example to demonstrate the above algorithms, discuss their main features and provide an interpretation of the new identification strategy.

3.4.2.1 Example

Consider the state-space model of equation (1.3) with the following matrices

$$\mathbf{A} = \begin{bmatrix} 0 & 0.5 \\ 0.5 & -0.5 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \qquad \mathbf{C} = \begin{bmatrix} 0 & 1 \end{bmatrix} \qquad D = 0 \tag{3.45}$$

The measured input-output data in equation (2.1) are simulated as follows. First we generate a white input sequence $\{\boldsymbol{u}(k)\}$ of length $l = 10,000$ (from a normal distribution with zero mean and standard deviation of 1) and two zero-mean gaussian noise sequences $\{\boldsymbol{w}_p(k)\}$ and $\{\boldsymbol{w}_m(k)\}$ with covariance

$$\mathbf{Q} = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} \times 10^{-2} \qquad R = 4 \times 10^{-2} \tag{3.46}$$

Said sequences are used to generate $\{\boldsymbol{y}(k)\}$ via equation (1.3). Such $\{\boldsymbol{u}(k)\}$ and $\{\boldsymbol{y}(k)\}$ can also be interpreted as input-output measurements with mutually uncorrelated zero-mean gaussian noise affecting the input-output channels with standard deviation of 0.1 and 0.2, respectively. The resulting signal-to-noise ratio is about 20 dB in both channels.

3.4.2.2 Estimation of Kalman output residuals

The algorithm starts with the choice of the parameter p. Let us assume that the system order is unknown but we have reason to believe it is small, say less than 5. Let us then choose $p = 20$ and run the first part of the identification method (steps 1 and 2), which is common to all the new OKID-based algorithms presented above. That leads to the estimation of the Kalman filter output residuals. Figure 3.1 compares the obtained estimates with the theoretical residuals coming from the Kalman filter in equation (3.1) with the gain computed from the true system and covariance matrices in equations (3.45) and (3.46) via equations (2.7c) and (2.7d) (algebraic

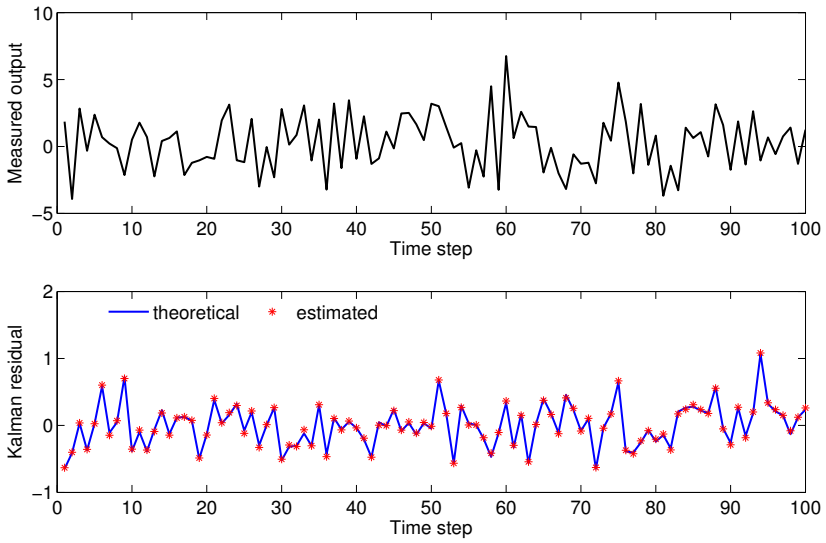


FIGURE 3.1: Estimation of Kalman filter output residuals.

Riccati equation). As part of the properties of the Kalman filter, the residuals are known to be a white process (lemma 8.5). It is then remarkable how it is possible to accurately estimate the time history of such a random process, as shown in Figure 3.1.

For the purpose of illustration, steps 3, 4 and 5 are executed for all the four algorithms described above to get the corresponding identified matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ and $\mathbf{K}$. The parameter i to be set in the DI and DP methods is chosen equal to 5, consistently with the above mentioned a priori belief on the system order. All the new algorithms are able to identify the right order ($n = 2$). It is here worth remarking that when noise corrupts the data, it is impossible to get exact identification. This fact generally makes the comparison of different methods and algorithms a difficult task, often addressed via lenghty numerical simulations whose generality is difficult to claim. The task is beyond the scope of this work. Table 3.1 reports the eigenvalues of the true $\mathbf{A}$ matrix and of the same matrix identified via the new algorithms. The identified values are shown in terms of mean and standard deviation of the results of a Monte Carlo simulation with 100 runs of the same example varying the noise sequences. All of the proposed algorithms provide good identification. None of them outperforms the others and neither form of the

Kalman filter seems to provide significantly better results, suggesting the proposed algorithms are all equivalent, at least in the example.

More interestingly, Table 3.1 shows how the OKID-based algorithms give better identification than the straight application of the corresponding deterministic methods. Running the DI and DP algorithms in appendix A directly on the measured $\{\boldsymbol{u}(k)\}$ and $\{\boldsymbol{y}(k)\}$ sequences give in general less accurate results than running them after the estimation of the Kalman filter residuals to identify the Kalman filter. This leads to the interpretation of the first part of the OKID approach as a pre-filtering stage. The Kalman filter embedded in the OKID core equation, equation (2.44), provides new input-output signals which are then passed to the second part of the new OKID approach, consisting in the identification of the Kalman filter from its estimated output (and residuals). Such pre-filtering lets the chosen deterministic identification method operate in the conditions for which it was formulated, i.e. with no noise or at least significantly attenuated noise. To emphasize the role played by the pre-filtering stage, Figures 3.2 and 3.3 show the plots of the normalized singular values arising from the Singular Value Decomposition (SVD) at the core of the DI and DP methods. Such SVDs are meant to split the zero and non-zero singular values, the number of the latter being the order of the system. If the input-output data are corrupted by noise, no singular value is exactly zero and the user might

TABLE 3.1: Eigenvalue comparison between true **A** and corresponding identified matrices (Monte Carlo simulation with 100 runs).

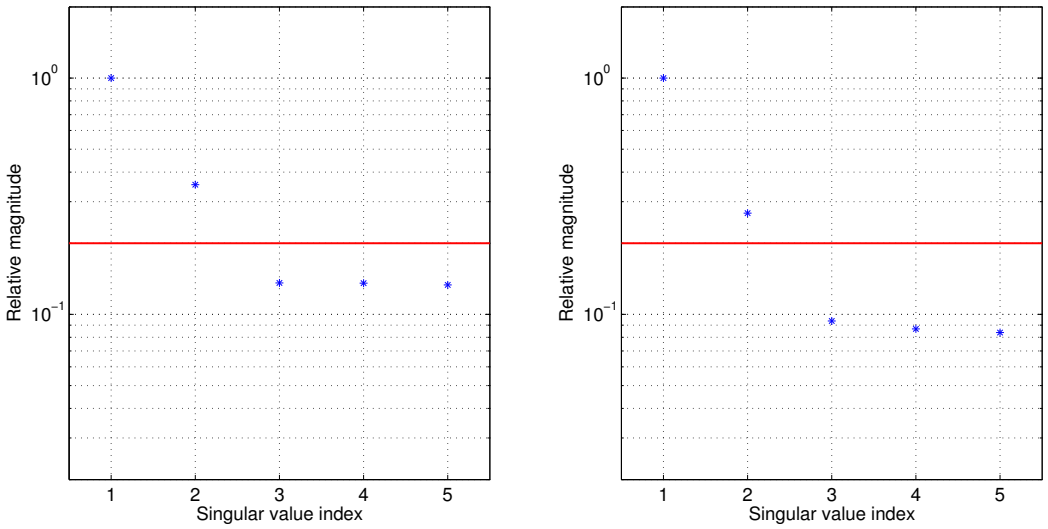
Method	Eigenvalue 1		Eigenvalue 2	
	mean	std. dev.	mean	std. dev.
True	-0.80902	-	0.30902	-
OKID/DIi	-0.80894	0.00097	0.30938	0.00265
OKID/DPi	-0.80896	0.00101	0.30935	0.00267
OKID/DIb	-0.80881	0.00098	0.30933	0.00266
OKID/DPb	-0.80826	0.00227	0.30915	0.00269
DI	-0.80864	0.00098	0.30760	0.00262
DP	-0.81375	0.00102	0.29896	0.00283

TABLE 3.2: Eigenvalue of **A** matrix identified by different algorithms with residual whitening.

Method	Eigenvalue 1	Eigenvalue 2
OKID/DIi	−0.8109258705999016	0.3077103118249301
OKID/DPi	−0.8109258705999016	0.3077103118249320
OKID/DIb	−0.8109258705999021	0.3077103118249284
OKID/DPb	−0.8109258705999010	0.3077103118249288

experience difficulties in deciding which singular values can be considered negligible and be discarded. Figure 3.2 shows the singular values of the DI and DP algorithms when applied directly, without pre-filtering. The separation line between zero and non-zero singular values is somewhat arguable. The pre-filtering gives rise to clearer plots, where two singular values stand out as being the ones to be considered different from zero, as shown for example for the algorithms based on the bar-form Kalman filter (Figure 3.3).

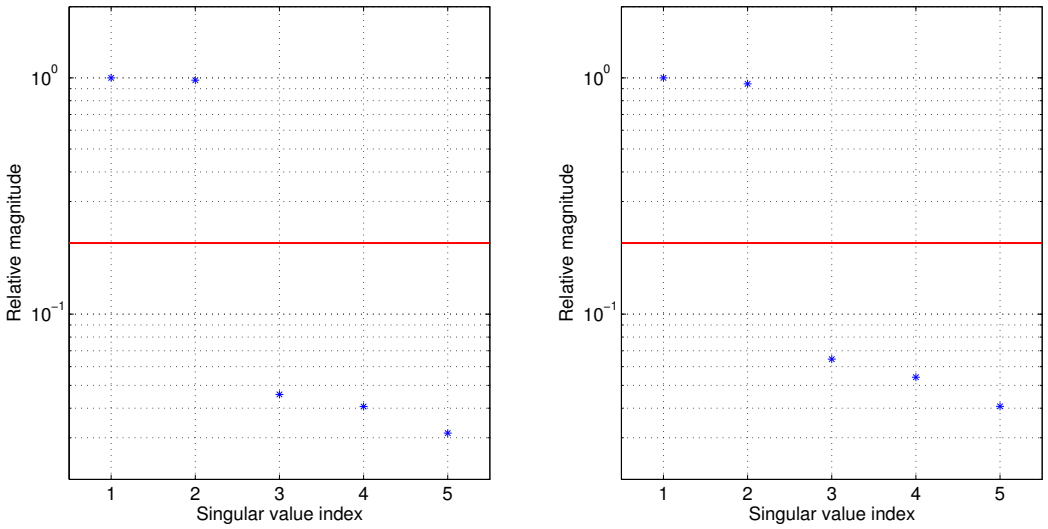
Note that the singular values considered to be negligible in the plots of Figure 3.3 are not exactly 0. Even though pre-filtering makes the order of the system clearly equal to 2, some noise is still present in the data fed to the the second part of the OKID-based algorithms. The OKID core equation relies on the assumption that $\bar{\mathbf{A}}^p$ in equation (2.37) is negligible, which is true for sufficiently large p. The approximation resulting from truncating p to a finite value gives then rise to some noise in the estimation of the residuals. Theoretically, increasing p asymptotically leads to no truncation error. In practice, p cannot grow indefinitely for numerical issues (condition number of the matrix to be pseudo-inverted in equation (2.46)) and because that would increase the number of parameters to be estimated and at the same time decrease the number of equations available in the LS problem of equation (2.44), reducing its overdeterminacy.



(A) DI algorithm.

(B) DP algorithm.

FIGURE 3.2: SVD of deterministic algorithms.



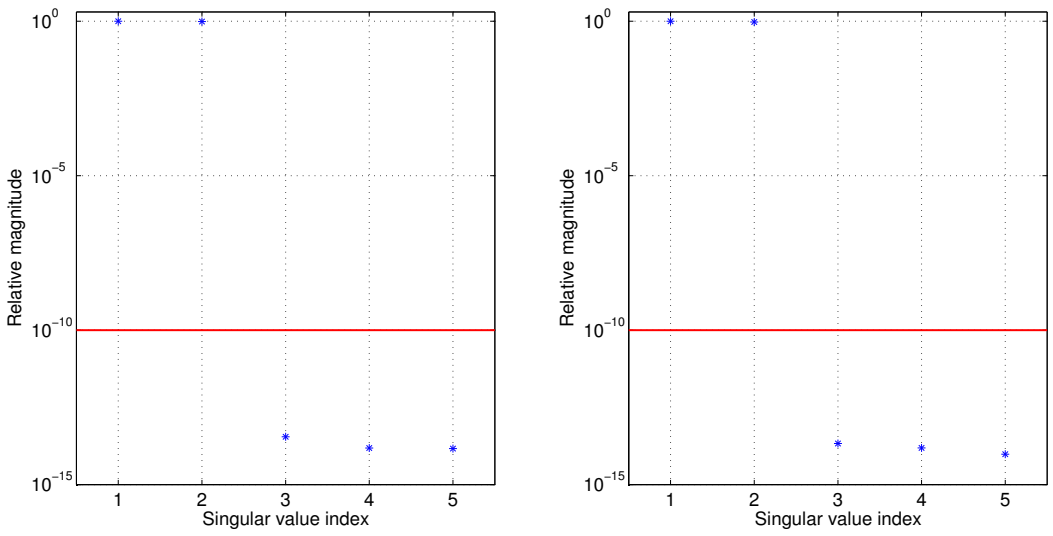
(C) OKID/DIb algorithm.

(D) OKID/DPb algorithm.

FIGURE 3.3: SVD of OKID algorithms based on Kalman filter bar form.

3.4.2.3 Exact residuals by whitening

An alternative to the classic OKID core equation (equation (2.44) in this work) was presented in Reference 34. The technique is called *residual whitening* and is used here to improve the estimate of the Kalman filter output residuals. The technique is outlined in section 5.5, where



(A) OKID/DIb algorithm. (B) OKID/DPb algorithm.

FIGURE 3.4: SVD of OKID algorithms based on Kalman filter bar form with residual whitening.

its relationship with the OKID core equation is discussed in more detail. Briefly, residual whitening can be interpreted as trading the truncation error of the classic OKID equation (due to neglecting $\mathbf{F}^p$, i.e. $\bar{\mathbf{A}}^p$, to get equation (2.40)) with the iteration convergence error of the GLS procedure used to solve the residual whitening equation. As opposed to the former, the latter can be made as small as desired just by running more iterations. We can then think of residual whitening as a technique to improve the estimation of the residuals and make the pre-filtering exact, i.e. yielding a noise-free set of data to be fed to the DI or DP method. To show the concept, we estimate the Kalman residuals by residual whitening with $p = 2$ and execute the steps 3 to 5 for all the proposed algorithms getting the SVD plots of Figure 3.4 and the eigenvalues of the identified $\mathbf{A}$ matrix summarized in Table 3.2. The zero singular values are now really such, since they are close to the working precision of Matlab®. Thanks to residual whitening, 15 order of magnitude separate the zero and non-zero singular values, making the selection of the right order of the system crystal clear. In conclusion, residual whitening with $p = n$ makes the pre-filtering exact and the SVD shows no trace of noise. As a consequence, the DI and DP methods are run on noise-free data and all the proposed algorithms provide the

same identified matrices, as shown by the eigenvalues of the identified $\mathbf{A}$. The numerical values in Table 3.2 differ only after the 15^{th} significant digit.

The first part of the OKID-based algorithms can now be interpreted as a conversion of the original combined identification problem of equation (1.3) into the deterministic problem of equation (3.1) or equation (3.5). The new approach consists then in converting the original problem, whose data are corrupted by noise, into a simpler noise-free problem which can be solved by any deterministic identification method. When the first part is solved approximately (e.g. due to truncation error in the classic OKID equation, limited number of iterations in the residual whitening technique, violation of the initial assumptions on the process and measurement noise), the error in the residual estimates makes the conversion not exact and the new identification problem is not completely noise-free, yet the noise is significantly reduced (pre-filtering).

For the sake of clarity, we used residual whitening in the example above to highlight the fact that the exact LS solution to the OKID equation would lead to completely noise-free identification of the observer. Residual whitening can generally be used to improve the estimates of the observer residuals, but it must be kept in mind that the LS problem of equation (5.13) is nonlinear (both $\bar{\mathbf{\Psi}}$ and $\mathbf{W}$ are unknowns) and the convergence of the GLS procedure to the (global) minimum is not always guaranteed. When the procedure converges, then residual whitening is a powerful technique to refine the identification.

As a last note, it is worth clarifying that it is impossible to estimate exactly the theoretical Kalman residuals from a finite-length record. This is due to the stochastic nature of the noise in the identification problem of equation (1.3). Even in numerical simulations, since the process and measurement noises are random, an infinitely-long record would be necessary to make them really satisfy the problem assumptions (in particular their whiteness). The consideration is

mainly of academic interest, since in real applications the noises can be of diverse nature and to some extent always violate the problem assumptions. Nevertheless it is important to realize that the residuals given by residual whitening are exact in the sense that they correspond to the LTI observer minimizing exactly the OKID equation with no truncation error. However, the finite nature of the record prevents the minimizing observer from being exactly the theoretical Kalman filter.

3.4.3 Numerical example

As a more realistic example, consider the lumped model of a four-story shear building, shown in Figure 3.5, with each mass equal to $m = 0.259$ kips-sec^2/in and each lateral spring of stiffness $k = 122.889$ kips/in. The building is also supposed to have viscous damping, quantified by a damping factor of $\zeta = 0.01$ for each of the 4 vibration modes. The assumption of modal damping makes it possible to recover the modal parameters (natural frequencies, damping factors and mode shapes) from the identified state-space model without conceptual difficulties. The force is applied in correspondence of the third floor via a zero-order-hold (ZOH) system with a sampling time of 0.01s. The excitation used in the example is a white signal normally distributed with zero mean and standard deviation of 1 and duration of 100s ($l = 10,000$). The input channel is affected by gaussian noise of standard deviation equal to 0.15, for a signal-to-noise ratio of about 16 dB. The lateral acceleration at each floor is measured, for a total of 4 outputs. Gaussian noise of standard deviation of 1 is present in each output channel, resulting in signal-to-noise ratios of about 13 db, 26 dB, 52 dB and 50 dB (from the ground up). The discrete-time state-space model of the structure of Figure 3.5 is therefore in the form of equation (1.3), with $n = 8$, $m = 1$ and $q = 4$. The process noise $\boldsymbol{w}_p$ is due to the noise in the input channel. Its covariance matrix

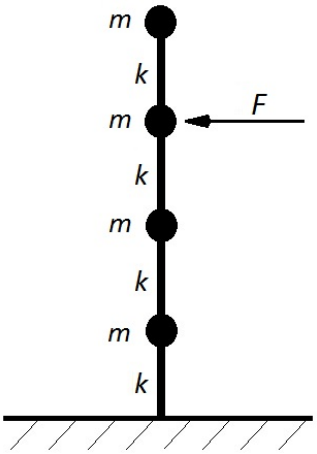
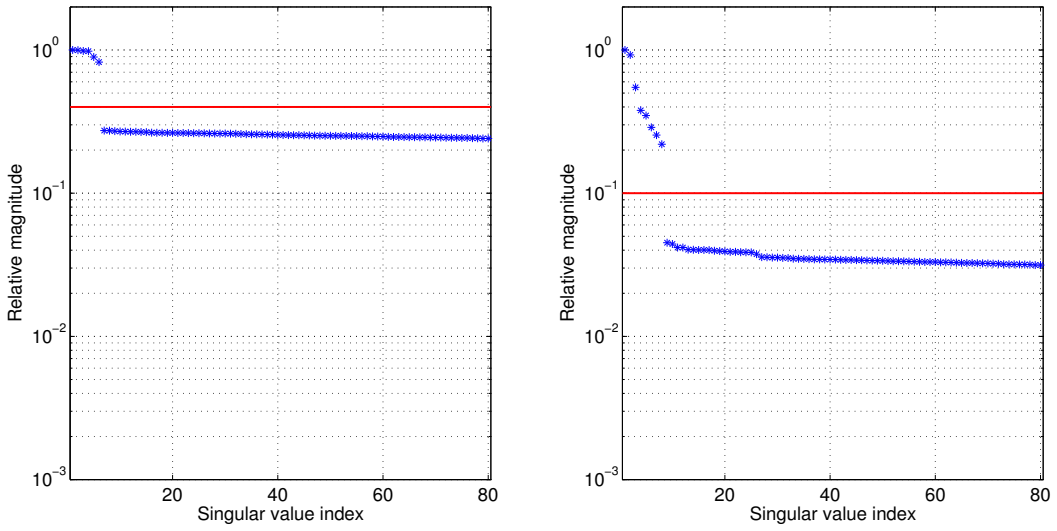


FIGURE 3.5: Lumped model of 4-story shear-type building.

is then $\mathbf{Q} = \mathbf{B}\mathbf{B}^{\mathrm{T}}$, whereas the covariance of the measurement noise $\boldsymbol{w}_m$ is the identity matrix.

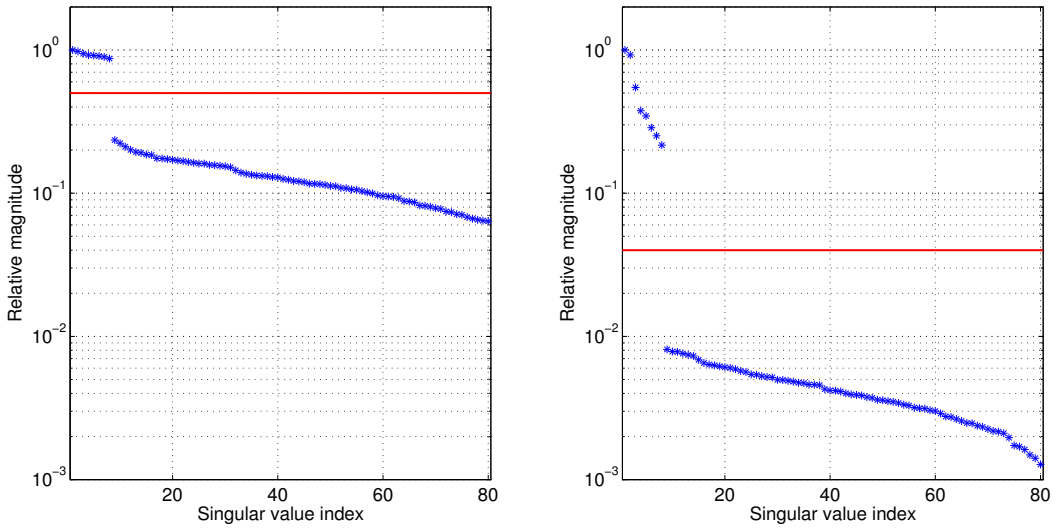
All the four variants of the algorithm described above are executed, with $p = 40$ in the OKID equation and $i = 20$ for the DI and DP algorithms. The identification results are reported in Table 3.3, in the form of natural frequencies and damping factors, together with the true values as well as those obtained via the direct application of the DI and DP methods and via traditional combined (deterministic input plus stochastic noise) methods such as OKID/ERA (section 3.3) and N4SID. N4SID (References 2, 35) is another very popular algorithm for the identification of combined linear systems. It belongs to the family of subspace methods and has been extensively used in industrial applications. The Matlab® code used here to run N4SID is the one provided in Reference 2[3]. The algorithms based on the innovation form of the Kalman filter perform sensibly better than their bar-form counterparts, in particular for OKID/DP. The accuracy of OKID/DIi and OKID/DPi is in line with OKID/ERAvi and N4SID. A very significant fact is that the OKID pre-filtering makes OKID/DIi and OKID/DPi generally perform better than DI and DP, as expected from the theoretical framework previously presented. This confirms the benefit of pre-filtering the data via the OKID equation, making the DI and DP algorithms work

[3] Also available at http://homes.esat.kuleuven.be/~smc/sysid/software/.



(A) DI algorithm. (B) DP algorithm.

FIGURE 3.6: SVD of deterministic algorithms.



(A) OKID/DIi algorithm. (B) OKID/DPi algorithm.

FIGURE 3.7: SVD of OKID algorithms based on Kalman filter innovation form.

in conditions closer to the ones for which they are formulated. For completeness, the SVD plots of some of the algorithms in Table 3.3 are reported in Figures 3.6 and 3.7. The advantage of OKID is evident with the DI method, leading to the correct identification of 4 vibration modes (Figure 3.7a), whereas without OKID pre-filtering the first mode is missed (Figure 3.6a). Its value in Table 3.3 is reported by forcing the selection of 8 non-zero singular values in SVD plots

TABLE 3.3: Identified natural frequencies (Hz) and damping factors of the structure of Fig 3.5 (Monte Carlo simulation, average over 100 runs).

Method	f_1	ζ_1	f_2	ζ_2	f_3	ζ_3	f_4	ζ_4
True	1.3948	0.0100	3.9721	0.0100	5.9447	0.0100	7.0122	0.0100
OKID/DIi	1.3955	0.0109	3.9725	0.0109	5.9450	0.0104	7.0122	0.0100
OKID/DPi	1.3958	0.0105	3.9730	0.0106	5.9451	0.0103	7.0121	0.0100
OKID/DIb	1.3979	0.0154	3.9718	0.0129	5.9458	0.0121	7.0126	0.0120
OKID/DPb	1.3653	0.0013	3.9991	0.0131	5.9499	0.0140	7.0197	0.0120
DI	1.3892	0.0120	3.9677	0.0106	5.9429	0.0105	7.0111	0.0104
DP	1.3951	0.0098	3.9754	0.0097	5.9453	0.0098	7.0129	0.0098
OKID/ERAvi	1.3956	0.0120	3.9732	0.0120	5.9458	0.0111	7.0125	0.0100
N4SID	1.3948	0.0102	3.9724	0.0103	5.9450	0.0101	7.0120	0.0100

like Figure 3.6a. In the DP case, the advantage of OKID shows up in pushing the negligible singular values down towards 0 (Figures 3.6b and 3.7b).

3.4.4 Experimental example

In order to demonstrate the new state-space model identification approach on real data, the experimental tests performed in the Engineering Institute (EI) at Los Alamos National Laboratory (LANL) and described in Reference 36 are considered in this section. The test structure (Figure 3.8) is a laboratory three-story building used as a damage-detection test-bed structure. It consists of aluminum columns and plates assembled using bolted joints, intended to form a four degree-of-freedom shear building.

An electrodynamic shaker provides a lateral excitation to the base floor along the center line of the structure. The structure and shaker are mounted together on an aluminum baseplate, and the entire system rests on rigid foam to minimize extraneous sources of unmeasured excitation from being introduced through the base of the system. A load cell attached at the end of a stinger measures the input force from the shaker to the structure. Four accelerometers attached at the center line of each floor on the opposite side from the excitation source measure the system response. The overall arrangement is intended to minimize the torsional excitation of

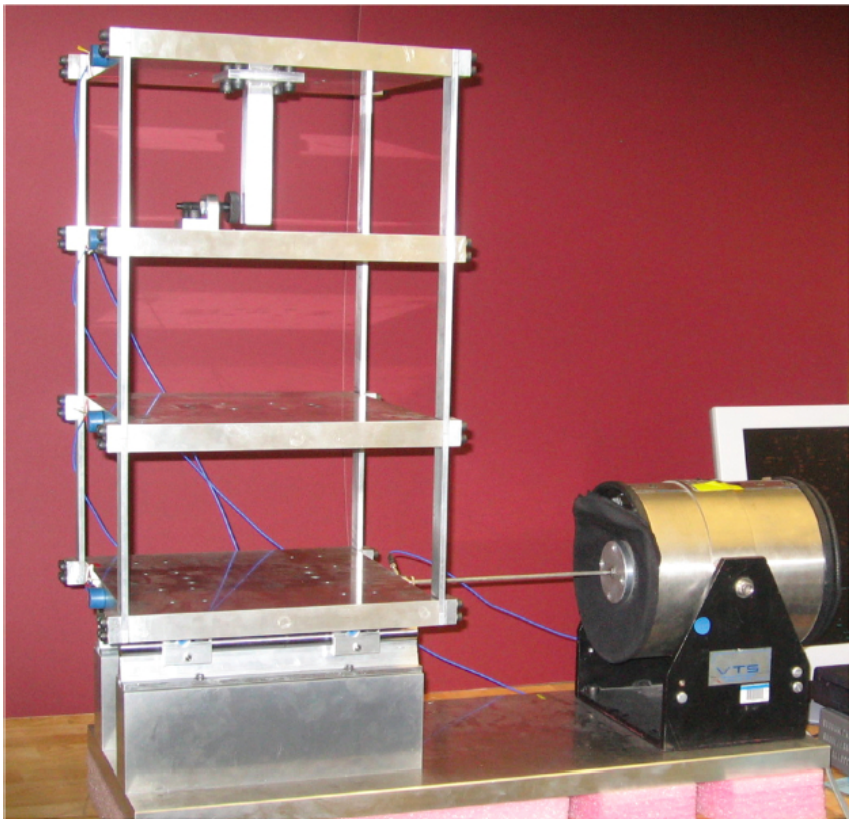


FIGURE 3.8: LANL test structure.

the system. In order to avoid the rigid body modes of the structure, a band-limited random excitation in the range of 20-150 Hz is used to excite the structure. The analog sensor signals are discretized with a sampling frequency of 320 Hz. The resulting time histories of 25.6 s in duration contain 8,192 time samples.

In the following example, we use the first dataset for the standard configuration of the test structure (denoted as state #1 in Reference 36) for system identification and the second dataset for the same configuration for validation of the identified model[4]. The OKID/DPi algorithm is applied for illustration. Choosing $p = 100$ and $i = 20$, the SVD of Figure 3.9 is obtained and clearly reveals the presence of 3 primary modes of vibration (largest 6 singular values) and 3 secondary modes (7^{th} to 12^{th} singular values). Retaining only the 6 largest singular values yields the following modal parameters: $f_1 = 30.72$Hz, $\zeta_1 = 2.2\%$, $f_2 = 54.80$Hz, $\zeta_2 = 0.78\%$,

[4]The data are available for download at http://institute.lanl.gov/ei/software-and-data/.

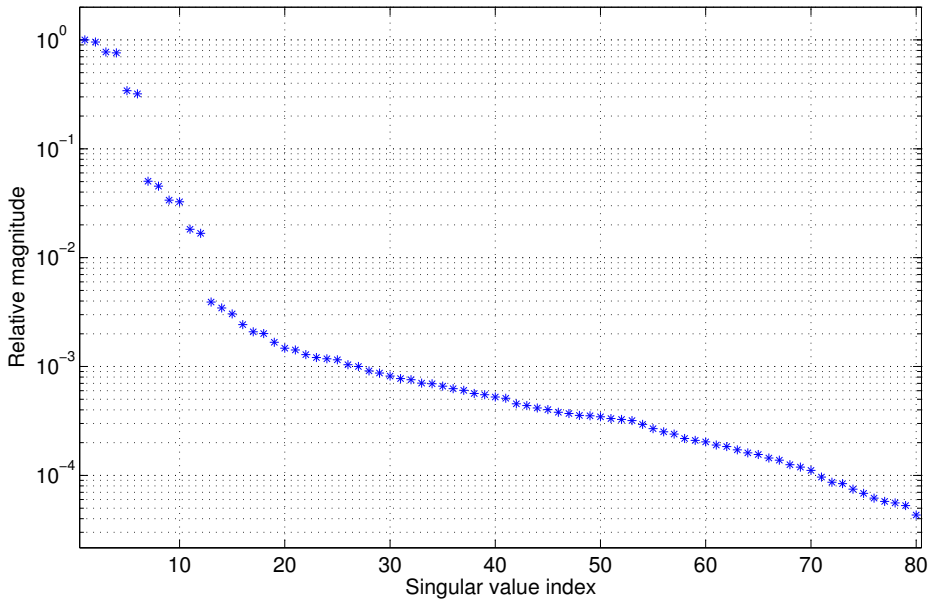


FIGURE 3.9: SVD in OKID/DPi.

$f_3 = 71.73$Hz, $\zeta_3 = 0.73\%$, which are in line with the values estimated in Reference 36 from the analysis of the experimental frequency response functions via the rational-fraction polynomial method (RFP). Retaining the 12 largest singular values in Figure 3.9, the same three modes are identified (with slightly modified modal parameters) and three additional modes appear with natural frequencies of 21.12 Hz, 41.05 Hz and 76.16 Hz. The additional modes can be explained as torsional modes of the test structure, weekly excited by the shaker and therefore hardly detectable in the frequency response analysis reported in Reference 36. Their presence is confirmed by the prediction capabilities of the identified six-mode model verified on the validation dataset (Figure 3.10), which are significantly better than the identified three-mode model. Figure 3.10 compares the outputs measured in the validation test and the outputs predicted by the identified system model when driven by the same measured input. This corresponds to multi-step-ahead prediction and the resulting output error is due not only to the identification error but also to the process and measurement noise affecting the data. The presented identification approach provides also the optimal one-step-ahead predictor (Kalman filter) associated with the identified

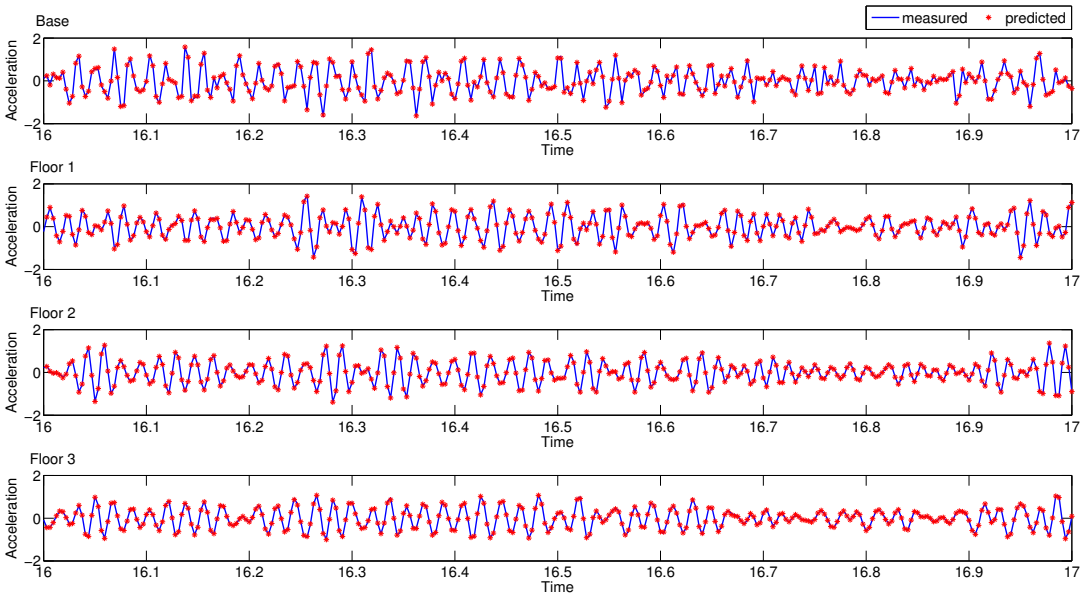


FIGURE 3.10: Validation of identified model.

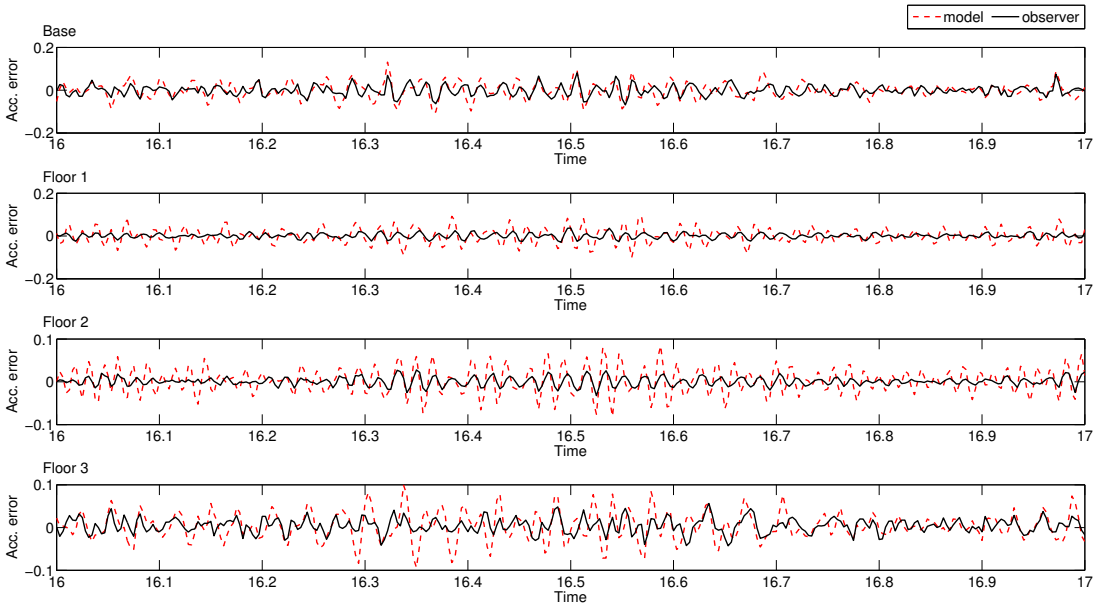


FIGURE 3.11: Validation of identified model and observer in terms of output prediction error.

model and the noise embedded in the data. Figure 3.11 shows both the output error from the system multi-step-ahead prediction and the output error from the Kalman filter one-step-ahead prediction. The latter is obviously smaller thanks to the additional information provided by the output measured at the previous time step. Figures 3.10 and 3.11 demonstrate how the

presented OKID-based algorithms are able to identify from real experimental data a state-space linear model and the associated optimal observer that can be effectively used for predicting the dynamic response of the system.

3.5 Conclusions

This chapter presented in detail several OKID algorithms for linear system identification from noisy input-output data. All of them start from the least-squares solution of the OKID core equation derived in chapter 2 but can then be classified into two families, those completing the identification from the estimated Markov parameters of the Kalman filter and those accomplishing the same task from the estimated output residuals of the Kalman filter. The traditional OKID/ERA method belongs to the former. The latter is a novel family of algorithms that numerical examples show to perform in line with OKID/ERA. In particular, in the given numerical and experimental examples, OKID/DPi proved to be the most accurate among all the OKID methods.

It is also worth remarking how the chapter discusses and demonstrates via numerical examples the conceptual result unveiled by the novel OKID approach based on output residuals. As better explained in chapter 5, the OKID core equation can be seen as a converter from combined (deterministic and stochastic) to purely deterministic system identification, i.e. as a powerful technique to simplify the identification problem. That is possible thanks to the filtering action that the Kalman filter embedded in the OKID equation performs. The interpretation is not only of theoretical interest, it also suggests that OKID can be applied to output-only system identification as well (chapter 4).

Chapter 4

Output-only observer/Kalman filter identification (O^3KID)

4.1 Introduction

The OKID core equation has been interpreted in chapter 3 as an implicit Kalman filter capable of converting an identification problem with stochastic signals (noise) into a purely deterministic identification problem. This suggests that OKID can also be applied to output-only system identification, i.e. to the case where only the output is measured and the unknown input can be approximated as white noise. As explained below, such case is typical in the identification of large structure in the area of civil engineering known as structural health monitoring (SHM). This application gives a chance to remark how system identification is a promising technique in civil engineering.

As the infrastructure system rapidly ages and maintenance and rehabilitation operations are more costly and urgent, Structural Health Monitoring (SHM) has become an active area of

research in civil engineering. Several system identification techniques have been developed over the past few decades and their application is growing with the availability of instrumentation on civil infrastructures. The purpose of system identification is the estimation of a mathematical model for the structure under consideration from the time histories of its response to environmental disturbances (e.g. earthquake, wind, traffic, etc.). From such a model, one can then compute the modal parameters of the structure (natural frequencies, damping factors, mode shapes) and use them for SHM, directly to detect damaged areas or indirectly to refine or update more detailed numerical models of the structure (e.g. a Finite Element Model, FEM).

The modal parameters are classically identified from a mathematical model representative of the structure in operational conditions, which describes the relationship between the excitation applied to the structure and the structure's vibrational response. However, especially in the case of civil infrastructures, it is often difficult to measure and control the excitation without disrupting the normal operations. The difficulties arise from the random character of the excitation both in the nature (e.g. wind, traffic) and in the way it is applied to the structure (e.g. distribution of wind pressure). In response to these obstacles, a variety of system identification techniques have been developed using only structural response information and are typically referred to as stochastic identification or output-only system identification. The standard assumption made in the development of output-only identification techniques is that the input is white and stationary. The success of such techniques obviously depends on how close the excitation is to the assumption. In the frequency domain, when the input is white, the power spectral density of the output itself can be considered to represent the dynamical properties of the system. Stochastic techniques in the frequency domain such as peak-picking (References 37, 38), frequency domain decomposition (References 38, 39) and maximum likelihood identification (References 40, 41) have then been developed. In the case of time-domain analysis, it has been shown that the

covariance functions of the system vibrational response uniquely characterize a dynamic system in a similar fashion to impulse response functions. The observation has led to the application to the stochastic case of classic system identification techniques that use impulse response functions (Reference 42). Such techniques are referred to as *covariance-driven* stochastic identification techniques, as opposed to other time-domain techniques known as *data-driven* that directly use the recorded output time histories. Among the latter, several stochastic subspace methods are available, i.e. algorithms based on well-established linear algebra concepts and robust numerical techniques. A notable example is presented in Reference 35 and a good general reference is Reference 2. A survey on output-only system identification methods is beyond the scope of this thesis.

This work is a completely new development, first presented at the Sixth World Conference on Structural Control and Monitoring in Barcelona in 2014 (Reference 43) and recently accepted for publication in Structural Control and Health Monitoring (Reference 44). Several new algorithms for output-only identification are proposed by extending to the output-only case the OKID approach presented in the previous chapters. The resulting algorithms are referred to as O^3KID (output-only OKID). Their derivation could be skipped by the simple observation that as a Kalman filter exists for the linear system in equation (4.1), it exists as well for the same model without input (i.e. $\boldsymbol{u} = \boldsymbol{0}$). With the general framework for OKID given in the previous chapters, the development of the O^3KID algorithms is then straightforward. Nevertheless, for the sake of completeness, the core O^3KID equation and the application of the deterministic algorithms to identify the Kalman filter are outlined in this chapter and specialized to the output-only case. Indeed, the complete absence of the deterministic input $\boldsymbol{u}$ allows one to tailor the DI and DP algorithms for a more elegant solution.

4.2 Problem description

Like any linear dynamical system, the mathematical model of a structure can be represented in the state-space form in equation (1.3). The vectors $\boldsymbol{w}_p \in \mathbb{R}^{n\times 1}$ and $\boldsymbol{w}_m \in \mathbb{R}^{q\times 1}$ represent noise in the process (e.g. noise in the input or unmodeled dynamics) and in the measurement (e.g. noise in the output measurement and, if $\mathbf{D} \neq \mathbf{0}$, in the input, too). The standard assumption is that the process and measurement noises are zero-mean white stationary processes, uncorrelated with $\boldsymbol{u}$ and $\boldsymbol{y}$, and with covariance $\mathbf{Q} \in \mathbb{R}^{n\times n}$ and $\mathbf{R} \in \mathbb{R}^{q\times q}$, respectively (section 2.2). When both input and output measurements are available (input-output system identification), OKID methods have been demonstrated to be very effective in identifying the matrices of structural models (References 20, 22, 23 and sections 3.4.3 and 3.4.4).

When the input measurements are not available but can be assumed to be white and stationary (standard assumption in the literature of output-only system identification for SHM), the system in (1.3) can be rewritten as

$$\boldsymbol{x}(k+1) = \mathbf{A}\boldsymbol{x}(k) + \boldsymbol{w}'_p(k) \tag{4.1a}$$

$$\boldsymbol{y}(k) = \mathbf{C}\boldsymbol{x}(k) + \boldsymbol{w}'_m(k) \tag{4.1b}$$

where $\boldsymbol{w}'_p \in \mathbb{R}^{n\times 1}$ and $\boldsymbol{w}'_m \in \mathbb{R}^{q\times 1}$ are zero-mean white stationary processes including the original process and measurement noises and the effect of the unknown input on the state equation ($\mathbf{B}\boldsymbol{u}(k)$) and on the measurement equation ($\mathbf{D}\boldsymbol{u}(k)$)

$$\boldsymbol{w}'_p(k) = \mathbf{B}\boldsymbol{u}(k) + \boldsymbol{w}_p \tag{4.2a}$$

$$\boldsymbol{w}'_m(k) = \mathbf{D}\boldsymbol{u}(k) + \boldsymbol{w}_m \tag{4.2b}$$

As a result $\boldsymbol{w}'_p$ and $\boldsymbol{w}'_m$ satisfy the same assumptions as the original process and measurement noise and will be referred to as such in the rest of the chapter. Although they are assumed to be uncorrelated with the output of the system, notice that when $\mathbf{D} \neq \mathbf{0}$ the input component embedded in $\boldsymbol{w}'_p$ and $\boldsymbol{w}'_m$ generally makes them correlated via a cross-covariance[1] matrix $\mathbf{S}' \in \mathbb{R}^{n \times q}$. This is indeed the case with structures where the input is force (or pressure) and the measured output is acceleration. The auto-covariance matrices are denoted by $\mathbf{Q}' \in \mathbb{R}^{n \times n}$ and $\mathbf{R}' \in \mathbb{R}^{q \times q}$, respectively.

The output-only system identification problem addressed in this chapter can be stated as follows. Given a set of length l of output data

$$\{\boldsymbol{y}(k)\} = \{\boldsymbol{y}(0), \boldsymbol{y}(1), \boldsymbol{y}(2), \ldots, \boldsymbol{y}(l-1)\} \tag{4.3}$$

measured from the system in (4.1) starting at some unknown initial state $\boldsymbol{x}(0)$ and driven by $\boldsymbol{w}'_p$ and $\boldsymbol{w}'_m$, the objective is to identify the state-space model (4.1), i.e. to find the matrices $\mathbf{A}$ and $\mathbf{C}$. Neither the noise sequences $\{\boldsymbol{w}'_p(k)\}$ and $\{\boldsymbol{w}'_m(k)\}$ nor their covariance matrices $\mathbf{Q}'$, $\mathbf{R}'$, $\mathbf{S}'$ are assumed to be known.

4.3 O^3KID core equation

Similar to OKID (input-output Observer/Kalman filter IDentification), O^3KID (Output-Only Observer/Kalman filter IDentification) consists of two main steps. First a set of algebraic equations is solved by least-squares (LS). That is totally analogous to the OKID core equation (2.44), but it is now referred to as the O^3KID core equation since its formulation is simplified by the

[1] Although not taken into account in section 2.4, noise cross-correlation does not affect the properties at the core of OKID described in section 2.3. The same is true for the absence of a deterministic input. Therefore, fact 0 is still valid also in the context of output-only systems such as equation (4.1) with cross-correlated noise. Hence OKID is applicable.

absence of the deterministic input $\boldsymbol{u}$. Then the Kalman filter associated with the system under consideration and the noise statistics embedded in the data is identified. Thanks to the close mathematical relationship between the system and the associated Kalman filter, the identification of the latter also solves the original problem, yielding the desired system matrices. In the case of input-output identification, OKID provides the matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ in equation (4.1) as well as the gain $\mathbf{K}$ of the associated Kalman filter. In the output-only case, only the matrices $\mathbf{A}$ and $\mathbf{C}$ can be identified for the system, as it is clear from equation (4.1). Nevertheless the approach in O^{3}KID is the same as for the input-output case and yields $\mathbf{K}$ as well. O^{3}KID provides valuable information for applications in structural health monitoring, as the modal parameters of the structure under consideration (natural frequencies, damping factors and mode shapes) can be computed from the identified matrices $\mathbf{A}$ and $\mathbf{C}$. Also, the resulting Kalman filter can be used for optimal state estimation or output filtering.

Some overlap with the notation in section 2.5 is preferred to the use of further letters and symbols. Similar notation also helps make clear the links between OKID and O^{3}KID.

Consider the steady-state Kalman filter[2] for the system in (4.1)

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{A}\hat{\boldsymbol{x}}(k) + \mathbf{K}\big(\boldsymbol{y}(k) - \hat{\boldsymbol{y}}(k)\big) \tag{4.5a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) \tag{4.5b}$$

[2] For brevity and clarity of presentation, the observer underlying the O^{3}KID approach has been introduced in equation (4.5) already in the form of a Kalman filter. This is justified by the fact that such observer is eventually proven to be the Kalman filter. For a more rigorous mathematical presentation, similar to what was done in section 2.5, the observer in equation (4.5) could have been introduced in the most general form of LTI observer for the output-only system in equation (4.1), i.e.

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{F}\hat{\boldsymbol{x}}(k) + \mathbf{K}\boldsymbol{y}(k) \tag{4.4a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) \tag{4.4b}$$

The fact that the steady-state Kalman filter is the unique observer in the form of equation (4.4) featuring the orthogonality property of the LS residuals of the O^{3}KID core equation would in the end prove that $\mathbf{F} = \mathbf{A} - \mathbf{KC}$, making equation (4.4) equivalent to equation (4.5).

where $\hat{\boldsymbol{x}} \in \mathbb{R}^{n\times 1}$ and $\hat{\boldsymbol{y}} \in \mathbb{R}^{q\times 1}$ are the observer state and output and $\mathbf{K} \in \mathbb{R}^{n\times q}$ is the observer gain. The observer in (4.5) is in the form of one-step-ahead predictor, i.e. it provides an estimate $\hat{\boldsymbol{x}}(k+1)$ of the next state $\boldsymbol{x}(k+1)$ from the current state estimate $\hat{\boldsymbol{x}}(k)$ and output measurement $\boldsymbol{y}(k)$. Since all the matrices are constant with time, equation (4.5) is a linear-time-invariant (LTI) model. Additionally, recalling the definition of observer output residual in equation (2.8), the Kalman filter in equation (4.5) is noticed to be in innovation form (section3.2).

Plugging equation (4.5b) into (4.5a) and equation (2.8) into (4.5b), the observer in (4.5) can be written in the equivalent form

$$\hat{\boldsymbol{x}}(k+1) = \bar{\mathbf{A}}\hat{\boldsymbol{x}}(k) + \mathbf{K}\boldsymbol{y}(k) \tag{4.6a}$$

$$\boldsymbol{y}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \boldsymbol{\epsilon}(k) \tag{4.6b}$$

where $\bar{\mathbf{A}} = \mathbf{A} - \mathbf{KC}$.

Propagating (4.6) forward in time by $p-1$ time steps (via repeated substitution) and then shifting the time index backward by p, we obtain

$$\hat{\boldsymbol{x}}(k) = \bar{\mathbf{A}}^p\hat{\boldsymbol{x}}(k-p) + \mathbf{T}\boldsymbol{v}(k) \tag{4.7}$$

where

$$\boldsymbol{v}(k) = \begin{bmatrix} \boldsymbol{y}(k-1) \\ \boldsymbol{y}(k-2) \\ \vdots \\ \boldsymbol{y}(k-p) \end{bmatrix} \tag{4.8a}$$

$$\mathbf{T} = \begin{bmatrix} \mathbf{K} & \bar{\mathbf{A}}\mathbf{K} & \dots & \bar{\mathbf{A}}^{p-2}\mathbf{K} & \bar{\mathbf{A}}^{p-1}\mathbf{K} \end{bmatrix} \tag{4.8b}$$

The stability of the Kalman filter (Reference 21) guarantees that $\bar{\mathbf{A}}^p$ becomes negligible for sufficiently large values of p ($p >> n$). Equation (4.7) yields then the following relation expressing the current state as a linear combination of solely past output values

$$\hat{\boldsymbol{x}}(k) = \mathbf{T}\boldsymbol{v}(k) \tag{4.9}$$

Plugging (4.9) into (4.6b), we obtain

$$\boldsymbol{y}(k) = \boldsymbol{\Phi}\boldsymbol{v}(k) + \boldsymbol{\epsilon}(k) \tag{4.10}$$

where

$$\boldsymbol{\Phi} = \begin{bmatrix} \mathbf{CK} & \mathbf{C\bar{A}K} & \dots & \mathbf{C\bar{A}}^{p-2}\mathbf{K} & \mathbf{C\bar{A}}^{p-1}\mathbf{K} \end{bmatrix} \tag{4.11}$$

Note that $\boldsymbol{\Phi}$ in equation (4.11) corresponds to the sequence of $\boldsymbol{\Phi}_j^{(2)}$, $j = 1, 2, \dots, p$, in equation (3.14b).

Equation (4.10) relates mesured output values, without the state appearing explicitly. Also, note that $\boldsymbol{\Phi} \in \mathbb{R}^{q \times qp}$ contains the sequence of Markov parameters (or unit pulse response) of the observer in bar form (see equation (4.21)). Equation (4.10) can be written for each time step $k = p, p+1, \dots, l-1$ of the measured data record, to obtain the following set of equations in matrix form

$$\mathbf{Y} = \boldsymbol{\Phi}\mathbf{V} + \mathbf{E} \tag{4.12}$$

where

$$\mathbf{Y} = \begin{bmatrix} \boldsymbol{y}(p) & \boldsymbol{y}(p+1) & \dots & \boldsymbol{y}(l-1) \end{bmatrix} \tag{4.13a}$$

$$\mathbf{V} = \begin{bmatrix} \boldsymbol{v}(p) & \boldsymbol{v}(p+1) & \dots & \boldsymbol{v}(l-1) \end{bmatrix} \tag{4.13b}$$

$$\mathbf{E} = \begin{bmatrix} \boldsymbol{\epsilon}(p) & \boldsymbol{\epsilon}(p+1) & \dots & \boldsymbol{\epsilon}(l-1) \end{bmatrix} \tag{4.13c}$$

Equation (4.12) is at the core of O^{3}KID. $\mathbf{Y}$ and $\mathbf{V}$ are known (from measurements), $\mathbf{\Phi}$ and $\mathbf{E}$ are not. By having $l - p > pq$ (more equations than unknowns) and considering $\mathbf{E}$ as an error term, it is possible to find the LS solution to (4.12)

$$\tilde{\mathbf{\Phi}} = \mathbf{Y}\mathbf{V}^{\mathrm{T}} \left(\mathbf{V}\mathbf{V}^{\mathrm{T}}\right)^{-1} = \mathbf{Y}\mathbf{V}^{\dagger} \tag{4.14}$$

where † denotes the Moore-Penrose pseudoinverse of a matrix, as well as the corresponding LS residuals

$$\tilde{\mathbf{E}} = \mathbf{Y} - \tilde{\mathbf{\Phi}}\mathbf{V} \tag{4.15}$$

Post-multiplying (4.12) by $\mathbf{V}^{\mathrm{T}}$ and replacing $\mathbf{\Phi}$ and $\mathbf{E}$ with their LS estimates $\tilde{\mathbf{\Phi}}$ and $\tilde{\mathbf{E}}$, we obtain

$$\mathbf{Y}\mathbf{V}^{\mathrm{T}} = \mathbf{Y}\mathbf{V}^{\mathrm{T}} \left(\mathbf{V}\mathbf{V}^{\mathrm{T}}\right)^{-1} \mathbf{V}\mathbf{V}^{\mathrm{T}} + \tilde{\mathbf{E}}\mathbf{V}^{\mathrm{T}} = \mathbf{Y}\mathbf{V}^{\mathrm{T}} + \tilde{\mathbf{E}}\mathbf{V}^{\mathrm{T}} \tag{4.16}$$

which implies that $\tilde{\mathbf{E}}\mathbf{V}^{\mathrm{T}} = \mathbf{0}$. From the definition of $\boldsymbol{v}(k)$, we conclude that

$$\sum_{k=p}^{l-1} \tilde{\boldsymbol{\epsilon}}(k)\boldsymbol{y}^{\mathrm{T}}(k-j) = \mathbf{0} \qquad j = 1, 2, ..., p \tag{4.17}$$

Since the stated assumptions make the process in (4.1) stationary, then the ergodic property applies and following exactly the same strategy as in section2.5 we can prove that the solution

to the LS problem in (4.12) yields an estimate of the Markov parameters and output residuals of the Kalman filter corresponding to the unknown system matrices $\mathbf{A}$, $\mathbf{C}$ and noise statistics $\mathbf{Q}'$, $\mathbf{R}'$, $\mathbf{S}'$ that generated the given output sequence $\{\boldsymbol{y}(k)\}$. The stability of the Kalman filter (Reference 21) also guarantees that the observer used to derive the O^{3}KID core equation is stable, as required for equation (4.9) to hold.

In summary, solving the O^{3}KID core equation (4.12) by LS yields an estimate for the sequence of the output residuals of the Kalman filter

$$\tilde{\mathbf{E}} = \begin{bmatrix} \tilde{\boldsymbol{\epsilon}}(p) & \tilde{\boldsymbol{\epsilon}}(p+1) & \tilde{\boldsymbol{\epsilon}}(p+2) & \dots & \tilde{\boldsymbol{\epsilon}}(l-1) \end{bmatrix} \tag{4.18}$$

and for the sequence of Markov parameters of the Kalman filter in bar form

$$\tilde{\bar{\mathbf{\Phi}}} = \begin{bmatrix} \tilde{\bar{\mathbf{\Phi}}}_1 & \tilde{\bar{\mathbf{\Phi}}}_2 & \dots & \tilde{\bar{\mathbf{\Phi}}}_{p-1} & \tilde{\bar{\mathbf{\Phi}}}_p \end{bmatrix} \tag{4.19}$$

where $\tilde{\bar{\mathbf{\Phi}}}_j \in \mathbb{R}^{q \times q}$ is an estimate for the j^{th} Markov parameter $\mathbf{C}\bar{\mathbf{A}}^{j-1}\mathbf{K}$.

4.4 Kalman filter identification via output residuals

Similar to OKID, the second step of O^{3}KID consists of the identification of the Kalman filter associated with the system to be identified and the noise statistics embedded in the data. Whereas the first step, i.e. solving the O^{3}KID core equation (4.12), is the same for all methods based on O^{3}KID, which differ by how they identify the Kalman filter. Similar to OKID, O^{3}KID methods can be classified in two families: the ones identifying the Kalman filter from its Markov parameters and those accomplishing the same task from its output residuals.

Recalling the definition of observer output residuals (2.8), (4.5) can be written as follows

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{A}\hat{\boldsymbol{x}}(k) + \mathbf{K}\boldsymbol{\epsilon}(k) \tag{4.20a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) \tag{4.20b}$$

which is the *innovation form* of the Kalman filter for the output-only system in equation (4.1). Equation (4.20) can also be looked at as the state-space model of a dynamic system with $\boldsymbol{\epsilon}$ as the input and $\hat{\boldsymbol{y}}$ as the output. Most importantly, no (unknown) noise term is present in (4.20). Therefore a new noise-free input-output identification problem can be formulated as follows. Given the sequences of $\boldsymbol{\epsilon}$ and $\hat{\boldsymbol{y}}$, identify the matrices of the system in (4.20), i.e. $\mathbf{A}$, $\mathbf{C}$ and $\mathbf{K}$. Yielding $\mathbf{A}$ and $\mathbf{C}$, the solution to the new problem would solve also the original identification problem.

From the first step of O^{3}KID, an estimate for the sequence of $\boldsymbol{\epsilon}(k)$ for $k = p, p+1, \ldots, l-1$ is indeed available, as well as for the sequence of $\hat{\boldsymbol{y}}(k)$, since $\boldsymbol{\epsilon}(k)$ and $\hat{\boldsymbol{y}}(k)$ are related via the definition of observer output residuals (2.8). We indeed converted the original identification problem (4.1), characterized by the presence of unknown signals ($\boldsymbol{w}'_p$ and $\boldsymbol{w}'_m$), into the simpler identification problem (4.20). *Any* method formulated for deterministic system identification can be used to address the new problem, solving the original problem at the same time. This gives rise to many O^{3}KID-based output-only identification algorithms, as many as the deterministic state-space model identification methods one can think of.

In this chapter, we demonstrate via examples two possible choices, namely the Deterministic Intersection (DI) and the Deterministic Projection (DP) methods described in appendix A and specialized below to the O^{3}KID case. The resulting O^{3}KID-based algorithms are referred to as O^{3}KID/DIi and O^{3}KID/DPi, where the lower-case letter i indicates that the underlying Kalman

filter is identified in its innovation form (4.20), distinguishing them from the following variant.

From (4.6), plugging (2.8) into (4.6b), we can write

$$\hat{\boldsymbol{x}}(k+1) = \bar{\mathbf{A}}\hat{\boldsymbol{x}}(k) + \mathbf{K}\boldsymbol{y}(k) \tag{4.21a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) \tag{4.21b}$$

which has already been referred to as the *bar form* of the Kalman filter. Similar to (4.20), notice how the dynamic system in (4.21) is purely deterministic, with input $\boldsymbol{y}(k)$ and output $\hat{\boldsymbol{y}}(k)$ that are both known (from measurement and from estimation, respectively). Again, *any* method for deterministic system identification can be applied to find $\bar{\mathbf{A}}$, $\mathbf{C}$ and $\mathbf{K}$, from which $\mathbf{A}$ can be recovered as $\mathbf{A} = \bar{\mathbf{A}} + KC$. Such variant, when coupled with DI and DP to identify the Kalman filter in bar form, gives rise to O^{3}KID/DIb and O^{3}KID/DPb. Their input-output versions, OKID/DIb and OKID/DPb, were presented in detail in section 3.4, but numerical examples suggest their performances are inferior to the algorithms based on the innovation form of the Kalman filter. Intuitively, the direct identification of $\mathbf{A}$ is preferable to the identification of $\bar{\mathbf{A}}$ and subsequent recovery of $\mathbf{A}$ from the estimated $\mathbf{K}$ and $\mathbf{C}$. Therefore, O^{3}KID/DIb and O^{3}KID/DPb are omitted in this chapter.

It is worth adding that other algorithms based on the same approach can be devised simply by replacing DI and DP by other deterministic methods. For instance, one could use the subspace Algorithm 1 and Algorithm 2 in Reference 2 or the algorithms from the superspace family (References 31–33). Also note that the approach based on the Kalman output residuals highlights the following aspect of O^{3}KID. The O^{3}KID core equation (4.12) allows one to convert the original stochastic identification problem, complicated by the presence of unknown signals ($\boldsymbol{w}'_p$ and $\boldsymbol{w}'_m$), into a new simpler, purely deterministic identification problem whose solution

includes the solution to the original problem. The intuitive interpretation is that the Kalman filter underlying (4.12) optimally filters the noise out of the problem, paralleling the same central role that the Kalman filter has in classic signal estimation (see section 3.4.2 and chapter 5). Since the record used to construct (4.12) is finite, the filtering action is not exact and neither are the resulting estimates of $\mathbf{A}$ and $\mathbf{C}$. Indeed, due to the stochastic nature of the problem addressed in this work, an infinite record of output data would be necessary to aim to exact identification of the system in (4.1). The fact that a Kalman filter exists for the system in (4.1) as well as for the system in (4.1) indeed inspired the extension of the OKID approach to the output-only case and led to O^{3}KID.

Although one could use the DI or DP algorithms in their original formulation, as described in appendix A, it is possible to modify them for a more elegant solution to take into account the special structure that the models in equations (4.20) and (4.21) have, i.e. no direct influence matrix in the observation equation. In the rest of the section, the DI and DP methods are then briefly outlined and specialized to the identification of the Kalman filter in (4.20) from the following data

$$\{\boldsymbol{\epsilon}(k)\} = \{\boldsymbol{\epsilon}(p), \boldsymbol{\epsilon}(p+1), \boldsymbol{\epsilon}(p+2), \ldots, \boldsymbol{\epsilon}(l-1)\} \tag{4.22a}$$

$$\{\hat{\boldsymbol{y}}(k)\} = \{\hat{\boldsymbol{y}}(p), \hat{\boldsymbol{y}}(p+1), \hat{\boldsymbol{y}}(p+2), \ldots, \hat{\boldsymbol{y}}(l-1)\} \tag{4.22b}$$

Note that in O^{3}KID/DIi and O^{3}KID/DPi, the true $\{\boldsymbol{\epsilon}\}$ and $\{\hat{\boldsymbol{y}}\}$ are not known and replaced by their estimates

$$\{\tilde{\boldsymbol{\epsilon}}(k)\} = \{\tilde{\boldsymbol{\epsilon}}(p), \tilde{\boldsymbol{\epsilon}}(p+1), \tilde{\boldsymbol{\epsilon}}(p+2), \ldots, \tilde{\boldsymbol{\epsilon}}(l-1)\} \tag{4.23a}$$

$$\{\tilde{\boldsymbol{y}}(k)\} = \{\tilde{\boldsymbol{y}}(p), \tilde{\boldsymbol{y}}(p+1), \tilde{\boldsymbol{y}}(p+2), \ldots, \tilde{\boldsymbol{y}}(l-1)\} \tag{4.23b}$$

where the latter sequence is obtained by $\tilde{\boldsymbol{y}}(k) = \boldsymbol{y}(k) - \tilde{\boldsymbol{\epsilon}}(k)$.

4.4.1 Deterministic intersection (DI) algorithm

The property at the core of the DI algorithm for deterministic system identification is that the Kalman state history can be expressed via the following two matrix relations

$$\hat{\mathbf{X}} = \mathbf{L}_p \mathbf{H}_p \qquad \hat{\mathbf{X}} = \mathbf{L}_f \mathbf{H}_f \tag{4.24}$$

where

$$\hat{\mathbf{X}} = \begin{bmatrix} \hat{\boldsymbol{x}}(p+i) & \hat{\boldsymbol{x}}(p+i+1) & \dots & \hat{\boldsymbol{x}}(l-i) \end{bmatrix} \tag{4.25a}$$

$$\mathbf{H}_p = \begin{bmatrix} \boldsymbol{\epsilon}(p) & \boldsymbol{\epsilon}(p+1) & \dots & \boldsymbol{\epsilon}(l-2i) \\ \hat{\boldsymbol{y}}(p) & \hat{\boldsymbol{y}}(p+1) & \dots & \hat{\boldsymbol{y}}(l-2i) \\ \boldsymbol{\epsilon}(p+1) & \boldsymbol{\epsilon}(p+2) & \dots & \boldsymbol{\epsilon}(l-2i+1) \\ \hat{\boldsymbol{y}}(p+1) & \hat{\boldsymbol{y}}(p+2) & \dots & \hat{\boldsymbol{y}}(l-2i+1) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\epsilon}(p+i-1) & \boldsymbol{\epsilon}(p+i) & \dots & \boldsymbol{\epsilon}(l-i-1) \\ \hat{\boldsymbol{y}}(p+i-1) & \hat{\boldsymbol{y}}(p+i) & \dots & \hat{\boldsymbol{y}}(l-i-1) \end{bmatrix} \tag{4.25b}$$

$$\mathbf{H}_f = \begin{bmatrix} \boldsymbol{\epsilon}(p+i) & \boldsymbol{\epsilon}(p+i+1) & \dots & \boldsymbol{\epsilon}(l-i) \\ \hat{\boldsymbol{y}}(p+i) & \hat{\boldsymbol{y}}(p+i+1) & \dots & \hat{\boldsymbol{y}}(l-i) \\ \boldsymbol{\epsilon}(p+i+1) & \boldsymbol{\epsilon}(p+i+2) & \dots & \boldsymbol{\epsilon}(l-i+1) \\ \hat{\boldsymbol{y}}(p+i+1) & \hat{\boldsymbol{y}}(p+i+2) & \dots & \hat{\boldsymbol{y}}(l-i+1) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\epsilon}(p+2i-1) & \boldsymbol{\epsilon}(p+2i) & \dots & \boldsymbol{\epsilon}(l-1) \\ \hat{\boldsymbol{y}}(p+2i-1) & \hat{\boldsymbol{y}}(p+2i) & \dots & \hat{\boldsymbol{y}}(l-1) \end{bmatrix} \tag{4.25c}$$

(4.25d)

and $\mathbf{L}_p$, $\mathbf{L}_f \in \mathbb{R}^{n \times 2qi}$ are two constant matrices. For (4.24) to hold, the parameter i must be such that $qi \geq n$. From a linear algebra perspective, (4.24) shows how the Kalman state sequence lies both in the row space of $\mathbf{H}_p$ and in the row space of $\mathbf{H}_f$. $\hat{\mathbf{X}}$ can then be reconstructed by intersection of the row spaces of $\mathbf{H}_p$ and $\mathbf{H}_f$, which can be accomplished by two singular value decompositions (SVDs). The most intuitive way to compute said intersection is to note that

(4.24) implies that

$$\mathbf{L}_p\mathbf{H}_p - \mathbf{L}_f\mathbf{H}_f = \mathbf{0} \tag{4.26}$$

which can be rewritten as

$$\begin{bmatrix}\mathbf{L}_p & -\mathbf{L}_f\end{bmatrix}\begin{bmatrix}\mathbf{H}_p \\ \\ \mathbf{H}_f\end{bmatrix} = \mathbf{0} \tag{4.27}$$

$\mathbf{L}_p$ and $\mathbf{L}_f$ can then be computed as the left singular vectors associated with the zero singular values of the comprehensive data matrix H, defined as the concatenation of the past data matrix $\mathbf{H}_p$ and the future data matrix $\mathbf{H}_f$

$$\mathbf{H} = \begin{bmatrix}\mathbf{H}_p \\ \\ \mathbf{H}_f\end{bmatrix} \tag{4.28}$$

The SVD of H is then needed

$$\mathbf{H} = \begin{bmatrix}\mathbf{U}_{11} & \mathbf{U}_{12} \\ \\ \mathbf{U}_{21} & \mathbf{U}_{22}\end{bmatrix}\begin{bmatrix}\mathbf{S}_{11} & \mathbf{0} \\ \\ \mathbf{0} & \mathbf{0}\end{bmatrix}\mathbf{V}^{\mathrm{T}} \tag{4.29}$$

where $\mathbf{S}_{11} \in \mathbb{R}^{(2qi+n)\times(2qi+n)}$ is a diagonal matrix with the non-zero singular values of $\mathbf{H}$, $\mathbf{V} \in \mathbb{R}^{(l-2i-p+1)\times(l-2i-p+1)}$ is the matrix with its right singular vectors, and the matrix $\mathbf{U}$ with the left singular vectors is partitioned into $\mathbf{U}_{11}, \mathbf{U}_{21} \in \mathbb{R}^{2qi\times(2qi+n)}$ and $\mathbf{U}_{12}, \mathbf{U}_{22} \in \mathbb{R}^{2qi\times(2qi-n)}$. By left-multiplying $\mathbf{H}$ with its left singular vectors corresponding to the zero singular values and recalling the orthogonality property of the singular vector matrix U, we obtain

$$\begin{bmatrix}\mathbf{U}_{12}^{\mathrm{T}} & \mathbf{U}_{22}^{\mathrm{T}}\end{bmatrix}\begin{bmatrix}\mathbf{H}_p \\ \\ \mathbf{H}_f\end{bmatrix} = \mathbf{0} \tag{4.30}$$

or equivalently

$$\mathbf{U}_{12}^{\mathrm{T}}\mathbf{H}_p = -\mathbf{U}_{22}^{\mathrm{T}}\mathbf{H}_f \tag{4.31}$$

Equation (4.31) shows that $\mathbf{U}_{12}^{\mathrm{T}}\mathbf{H}_p$ (or $-\mathbf{U}_{22}^{\mathrm{T}}\mathbf{H}_f$) represents the required intersection of the row spaces of $\mathbf{H}_p$ and $\mathbf{H}_f$. However, $\mathbf{U}_{12}^{\mathrm{T}}\mathbf{H}_p$ contains $2qi-n$ row vectors, only n of which are linearly independent. Another SVD can be taken to compute a basis of n linearly independent vectors for its row space. Such basis provides the sequence of the Kalman state $\hat{\mathbf{X}}$ and the identification of the Kalman filter in (4.20) can then be completed by solving the following sets of linear equations

$$\begin{bmatrix} \hat{\boldsymbol{x}}(p+i+1) & \dots & \hat{\boldsymbol{x}}(l-i) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{K} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}}(p+i) & \dots & \hat{\boldsymbol{x}}(l-i-1) \\ \tilde{\boldsymbol{\epsilon}}(p+i) & \dots & \tilde{\boldsymbol{\epsilon}}(l-i-1) \end{bmatrix} \tag{4.32a}$$

$$\begin{bmatrix} \hat{\boldsymbol{y}}(p+i) & \dots & \hat{\boldsymbol{y}}(l-i-1) \end{bmatrix} = \mathbf{C} \begin{bmatrix} \hat{\boldsymbol{x}}(p+i) & \dots & \hat{\boldsymbol{x}}(l-i-1) \end{bmatrix} \tag{4.32b}$$

Since the true $\boldsymbol{\epsilon}$ and $\hat{\boldsymbol{y}}$ are not known and only their estimates are available from the OKID equation (4.12), the above equations should be solved in a LS sense. Note that the traditional solution scheme of DI would solve (4.32) at once as in equation (A.26), identifying also a direct influence matrix. In the Kalman filter in (4.20), there is no direct influence matrix and such information can be included in the identification process by solving (4.32) instead of (A.26). Obviously, in the latter case the direct influence matrix of the Kalman filter model is expected to be found equal to $\mathbf{0}$. As verified by numerical and experimental examples, indeed it turns out to be negligible and existing DI codes such as the one provided with Reference 2and described in appendix A can be used without modification. Additionally, note that the LS estimate of $\mathbf{A}$ is independent from the presence of a direct influence matrix in equation (A.26). If one is only interested in the natural frequencies and damping factors of the system, only (4.32a) needs to be solved.

In Reference 29 a more robust and computationally efficient way to complete the intersection of $\mathbf{H}_p$ and $\mathbf{H}_f$ and construct a LS problem equivalent to (4.32) is provided. With reference to

the identification of the model in (4.20), the algorithm in Reference 29 can be summarized and modified to take into account the absence of a direct influence matrix as follows.

DI algorithm for output-only Kalman filter identification

1. construct the matrices $\mathbf{H}_p$ and $\mathbf{H}_f$ in (4.25b) and (4.25c) choosing i greater than q times the assumed order of the system to be identified, and concatenate them into a single matrix $\mathbf{H}$ as defined in (4.28)

2. take the SVD of $\mathbf{H}$

$$\mathbf{H} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}} = \begin{bmatrix} \mathbf{U}_{11} & \mathbf{U}_{12} \\ \mathbf{U}_{21} & \mathbf{U}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{S}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \mathbf{V}^{\mathrm{T}} \tag{4.33}$$

3. calculate $\mathbf{U}_q$ from the SVD of $\mathbf{U}_{12}^{\mathrm{T}}\mathbf{U}_{11}\mathbf{S}_{11}$

$$\mathbf{U}_{12}^{\mathrm{T}}\mathbf{U}_{11}\mathbf{S}_{11} = \begin{bmatrix} \mathbf{U}_q & \mathbf{U}_r \end{bmatrix} \begin{bmatrix} \mathbf{S}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}_q^{\mathrm{T}} \\ \mathbf{V}_r^{\mathrm{T}} \end{bmatrix} \tag{4.34}$$

4. solve by LS the following set of equations for $\mathbf{A}$ and $\mathbf{K}$

$$\mathbf{U}_q^{\mathrm{T}}\mathbf{U}_{12}^{\mathrm{T}}\mathbf{U}(2q+1:2q(i+1),1:2qi+n)\mathbf{S}_{11} = \begin{bmatrix} \mathbf{A} & \mathbf{K} \end{bmatrix} \begin{bmatrix} \mathbf{U}_q^{\mathrm{T}}\mathbf{U}_{12}^{\mathrm{T}}\mathbf{U}(1:2qi,1:2qi+n)\mathbf{S}_{11} \\ \mathbf{U}(2qi+1:2qi+q),1:2qi+n)\mathbf{S}_{11} \end{bmatrix} \tag{4.35}$$

5. solve by LS the following set of equations for $\mathbf{C}$

$$\mathbf{U}(2qi+q+1:2q(i+1),1:2qi+n)\mathbf{S}_{11} = \mathbf{C}\mathbf{U}_q^{\mathrm{T}}\mathbf{U}_{12}^{\mathrm{T}}\mathbf{U}(1:2qi,1:2qi+n)\mathbf{S}_{11} \tag{4.36}$$

where the standard Matlab® notation has been used to indicate specific submatrices of $\mathbf{U}$. For example, $\mathbf{U}(a:b, c:d)$ indicates the submatrix of $\mathbf{U}$ at the intersection of rows $\mathbf{A}$ to $\mathbf{B}$ and columns $\mathbf{C}$ to $\mathbf{D}$.

4.4.2 Deterministic projection (DP) algorithm

As an alternative, the Kalman filter in (4.20) can be identified by the Deterministic Projection (DP) algorithm. The equation at the core of the method is the following

$$\hat{\mathbf{Y}}_h = \mathbf{\Gamma}\hat{\mathbf{X}} + \mathbf{H}_t\mathbf{E}_h \tag{4.37}$$

where $\hat{\mathbf{Y}}_h$ and $\mathbf{E}_h$ are matrices with Kalman output and residual data arranged according to a block Hankel structure, $\mathbf{H}_t$ is a block Toeplitz lower-triangular matrix that contains the first i Markov parameters of the Kalman filter in innovation form, $\mathbf{\Gamma}$ is the extended observability

matrix and $\hat{\mathbf{X}}$ is the sequence of Kalman states

$$\hat{\mathbf{Y}}_h = \begin{bmatrix} \hat{\boldsymbol{y}}(p) & \hat{\boldsymbol{y}}(p+1) & \dots & \hat{\boldsymbol{y}}(l-i) \\ \hat{\boldsymbol{y}}(p+1) & \hat{\boldsymbol{y}}(p+2) & \dots & \hat{\boldsymbol{y}}(l-i+1) \\ \vdots & \vdots & \ddots & \vdots \\ \hat{\boldsymbol{y}}(p+i-1) & \hat{\boldsymbol{y}}(p+i) & \dots & \hat{\boldsymbol{y}}(l-1) \end{bmatrix} \tag{4.38a}$$

$$\mathbf{E}_h = \begin{bmatrix} \boldsymbol{\epsilon}(p) & \boldsymbol{\epsilon}(p+1) & \dots & \boldsymbol{\epsilon}(l-i) \\ \boldsymbol{\epsilon}(p+1) & \boldsymbol{\epsilon}(p+2) & \dots & \boldsymbol{\epsilon}(l-i+1) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{\epsilon}(p+i-1) & \boldsymbol{\epsilon}(p+i) & \dots & \boldsymbol{\epsilon}(l-1) \end{bmatrix} \tag{4.38b}$$

$$\mathbf{H}_t = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{CK} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{CAK} & \mathbf{CK} & \mathbf{0} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{i-2}\mathbf{K} & \mathbf{CA}^{i-3}\mathbf{K} & \mathbf{CA}^{i-4}\mathbf{K} & \dots & \mathbf{0} \end{bmatrix} \tag{4.38c}$$

$$\hat{\mathbf{X}} = \begin{bmatrix} \hat{\boldsymbol{x}}(p) & \hat{\boldsymbol{x}}(p+1) & \dots & \hat{\boldsymbol{x}}(l-i) \end{bmatrix} \tag{4.38d}$$

$$\boldsymbol{\Gamma} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{i-1} \end{bmatrix} \tag{4.38e}$$

The key observation in the DP method is of geometrical nature. Under general conditions (Reference 30), the dimension of the projection of the row space of $\hat{\mathbf{Y}}_h$ on the orthogonal complement of the row space of $\mathbf{E}_h$ provides the order of the system and its column space yields

an estimate of $\mathbf{\Gamma}$. Indeed the estimation of the observability matrix is at the core of the DP method, similar to how the reconstruction of the Kalman state history is the keystone of the DI method. Once $\mathbf{\Gamma}$ is estimated, $\mathbf{C}$ can be read as the first q rows of $\mathbf{\Gamma}$ whereas $\mathbf{A}$ can be computed thanks to the following relationship with $\mathbf{\Gamma}$

$$\underline{\mathbf{\Gamma}}\mathbf{A} = \bar{\mathbf{\Gamma}} \tag{4.39}$$

where $\underline{\mathbf{\Gamma}}$ and $\bar{\mathbf{\Gamma}}$ are obtained from $\mathbf{\Gamma}$ by deleting the last and first q rows, respectively.

Let $\mathbf{\Pi}_{\mathbf{E}_h^\perp}$ be the matrix that projects the row space of a matrix onto the row space of the orthogonal complement to the row space of $\mathbf{E}_h$. The projection of $\hat{\mathbf{Y}}_h$ onto $\mathbf{E}_h^\perp$ is denoted by $\hat{\mathbf{Y}}_h/\mathbf{E}_h^\perp$ and can be expressed as $\hat{\mathbf{Y}}_h/\mathbf{E}_h^\perp = \hat{\mathbf{Y}}_h\mathbf{\Pi}_{\mathbf{E}_h^\perp}$. Post-multiplying both sides of (4.37) by $\Pi_{\mathbf{E}_h^\perp}$, we obtain

$$\hat{\mathbf{Y}}_h/\mathbf{E}_h^\perp = \mathbf{\Gamma}\hat{\mathbf{X}}\mathbf{\Pi}_{\mathbf{E}_h^\perp} \tag{4.40}$$

because $\mathbf{E}_h\mathbf{\Pi}_{\mathbf{E}_h^\perp}$ is null by definition. Take the SVD of $\hat{\mathbf{Y}}_h/\mathbf{E}_h^\perp$

$$\hat{\mathbf{Y}}_h/\mathbf{E}_h^\perp = \begin{bmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{bmatrix}\begin{bmatrix}\mathbf{S}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{V}_1^{\mathrm{T}} \\ \mathbf{V}_2^{\mathrm{T}}\end{bmatrix} \tag{4.41}$$

where $\mathbf{S}_1 \in \mathbb{R}^{n\times n}$ is a diagonal matrix containing the non-zero singular values, $\mathbf{U}_1 \in \mathbb{R}^{qi\times n}$ and $\mathbf{U}_2 \in \mathbb{R}^{qi\times(qi-n)}$ contain the left singular vectors, and $\mathbf{V}_1 \in \mathbb{R}^{(l-i-p+1)\times n}$ and $\mathbf{V}_2 \in \mathbb{R}^{(l-i-p+1)\times(l-i-p+1-n)}$ contain the right singular vectors. $\mathbf{U}_1$ is then an orthonormal basis for the column space of $\hat{\mathbf{Y}}_h/\mathbf{E}_h^\perp$ and can be taken as an estimate of the extended observability matrix $\mathbf{\Gamma}$. The matrix $\mathbf{C}$ can be read as the first q rows of $\mathbf{U}_1$ and $\mathbf{A}$ can be estimated as

$$\mathbf{A} = \underline{\mathbf{U}}_1^\dagger\bar{\mathbf{U}}_1 \tag{4.42}$$

where $\underline{\mathbf{U}}_1$ and $\bar{\mathbf{U}}_1$ are defined similarly to $\underline{\mathbf{\Gamma}}$ and $\bar{\mathbf{\Gamma}}$.

The identification of $\mathbf{A}$ and $\mathbf{C}$ solves the original problem and allows one to compute the natural frequencies, the damping factors and the mode shapes for the structure under consideration. If it is also desired to identify the Kalman gain $\mathbf{K}$, an extra step is necessary. By the orthonormality properties of the SVD, the columns of $\mathbf{U}_2$ are orthonormal to the columns of $\mathbf{U}_1$, i.e. to the columns of $\mathbf{\Gamma}$. Pre-multiplying (4.37) by $\mathbf{U}_2^{\mathrm{T}}$ and post-multiplying by $\mathbf{E}_h^{\dagger}$, we obtain

$$\mathbf{U}_2^{\mathrm{T}}\hat{\mathbf{Y}}_h\mathbf{E}_h^{\dagger} = \mathbf{U}_2^{\mathrm{T}}\mathbf{H}_t \tag{4.43}$$

which can be rewritten as an overdetermined set of linear equations in the unknown $\mathbf{K}$ as follows. Denote by $\mathbf{M}_j \in \mathbb{R}^{(qi-n)\times q}$ and $\mathbf{L}_j \in \mathbb{R}^{(qi-n)\times q}$ the block columns of $\mathbf{U}_2^{\mathrm{T}}\hat{\mathbf{Y}}_h\mathbf{E}_h^{\dagger}$ and $\mathbf{U}_2^{\mathrm{T}}$, respectively, i.e.

$$\mathbf{U}_2^{\mathrm{T}}\mathbf{Y}_h\mathbf{E}_h^{\dagger} = \begin{bmatrix}\mathbf{M}_1 & \mathbf{M}_2 & \dots & \mathbf{M}_i\end{bmatrix} \tag{4.44a}$$

$$\mathbf{U}_2^{\mathrm{T}} = \begin{bmatrix}\mathbf{L}_1 & \mathbf{L}_2 & \dots & \mathbf{L}_i\end{bmatrix} \tag{4.44b}$$

Equation (4.43) can then be written as

$$\begin{bmatrix}\mathbf{M}_1\\ \mathbf{M}_2\\ \vdots\\ \mathbf{M}_{i-1}\end{bmatrix} = \begin{bmatrix}\mathbf{L}_2 & \mathbf{L}_3 & \dots & \mathbf{L}_i\\ \mathbf{L}_3 & \mathbf{L}_4 & \dots & \mathbf{0}\\ \vdots & \vdots & \dots & \vdots\\ \mathbf{L}_i & \mathbf{0} & \dots & \mathbf{0}\end{bmatrix}\underline{\mathbf{U}}_1\mathbf{K} \tag{4.45}$$

and solved by LS for $\mathbf{K}$. Note that, similar to the DI algorithm, the classic implementation of the DP method (appendix A) would attempt to identify also a direct influence matrix , which is however known to be null for the system in (4.20). Existing codes for the classic DP algorithm

can be used without modification and yield a direct influence matrix approximately null, as verified by numerical examples. The version adopted in the examples on O^3KID/DPi illustrated in this chapter is tailored for the identification of the state-space model in equation (4.20). The corresponding DP algorithm is summarized below step by step for ease of implementation.

DP algorithm for output-only Kalman filter identification

1. construct the matrices $\hat{\mathbf{Y}}_h$ and $\mathbf{E}_h$ as in (4.38)
2. project[3] $\hat{\mathbf{Y}}_h$ on $\mathbf{E}_h^\perp$ to obtain $\hat{\mathbf{Y}}_h/\mathbf{E}_h^\perp$
3. calculate $\mathbf{U}_1$ and $\mathbf{U}_2$ from the SVD of $\hat{\mathbf{Y}}_h/\mathbf{E}_h^\perp$ as in (4.41)
4. read $\mathbf{C}$ as the first q rows of $\mathbf{U}_1$
5. compute $\mathbf{A}$ from (4.42)
6. if desired, solve by LS (4.45) to find the Kalman gain $\mathbf{K}$

4.5 Kalman filter identification via Markov parameters

The Kalman filter at the core of the O^3KID approach can be identified from its Markov parameters, whose estimate $\tilde{\mathbf{\Phi}}$ is obtained from the LS solution of the O^3KID core equation. Similar to output-only OKID, before applying ERA or ERA-DC, the following preliminary operation has to be done on the estimated Kalman filter Markov parameters. The latter refer to the Kalman filter in the bar form (4.21). With simple algebraic manipulation, it is possible to demonstrate that the sequence $\mathbf{\Phi}_j = \mathbf{C}\bar{\mathbf{A}}^{j-1}\mathbf{K}$ can be converted into the sequence $\mathbf{\Psi}_j = \mathbf{C}\mathbf{A}^{j-1}\mathbf{K}$, which

[3] The projection of $\hat{\mathbf{Y}}_h$ on $\mathbf{E}_h^\perp$ can be computed in closed form via the projection operator $\mathbf{\Pi}_{\mathbf{E}_h^\perp} = I - \mathbf{E}_h^{\mathrm{T}}\left(\mathbf{E}_h\mathbf{E}_h^{\mathrm{T}}\right)^\dagger \mathbf{E}_h$ or via numerically more robust techniques such as QR decomposition. The code for DP available at http://homes.esat.kuleuven.be/~smc/sysid/software/ relies on the latter, as illustrated in chapter 6 of Reference 2. In the examples illustrated in this chapter, the same technique based on QR decomposition is used.

corresponds to the Markov parameters of the Kalman filter in its innovation form (4.20). Additionally, it can be noticed that the sequence of estimated $\mathbf{C}\bar{\mathbf{A}}^{j-1}\mathbf{K}$ is finite ($j = 1, 2, ..., p$) only in appearance. For $j > p$, $\mathbf{C}\bar{\mathbf{A}}^{j-1}K$ can be considered to be equal to $\mathbf{0}$ as assumed in the derivation of the O^{3}KID core equation. In other words, the sequence of estimated Markov parameters $\tilde{\mathbf{\Phi}}_j$ can be extended to an arbitrary value $j = N > p$ simply by padding $\tilde{\mathbf{\Phi}}$ in (4.19) with zeros. The estimate for the Markov parameters of the Kalman filter in innovation form can then be computed as

$$\tilde{\mathbf{\Psi}}_1 = \tilde{\mathbf{\Phi}}_1 \tag{4.46a}$$

$$\tilde{\mathbf{\Psi}}_j = \tilde{\mathbf{\Phi}}_j + \sum_{h=1}^{j-1} \tilde{\mathbf{\Phi}}_h \tilde{\mathbf{\Psi}}_{j-h} \quad \text{for } j = 2, 3, \ldots, N \tag{4.46b}$$

From the Markov parameters $\mathbf{\Psi}_j = CA^{j-1}K$, $j = 1, 2, \ldots, N$, ERA or ERA-DC can be applied to identify the desired matrices $\mathbf{A}$, $\mathbf{C}$ and $\mathbf{K}$. The resulting algorithms are referred to as O^{3}KID/ERAi and O^{3}KID/ERA-DCi. It is worth noting how the Markov parameters $\mathbf{\Phi}_j = C\bar{\mathbf{A}}^{j-1}K$ could also have been used as an input to ERA or ERA-DC, leading to the identification of $\bar{\mathbf{A}}$, $\mathbf{C}$ and $\mathbf{K}$, from which $\mathbf{A}$ could then be recovered as $\mathbf{A} = \bar{\mathbf{A}} + \mathbf{KC}$. However, similar considerations to what observed for the identification of the Kalman filter via output residuals apply and the direct identification of $\mathbf{A}$ turns out to be more accurate. O^{3}KID/ERAb and O^{3}KID/ERA-DCb are then not illustrated in this chapter.

In contrast with the identification via observer output residuals, the use of Markov parameters is completely insensitive to the absence of a direct influence matrix in equation (4.20).

4.6 O^3KID algorithms

The detailed steps for O^3KID are described below. The user can choose between identifying the underlying Kalman filter from its Markov parameters or from its output residuals. Furthermore, within these two families, any deterministic state-space model identification algorithm can be chosen other than the ones illustrated in this work (DI, DP, ERA, ERA-DC).

The input to the following O^3KID-based algorithms is the sequence of measured system output $\{\boldsymbol{y}(k)\}$ of length l in (4.3). The output of the algorithm is the set of matrices $\mathbf{A}$, $\mathbf{C}$ and $\mathbf{K}$.

1. construct the matrices $\mathbf{Y}$ and $\mathbf{V}$ in (4.13a) and (4.13b), choosing p sufficiently larger than the assumed order n of the system (4.1) to be identified (typically, 20 times larger)

2. solve equation (4.12)

$$\tilde{\mathbf{\Phi}} = Y\mathbf{V}^\dagger \tag{4.47}$$

For algorithms based on Kalman filter output residuals

3. compute

$$\begin{bmatrix} \tilde{\boldsymbol{y}}(p) & \tilde{\boldsymbol{y}}(p+1) & \dots & \tilde{\boldsymbol{y}}(l-1) \end{bmatrix} = \tilde{\mathbf{\Phi}}\mathbf{V} \tag{4.48a}$$

$$\begin{bmatrix} \tilde{\boldsymbol{\epsilon}}(p) & \tilde{\boldsymbol{\epsilon}}(p+1) & \dots & \tilde{\boldsymbol{\epsilon}}(l-1) \end{bmatrix} = \mathbf{Y} - \tilde{\mathbf{\Phi}}\mathbf{V} \tag{4.48b}$$

4. execute, with input $\tilde{\boldsymbol{\epsilon}}(k)$ and $\tilde{\boldsymbol{y}}(k)$, $k = p, p+1, \dots, l-1$, any of the following algorithms

   - DI (in this section or in section A.3) for O^3KID/DIi
   - DP (in this section or in section A.4) for O^3KID/DPi

- any other algorithm for deterministic state-space model identification from arbitrary excitation

5. read the output matrices **A**, **C** and, if desired, **K**

For algorithms based on Kalman filter Markov parameters

3. partition $\tilde{\mathbf{\Phi}}$ in submatrices of q columns each

$$\tilde{\mathbf{\Phi}} = \begin{bmatrix} \tilde{\mathbf{\Phi}}_1 & \tilde{\mathbf{\Phi}}_2 & \dots & \tilde{\mathbf{\Phi}}_{p-1} & \tilde{\mathbf{\Phi}}_p \end{bmatrix} \tag{4.49}$$

4. compute $\tilde{\mathbf{\Psi}}_j$ for $j = 1, 2, \dots, N$ from (4.46) with $\tilde{\mathbf{\Phi}}_j = 0$ for $j > p$, choosing N such that the sequence $\tilde{\mathbf{\Psi}}_j$ covers a significant portion of the impulse response of the system

5. execute, with input $\tilde{\mathbf{\Psi}}_j$, $j = 1, 2, ..., N$, any of the following algorithms

   - ERA (section A.1) for O^{3}KID/ERAi
   - ERA-DC (section A.2) for O^{3}KID/ERA-DCi
   - any other algorithm for deterministic state-space model identification from unit pulse response (Markov parameters)

6. read the output matrices **A**, **C** and, if desired, **K**

4.7 Numerical examples

The following numerical examples aim to illustrate the correctness and effectiveness of the output-only identification approach proposed in this chapter. The results obtained from the four different O^{3}KID algorithms are reported and compared to highlight some of their features.

4.7.1 Four-story shear-type building

Consider the same lumped model of a four-story shear-type building shown in Figure 3.5 and described in section 3.4.3. Since the excitation is a white process normally distributed with zero mean and standard deviation of 1, the input satisfies the assumptions made in section 4.2. The gaussian noise that might affect the input channel is not explicitly modeled here, as it can be considered included in the unmeasured excitation mentioned above. With regard to the structural response, the lateral acceleration at each floor is measured, for a total of 4 outputs. Zero-mean gaussian noise is present in each output channel, with standard deviation expressed as a percentage $\sigma_{m\%}$ of the standard deviation σ_{y_i} of the corresponding true system output y_i. The complete discrete-time state-space model of the structure is therefore in the form of equation (1.3) with $n = 8$, $m = 1$ and $q = 4$. The process noise $\boldsymbol{w}_p$ is not explicitly modeled ($\mathbf{Q} = \mathbf{0}$), whereas the covariance of the measurement noise $\boldsymbol{w}_m$ is the diagonal matrix

$$\mathbf{R} = \sigma_{m\%}^2 \begin{bmatrix} \sigma_{y_1}^2 0 & 0 & 0 & \\ 0 & \sigma_{y_2}^2 & 0 & 0 \\ 0 & 0 & \sigma_{y_3}^2 & 0 \\ 0 & 0 & 0 & \sigma_{y_4}^2 \end{bmatrix} \tag{4.50}$$

In the output-only system identification problem addressed in this chapter, only the output (corrupted by noise) is assumed to be measured. The contribution of the input to the state and measurement dynamics is absorbed in the noise terms $\boldsymbol{w}_p$ and $\boldsymbol{w}_m$ in (4.1), which then have covariance matrices $\mathbf{Q}' = \mathbf{B}\mathbf{B}^{\mathrm{T}}$ and $\mathbf{R}' = \mathbf{D}\mathbf{D}^{\mathrm{T}} + \mathbf{R}$, respectively. The model to be identified consists in the matrices $\mathbf{A}$ and $\mathbf{C}$.

4.7.2 Full set of sensors

The following study is based on Monte Carlo simulations of 100 runs each and different measurement noise levels $\sigma_{m\%}$. In each Monte Carlo simulation, the measurement noise level is kept constant as well as the standard deviation of the excitation. Independent sequences for $\boldsymbol{w}_p$ and $\boldsymbol{w}_m$ are generated for each run and the corresponding output sequences obtained from (4.1) are fed to the illustrated algorithms to identify the matrices $\mathbf{A}$ and $\mathbf{C}$. In each of the four O^{3}KID-based algorithms, the SVD of a data matrix needs to be performed. The system order is revealed at this stage by the number of singular values that can be considered different from zero. In deterministic system identification (noise-free input and output data), the separation between non-zero and zero singular values is an extremely simple task since the latter are indeed numerically zero. The presence of noise $\boldsymbol{w}_p$ and $\boldsymbol{w}_m$ prevents the singular values from being exactly zero and the user is asked to select the singular values that are sufficiently small to be considered zero and discarded. As an example, Figure 4.1 shows the singular value plots obtained for each O^{3}KID-based algorithm for a single run of the Monte Carlo simulation with $\sigma_{m\%} = 10\%$. Note how for O^{3}KID/DPi, O^{3}KID/ERAi and O^{3}KID/ERA-DCi there is a clear gap between the 8 largest singular values and the others. The order $n = 8$ is then correctly selected without difficulty. In contrast, the singular value plot in O^{3}KID/DIi does not reveal the order of the system as clearly. In the Monte Carlo simulation, in order to compare the results from the different methods, the number of non-zero singular values chosen in O^{3}KID/DIi is forced to 8.

From the identified matrices $\mathbf{A}$ and $\mathbf{C}$, the modal parameters of the structures can be computed as follows. By using the eigenvectors of $\mathbf{A}$, transform the identified system into modal

coordinates. The resulting state-space model has system and output matrices

$$\mathbf{\Lambda} = \boldsymbol{\phi}^{-1}\mathbf{A}\boldsymbol{\phi} \tag{4.51a}$$

$$\mathbf{\Omega} = \mathbf{C}\boldsymbol{\phi} \tag{4.51b}$$

where the columns of $\boldsymbol{\phi} \in \mathbb{R}^{n\times n}$ correspond to the eigenvectors of $\mathbf{A}$ and $\mathbf{\Lambda}$ is a diagonal matrix with the eigenvalues of $\mathbf{A}$, namely $\lambda_1, \lambda_2, \ldots, \lambda_n$. When the system has no rigid body modes and all its vibrational modes are underdamped, the eigenvalues λ_i appear in complex conjugate pairs, i.e.

$$\lambda_{2i} = a_{2i} + \mathrm{j}b_{2i} = \lambda_{2i-1}^{*} \quad \text{for } i = 1, 2, \ldots, n/2 \tag{4.52}$$

where a_{2i} and b_{2i} represent the real and imaginary parts of λ_{2i}, respectively, and the superscript $*$ denotes the complex conjugate. Note that the order of the eigenvalues and the corresponding eigenvectors can be changed arbitrarily, provided that the re-ordering is done consistently and simultaneously for both. In general, each pair $a_{2i} \pm \mathrm{j}b_{2i}$ $(i = 1, 2, \ldots, n/2)$ corresponds to a second-order vibration mode. Under the standard zero-order-hold assumption, the corresponding undamped natural frequencies f_i (in Hz) and damping factors ζ_i (dimensionless ratio) can be computed as

$$f_i = \mathrm{mag}\left(\frac{\log\lambda_i}{2\pi\Delta t}\right) \tag{4.53a}$$

$$\zeta_i = -\cos\left(\mathrm{phase}\left(\log\lambda_i\right)\right) \tag{4.53b}$$

where Δt is the sampling time in seconds of the measured data. The mode shapes are given by the columns of $\mathbf{\Omega}$. Since the system is classically damped, for each mode the phase between different degrees of freedom is expected to be either 0 or π. Due to the noise $\boldsymbol{w}_p$ and $\boldsymbol{w}_m$, the estimated mode shapes will generally have some additional phase difference, which can be

TABLE 4.1: Numerical example – Identified natural frequencies (Hz) of the structure in Fig. 3.5 (Monte Carlo simulation, average $\bar{f}_i$ and standard deviation σ_i^f, $i = 1, 2, 3, 4$, over 100 runs) for different levels of measurement noise $\sigma_{m\%}$.

$\sigma_{m\%}$	Method	$\bar{f}_1$	σ_1^f	$\bar{f}_2$	σ_2^f	$\bar{f}_3$	σ_3^f	$\bar{f}_4$	σ_4^f
	True	1.395	–	3.972	–	5.945	–	7.012	–
	O^3KID/DIi	1.369	0.133	3.963	0.040	5.938	0.032	7.007	0.027
	O^3KID/DPi	1.397	0.019	3.974	0.010	5.945	0.012	7.013	0.013
0%	O^3KID/ERAi	1.398	0.007	3.973	0.009	5.946	0.012	7.012	0.012
	O^3KID/ERA-DCi	1.398	0.007	3.973	0.009	5.946	0.012	7.012	0.012
	N4SID	1.395	0.005	3.974	0.008	5.945	0.011	7.013	0.012
	O^3KID/DIi	1.390	0.006	3.970	0.008	5.943	0.011	7.011	0.011
	O^3KID/DPi	1.400	0.005	3.974	0.008	5.945	0.011	7.014	0.011
1%	O^3KID/ERAi	1.398	0.007	3.973	0.009	5.946	0.012	7.012	0.012
	O^3KID/ERA-DCi	1.398	0.007	3.973	0.009	5.946	0.012	7.012	0.012
	N4SID	1.395	0.005	3.974	0.008	5.945	0.011	7.014	0.012
	O^3KID/DIi	1.406	0.024	3.974	0.009	5.946	0.012	7.013	0.012
	O^3KID/DPi	1.401	0.005	3.974	0.008	5.945	0.011	7.014	0.011
10%	O^3KID/ERAi	1.397	0.007	3.973	0.009	5.947	0.012	7.013	0.012
	O^3KID/ERA-DCi	1.397	0.007	3.973	0.009	5.947	0.012	7.013	0.012
	N4SID	1.396	0.008	3.973	0.008	5.946	0.011	7.014	0.011
	O^3KID/DIi	1.400	0.006	3.972	0.008	5.946	0.011	7.013	0.012
	O^3KID/DPi	1.400	0.005	3.973	0.008	5.946	0.011	7.014	0.011
20%	O^3KID/ERAi	1.396	0.006	3.973	0.009	5.948	0.012	7.013	0.012
	O^3KID/ERA-DCi	1.396	0.006	3.973	0.009	5.948	0.012	7.013	0.012
	N4SID	1.397	0.009	3.973	0.008	5.946	0.011	7.014	0.012

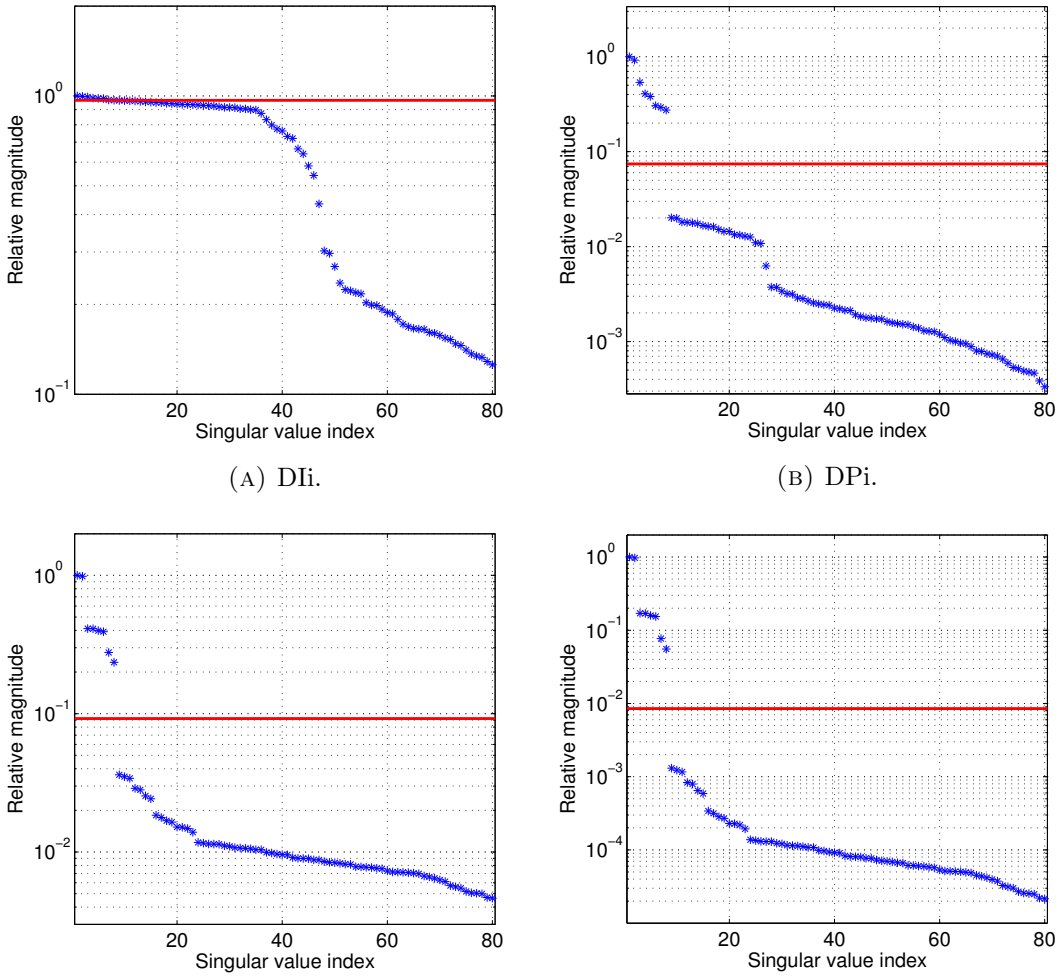
eliminated in the normalization process by finding the closest real mode shape.

Tables 4.1 and 4.2 report the average of the natural frequencies and damping factors identified in each Monte Carlo simulation, with different noise levels $\sigma_{m\%}$. All the four O^3KID algorithms are executed with $p = 100$ in the O^3KID equation, $i = 20$ in DI and DP and $N = 200$ in ERA and ERA-DC. The results from N4SID ($i = 20$) are also reported as a benchmark. N4SID (References 2, 35) is a an algorithm for system identification in the presence of noise that, similar to OKID, can be applied to both the input-output and output-only cases. As expected from the analysis of its singular value plot, the performance of O^3KID/DIi is inferior to the other algorithms. Remarkably, in most of the runs, O^3KID/DIi is able to identify the correct modes. However, the lower accuracy and precision, in particular for the first mode, reflect the fact that in Figure 4.1a the singular values corresponding to true and spurious modes are very close to each other. The other O^3KID algorithms provide estimates of natural

TABLE 4.2: Numerical example – Identified damping factors of the structure in Fig. 3.5 (Monte Carlo simulation, average $\bar{\zeta}_i$ and standard deviation σ_i^{ζ}, $i = 1, 2, 3, 4$, over 100 runs) for different levels of measurement noise $\sigma_{m\%}$.

$\sigma_{m\%}$	Method	$\bar{\zeta}_1$	σ_1^{ζ}	$\bar{\zeta}_2$	σ_2^{ζ}	$\bar{\zeta}_3$	σ_3^{ζ}	$\bar{\zeta}_4$	σ_4^{ζ}
	True	0.0100	–	0.0100	–	0.0100	–	0.0100	–
	O^3KID/DIi	0.0336	0.0831	0.0127	0.0094	0.0113	0.0038	0.0104	0.0022
	O^3KID/DPi	0.0102	0.0186	0.0103	0.0023	0.0102	0.0018	0.0099	0.0017
0%	O^3KID/ERAi	0.0246	0.0063	0.0076	0.0022	0.0090	0.0019	0.0094	0.0017
	O^3KID/ERA-DCi	0.0246	0.0063	0.0076	0.0022	0.0090	0.0019	0.0094	0.0017
	N4SID	0.0111	0.0047	0.0103	0.0020	0.0101	0.0018	0.0100	0.0015
	O^3KID/DIi	0.0141	0.0060	0.0108	0.0021	0.0105	0.0018	0.0102	0.0015
	O^3KID/DPi	0.0115	0.0047	0.0102	0.0020	0.0102	0.0018	0.0099	0.0015
1%	O^3KID/ERAi	0.0246	0.0063	0.0076	0.0022	0.0090	0.0019	0.0095	0.0017
	O^3KID/ERA-DCi	0.0246	0.0063	0.0076	0.0022	0.0090	0.0019	0.0095	0.0017
	N4SID	0.0114	0.0049	0.0103	0.0020	0.0102	0.0018	0.0100	0.0015
	O^3KID/DIi	0.0510	0.0390	0.0118	0.0028	0.0107	0.0020	0.0104	0.0017
	O^3KID/DPi	0.0111	0.0045	0.0101	0.0020	0.0101	0.0018	0.0099	0.0015
10%	O^3KID/ERAi	0.0230	0.0064	0.0078	0.0022	0.0091	0.0019	0.0095	0.0017
	O^3KID/ERA-DCi	0.0230	0.0064	0.0078	0.0022	0.0091	0.0019	0.0095	0.0017
	N4SID	0.0115	0.0050	0.0104	0.0021	0.0102	0.0018	0.0100	0.0015
	O^3KID/DIi	0.0111	0.0045	0.0101	0.0021	0.0102	0.0018	0.0100	0.0015
	O^3KID/DPi	0.0111	0.0045	0.0100	0.0020	0.0101	0.0018	0.0099	0.0015
20%	O^3KID/ERAi	0.0189	0.0064	0.0081	0.0022	0.0093	0.0019	0.0096	0.0017
	O^3KID/ERA-DCi	0.0189	0.0064	0.0081	0.0022	0.0093	0.0019	0.0096	0.0017
	N4SID	0.0115	0.0052	0.0104	0.0021	0.0102	0.0018	0.0100	0.0015

frequencies and damping factors that are extremely close to the true values. Both the average estimation error and the standard deviation are below 1%. The most critical mode from the identification point of view seems to be the first one. Note how N4SID is generally more accurate than O^3KID-based methods but as measurement noise increases the average error and standard deviation of N4SID estimates get worse, whereas O^3KID-based methods are not affected. Also note how O^3KID/DPi performs significantly better in the estimation of damping factors than O^3KID/ERAi and O^3KID/ERA-DCi. The two algorithms based on Kalman filter Markov parameters provide very similar estimates of natural frequencies and damping factors (the difference is associated with digits beyond those reported in the tables), but ERA-DC performs consistently better than ERA. The same is true for the estimates of the mode shapes and can be related to the ability of ERA-DC to better separate zero and non-zero singular values (Figures 4.1c and 4.1d). The mode shapes shown in Figure 4.2 are the averages obtained from



(A) DIi.

(B) DPi.

(C) ERAi (80 largest singular values).

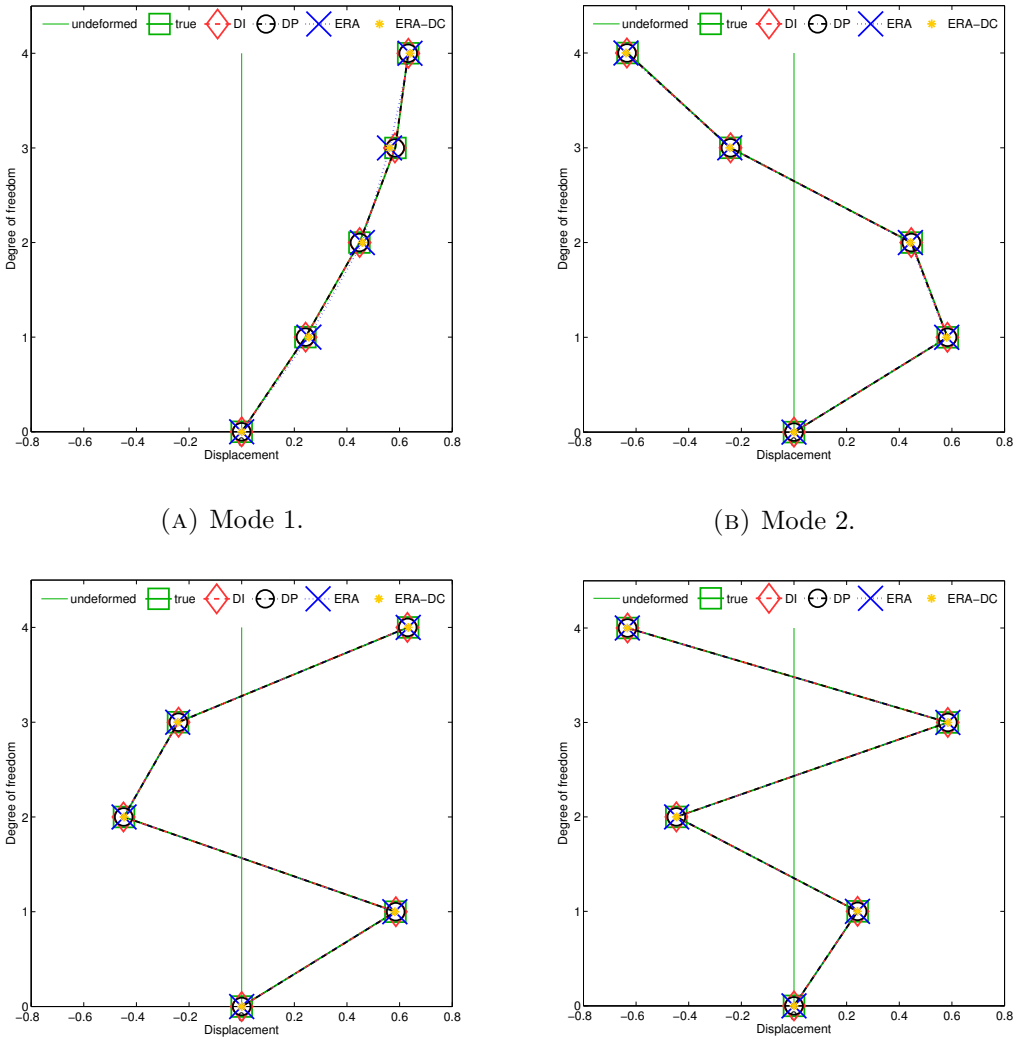
(D) ERA-DCi (80 largest singular values).

FIGURE 4.1: Numerical example – Singular value plots for O^{3}KID algorithms ($\sigma_{m\%} = 10\%$).

the Monte Carlo simulation with $\sigma_{m\%} = 10\%$ after normalization (unit eigenvectors). They are identified with great accuracy by all the proposed algorithms, in particular by the ones based on the estimation of the Kalman output residuals.

4.7.3 Reduced set of sensors

As an additional example to demonstrate O^{3}KID, consider the same structure in Figure 3.5 and the same study in section 4.7.2, this time assuming that the lateral acceleration is measured only in correspondence of the second and fourth floor. The four-degree-of-freedom model has



(A) Mode 1.

(B) Mode 2.

(C) Mode 3.

(D) Mode 4.

FIGURE 4.2: Numerical example – Comparison between true and identified modes (Monte Carlo simulation, average over 100 runs with $\sigma_{m\%} = 10\%$).

then the same $\mathbf{A} \in \mathbb{R}^{8\times8}$ matrix and reduced $\mathbf{C} \in \mathbb{R}^{2\times8}$ matrix. As shown in Figure 4.3, all O^{3}KID algorithms are able to identify the correct order of the system, even the variant based on DI. The latter produces indeed a very clear singular value plot, where the gap between zero and non-zero singular values is not large in magnitude but is very well defined, being the non-zero singular values very close to each other. O^{3}KID/ERA-DCi confirms its ability to separate the singular values better than O^{3}KID/ERAi.

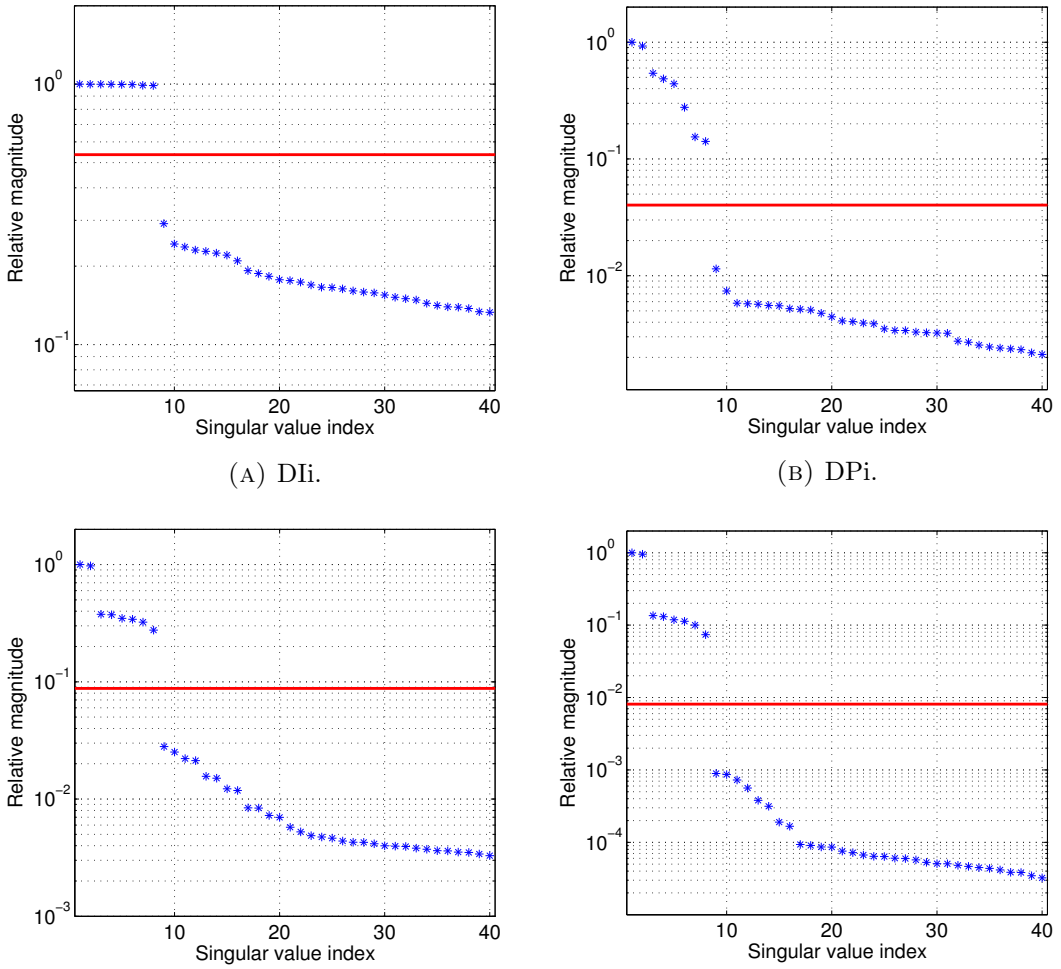
Tables 4.3 and 4.4 show how all the proposed algorithms successfully identify the four-story

TABLE 4.3: Numerical example with reduced set of sensors – Identified natural frequencies (Hz) of the structure in Fig. 3.5 (Monte Carlo simulation, average $\bar{f}_i$ and standard deviation σ_i^f, $i = 2, 4$, over 100 runs) for different levels of measurement noise $\sigma_{m\%}$.

$\sigma_{m\%}$	Method	$\bar{f}_1$	σ_1^f	$\bar{f}_2$	σ_2^f	$\bar{f}_3$	σ_3^f	$\bar{f}_4$	σ_4^f
	True	1.395	–	3.972	–	5.945	–	7.012	–
	O^3KID/DIi	1.395	0.006	3.973	0.008	5.946	0.011	7.013	0.011
	O^3KID/DPi	1.396	0.006	3.974	0.008	5.945	0.011	7.013	0.011
0%	O^3KID/ERAi	1.427	0.008	3.961	0.012	5.938	0.012	7.007	0.013
	O^3KID/ERA-DCi	1.427	0.009	3.961	0.012	5.938	0.012	7.007	0.013
	N4SID	1.396	0.006	3.973	0.008	5.945	0.011	7.013	0.011
	O^3KID/DIi	1.396	0.006	3.972	0.008	5.945	0.011	7.013	0.011
	O^3KID/DPi	1.398	0.006	3.975	0.008	5.945	0.011	7.013	0.011
1%	O^3KID/ERAi	1.505	0.353	4.060	0.352	5.963	0.185	7.023	0.070
	O^3KID/ERA-DCi	1.505	0.353	4.060	0.352	5.963	0.185	7.023	0.070
	N4SID	1.396	0.013	3.973	0.008	5.945	0.011	7.013	0.011
	O^3KID/DIi	1.396	0.005	3.973	0.008	5.945	0.011	7.013	0.011
	O^3KID/DPi	1.396	0.006	3.975	0.008	5.946	0.012	7.013	0.011
10%	O^3KID/ERAi	1.405	0.009	3.974	0.011	5.943	0.013	7.013	0.014
	O^3KID/ERA-DCi	1.405	0.009	3.974	0.011	5.943	0.013	7.013	0.014
	N4SID	1.397	0.017	3.973	0.008	5.945	0.012	7.013	0.012
	O^3KID/DIi	1.396	0.005	3.973	0.008	5.945	0.012	7.013	0.011
	O^3KID/DPi	1.396	0.005	3.976	0.008	5.946	0.012	7.013	0.012
20%	O^3KID/ERAi	1.393	0.007	3.979	0.011	5.943	0.013	7.014	0.014
	O^3KID/ERA-DCi	1.393	0.007	3.979	0.011	5.943	0.013	7.014	0.014
	N4SID	1.397	0.020	3.973	0.010	5.945	0.012	7.013	0.012

TABLE 4.4: Numerical example with reduced set of sensors – Identified damping factors of the structure in Fig. 3.5 (Monte Carlo simulation, average $\bar{\zeta}_i$ and standard deviation σ_i^ζ, $i = 2, 4$, over 100 runs) for different levels of measurement noise $\sigma_{m\%}$.

$\sigma_{m\%}$	Method	$\bar{\zeta}_1$	σ_1^ζ	$\bar{\zeta}_2$	σ_2^ζ	$\bar{\zeta}_3$	σ_3^ζ	$\bar{\zeta}_4$	σ_4^ζ
	True	0.0100	–	0.0100	–	0.0100	–	0.0100	–
	O^3KID/DIi	0.0123	0.0047	0.0111	0.0022	0.0110	0.0020	0.0108	0.0018
	O^3KID/DPi	0.0117	0.0053	0.0103	0.0020	0.0102	0.0018	0.0100	0.0015
0%	O^3KID/ERAi	0.0107	0.0061	0.0106	0.0026	0.0101	0.0021	0.0099	0.0018
	O^3KID/ERA-DCi	0.0107	0.0061	0.0106	0.0026	0.0101	0.0021	0.0099	0.0018
	N4SID	0.0112	0.0055	0.0104	0.0020	0.0102	0.0018	0.0100	0.0015
	O^3KID/DIi	0.0119	0.0050	0.0103	0.0020	0.0102	0.0018	0.0101	0.0016
	O^3KID/DPi	0.0109	0.0048	0.0101	0.0019	0.0100	0.0018	0.0099	0.0015
1%	O^3KID/ERAi	0.0105	0.0148	0.0129	0.0084	0.0093	0.0023	0.0071	0.0036
	O^3KID/ERA-DCi	0.0105	0.0148	0.0129	0.0084	0.0093	0.0023	0.0071	0.0036
	N4SID	0.0113	0.0060	0.0104	0.0020	0.0102	0.0019	0.0100	0.0015
	O^3KID/DIi	0.0121	0.0052	0.0104	0.0020	0.0103	0.0019	0.0101	0.0016
	O^3KID/DPi	0.0117	0.0053	0.0102	0.0019	0.0101	0.0018	0.0099	0.0015
10%	O^3KID/ERAi	0.0154	0.0061	0.0103	0.0026	0.0101	0.0019	0.0095	0.0019
	O^3KID/ERA-DCi	0.0154	0.0061	0.0103	0.0026	0.0101	0.0019	0.0095	0.0019
	N4SID	0.0110	0.0067	0.0103	0.0020	0.0103	0.0019	0.0100	0.0016
	O^3KID/DIi	0.0124	0.0056	0.0106	0.0021	0.0105	0.0019	0.0101	0.0016
	O^3KID/DPi	0.0118	0.0056	0.0103	0.0020	0.0102	0.0019	0.0099	0.0015
20%	O^3KID/ERAi	0.0143	0.0060	0.0105	0.0025	0.0107	0.0019	0.0100	0.0019
	O^3KID/ERA-DCi	0.0143	0.0060	0.0105	0.0025	0.0107	0.0019	0.0100	0.0019
	N4SID	0.0107	0.0074	0.0101	0.0025	0.0103	0.0020	0.0101	0.0016



(A) DIi.

(B) DPi.

(C) ERAi (40 largest singular values).

(D) ERA-DCi (40 largest singular values).

FIGURE 4.3: Numerical example with reduced set of sensors – Singular value plots for O^3KID algorithms ($\sigma_{m\%} = 10\%$).

structure even when the acceleration is measured at two floors only. In particular, notice how the algorithms based on the estimation of the Kalman output residuals provide very precise (small standard deviation) estimates at all noise levels. The corresponding estimated mode shapes match very well the true ones, as far as the measured degrees of freedom. The minimum condition to reconstruct the complete mode shapes is that each degree of freedom is instrumented (either with an actuator or a sensor), as proven in Reference 45. In output-only system identification, this requires the response at each degree of freedom to be measured, making it impossible to reconstruct the complete mode shapes in case of reduced set of sensors.

4.8 Experimental example

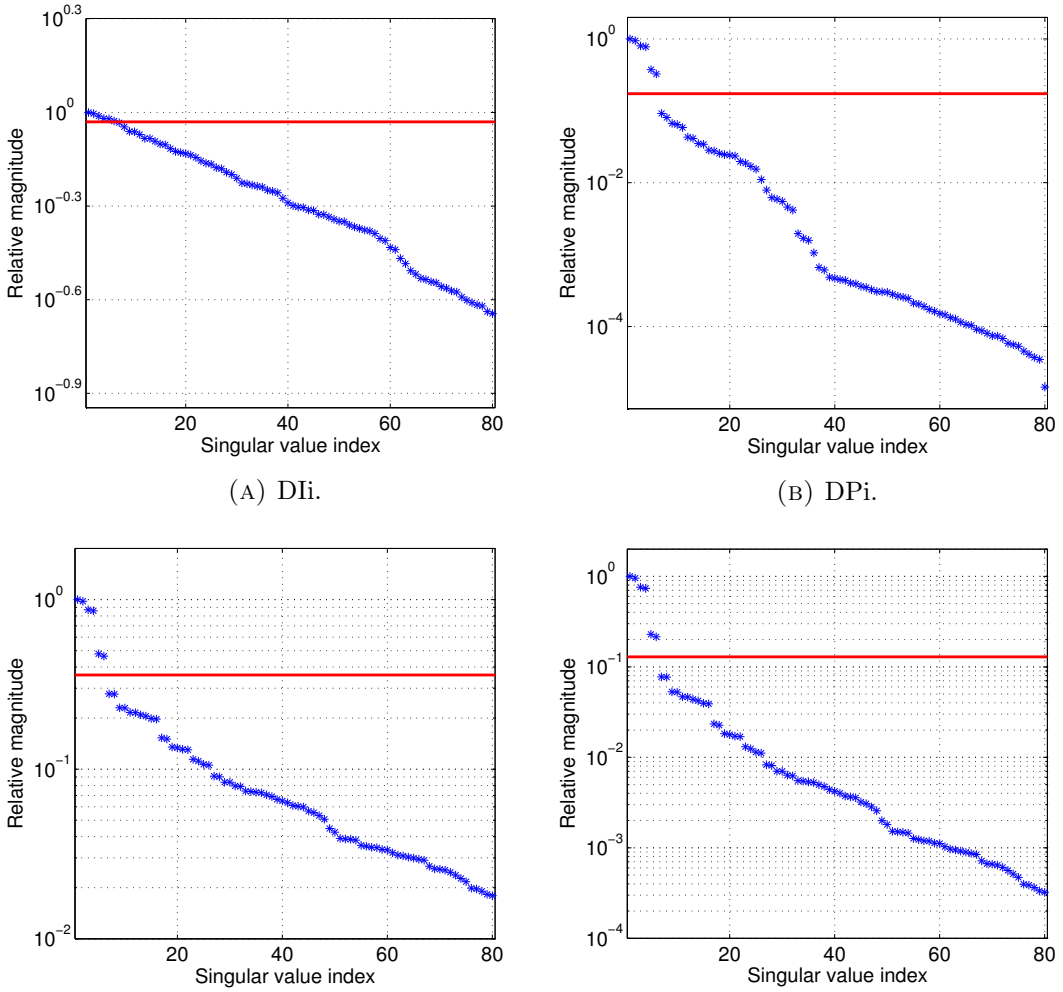
For the demonstration of O^3KID on real data, the same experimental tests performed in the Engineering Institute (EI) at Los Alamos National Laboratory (LANL) and used in chapter 3 are considered in this section.

Recall that the excitation is band-limited in the range of 20-150 Hz. Strictly speaking, such input does not satisfy the assumptions in section 4.2, but it is closer to the excitation that can occur in real applications (finite-power spectrum). Additionally, the excitation measurements allow one to estimate with greater accuracy the natural frequencies and damping factors, providing a gold standard against which the modal parameters identified by O^3KID can be compared. These facts make the example an ideal test bench for the proposed output-only identification methods.

In the following example, we use the first dataset for the standard configuration of the test structure (denoted as state #1 in Reference 36)[4]. Although the excitation input is measured, it is not fed to the O^3KID algorithms. These are applied choosing $p = 100$, $i = 20$ and $N = 200$. The resulting SVD plots are shown in Figure 4.4 and lead to the conclusion that the dynamic response of the structure is characterized by three vibration modes. Notice how, similar to the example in Figure 4.1, O^3KID/DIi does not show a clear cut between non-zero and zero singular values. In order to compare the identification results, the 6 largest singular values are retained as suggested by the SVD of the other algorithms.

Table 4.5 reports the natural frequencies and damping factors identified via O^3KID. The results are close to the estimates reported in Reference 36 and obtained by an input-output modal

[4]The data are available for download at http://institute.lanl.gov/ei/software-and-data/.



(A) DIi.

(B) DPi.

(C) ERA (80 largest singular values).

(D) ERA-DC (80 largest singular values).

FIGURE 4.4: LANL test structure – Singular value plots for O^3KID algorithms.

TABLE 4.5: LANL test structure – Identified natural frequencies (Hz) and damping factors, compared with the values reported by LANL.

Method	f_1	ζ_1	f_2	ζ_2	f_3	ζ_3
Input-Output RFP (LANL benchmark)	30.7	0.063	54.2	0.020	70.7	0.010
O^3KID/DIi	31.063	0.068	54.472	0.015	71.475	0.007
O^3KID/DPi	31.380	0.065	54.709	0.020	71.534	0.010
O^3KID/ERAi	31.176	0.045	54.445	0.017	71.429	0.008
O^3KID/ERA-DCi	31.176	0.045	54.445	0.017	71.429	0.008
N4SID	32.122	0.064	54.822	0.019	71.605	0.010

identification technique formulated in the frequency domain (Rational-Fraction Polynomial, RFP).

TABLE 4.6: LANL test structure with modified column stiffness – Identified natural frequencies (Hz) and damping factors.

Method	f_1	ζ_1	f_2	ζ_2	f_3	ζ_3
O^3KID/DIi	32.180	0.246	54.646	0.026	62.054	0.015
O^3KID/DPi	28.032	0.061	54.624	0.019	62.179	0.012
O^3KID/ERAi	27.955	0.021	54.413	0.019	62.179	0.011
O^3KID/ERA-DCi	27.955	0.021	54.413	0.019	62.179	0.011
N4SID	28.700	0.018	54.737	0.018	62.217	0.012

To further demonstrate the proposed algorithms, Table 4.6 shows the natural frequencies and damping factors obtained from another dataset in Reference 36, the one denoted as state #7 and characterized by reduced stiffness in two of the columns supporting the second floor. As expected, apart from O^3KID/DIi, the identified natural frequencies decrease whereas the damping factors do not change significantly.

4.9 Conclusions

The well-established OKID method for system identification from input-output data has been extended to the case where only the output measurements are available and the input can be considered to be a white process. Not only the traditional OKID algorithms completing the identification via the Kalman filter Markov parameters by ERA or ERA-DC have been specialized to the output-only case, but also new algorithms based on the estimation of the Kalman output residuals and on final identification by subspace methods such as Deterministic Intersection (DI) and Deterministic Projection (DP) have been formulated. All of the proposed algorithms have been illustrated via examples on both numerical and experimental data, which demonstrated how O^3KID is an effective approach for output-only system identification. In particular, O^3KID/DPi proved to be very reliable and accurate and its performance are in line with or better than existing methods for output-only system identification methods such as

N4SID. Similar to the family of subspace methods, OKID and O^3KID provide then a unified framework for input-output and output-only system identification and a useful tool for structural health monitoring.

Chapter 5

OKID as a unified framework for linear system identification

5.1 Introduction

The objective of this chapter is to provide an overview of OKID, without getting lost in the mathematical details. Whereas chapters 2 and 3 were meant to rigorously prove, describe and demonstrate the validity of the traditional and novel OKID algorithms, this chapter is intended to focus on the general and unified framework that OKID provides for system identification. Starting from a very high-level view of OKID, the detail level is gradually increased and, with the help of diagrams, the relationships among all the algorithms presented in chapter 3 are highlighted. Additionally, the connection between OKID and system identification methods based on ARX models (autoregressive models with exogenous input) and ARMAX (autoregressive-moving-average models with exogenous input) is discussed. In particular, the latter provides an alternative OKID core equation, i.e. it can improve the first step in the OKID approach.

5.2 OKID as a two-step identification process

All the algorithms presented in chapter 2 start with the construction of the LS problem in equation (2.44) and the corresponding solution. This LS problem then earned the name of *OKID core equation* as it represents the connecting link among the algorithms presented in chapter 3. The interpretation given in section 3.4 to the algorithms based on the estimation of the Kalman filter output residuals highlights how the use of the Kalman filter equations to derive the OKID core equation serves as a filter to the noisy input-output data obtained from measurements. The first step of OKID can then be interpreted as a filtering stage, to eliminate (or at least reduce) the undesired effect of noise on the identification of the matrices **A**, **B**, **C**, **D**. The concept is thoroughly discussed and illustrated in section 3.4.2.

After noise is filtered out of the data, the second step of OKID takes place and consists in the identification of the Kalman filter associated with the system to be identified and the statistical characteristics of the noise embedded in the measured input-output data. Whereas the first step provides some preliminary information on such a Kalman filter, either its unite pulse response (which is a deterministic and distinctive characteristic of the Kalman filter, its signature) or its output residuals (which serve to construct a new, purely deterministic identification problem), the second step completes the identification of the Kalman filter itself. Thanks to the close relationship between the Kalman filter model, equation (3.1) or (3.5), and the system model, equation (1.3), the matrices of the latter can be easily recovered, solving the original identification problem. The two-step process described above is summarized in the diagram in Figure 5.1. All the mathematical details are found in chapter 2 for step 1 and chapter 3 for step 2.

It is at this stage interesting to point out how the identification of the Kalman filter is not only

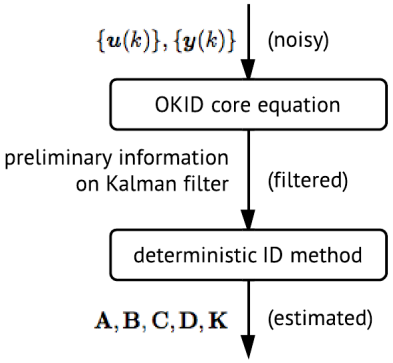


FIGURE 5.1: Two main steps of OKID.

appealing for the reasons mentioned in section 2.2, it is actually necessary in order to neutralize the detrimental effect that noise has on the identification process. The gain $\mathbf{K}$ can be explicitly found or not in OKID (depending on the chosen algorithm), nevertheless its presence is crucial in the formulation of OKID. It is indeed the Kalman filter that yields the OKID core equation, allowing one to find an asymptotically unbiased estimate of variables somewhat related to the system to be identified. In summary, OKID establishes the Kalman filter as the right tool to harness process and measurement noise in system identification. The optimal properties of the Kalman filter in signal estimation play a crucial role in system identification as well, as they allow the construction of a LS problem with white noise (as desired for unbiasedness) despite the process noise passing through the system becomes colored at the output measurement. The Kalman filter can then be considered to be the bridge between combined (i.e. from noisy data) and deterministic (i.e. from noise-free data) system identification.

The practical consequence of such a role for the Kalman filter in system identification is that OKID allows any deterministic system identification method to become combined. As mentioned in chapter 1, identification methods are referred to as *deterministic* when they are formulated without taking into account noise in the data, i.e. they are based on equation (1.1) instead of equation (1.3). Identification methods arising from the latter, like OKID algorithms, are called

combined since they take into account both the deterministic drive $\boldsymbol{u}$ and the stochastic drive $\boldsymbol{w}_p$, $\boldsymbol{w}_m$. The robustness of deterministic identification methods to noise in the data is then entirely in the hands of the numerical techniques used in the implementation. Noise generally makes the models estimated by deterministic methods biased. The benefit of the OKID core equation is that it can filter out the noise embedded in the measured data and let the identification get completed by deterministic identification methods, which then operate closer to the conditions for which they were formulated. By preliminary application of the OKID core equation and the construction of a new identification problem where the Kalman filter instead of the original system is identified, any deterministic identification method can result in a combined algorithm able to optimally handle noise.

Finally, it is worth commenting on the sources of estimation error in OKID-based algorithms, namely the truncation error in the OKID core equation and the finite-record error. In the derivation of equation (2.44), passing from equation (2.37) to equation (2.40) it is assumed that $\mathbf{F}^p$, which eventually turns out to correspond to $\bar{\mathbf{A}}^p$, is null. However, the Kalman gain $\mathbf{K}$ does not place the eigenvalues of $\bar{\mathbf{A}}^p$ at the origin as a deadbeat observer would. The presence of noise indeed makes the decay of the term $\bar{\mathbf{A}}^p$ gradual. Although it converges to zero as p is increased, it will never be identically zero. Hence, to explicitly remark the corresponding source of approximation, we should rewrite equation (2.41) as

$$\boldsymbol{y}(k) = \boldsymbol{\Phi}\boldsymbol{v}(k) + \boldsymbol{\epsilon}(k) + \boldsymbol{\epsilon}_t(k) \tag{5.1}$$

where $\boldsymbol{\epsilon}_t(k) = \mathbf{C}\bar{\mathbf{A}}^p\boldsymbol{x}(k-p)$ is the truncation error arising from the approximation made to derive equation (2.41). Correspondingly, the OKID core equation should be written as

$$\mathbf{Y} = \boldsymbol{\Phi}\mathbf{V} + \mathbf{E} + \mathbf{E}_t \tag{5.2}$$

where $\mathbf{E}_t$ is defined, similar to $\mathbf{E}$, as

$$\mathbf{E}_t = \begin{bmatrix} \boldsymbol{\epsilon}_t(0) & \boldsymbol{\epsilon}_t(1) & \dots & \boldsymbol{\epsilon}_t(l-p-1) \end{bmatrix} \tag{5.3}$$

When solving equation (5.2) via LS by considering the unknown terms $\mathbf{E}$ and $\mathbf{E}_t$ as error terms, the fact that $\boldsymbol{\epsilon}_t(k)$ is not white causes a bias in the estimate for $\boldsymbol{\Phi}$. The second source of estimation error is due to the finite length of the data record. Even assuming that there is no truncation error (or, as in practice, that p is chosen sufficiently large to make the truncation error negligible), the estimator for $\boldsymbol{\Phi}$ given by the LS solution to equation (2.44) is asymptotically unbiased. In practice, this means that an infinitely long record (infinite l) would be necessary to find the true value of $\boldsymbol{\Phi}$. The finite-record error is intrinsic to the problem described in section 2.2, due to the stochastic nature of the noise. The truncation error arises instead in the proposed approach to system identification, it is intrinsic to OKID. Nevertheless, as anticipated in section 3.4.2.3 and presented in more detail in section 5.5, the truncation error can be eliminated by a technique known as residual whitening.

5.3 Main variants within the framework of OKID

5.3.1 Input to the second step

After solving equation (2.44), two main approaches can be taken to identify the Kalman filter underlying the OKID core equation. The LS solution $\tilde{\boldsymbol{\Phi}}$ in equation (2.46) comes with the LS residuals $\tilde{\mathbf{E}}$ in equation (2.47). As rigorously proven in section 2.5, they correspond to the Kalman filter Markov parameters (or unit pulse response) and output residuals, respectively. Depending on which of the two new pieces of information one decides to focus on, two different OKID approaches arise, as illustrated in Figure 5.2. The left branch is based on the estimated

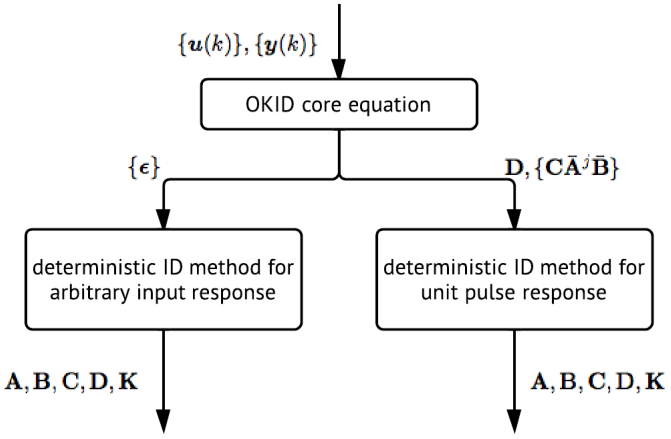


FIGURE 5.2: Two main approaches within OKID.

output residuals and identifies the underlying Kalman filter by any identification method able to find the state-space model of a linear system from its noise-free response to an arbitrary (and sufficiently rich) input sequence. This approach is described in detail in section 3.4. The right branch works on the estimate of the unit pulse response and identifies the underlying Kalman filter by any identification method able to find the state-space model of a linear system from its noise-free response to a unit pulse. This approach is described in detail in section 3.3.

5.3.2 Deterministic identification method

Within each branch in Figure 5.2, different methods can be chosen. Among the methods for system identification from arbitrary input response, the choice is virtually unlimited. In this work, popular and well-established methods such as deterministic intersection (DI) and deterministic projection (DP) have been chosen as representative to illustrate and demonstrate the proposed OKID approach. Many other choices are possible though, for example the subspace Algorithm 1 and Algorithm 2 in Reference 2 or the algorithms from the superspace family (References 31–33). Since the filtering action performed in the first step of OKID is not exact, one

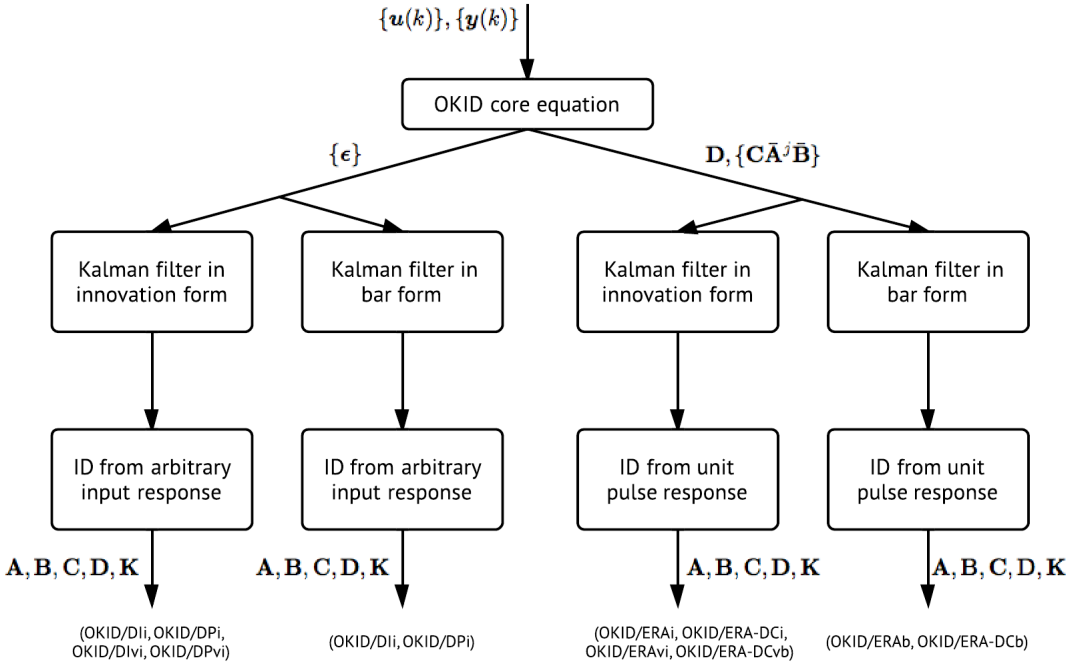


FIGURE 5.3: Two forms for the Kalman filter to be identified.

could also think of using a combined system identification method to better handle the filtering (estimation) error. Such a method could be OKID itself for an approach which could be named OKID^2. This idea is not developed here though, it is left as future work.

As to the methods for system identification from unit pulse response, eigensystem realization algorithm (ERA) and its improved version with data correlation (ERA-DC) have been chosen, partly for historical reasons (originally OKID was developed in conjunction with those methods) and also because ERA is the most established approach when a state-space realization is needed from a unit pulse response.

5.3.3 Kalman filter form

After deciding the approach to take and the specific method to use for the identification of the Kalman filter, the next question is about which form of the Kalman filter model one wants to

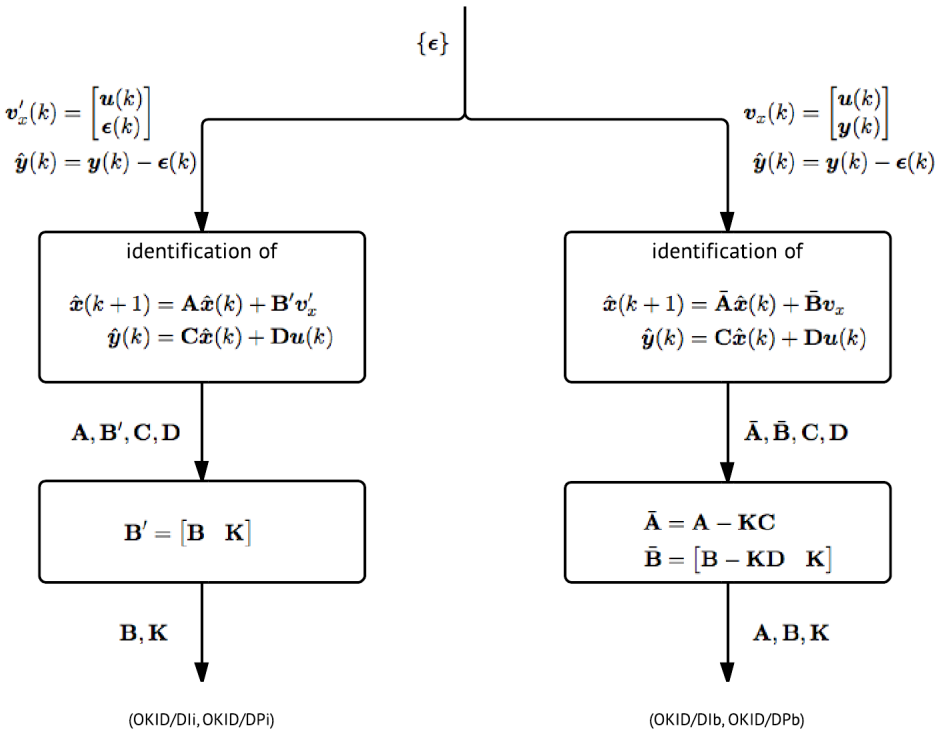


FIGURE 5.4: Detailed diagram for algorithms based on output residuals.

directly identify (Figure 5.3). The Kalman filter can be represented by two different although equivalent state space models, which were introduced in section 3.2 and in this work are referred to as *innovation form*, equations (3.1) or (3.2), and *bar form*, equations (3.5) and (3.7). Figure 5.3 illustrates the options and the resulting OKID algorithms. As the two Kalman filter forms are equivalent, it is always possible to pass from one to the other after their identification. However, as numerical experiments suggest (see for example section 3.4.3), the innovation form seems to be preferred. Intuitively, it makes indeed sense to identify the Kalman filter in the form the closest to the original dynamic system to be identified, and the innovation form in equation (3.1) is indeed the closest possible to equation (1.3). All the matrices match, apart from the former having an additional matrix, the Kalman gain $\mathbf{K}$, in its state equation.

Figures 5.4 and 5.5 show a more detailed diagram than Figures 5.1-5.3. For space reasons, the

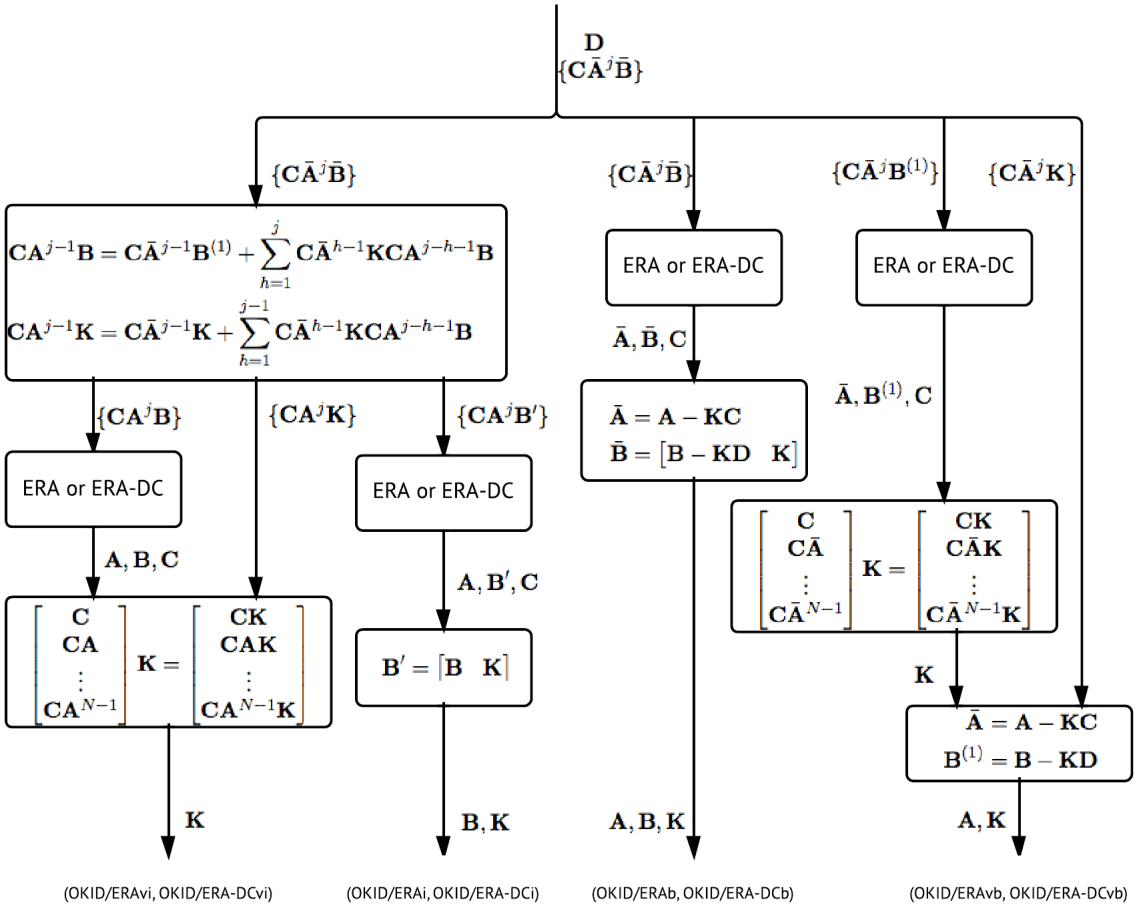


FIGURE 5.5: Detailed diagram for algorithms based on Markov parameters.

diagram is split in its two main parallel branches, shown in separate figures. The notation in the figures is consistent with the definitions given in chapter 3.

5.4 OKID and identification of ARX models

The framework laid out so far for OKID is quite broad, as it gives rise to many possible algorithms for system identification in the presence of noise. However it has been strictly limited to state-space model identification. A second glance at equation (2.41), which originates the OKID core equation, reveals that it is based on an autoregressive model with exogenous input

(ARX), affected by white error $\boldsymbol{\epsilon}$ (which can be regarded as playing the role of measurement noise). More precisely, as the current output depends on input and output up to p time steps back, the ARX model order is p and the model can be denoted by ARX(p). The LS solution to the core OKID equation then yields the following ARX(p) model

$$\boldsymbol{y}(k) = \boldsymbol{\Phi}_1^{(2)}\boldsymbol{y}(k-1) + \ldots + \boldsymbol{\Phi}_p^{(2)}\boldsymbol{y}(k-p) + \boldsymbol{\Phi}_0^{(1)}\boldsymbol{u}(k) + \ldots + \boldsymbol{\Phi}_p^{(1)}\boldsymbol{u}(k-p) \tag{5.4}$$

An ARX model predicts the output of the system from its previous outputs and current and previous inputs. It is a simpler description than a state-space model as it does not involve the state, but it is perfectly equivalent in terms of input-output mapping. When a simple input-output mapping is desired, it is then sufficient to identify an ARX model, without the need to go after a more complex representation of the system such as a state-space model. Therefore ARX models are quite popular in engineering practice. A comparison between pros and cons of ARX and state-space models is beyond the scope of this work.

The theory behind OKID (chapter 2) shows however something very interesting on ARX model identification. When data are corrupted by process and measurement noise, like in equation (1.3), then the ARX model to be identified should be of high order, typically much higher than the actual system order n. In fact the ARX model that one identifies is not the mathematical model of the dynamic system but the model of its optimal observer. The point can be illustrated via equation (5.1). If the data contain no noise, we know that the optimal steady-state observer (the one minimizing the norm squared of the state estimation error) is a deadbeat observer. By choosing $p = n$ the LS solution to equation (2.44) yields an ARX(n) model with the gain $\mathbf{K}$ embedded in $\boldsymbol{\Phi}$ corresponding to the deadbeat gain, i.e. being such that $\bar{\mathbf{A}}^j = \mathbf{0}$ for all $j \geq n$. The truncation error $\boldsymbol{\epsilon}_t(k) = \mathbf{C}\bar{\mathbf{A}}^n\boldsymbol{x}(k-n)$ is null and so is the output residual $\boldsymbol{\epsilon}(k)$. Therefore, with noise-free input-output data it makes sense to look for an ARX model of order

corresponding to the order of the actual system. Such a model exactly fits (reproduces) the data with the lowest possible order. The presence of the deadbeat gain in the coefficients of the ARX model suggests that the ARX model is actually the model of the observer. The question about whether such ARX model represents the system or its deadbeat observer is however irrelevant. In the absence of noise and in steady state, both the system and the observer state-space models reproduce the same (measured) output sequence. The fact can be confusing at first. If one looks at the observer model in equation (3.5), it is not intuitive that there exists a gain $\mathbf{K}$ such that (in steady state) the output of the observer is the same as that of the system in equation (1.1). Equations (1.1) and (3.5) just looks different, their system matrix is not the same, neither are their Markov parameters. Nevertheless they are driven by different inputs, and the additional input in equation (3.5), modulated by the deadbeat gain, compensates exactly for the difference in the system matrix. A mathematical argument to convince the skeptical reader is the following. The observer state-space models in equations (3.1) and (3.5) are equivalent (the latter was derived from the former), hence they reproduce the same time history $\hat{\boldsymbol{y}}$. In steady-state and absence of noise, the deadbeat output residuals are identically equal to zero, i.e. $\boldsymbol{\epsilon}(k) = \mathbf{0}$ and the term $\mathbf{K}\boldsymbol{\epsilon}(k)$ in equation (3.1a) vanishes. The state-space model in equation (3.1) then becomes the same as the state-space model in equation (1.1). Hence, in steady-state and absence of noise, the system and deadbeat observer models are the same, regardless of being in state-space or ARX form. In fact the derivation of equation (2.41) in section 2.5, appropriately modified for the noise-free case, is a very elegant and intuitive way to pass from a deterministic system state-space model to the corresponding ARX model.

When process and measurement noises corrupt the data, the distinction between system and optimal observer model becomes more relevant. The steady-state observer minimizing the norm squared of the the state estimation error is the Kalman filter, whose matrix $\mathbf{K}$ makes the ARX

coefficients $\mathbf{\Phi}_j$ smaller and smaller as j is increased, but the convergence is asymptotic. There is no threshold for j after which $\mathbf{C}\bar{\mathbf{A}}^{j-1}$ is identically equal to zero. Hence the ARX model of the Kalman filter derived in equation (2.41) is of infinite order. In practice, one can truncate it to a large value of p. Nevertheless, the order of the actual system is $n \ll p$, and we know there does exist an ARX(n) model equivalent to the system state-space model. We want to find the ARX model of the system but the well established LS estimator, equation (2.46), yields the ARX model of the Kalman filter. And if we force $p = n$ in equation (2.46), we get a biased estimate for the ARX coefficients due to the truncation error $\boldsymbol{\epsilon}_t(k)$ in equation (5.1) being neither negligible nor white. This fact is not surprising from an OKID perspective. The OKID framework says that the presence of noise in the data prevents us from getting direct access to the system model. The corresponding Kalman filter needs to be identified and the matrices of the system are then recovered from the Kalman filter matrices. In summary, the OKID framework also provides the right interpretation to the ARX models that one can identify from input-output data.

The fact that the Kalman filter is the bridge between noisy input-output data and the system to be identified can be extend beyond OKID and ARX model identification. It suggests that independently from the method used or the model chosen to identify a linear system, when the measured data are affected by process and measurement noise, any sort of LS minimization used to identify the model needs to be done on the Kalman filter parameters and not on the system itself. This claim, although supported from the theory presented in this work, needs to be confirmed via numerical simulations. A good test could be on OKID itself as follows. Acknowledging the presence of truncation error in algorithms of OKID/ERA type, References 23, 46 proposed the use of sequential quadratic programming to refine the state-space model identified

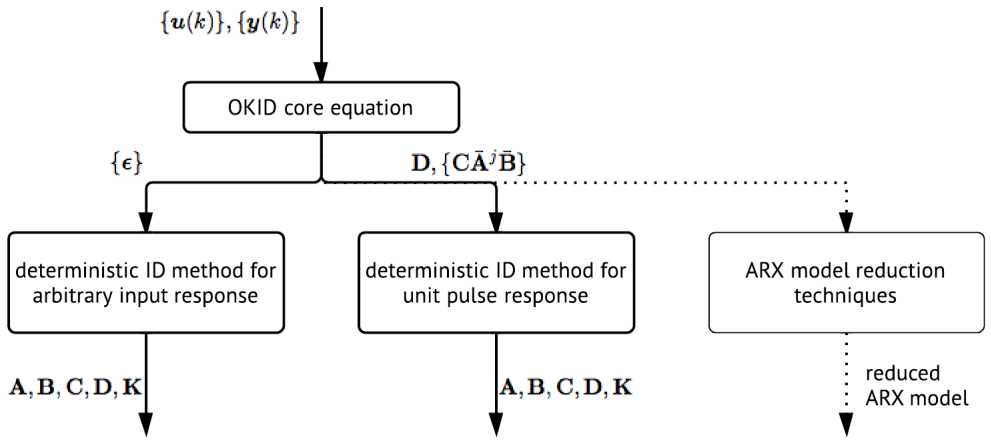


FIGURE 5.6: Link between OKID and methods for ARX model identification.

by OKID/ERA, potentially affected by the truncation error. The refinement consists in minimizing a LS-type objective function for the entries of the matrices of the system state-space model. The problem is obviously nonlinear since the state is also unknown and multiplied by $\mathbf{A}$ and $\mathbf{C}$. For the objective function to be quadratic and hence simple to be minimized, an iterative procedure is adopted. Using as an initial guess the state-space model identified via OKID/ERA with large yet finite p, the objective function is replaced by its quadratic approximation and then minimized. Around the obtained solution, a new quadratic approximation for the objective function can be constructed and another iteration performed. Provided that the initial guess is sufficiently close to the true solution so that the quadratic approximation is valid, the procedure converges to the actual state-space model minimizing the sum of the squares of the error between the measured output and the output predicted by the model. The argument developed above suggests that the correct (unbiased) approach would be to construct the objective function from the Kalman filter state-space model instead of the system state-space model. The study is not addressed in this thesis, it is left as future work.

The link between OKID and ARX model identification leads to another interesting interpretation of OKID algorithms. The first step of OKID identifies a high-order ARX(p) model and the

second step consists in the reduction of such a model to ARX(n), as the identified state-space model (of order n) can be converted to ARX(n) without difficulty. However, in the view of the author, this interpretation does not provide a clear picture. Thinking in terms of noise adding modes (eigensolutions) to the identified model that then need to be eliminated is an abstract and debatable argument. The interpretation of the process via observers provides a rigorous, well defined and intuitive framework to better understand the identification process.

Figure 5.6 schematically shows the link between the OKID framework presented here and ARX model identification methods

5.5 Residual whitening and identification of ARMAX models

In section 5.2 it has been highlighted how the OKID core equation (2.44) is actually approximated, it suffers from a truncation error converging to zero as the parameter p is increased. In Reference 34 an alternative equation was proposed for the purpose of reducing the value of p. Besides requiring more computational effort, increasing p also means fewer equations in the LS problem in equation (2.44), which is detrimental in particular when the available data record is short. In this section we notice another aspect of such an alternative equation, i.e. the fact that it is exact, no truncation is involved. That gives the potential for more accurate system identification.

The derivation proposed below is different from the one in Reference 34 and provides additional insight to OKID. Consider the Kalman filter in innovation form, equation (3.1). We can think of using an observer to estimate its state. Supposing the residual $\boldsymbol{\epsilon}(k)$ is known, the Kalman filter in equation (3.1) is a purely deterministic system. Its deadbeat observer can be written

itself in innovation form as

$$\boldsymbol{\chi}(k+1) = \mathbf{A}\boldsymbol{\chi}(k) + \mathbf{B}\boldsymbol{u}(k) + \mathbf{K}\boldsymbol{\epsilon}(k) + \mathbf{G}\left(\hat{\boldsymbol{y}}(k) - \boldsymbol{\zeta}(k)\right) \tag{5.5a}$$

$$\boldsymbol{\zeta}(k) = \mathbf{C}\boldsymbol{\chi}(k) + \mathbf{D}\boldsymbol{u}(k) \tag{5.5b}$$

where $\boldsymbol{\chi}$ and $\boldsymbol{\zeta}$ are the observer estimates for the Kalman filter state $\hat{\boldsymbol{x}}$ and $\hat{\boldsymbol{y}}$ output of the Kalman filter and $\mathbf{G}$ is the deadbeat gain of the observer. Note that in equation (5.5) we wrote such an observer in innovation form, i.e. adding to the state equation of the system to be observed (the Kalman filter in this case) the output error term multiplied by a gain. By plugging equation (5.5b) into equation (5.5a) and recalling equation (2.8), the state equation (5.5a) can equivalently be written as

$$\boldsymbol{\chi}(k+1) = \bar{\mathbf{A}}_G\boldsymbol{\chi}(k) + \bar{\mathbf{B}}_G\boldsymbol{v}_x(k) + \bar{\mathbf{M}}_G\boldsymbol{\epsilon}(k) \tag{5.6}$$

where

$$\bar{\mathbf{A}}_G = \mathbf{A} - \mathbf{GC} \tag{5.7a}$$

$$\bar{\mathbf{B}}_G = \begin{bmatrix} \mathbf{B} - \mathbf{GD} & \mathbf{G} \end{bmatrix} \tag{5.7b}$$

$$\bar{\mathbf{M}}_G = \mathbf{K} - \mathbf{G} \tag{5.7c}$$

$$\tag{5.7d}$$

Note that $\boldsymbol{v}_x$ is the same as in classic OKID, defined in equation (2.34b); $\bar{\mathbf{A}}_G$ and $\bar{\mathbf{B}}_G$ have exactly the same structure as $\bar{\mathbf{A}}$ and $\bar{\mathbf{B}}$ in equations (3.6) and (3.8a) but the gain is the deadbeat gain, indicated as $\mathbf{G}$ to avoid confusion, instead of the Kalman gain. Propagating (5.6) forward by p

time steps, we obtain

$$\boldsymbol{\chi}(k+1) = \bar{\mathbf{A}}_G^p \boldsymbol{\chi}(k-p) + \mathbf{T}_G \boldsymbol{z}(k) + \mathbf{S}_G \boldsymbol{w}(k) \tag{5.8}$$

where

$$\mathbf{T}_G = \begin{bmatrix} \bar{\mathbf{B}}_G & \bar{\mathbf{A}}_G \bar{\mathbf{B}}_G & \dots & \bar{\mathbf{A}}_G^{p-1} \bar{\mathbf{B}}_G \end{bmatrix} \tag{5.9a}$$

$$\boldsymbol{z}(k) = \begin{bmatrix} \boldsymbol{v}_x(k-1) \\ \boldsymbol{v}_x(k-2) \\ \vdots \\ \boldsymbol{v}_x(k-p) \end{bmatrix} \tag{5.9b}$$

$$\mathbf{S}_G = \begin{bmatrix} \bar{\mathbf{M}}_G & \bar{\mathbf{A}}_G \bar{\mathbf{M}}_G & \dots & \bar{\mathbf{A}}_G^{p-1} \bar{\mathbf{M}}_G \end{bmatrix} \tag{5.9c}$$

$$\boldsymbol{w}(k) = \begin{bmatrix} \boldsymbol{\epsilon}(k-1) \\ \boldsymbol{\epsilon}(k-2) \\ \vdots \\ \boldsymbol{\epsilon}(k-p) \end{bmatrix} \tag{5.9d}$$

Note the similarity between the above matrices and vector and the ones in classic OKID. Assuming that $\mathbf{A}$ and $\mathbf{C}$ form an observable pair, from control theory we know that there indeed exists a deadbeat gain $\mathbf{G}$ such that $\bar{\mathbf{A}}_G^p = \mathbf{0}$ for any $p \geq n$. We can then write

$$\boldsymbol{\chi}(k+1) = \mathbf{T}_G \boldsymbol{z}(k) + \mathbf{S}_G \boldsymbol{w}(k) \tag{5.10}$$

Additionally, after n time steps, we know that the deadbeat observer gets into steady state and the estimation error becomes identically zero. Hence, $\boldsymbol{\chi}(k) = \hat{\boldsymbol{x}}$ and $\boldsymbol{\zeta}(k) = \hat{\boldsymbol{y}}$. Plugging

equation (5.10) into equation (5.5b) and recalling equation (2.8), we obtain

$$\boldsymbol{y}(k) = \bar{\boldsymbol{\Phi}}\boldsymbol{v}(k) + \bar{\boldsymbol{\Psi}}\boldsymbol{w}(k) + \boldsymbol{\epsilon}(k) \tag{5.11}$$

where

$$\bar{\boldsymbol{\Phi}} = \begin{bmatrix} \mathbf{C}\bar{\mathbf{B}}_G & \mathbf{C}\bar{\mathbf{A}}_G\bar{\mathbf{B}}_G & \dots & \mathbf{C}\bar{\mathbf{A}}_G^{p-1}\bar{\mathbf{B}}_G \end{bmatrix} \tag{5.12a}$$

$$\bar{\boldsymbol{\Psi}} = \begin{bmatrix} \mathbf{C}\bar{\mathbf{M}}_G & \mathbf{C}\bar{\mathbf{A}}_G\bar{\mathbf{M}}_G & \dots & \mathbf{C}\bar{\mathbf{A}}_G^{p-1}\bar{\mathbf{M}}_G \end{bmatrix} \tag{5.12b}$$

Equation (5.11) relates the measured input and output of the system without the state appearing explicitly, in a very similar fashion to equation (2.44). However, two differences should be noticed about equation (5.11). First of all, no approximation has been made in its derivation. Secondly, the past residuals appear in it via $\boldsymbol{w}(k)$. They are unknown and multiplied by the unknown coefficients in $\bar{\boldsymbol{\Psi}}$, making the following LS problem nonlinear. Writing equation (5.11) for all $k \geq p$ and putting everything together, we obtain

$$\mathbf{Y} = \bar{\boldsymbol{\Phi}}\mathbf{V} + \bar{\boldsymbol{\Psi}}\mathbf{W} + \mathbf{E} \tag{5.13}$$

where, $\mathbf{Y}$, $\mathbf{V}$ and $\mathbf{E}$ are the same as in equation (2.45) and similarly the matrix $\mathbf{W}$ is defined as

$$\mathbf{W} = \begin{bmatrix} \boldsymbol{w}(p) & \boldsymbol{w}(p+1) & \dots & \boldsymbol{w}(l-1) \end{bmatrix} \tag{5.14}$$

The way we wrote equation (5.13) lends itself to a LS solution, as the error term $\boldsymbol{\epsilon}(k)$ is the Kalman filter output residual and therefore it is zero-mean, white and minimum in a LS sense. The LS solution to equation (5.13) yields residuals that are orthogonal not only to the current input and past input and output (proving they really are the Kalman output residuals), but

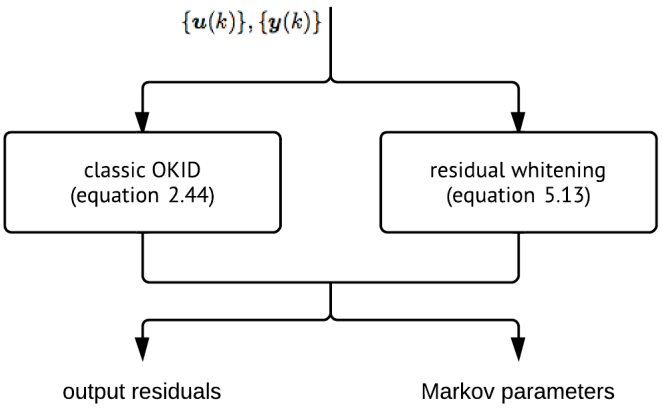


FIGURE 5.7: Alternative OKID core equations.

also to the past residuals. The latter implies that the estimated residuals are explicitly forced by the LS solution to be white, from which the name of the technique. The value of p to make equation (5.13) hold without any approximation just needs to be equal to n or larger. On the other side, since both $\bar{\mathbf{\Psi}}$ and $\mathbf{W}$ are unknown, the LS problem is nonlinear and cannot be solved in closed form as equation (2.44). One way to solve it is as follows. $\mathbf{W}$ is initially unknown, equation (5.13) must be solved iteratively, updating $\mathbf{W}$ with the residuals $\mathbf{E}$ estimated at each iteration by LS. The procedure belongs to the broad class of generalized least squares (GLS, References 34, 47, 48).

In summary, the residual whitening technique can be interpreted as trading the truncation error of the classic OKID equation (due to neglecting $\mathbf{F}^p$, i.e. $\bar{\mathbf{A}}^p$, to get equation (2.40)) with the iteration convergence error of the GLS procedure. As opposed to the former, the latter can be made as small as desired just by running more iterations. We can then think of residual whitening as a technique to improve the estimation of the residuals and make the OKID pre-filtering exact, i.e. yielding a noise-free set of data to be fed to the DI or DP method. The example in section 3.4.2.3 illustrates the concept.

Figure 5.7 shows how equation (5.13) is an alternative to equation (2.44) for the estimation of the Kalman filter output residuals and Markov parameters. The second step of OKID is the same for algorithms based on the output residuals, whereas it needs some minor modifications for algorithms based on Markov parameters (see Reference 34).

5.6 Conclusions

This chapter has provided an overview of OKID in the light of the new developments presented mainly in chapter 3. It is now clear how OKID is indeed a general and unifying approach to linear system identification from noisy measurements. The OKID framework extends the applicability of any deterministic identification method to the case with noisy data and has a clear connection with other techniques for system identification based on time-series models such as ARX and ARMAX. The high-level view and intuitive interpretation of OKID given in this chapter is expected to help the reader navigate the complex world of system identification.

Chapter 6

Deterministic bilinear system identification with arbitrary input

6.1 Introduction

In this chapter we shift the topic from linear to bilinear system identification. As mentioned in section 1.2, bilinear state-space model identification is of interest for two main reasons. Some physical systems are inherently bilinear and bilinear models of high dimension can approximate a broad class of nonlinear systems. Nevertheless, no well-established technique for bilinear system identification is available yet.

Recently a deterministic bilinear model identification method was presented in Reference 33, by extending the interaction matrix formulation that was originally developed for linear state-space models. A linear relationship between the bilinear system state and past input-output data was obtained and used to rewrite the bilinear model as an equivalent linear model (ELM). The ELM can be identified with any linear identification method, from which the original bilinear model is

then recovered. In contrast to other attempts to transform the bilinear model into an ELM, e.g. by expanding the bilinear system state with sinusoidal basis functions by perturbation theory (Reference 49), a benefit of the method in Reference 33 is the freedom in choosing the form of input excitation as long as it is sufficiently rich. The same advantage and the fact that data from a single experiment are sufficient make the approach more appealing than the one adopted by Juang, based on multiple pulses over multiple experiments(References 11, 12).

The content of this chapter was presented at the 23rd AAS/AIAA Spaceflight Mechanics Meeting in Kauai, HI, in 2013 (Reference 50) and published in Nonlinear Dynamics in 2014 (Reference 51). Its first fundamental contribution is a formal proof for the existence of the interaction matrices that were originally postulated in References 33, 52 where they were used to linearly express the causal relationship between the state of a bilinear system at the current time step in terms of past input-output measurements. The second contribution is showing that the bilinear state can also be linearly related to future and current input-output data, again via the interaction matrices. The derivation of the anticausal state representation is less straightforward than the causal version established in Reference 33, but makes it possible to develop an intersection subspace (IS) approach to discrete-time bilinear system identification. The third contribution of this chapter is therefore the IS approach where the bilinear system state history is reconstructed by intersecting two input-output vector spaces. The bilinear state-space model matrices can then be identified by the classic least-squares method. To our knowledge, this is the first application to bilinear systems of an algorithm based on intersection of vector spaces, which has proven to be very effective in linear system identification (References 2, 29, 53). Note that in the proposed IS method it is not even necessary to identify an ELM or explicitly compute the interaction matrices. Additionally, establishing that an anticausal linear relation also exists between the bilinear state and the input-output data gives rise to an anticausal version

of the ELM-based method presented in Reference 33. This new development serves as a useful complement to the original method. Finally, other linear input-output-to-state representations are also developed and their benefits are discussed, resulting in a wide range of algorithms for the identification of discrete-time bilinear systems that require data from a single experiment without specific restrictions on the form of excitation input to be used.

This chapter also introduces concepts such as input-output-to-state-representations (IOSRs), ELM and IS methods and interaction matrices that are at the core of the next chapters, too. In particular, it is worth mentioning how the identification methods presented in this chapter and chapter 7 are derived via interaction matrices. This is indeed the same technique that was originally used to develop OKID for linear systems, although OKID is derived in chapter 2 via the more intuitive concept of observer. Instead of interaction matrices, one could indeed use state observers to derive the proposed bilinear system identification methods. In chapter 8 the link between interaction matrices and observer gains will be rigorously established for bilinear systems as well, proving that the novel methods for bilinear system identification developed in this work are an extension of OKID to bilinear systems.

As a last note, notation in chapters 6-9 is completely redefined to avoid confusion with the notation adopted in the linear case (chapters 2-5). Nevertheless, some overlap remarks the connection between linear and bilinear system identification methods described in this work.

6.2 Problem statement

Consider an n-state, single-input, q-output bilinear system in state-space form

$$\boldsymbol{x}(k+1) = \mathbf{A}\boldsymbol{x}(k) + \mathbf{N}\boldsymbol{x}(k)u(k) + \mathbf{B}u(k) \tag{6.1a}$$

$$\boldsymbol{y}(k) = \boldsymbol{x}(k) + \mathbf{D}u(k) \tag{6.1b}$$

where $\boldsymbol{x} \in \mathbb{R}^{n\times 1}$ is the state vector, $u \in \mathbb{R}$ is the input, $\boldsymbol{y} \in \mathbb{R}^{q\times 1}$ is the output vector and $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{N} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times 1}$, $\mathbf{C} \in \mathbb{R}^{q\times n}$, $\mathbf{D} \in \mathbb{R}^{q\times 1}$ are the matrices governing the dynamics of the system. A single set of length l of input-output data that starts from some unknown initial state $\boldsymbol{x}(0)$ is given

$$\{u(k)\} = \{u(0),\ u(1),\ u(2),\ ...,\ u(l-1)\} \tag{6.2a}$$

$$\{\boldsymbol{y}(k)\} = \{\boldsymbol{y}(0),\ \boldsymbol{y}(1),\ \boldsymbol{y}(2),\ ...,\ \boldsymbol{y}(l-1)\} \tag{6.2b}$$

The objective is to identify the system of Eq. (6.1) with the input-output data provided in Eq. (2.1). The data is assumed to be of sufficient length and richness so that the system of Eq. (6.1) can be correctly identified. For simplicity, we focus on the single-input case. Extension to the multi-input case can be made without conceptual difficulties.

6.3 General approach

Two different approaches to bilinear system identification are adopted in this work. Both of them are based on the following linear relationship between the state $\boldsymbol{x}(k)$ of the bilinear system

and a superstate $\boldsymbol{z}(k)$ made of input-output data only

$$\boldsymbol{x}(k) = \mathbf{T}\boldsymbol{z}(k) \tag{6.3}$$

where $\mathbf{T}$ is a constant matrix. In this and the following chapters, relations of the form of equation (6.3) play a central role and are referred to as *input-output-to-state representations (IOSRs)*. It is worth noting the similarities with equation (2.40), which is indeed an IOSR for linear systems. Depending on the specific choice of IOSR or, equivalently, on the specific definition chosen for the superstate $\boldsymbol{z}(k)$, several identification algorithms of ELM and IS type can be devised.

6.3.1 Equivalent linear model (ELM) method

If a relation like equation (6.3) were available for bilinear systems, the identification problem could be dramatically simplified by rewriting the original bilinear system of equation (6.1) in the form of a linear system. Substituting equation (6.3) into equation (6.1a) we obtain the following *equivalent linear model (ELM)*

$$\boldsymbol{x}(k+1) = \mathbf{A}\boldsymbol{x}(k) + \mathbf{B}_{ELM}\boldsymbol{w}(k) \tag{6.4a}$$

$$\boldsymbol{y}(k) = \mathbf{C}\boldsymbol{x}(k) + \mathbf{D}u(k) \tag{6.4b}$$

where

$$\mathbf{B}_{ELM} = \begin{bmatrix}\mathbf{B} & \mathbf{NT}\end{bmatrix} \tag{6.5}$$

and $\boldsymbol{w}(k)$ is defined as the input to the ELM

$$\boldsymbol{w}(k) = \begin{bmatrix} u(k) \\ \boldsymbol{z}(k)u(k) \end{bmatrix} \tag{6.6}$$

The ELM is a linear state-space model with known input $\{\boldsymbol{w}(k)\}$ and output $\{\boldsymbol{y}(k)\}$ history. Thus it can be identified by any linear identification method and then used to recover the original bilinear system matrices. The ELM approach was first presented in References 33, 49. The latter relied on sinusoidal basis functions to obtain an approximate IOSR. The former introduced the concept of interaction matrices to derive a specific IOSR. In this work, the IOSR in Reference 33 is referred to as *causal IOSR* because the state at the current time step is related to the current superstate which is defined in terms of past input and past output data only.

6.3.2 Intersection subspace (IS) method

If one were able to find two independent IOSRs of the form of equation (6.3), then another approach, based on the direct reconstruction of the bilinear state history, would be possible. Subspace methods have been very successful in linear system identification (see Reference 2 for a detailed presentation) but are very general. In particular, their versions based on the intersection of two vector spaces (Reference 53) can be applied to reconstruct the state of any kind of system for which two relations like equation (6.3) can be obtained. The specialization of the IS method for bilinear systems is a new development. Note how the IS method presented here relies on the same principles as the algorithm referred to as DI in appendix A. IS is indeed the extension of DI to bilinear systems, whose main challenge is the development of IOSRs for bilinear systems, which is addressed in sections 6.4 and 6.5.

6.3.2.1 Superspaces

Assume the following IOSRs are available for the bilinear system of equation (6.1)

$$\boldsymbol{x}(k) = \mathbf{T}_a \boldsymbol{z}_a(k) \qquad \boldsymbol{x}(k) = \mathbf{T}_b \boldsymbol{z}_b(k) \tag{6.7}$$

and define the following matrices

$$\mathbf{X} = \begin{bmatrix} \boldsymbol{x}(k_i) & \boldsymbol{x}(k_i+1) & \boldsymbol{x}(k_i+2) & \dots & \boldsymbol{x}(k_f) \end{bmatrix} \tag{6.8}$$

$$\mathbf{Z}_a = \begin{bmatrix} \boldsymbol{z}_a(k_i) & \boldsymbol{z}_a(k_i+1) & \boldsymbol{z}_a(k_i+2) & \dots & \boldsymbol{z}_a(k_f) \end{bmatrix} \tag{6.9}$$

$$\mathbf{Z}_b = \begin{bmatrix} \boldsymbol{z}_b(k_i) & \boldsymbol{z}_b(k_i+1) & \boldsymbol{z}_b(k_i+2) & \dots & \boldsymbol{z}_b(k_f) \end{bmatrix} \tag{6.10}$$

where k_i and k_f are the initial and final time steps for which equation (6.7) holds. Then we can write

$$\mathbf{X} = \mathbf{T}_a \mathbf{Z}_a \qquad \mathbf{X} = \mathbf{T}_b \mathbf{Z}_b \tag{6.11}$$

The row space of the (minimum-dimension) state $\mathbf{X}$ is a subspace of the row space of $\mathbf{Z}_a$ and also a subspace of the row space of $\mathbf{Z}_b$. The row space of $\mathbf{X}$ must then lie in the intersection between the row spaces of the two superspaces $\mathbf{Z}_a$ and $\mathbf{Z}_b$. The problem of reconstructing the state history $\{\boldsymbol{x}(k)\}$ is therefore reduced to finding the intersection of two vector spaces.

6.3.2.2 Intersection

The intersection of the two vector spaces given by the row spaces of $\mathbf{Z}_a$ and $\mathbf{Z}_b$ is spanned by common row vectors $\boldsymbol{h}_i$. The latter can be expressed as linear combinations of the rows of $\mathbf{Z}_a$ or of the rows of $\mathbf{Z}_b$

$$\boldsymbol{h}_i = \boldsymbol{a}_i^{\mathrm{T}} \mathbf{Z}_a = \boldsymbol{b}_i^{\mathrm{T}} \mathbf{Z}_b \tag{6.12}$$

where $\boldsymbol{a}_i$ and $\boldsymbol{b}_i$ are column vectors with the corresponding linear combination coefficients. Defining

$$\mathbf{R} = \begin{bmatrix} \mathbf{Z}_a^{\mathrm{T}} & \mathbf{Z}_b^{\mathrm{T}} \end{bmatrix} \tag{6.13a}$$

$$\boldsymbol{c}_i = \begin{bmatrix} \boldsymbol{a}_i \\ -\boldsymbol{b}_i \end{bmatrix} \tag{6.13b}$$

we can rewrite equation (6.12) as

$$\mathbf{R}\boldsymbol{c}_i = \mathbf{0} \tag{6.14}$$

which shows that the column vectors $\boldsymbol{c}_i$ lie in the null space of $\mathbf{R}$ and therefore can be conveniently found by Singular Value Decomposition (SVD) as the right singular vectors associated with the zero singular values of $\mathbf{R}$. Note that all the possible pair combinations of basis vectors of the null space of $\mathbf{Z}_a^{\mathrm{T}}$ and of the null space of $\mathbf{Z}_b^{\mathrm{T}}$ satisfy equation (6.14). In general, the null space of $\mathbf{R}$ then has dimension $m \geq n$ and another step is necessary to get a basis for the intersection subspace (i.e. n linearly independent $\boldsymbol{h}_i$ vectors). This can be easily achieved by another SVD as follows. Knowledge of $\boldsymbol{c}_i$'s allows one to compute the corresponding $\boldsymbol{h}_i$'s through equation (6.13b) and either of equalities in equation (6.12). Stacking the m common row vectors $\boldsymbol{h}_i$'s in a matrix $\mathbf{H}$, the n basis vectors $\mathbf{X}_i$'s can be found as the right singular vectors associated with the non-zero singular values of $\mathbf{H}$. A basis for the actual bilinear state space is obtained, which means the state history of the bilinear system is now known.

6.3.2.3 Direct identification

Once the state history is reconstructed, the identification problem is dramatically simplified and can be solved by the classic least-squares method. Defining

$$\mathbf{U} = \begin{bmatrix} u(k_i) & \dots & u(k_f - 1) \end{bmatrix} \tag{6.15a}$$

$$\mathbf{Y} = \begin{bmatrix} \boldsymbol{y}(k_i) & \dots & \boldsymbol{y}(k_f - 1) \end{bmatrix} \tag{6.15b}$$

$$\mathbf{X}_- = \begin{bmatrix} \boldsymbol{x}(k_i) & \dots & \boldsymbol{x}(k_f - 1) \end{bmatrix} \tag{6.15c}$$

$$\mathbf{X}_+ = \begin{bmatrix} \boldsymbol{x}(k_i + 1) & \dots & \boldsymbol{x}(k_f) \end{bmatrix} \tag{6.15d}$$

$$\mathbf{P}_{XU} = \begin{bmatrix} \boldsymbol{x}(k_i)u(k_i) & \dots & \boldsymbol{x}(k_f - 1)u(k_f - 1) \end{bmatrix} \tag{6.15e}$$

from equation (6.1) we can write

$$\mathbf{X}_+ = \begin{bmatrix} \mathbf{A} & \mathbf{B} & \mathbf{N} \end{bmatrix} \begin{bmatrix} \mathbf{X}_- \\ \mathbf{U} \\ \mathbf{P}_{XU} \end{bmatrix} \tag{6.16}$$

$$\mathbf{Y} = \begin{bmatrix} \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{X}_- \\ \mathbf{U} \end{bmatrix} \tag{6.17}$$

and recover **A**, **B**, **N**, **C** and **D** via pseudo-inversion (Moore-Penrose)

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} & \mathbf{N} \end{bmatrix} = \mathbf{X}_+ \begin{bmatrix} \mathbf{X}_- \\ \mathbf{U} \\ \mathbf{P}_{XU} \end{bmatrix}^\dagger \tag{6.18}$$

$$\begin{bmatrix} \mathbf{C} & \mathbf{D} \end{bmatrix} = \mathbf{Y} \begin{bmatrix} \mathbf{X}_- \\ \mathbf{U} \end{bmatrix}^\dagger \tag{6.19}$$

Note that the state reconstructed by intersection is not necessarily in the original coordinate system, and so will be the identified bilinear model matrices **A**, **B**, **N**, **C**, **D**. As usual with state-space formulation, the change in coordinate system does not affect the identified model validity. Also, a sufficient richness condition for the input excitation is that the rank of the state-input matrix to be pseudo-inverted in equation (6.18) is full.

6.3.3 Comparison

The comparison between the ELM and IS methods is more subtle than it appears at a first glance. In this work, we limit ourselves to the following considerations. The core of the IS method is state reconstruction, which then makes it trivial to estimate the dynamic system matrices. The IS method can be applied for state reconstruction directly on the bilinear model of equation (6.1) or on the ELM of equation (6.4); in the latter case, the IS method is used as a purely linear identification algorithm. The application of the IS method to the bilinear model of equation (6.1) is more direct but it requires the existence of at least two IOSRs for the bilinear model.

6.4 Useful concepts

Before plunging into the algebraic derivation of IOSRs, a few concepts useful throughout the rest of this and the following chapters are presented here below.

6.4.1 Interaction matrices

The concept of interaction matrices is at the core of this work. The interaction matrices were originally formulated by Minh Q. Phan in the context of linear system identification of lightly-damped large flexible space structures. The dynamics of such structures can be described by the system Markov parameters. The identification of these Markov parameters can become difficult because of the very large number of system Markov parameters that must be solved for. The interaction matrix provides a mechanism to compress an infinite sequence of system Markov parameters into a finite sequence that can be easily identified from input-output measurements (References 20, 27). The compression can be exact and extremely efficient in the case of noise-free input-output data. The system Markov parameters are then recovered from the compressed Markov parameters, and remarkably, the recovery can be achieved without having to know the interaction matrix required for the compression in the first place. Later development revealed that the interaction matrix in the state-space system identification problem could be interpreted as a Kalman filter gain that is optimal with respect to the system and the (unknown) process and noise statistics embedded in the input-output data. This development led to the OKID method (Reference 20) and to the subsequent generalization of OKID as a unified approach to linear system identification presented in chapters 2-5. Although in system identification the interaction matrices turn out to be observer gains, they are more general and have found applications in other areas as well (e.g./ References 54–58). For a survey, see Reference 59.

The interaction matrices were recently proposed for bilinear system identification in Reference 33. For the sake of clarity, the key steps to derive the IOSR are reported here below (for more details about the derivation, see Reference 33). Consider the bilinear system of equation (6.1) and add and subtract the terms $\mathbf{M}'\boldsymbol{y}(k)$ and $\mathbf{M}''\boldsymbol{y}(k)u(k)$ where $\mathbf{M}'$ and $\mathbf{M}''$ are interaction matrices, getting

$$\begin{aligned}
\boldsymbol{x}(k+1) &= \mathbf{A}\boldsymbol{x}(k) + \mathbf{N}\boldsymbol{x}(k)u(k) + \mathbf{B}u(k) + \mathbf{M}'\boldsymbol{y}(k) \\
&\quad - \mathbf{M}'\boldsymbol{y}(k) + \mathbf{M}''\boldsymbol{y}(k)u(k) - \mathbf{M}''\boldsymbol{y}(k)u(k) \\
&= \mathbf{A}\boldsymbol{x}(k) + \mathbf{N}\boldsymbol{x}(k)u(k) + \mathbf{B}u(k) + \mathbf{M}'\mathbf{C}\boldsymbol{x}(k) + \mathbf{M}'\mathbf{D}u(k) - \mathbf{M}'\boldsymbol{y}(k) \\
&\quad + \mathbf{M}''\mathbf{C}\boldsymbol{x}(k)u(k) + \mathbf{M}''\mathbf{D}u^2(k) - \mathbf{M}''\boldsymbol{y}(k)u(k) \\
&= \left(\mathbf{A} + \mathbf{M}'\mathbf{C}\right)\boldsymbol{x}(k) + \left(\mathbf{N} + \mathbf{M}''\mathbf{C}\right)\boldsymbol{x}(k)u(k) + \left(\mathbf{B} + \mathbf{M}'\mathbf{D}\right)u(k) \\
&\quad - \mathbf{M}'\boldsymbol{y}(k) + \mathbf{M}''\mathbf{D}u^2(k) - \mathbf{M}''\boldsymbol{y}(k)u(k)
\end{aligned} \tag{6.20}$$

which can be rewritten as

$$\boldsymbol{x}(k+1) = \bar{\mathbf{A}}\boldsymbol{x}(k) + \bar{\mathbf{N}}\boldsymbol{x}(k)u(k) + \bar{\mathbf{B}}\boldsymbol{v}(k) \tag{6.21}$$

where

$$\bar{\mathbf{A}} = \mathbf{A} + \mathbf{M}'\mathbf{C} \tag{6.22a}$$

$$\bar{\mathbf{N}} = \mathbf{N} + \mathbf{M}''\mathbf{C} \tag{6.22b}$$

$$\bar{\mathbf{B}} = \begin{bmatrix} \mathbf{B} + \mathbf{M}'\mathbf{D} & -\mathbf{M}' & \mathbf{M}''\mathbf{D} & -\mathbf{M}'' \end{bmatrix} \tag{6.22c}$$

$$\boldsymbol{v}(k) = \begin{bmatrix} u(k) \\ \boldsymbol{y}(k) \\ u^2(k) \\ \boldsymbol{y}(k)u(k) \end{bmatrix} \tag{6.22d}$$

The interaction matrices convert the bilinear model $\mathbf{A}$, $\mathbf{N}$, $\mathbf{B}$ of equation (6.1a) into an equivalent bilinear model $\bar{\mathbf{A}}$, $\bar{\mathbf{N}}$, $\bar{\mathbf{B}}$, equation (6.21). The observation equation, equation (6.1b), does not change. The freedom introduced by $\mathbf{M}'$ and $\mathbf{M}''$ will be used to impose conditions to express the state at the current time step k solely in terms of past input and output data. In the following, for brevity of presentation, we will terminate the propagation of equation (6.21) at time step $k+2$, but it is possible to generalize the termination at any time step $k+p$. Propagating equation (6.21) one step forward, we get

$$\begin{aligned} \boldsymbol{x}(k+2) &= \bar{\mathbf{A}}\boldsymbol{x}(k+1) + \bar{\mathbf{N}}\boldsymbol{x}(k+1)u(k+1) + \bar{\mathbf{B}}\boldsymbol{v}(k+1) \\ &= \bar{\mathbf{A}}^2\boldsymbol{x}(k) + \bar{\mathbf{A}}\bar{\mathbf{N}}\boldsymbol{x}(k)u(k) + \bar{\mathbf{A}}\bar{\mathbf{B}}\boldsymbol{v}(k) + \bar{\mathbf{N}}\bar{\mathbf{A}}\boldsymbol{x}(k)u(k+1) \\ &\quad + \bar{\mathbf{N}}^2\boldsymbol{x}(k)u(k)u(k+1) + \bar{\mathbf{N}}\bar{\mathbf{B}}\boldsymbol{v}(k)u(k+1) + \bar{\mathbf{B}}\boldsymbol{v}(k+1) \end{aligned} \tag{6.23}$$

As will be proven later, suppose the sum of all the terms in equation (6.23) depending on the state $\boldsymbol{x}(k)$ vanishes, i.e.

$$\bar{\mathbf{A}}^2 + \bar{\mathbf{A}}\bar{\mathbf{N}}u(k) + \bar{\mathbf{N}}\bar{\mathbf{A}}u(k+1) + \bar{\mathbf{N}}^2u(k)u(k+1) = \mathbf{0} \tag{6.24}$$

then equation (6.23) becomes

$$\boldsymbol{x}(k+2) = \bar{\mathbf{A}}\bar{\mathbf{B}}\boldsymbol{v}(k) + \bar{\mathbf{N}}\bar{\mathbf{B}}\boldsymbol{v}(k)u(k+1) + \bar{\mathbf{B}}\boldsymbol{v}(k+1) \tag{6.25}$$

and can be rewritten in the form of an IOSR as

$$\boldsymbol{x}(k+2) = \mathbf{T}_2\boldsymbol{z}_2(k+2) \tag{6.26}$$

where

$$\mathbf{T}_2 = \begin{bmatrix}\bar{\mathbf{A}}\bar{\mathbf{B}} & \bar{\mathbf{N}}\bar{\mathbf{B}} & \bar{\mathbf{B}}\end{bmatrix} \tag{6.27a}$$

$$\boldsymbol{z}_2(k+2) = \begin{bmatrix}\boldsymbol{v}(k) \\ \boldsymbol{v}(k)u(k+1) \\ \boldsymbol{v}(k+1)\end{bmatrix} \tag{6.27b}$$

and the subscript 2 remarks that the state depends on input-output data 2 steps back in time. Equation (6.26), or equivalently equation (6.25), expresses the state of the bilinear model of equation (6.1) in terms of previous input-output data only. Shifting the time index backward by 2 steps, equation (6.26) takes the same form as equation (6.3).

As previously mentioned, equation (6.23) can be propagated further in time, leading to a higher-order ($p > 2$) IOSR of the form

$$\boldsymbol{x}(k) = \mathbf{T}_p \boldsymbol{z}_p(k) \tag{6.28}$$

In Reference 33, the general pattern to construct the superstate $\boldsymbol{z}_p(k)$ for the above IOSR from input-output data is derived. Generalizing equation (6.24), the condition to be satisfied to get to equation (6.28) is that the sum of all the terms depending on $\boldsymbol{x}(k)$ in the propagated equation vanishes. This condition will be symbolically indicated as

$$\boldsymbol{\mathcal{S}}_p\left(\bar{\mathbf{A}}, \bar{\mathbf{N}}, \{u(k) \ \ldots \ u(k+p-1)\}\right) = \mathbf{0} \tag{6.29}$$

and for $p = 2$ it takes the explicit form of equation (6.24).

6.4.2 Exact input-output-to-state representations and ideal bilinear systems

The condition for equation (6.24) to be identically satisfied for any input history $\{u(k)\}$ is

$$\bar{\mathbf{A}}^2 = \bar{\mathbf{A}}\bar{\mathbf{N}} = \bar{\mathbf{N}}\bar{\mathbf{A}} = \bar{\mathbf{N}}^2 = \mathbf{0} \tag{6.30}$$

To identically satisfy the generalized condition of equation (6.29), it is necessary that all the possible products of order p of matrices $\bar{\mathbf{A}}$ and $\bar{\mathbf{N}}$ are zero, which in this chapter is more compactly indicated as

$$\boldsymbol{\mathcal{C}}_p\left(\bar{\mathbf{A}}, \bar{\mathbf{N}}\right) = \mathbf{0} \tag{6.31}$$

In References 33, 52 the existence of interaction matrices $\mathbf{M}'$ and $\mathbf{M}''$ such that equation (6.31) is satisfied was only postulated. It is indeed a matter of fact that there exist bilinear systems for which equation (6.31) holds for some p. In this work, they are referred to as *ideal bilinear*

systems and a couple of examples are given in section 6.7. For an ideal bilinear system it is thus possible to get an exact IOSR and achieve (apart from noise) exact identification via the ELM or the IS methods. It is worth remarking that the concept of ideal bilinear system is used only for the purpose of theoretical considerations, and the techniques developed in this work apply to arbitrary bilinear systems of the form of equation (6.1), as proven by the next theorem and shown in the final examples.

6.4.3 Convergence of non-ideal bilinear systems

The following theorem provides rigorous justification for the application to *any* bilinear system of ELM and IS identification methods based on IOSRs derived via interaction matrices.

Theorem 6.1. *Given any bilinear system of the form of equation (6.1), if* $(\mathbf{A}, \mathbf{C})$ *is a detectable pair then there exist interaction matrices* $\mathbf{M}'$ *and* $\mathbf{M}''$ *and a value* γ *such that, for* $|u(k)| < \gamma$ *for all* k*, equation (6.29) is asymptotically satisfied as* p *increases.*

In particular, the theorem ensures that, even though equation (6.31) is not identically satisfied, the resulting approximation error in the IOSR converges to zero as the order p is increased, provided the input is kept within a certain threshold. The proof has its roots in the similarity between the present problem and the work of Bouazza et al. (Reference 60), who demonstrated the convergence of a Luenberger-like observer for a class of discrete-time nonlinear systems which can be shown to include bilinear systems of the form of equation (6.1).

Proof. Consider equation (6.1a). Introducing interaction matrices $\mathbf{M}'$ and $\mathbf{M}''$, we were able to write equation (6.21), which was then propagated forward by one time step to obtain $\boldsymbol{x}(k+2)$ in equation (6.23). Note that the terms containing the state $\boldsymbol{x}(k)$ in equation (6.23), which were eliminated via the condition of equation (6.24), do not involve the input $\boldsymbol{v}(k)$. Also, note

that the terms depending on $\boldsymbol{v}(k)$ in equation (6.23) do not contain the state $\boldsymbol{x}(k)$. These observations can be generalized to propagation in time up to any step $k+p$. Since the objective is to prove convergence to zero of the sum of the terms in equation (6.24) and in the more general equation (6.29), it is sufficient to study the convergence to the origin of the following dynamic system

$$\hat{\boldsymbol{x}}(k+1) = \bar{\mathbf{A}}\hat{\boldsymbol{x}}(k) + \bar{\mathbf{N}}\hat{\boldsymbol{x}}(k)u(k) \tag{6.32}$$

as explained in what follows. Propagating equation (6.32) forward, we obtain

$$\hat{\boldsymbol{x}}(k+p) = \boldsymbol{\mathcal{S}}_p\left(\bar{\mathbf{A}}, \bar{\mathbf{N}}, \{u(k) \ \ldots \ u(k+p-1)\}\right)\hat{\boldsymbol{x}}(k) \tag{6.33}$$

If we prove that, for any arbitrary $\hat{\boldsymbol{x}}(k)$, $\hat{\boldsymbol{x}}(k+p)$ converges to zero as p increases, then convergence of $\boldsymbol{\mathcal{S}}_p$ in equation (6.33) to zero will follow.

Recall the definition of $\bar{\mathbf{A}}$, equation (6.22), and assume $\mathbf{A}$ and $\mathbf{C}$ form a detectable pair. Then there exists a positive definite matrix $\mathbf{S}$ such that

$$\bar{\mathbf{A}}^{\mathrm{T}}\mathbf{S}\bar{\mathbf{A}} + \mathbf{I} = \mathbf{S} \tag{6.34}$$

Define the weighted norm

$$\hat{X}(k) = \hat{\boldsymbol{x}}^{\mathrm{T}}(k)\mathbf{S}\hat{\boldsymbol{x}} \tag{6.35}$$

which will be the measure of convergence to zero of $\hat{\boldsymbol{x}}(k)$. Even more conveniently, define the increment of the above weighted norm

$$\Delta\hat{X}(k) = \hat{X}(k+1) - \hat{X}(k) \tag{6.36}$$

Plugging equation (6.32) into the first term of equation (6.36) and equation (6.34) into the second term we obtain

$$\begin{aligned}
\Delta\hat{X}(k) &= \hat{\boldsymbol{x}}^{\mathrm{T}}(k+1)\mathbf{S}\hat{\boldsymbol{x}}(k+1) - \hat{\boldsymbol{x}}^{\mathrm{T}}(k)\mathbf{S}\hat{\boldsymbol{x}} \\
&= \left(\bar{\mathbf{A}}\hat{\boldsymbol{x}}(k) + \bar{\mathbf{N}}\hat{\boldsymbol{x}}(k)u(k)\right)^{\mathrm{T}}\mathbf{S}\left(\bar{\mathbf{A}}\hat{\boldsymbol{x}}(k) + \bar{\mathbf{N}}\hat{\boldsymbol{x}}(k)u(k)\right) - \hat{\boldsymbol{x}}^{\mathrm{T}}(k)\left(\bar{\mathbf{A}}^{\mathrm{T}}\mathbf{S}\bar{\mathbf{A}} + \mathbf{I}\right)\hat{\boldsymbol{x}}(k) \\
&= u^2(k)\hat{\boldsymbol{x}}^{\mathrm{T}}(k)\bar{\mathbf{N}}^{\mathrm{T}}\mathbf{S}\bar{\mathbf{N}}\hat{\boldsymbol{x}}(k) + 2u(k)\hat{\boldsymbol{x}}^{\mathrm{T}}(k)\bar{\mathbf{A}}^{\mathrm{T}}\mathbf{S}\bar{\mathbf{N}}\hat{\boldsymbol{x}}(k) - \hat{\boldsymbol{x}}^{\mathrm{T}}(k)\hat{\boldsymbol{x}}(k) \\
&= -||\hat{\boldsymbol{x}}(k)||^2\Bigg(\frac{-1}{||\hat{\boldsymbol{x}}(k)||^2}u^2(k)\hat{\boldsymbol{x}}^{\mathrm{T}}(k)\bar{\mathbf{N}}^{\mathrm{T}}\mathbf{S}\bar{\mathbf{N}}\hat{\boldsymbol{x}}(k) \\
&\quad - \frac{1}{||\hat{\boldsymbol{x}}(k)||^2}2u(k)\hat{\boldsymbol{x}}^{\mathrm{T}}(k)\bar{\mathbf{A}}^{\mathrm{T}}\mathbf{S}\bar{\mathbf{N}}\hat{\boldsymbol{x}}(k) + 1\Bigg)
\end{aligned} \tag{6.37}$$

from which it is clear that $\hat{X}(k)$ will converge to zero if the term in parentheses in equation (6.37) is positive, i.e. if

$$\frac{1}{||\hat{\boldsymbol{x}}(k)||^2}u^2(k)\hat{\boldsymbol{x}}(k)^{\mathrm{T}}\bar{\mathbf{N}}^{\mathrm{T}}\mathbf{S}\bar{\mathbf{N}}\hat{\boldsymbol{x}}(k) + \frac{1}{||\hat{\boldsymbol{x}}(k)||^2}2u(k)\hat{\boldsymbol{x}}(k)^{\mathrm{T}}\bar{\mathbf{A}}^{\mathrm{T}}\mathbf{S}\bar{\mathbf{N}}\hat{\boldsymbol{x}}(k) < 1 \tag{6.38}$$

By basic matrix norm properties, the following inequalities follow

$$\begin{aligned}
u^2(k)\hat{\boldsymbol{x}}(k)^{\mathrm{T}}\bar{\mathbf{N}}^{\mathrm{T}}\mathbf{S}\bar{\mathbf{N}}\hat{\boldsymbol{x}}(k) &\leq ||u^2(k)\hat{\boldsymbol{x}}(k)^{\mathrm{T}}\bar{\mathbf{N}}^{\mathrm{T}}\mathbf{S}\bar{\mathbf{N}}\hat{\boldsymbol{x}}(k)|| \\
&\leq u^2(k)||\hat{\boldsymbol{x}}(k)||^2||\bar{\mathbf{N}}||^2||\mathbf{S}||
\end{aligned} \tag{6.39}$$

$$\begin{aligned}
u(k)\hat{\boldsymbol{x}}(k)^{\mathrm{T}}\bar{\mathbf{A}}^{\mathrm{T}}\mathbf{S}\bar{\mathbf{N}}\hat{\boldsymbol{x}}(k) &\leq ||u(k)\hat{\boldsymbol{x}}(k)^{\mathrm{T}}\bar{\mathbf{A}}^{\mathrm{T}}\mathbf{S}\bar{\mathbf{N}}\hat{\boldsymbol{x}}(k)|| \\
&\leq |u(k)|\ ||\hat{\boldsymbol{x}}(k)||^2||\bar{\mathbf{A}}||\ ||\bar{\mathbf{N}}||\ ||\mathbf{S}||
\end{aligned} \tag{6.40}$$

and allow us to rewrite the convergence condition of equation (6.38) as

$$u^2(k)\ ||\bar{\mathbf{N}}||^2||\mathbf{S}|| + 2|u(k)|\ ||\bar{\mathbf{A}}||\ ||\bar{\mathbf{N}}||\ ||\mathbf{S}|| < 1 \tag{6.41}$$

which is always possible to satisfy by choosing $|u(k)| < \gamma$ for all k, with sufficiently small γ. By equation (6.35), convergence of $\hat{X}(k)$ implies convergence of $\hat{\boldsymbol{x}}(k)$, as desired. □

The above theorem is a fundamental finding that provides theoretical justification of the overall approach to bilinear system identification proposed in this chapter, and in particular of the application of the concept of interaction matrices to bilinear systems. In the rest of the work, we will sometimes refer to *exact IOSRs* and *approximate IOSRs* to remark the fact that not all bilinear systems satisfy equation (6.29) identically, but increasing the IOSR order p ensures that the approximation error converges to zero. It is worth remarking that the requirement of bounded input concerns the excitation to get input-output data for identification. Once the bilinear model is identified, it can be used to predict the output of the real bilinear system for inputs higher than the above mentioned bound.

6.5 Input-output-to-state representations

To implement the IS identification method it is necessary to have two independent Input-Output-to-State Representations (IOSRs). Another IOSR is necessary, other than the one derived above, equation (6.28). Additionally, even when applying the ELM method, some IOSRs can be more favorable than others for certain bilinear systems. Thus it is of interest to find other IOSRs, i.e. other linear relations between the bilinear system state and a superstate made of input-output data only.

6.5.1 Causal IOSR

A causal IOSR is here defined as a representation of the form of equation (6.3) where the state depends on past (and current, at most) input-output data only, i.e. $\boldsymbol{x}(k)$ depends on $u(i)$ and

$\boldsymbol{y}(i)$ with $i \leq k$. The example of IOSR derived above is causal. Its key equations are rewritten as follows introducing the subscript c where needed to remark the *causality* of the representation. Equation (6.21) becomes

$$\boldsymbol{x}(k+1) = \bar{\mathbf{A}}_c \boldsymbol{x}(k) + \bar{\mathbf{N}}_c \boldsymbol{x}(k) u(k) + \bar{\mathbf{B}}_c \boldsymbol{v}(k) \tag{6.42}$$

where

$$\bar{\mathbf{A}}_c = \mathbf{A} + \mathbf{M}_c' \mathbf{C} \tag{6.43a}$$

$$\bar{\mathbf{N}}_c = \mathbf{N} + \mathbf{M}_c'' \mathbf{C} \tag{6.43b}$$

$$\bar{\mathbf{B}}_c = \begin{bmatrix} \mathbf{B} + \mathbf{M}_c' \mathbf{D} & -\mathbf{M}_c' & \mathbf{M}_c'' \mathbf{D} & -\mathbf{M}_c'' \end{bmatrix} \tag{6.43c}$$

$$\boldsymbol{v}_c(k) = \begin{bmatrix} u(k) \\ \boldsymbol{y}(k) \\ u^2(k) \\ \boldsymbol{y}(k) u(k) \end{bmatrix} \tag{6.43d}$$

and the causal IOSR of equation (6.28) is denoted by

$$\boldsymbol{x}(k) = \mathbf{T}_{p,c} \boldsymbol{z}_{p,c}(k) \tag{6.44}$$

where the superstate $\boldsymbol{z}_{p,c}(k)$ depends on input-output data at steps $k-1$, $k-2$, ... , $k-p$ only. Letting $nCk = \binom{n}{k}$ denote the combinations of k out of n terms, commonly referred to as n-choose-k, the general pattern for the entries of the column vector $\boldsymbol{z}_{p,c}(k+p)$ is (Reference 33):

- $\boldsymbol{v}_c(k)$, $\boldsymbol{v}_c(k+1)$, ... , $\boldsymbol{v}_c(k+p-1)$

- $\boldsymbol{v}_c(k)$ multiplied with products of $u(k+1)$ to $u(k+p-1)$ in all possible combinations $(p-1)C1$, $(p-1)C2$, ... , $(p-1)C(p-1)$ of $\{u(k+1), u(k+2), \ldots, u(k+p-1)\}$
- $\boldsymbol{v}_c(k+1)$ multiplied with products of $u(k+2)$ to $u(k+p-1)$ in all possible combinations $(p-2)C1$, $(p-2)C2$, ... , $(p-2)C(p-2)$ of $\{u(k+2), u(k+3), \ldots, u(k+p-1)\}$

  $\vdots$
- $\boldsymbol{v}_c(k+p-3)$ multiplied with products of $u(k+p-2)$ and $u(k+p-1)$ in all possible combinations $2C1$, $2C2$ of $\{u(k+3), u(k+4), \ldots, u(k+p-1)\}$
- $\boldsymbol{v}_c(k+p-2)$ multiplied with $1C1$ of $u(k+p-1)$, which of course is $u(k+p-1)$

To obtain an expression for $\boldsymbol{z}_{p,c}(k)$, we simply shift the time indices of $\boldsymbol{z}_{p,c}(k+p)$ backwards by p time steps.

6.5.2 Anticausal IOSR

An anticausal IOSR is here defined as a representation of the form of equation (6.3) where the state depends on future (and current, at most) input-output data only, i.e. $\boldsymbol{x}(k)$ depends on $u(i)$ and $\boldsymbol{y}(i)$ with $i \geq k$.

Rewrite equation (6.1) as

$$\boldsymbol{x}(k) = \mathbf{A}^{-1}\boldsymbol{x}(k+1) - \mathbf{A}^{-1}\mathbf{N}\boldsymbol{x}(k)u(k) - \mathbf{A}^{-1}\mathbf{B}u(k) \tag{6.45a}$$

$$\boldsymbol{y}(k) = \mathbf{C}\boldsymbol{x}(k) + \mathbf{D}u(k) \tag{6.45b}$$

Let $\mathbf{M}'_a$ and $\mathbf{M}''_a$ be *another* pair of interaction matrices, where the subscript a stands for *anticausal*, and add and subtract the terms $\mathbf{M}'_a\boldsymbol{y}(k+1)$ and $\mathbf{M}''_a\boldsymbol{y}(k)u(k)$ in (6.45a), getting

$$\begin{aligned}
\boldsymbol{x}(k) &= \mathbf{A}^{-1}\boldsymbol{x}(k+1) - \mathbf{A}^{-1}\mathbf{N}\boldsymbol{x}(k)u(k) - \mathbf{A}^{-1}\mathbf{B}u(k) + \mathbf{M}'_a\boldsymbol{y}(k+1) \\
&\quad - \mathbf{M}'_a\boldsymbol{y}(k+1) + \mathbf{M}''_a\boldsymbol{y}(k)u(k) - \mathbf{M}''_a\boldsymbol{y}(k)u(k) \\
&= \mathbf{A}^{-1}\boldsymbol{x}(k+1) - \mathbf{A}^{-1}\mathbf{N}\boldsymbol{x}(k)u(k) - \mathbf{A}^{-1}\mathbf{B}u(k) + \mathbf{M}'_a\mathbf{C}\boldsymbol{x}(k+1) \\
&\quad + \mathbf{M}'_a\mathbf{D}u(k+1) - \mathbf{M}'_a\boldsymbol{y}(k+1) + \mathbf{M}''_a\mathbf{C}\boldsymbol{x}(k)u(k) + \mathbf{M}''_a\mathbf{D}u^2(k) - \mathbf{M}''_a\boldsymbol{y}(k)u(k) \\
&= \left(\mathbf{A}^{-1} + \mathbf{M}'_a\mathbf{C}\right)\boldsymbol{x}(k+1) - \left(\mathbf{A}^{-1}\mathbf{N} - \mathbf{M}''_a\mathbf{C}\right)\boldsymbol{x}(k)u(k) - \mathbf{A}^{-1}\mathbf{B}u(k) \\
&\quad + \mathbf{M}'_a\mathbf{D}u(k+1) - \mathbf{M}'_a\boldsymbol{y}(k+1) + \mathbf{M}''_a\mathbf{D}u^2(k) - \mathbf{M}''_a\boldsymbol{y}(k)u(k)
\end{aligned} \tag{6.46}$$

which can be rewritten as

$$\boldsymbol{x}(k) = \bar{\mathbf{A}}_a\boldsymbol{x}(k+1) + \bar{\mathbf{N}}_a\boldsymbol{x}(k)u(k) + \bar{\mathbf{B}}_a\boldsymbol{v}_a(k+1) \tag{6.47}$$

where

$$\bar{\mathbf{A}}_a = \mathbf{A}^{-1} + \mathbf{M}'_a\mathbf{C} \tag{6.48a}$$

$$\bar{\mathbf{N}}_a = -\mathbf{A}^{-1}\mathbf{N} + \mathbf{M}''_a\mathbf{C} \tag{6.48b}$$

$$\bar{\mathbf{B}}_a = -\begin{bmatrix}\mathbf{A}^{-1}\mathbf{B} & -\mathbf{M}'_a\mathbf{D} & \mathbf{M}'_a & -\mathbf{M}''_a\mathbf{D} & \mathbf{M}''_a\end{bmatrix} \tag{6.48c}$$

$$\boldsymbol{v}_a(k+1) = \begin{bmatrix} u(k) \\ u(k+1) \\ \boldsymbol{y}(k+1) \\ u^2(k) \\ \boldsymbol{y}(k)u(k) \end{bmatrix} \tag{6.48d}$$

Rewriting equation (6.47) for the next time step

$$\boldsymbol{x}(k+1) = \bar{\mathbf{A}}_a \boldsymbol{x}(k+2) + \bar{\mathbf{N}}_a \boldsymbol{x}(k+1)u(k+1) + \bar{\mathbf{B}}_a \boldsymbol{v}_a(k+2) \tag{6.49}$$

and plugging equation (6.47) and equation (6.49) into equation (6.47) itself, we get

$$\begin{aligned}
\boldsymbol{x}(k) &= \bar{\mathbf{A}}_a\big(\bar{\mathbf{A}}_a \boldsymbol{x}(k+2) + \bar{\mathbf{N}}_a \boldsymbol{x}(k+1)u(k+1) + \bar{\mathbf{B}}_a \boldsymbol{v}_a(k+2)\big) \\
&\quad + \bar{\mathbf{N}}_a\big(\bar{\mathbf{A}}_a \boldsymbol{x}(k+1) + \bar{\mathbf{N}}_a \boldsymbol{x}(k)u(k) + \bar{\mathbf{B}}_a \boldsymbol{v}_a(k+1)\big)u(k) + \bar{\mathbf{B}}_a \boldsymbol{v}_a(k+1) \\
&= \bar{\mathbf{A}}_a^2 \boldsymbol{x}(k+2) + \bar{\mathbf{A}}_a \bar{\mathbf{N}}_a \boldsymbol{x}(k+1)u(k+1) + \bar{\mathbf{A}}_a \bar{\mathbf{B}}_a \boldsymbol{v}_a(k+2) + \bar{\mathbf{N}}_a \bar{\mathbf{A}}_a \boldsymbol{x}(k+1)u(k) \\
&\quad + \bar{\mathbf{N}}_a^2 \boldsymbol{x}(k)u(k)^2 + \bar{\mathbf{N}}_a \bar{\mathbf{B}}_a \boldsymbol{v}_a(k+1)u(k) + \bar{\mathbf{B}}_a \boldsymbol{v}_a(k+1)
\end{aligned} \tag{6.50}$$

Assuming, similarly to equation (6.30),

$$\bar{\mathbf{A}}_a^2 = \bar{\mathbf{A}}_a \bar{\mathbf{N}}_a = \bar{\mathbf{N}}_a \bar{\mathbf{A}}_a = \bar{\mathbf{N}}_a^2 = \mathbf{0} \tag{6.51}$$

we get from equation (6.50)

$$\boldsymbol{x}(k) = \bar{\mathbf{A}}_a \bar{\mathbf{B}}_a \boldsymbol{v}_a(k+2) + \bar{\mathbf{N}}_a \bar{\mathbf{B}}_a \boldsymbol{v}_a(k+1)u(k) + \bar{\mathbf{B}}_a \boldsymbol{v}_a(k+1) \tag{6.52}$$

which we can rewrite as

$$\boldsymbol{x}(k) = \mathbf{T}_{2,a} \boldsymbol{z}_{2,a}(k) \tag{6.53}$$

where

$$\mathbf{T}_{2,a} = \begin{bmatrix} \bar{\mathbf{A}}_a\bar{\mathbf{B}}_a & \bar{\mathbf{N}}_a\bar{\mathbf{B}}_a & \bar{\mathbf{B}}_a \end{bmatrix} \tag{6.54a}$$

$$\boldsymbol{z}_{2,a}(k) = \begin{bmatrix} \boldsymbol{v}_a(k+2) \\ \boldsymbol{v}_a(k+1)u(k) \\ \boldsymbol{v}_a(k+1) \end{bmatrix} \tag{6.54b}$$

Equation (6.53), or equivalently equation (6.52), expresses the state of the bilinear model in terms of current and future input-output data only, and is an anticausal IOSR specialized for $p = 2$.

The rationale in the above derivation is to find an expression for the state written in terms of input-output data only, via an assumption similar to the one made for the causal representation, i.e. eliminating the dependance of $\boldsymbol{x}(k)$ on the states at any other time steps. The aim is the same as in the causal representation, however the derivation is less straightforward. To better show the approach to find higher-order anticausal IOSRs, the derivation for $p = 3$ is presented as well. After rewriting equation (6.49) for the next time step

$$\boldsymbol{x}(k+2) = \bar{\mathbf{A}}_a\boldsymbol{x}(k+3) + \bar{\mathbf{N}}_a\boldsymbol{x}(k+2)u(k+2) + \bar{\mathbf{B}}_a\boldsymbol{v}_a(k+3) \tag{6.55}$$

plug equations (6.47, 6.49, 6.55) into equation (6.50) to get an expression for $\boldsymbol{x}(k)$ whose (unknown) state-dependent terms all contain 3^{rd}-order products of matrices $\bar{\mathbf{A}}_a$ and $\bar{\mathbf{N}}_a$

$$\begin{aligned}\boldsymbol{x}(k) =&\bar{\mathbf{A}}_a^3\boldsymbol{x}(k+3) + \bar{\mathbf{A}}_a^2\bar{\mathbf{N}}_a\boldsymbol{x}(k+2)u(k+2) + \bar{\mathbf{A}}_a\bar{\mathbf{N}}_a\bar{\mathbf{A}}_a\boldsymbol{x}(k+2)u(k+1)\\ &+ \bar{\mathbf{N}}_a\bar{\mathbf{A}}_a^2\boldsymbol{x}(k+2)u(k) + \bar{\mathbf{A}}_a\bar{\mathbf{N}}_a^2\boldsymbol{x}(k+1)u^2(k+1)\\ &+ \bar{\mathbf{N}}_a\bar{\mathbf{A}}_a\bar{\mathbf{N}}_a\boldsymbol{x}(k+1)u(k+1)u(k) + \bar{\mathbf{N}}_a^2\bar{\mathbf{A}}_a\boldsymbol{x}(k+1)u^2(k) + \bar{\mathbf{N}}_a^3\boldsymbol{x}(k)u^3(k)\\ &+ \bar{\mathbf{A}}_a^2\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+3) + \bar{\mathbf{A}}_a\bar{\mathbf{N}}_a\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+2)u(k+1) + \bar{\mathbf{N}}_a\bar{\mathbf{A}}_a\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+2)u(k)\\ &+ \bar{\mathbf{A}}_a\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+2) + \bar{\mathbf{N}}_a^2\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+1)u^2(k) + \bar{\mathbf{N}}_a\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+1)u(k) + \bar{\mathbf{B}}_a\boldsymbol{v}_a(k+1)\end{aligned} \tag{6.56}$$

Assuming that the terms with matrix products given by all the possible combinations of $\bar{\mathbf{A}}$ and $\bar{\mathbf{N}}$ of 3^{rd} order vanish, we get

$$\begin{aligned}\boldsymbol{x}(k) =&\bar{\mathbf{A}}_a^2\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+3) + \bar{\mathbf{A}}_a\bar{\mathbf{N}}_a\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+2)u(k+1) + \bar{\mathbf{N}}_a\bar{\mathbf{A}}_a\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+2)u(k)\\ &+ \bar{\mathbf{A}}_a\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+2) + \bar{\mathbf{N}}_a^2\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+1)u^2(k) + \bar{\mathbf{N}}_a\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+1)u(k) + \bar{\mathbf{B}}_a\boldsymbol{v}_a(k+1)\end{aligned} \tag{6.57}$$

which can be rewritten as

$$\boldsymbol{x}(k) = \mathbf{T}_{3,a}\boldsymbol{z}_{3,a}(k) \tag{6.58}$$

where

$$\mathbf{T}_{3,a} = \begin{bmatrix}\bar{\mathbf{A}}_a^2\bar{\mathbf{B}}_a & \bar{\mathbf{A}}_a\bar{\mathbf{N}}_a\bar{\mathbf{B}}_a & \bar{\mathbf{N}}_a\bar{\mathbf{A}}_a\bar{\mathbf{B}}_a & \bar{\mathbf{N}}_a^2\bar{\mathbf{B}}_a & \bar{\mathbf{A}}_a\bar{\mathbf{B}}_a & \bar{\mathbf{N}}_a\bar{\mathbf{B}}_a & \bar{\mathbf{B}}_a\end{bmatrix} \tag{6.59}$$

$$\boldsymbol{z}_{3,a}(k) = \begin{bmatrix} \boldsymbol{v}_a(k+3) \\ \boldsymbol{v}_a(k+2)u(k+1) \\ \boldsymbol{v}_a(k+2)u(k) \\ \boldsymbol{v}_a(k+1)u^2(k) \\ \boldsymbol{v}_a(k+2) \\ \boldsymbol{v}_a(k+1)u(k) \\ \boldsymbol{v}_a(k+1) \end{bmatrix} \tag{6.60}$$

Following the same approach, anticausal IOSRs can be obtained for any arbitrary value of p and are denoted by

$$\boldsymbol{x}(k) = \mathbf{T}_{p,a}\boldsymbol{z}_{p,a}(k) \tag{6.61}$$

where the superstate $\boldsymbol{z}_{p,a}(k)$ is defined as the column vector with the following entries

- $\boldsymbol{v}_a(k+1)u^i(k)$ for all $i = 0, 1, \ldots, p-1$
- $\boldsymbol{v}_a(k+2)u^{i_1}(k)u^{i_2}(k+1)$ for all $i_1, i_2 \geq 0$ and $i_1 + i_2 \leq p-2$

  $\vdots$
- $\boldsymbol{v}_a(k+r)u^{i_1}(k)u^{i_2}(k+1) \ldots u^{i_p}(k+p-r)$ for all $i_j \geq 0$ and $\sum_{j=1}^{r} i_j \leq p-r$

  $\vdots$
- $\boldsymbol{v}_a(k+p)$

and depends on input-output data at steps k, $k+1$, ... , $k+p$ only. Note that the above definition of superstate $\boldsymbol{z}_{p,a}(k)$ with $\boldsymbol{v}_a(k)$ given by equation (6.48d) leads to some redundancy in the entries of $\boldsymbol{z}_{p,a}(k)$, which is suggested to be eliminated when implementing the desired identification algorithm.

It is worth remarking that the condition for a bilinear system to have an exact anticausal IOSR is formally the same as for the causal IOSR, but it involves $\bar{\mathbf{A}}_a$ and $\bar{\mathbf{N}}_a$ instead of $\bar{\mathbf{A}}_c$ and $\bar{\mathbf{N}}_c$, i.e.

$$\boldsymbol{\mathcal{C}}_p\left(\bar{\mathbf{A}}_a, \bar{\mathbf{N}}_a\right) = \mathbf{0} \tag{6.62}$$

A bilinear system ideal in *backward* sense is defined as a system satisfying equation (6.62) for some p.

To be rigorous, the previous theorem about convergence for increasing p refers to the causal IOSR and its extension the anticausal IOSR would need modifying the proof. Numerical evidence indeed shows convergence of the anticausal IOSR; however, the extension of the theorem is unnecessary thanks to the mixed-anticausal IOSR that will be presented next.

6.5.3 Considerations on causal and anticausal IOSRs

Having so far developed two IOSRs only, no alternative is possible when using the IS method. In contrast, the ELM method requires one IOSR only, therefore asking which IOSR is preferred is of interest.

Consider two extreme examples. If the system to be identified was an ideal bilinear model (in forward sense), then the causal IOSR would be the preferred choice since, being exact, it would in turn give an exact ELM and exact identification (apart from noise). Similarly, if the system in question was an ideal bilinear model in backward sense, then the anticausal IOSR would have to be chosen. A more realistic example concerns a non-ideal bilinear system, whose causal (anticausal) IOSR converges to the actual dynamics faster than its anticausal (causal) IOSR as p increases. A preferred choice would still exist and be the causal (anticausal) IOSR. An a priori optimal choice is not possible. Obviously, it is very hard to know how close a system is

to being ideal before its identification. Still, having different IOSRs at our disposal is helpful, allowing us to try different ELMs and then choose the one converging faster, with apparent computational advantages.

It is worth remarking that when the IS identification method is used, the order p of the causal and anticausal representation can be different ($p_c \neq p_a$). The ideal situation would be to have an ideal bilinear system in both forward and backward sense, which is a rare (if at all possible) occurrence. Four interaction matrices ($\mathbf{M}'_c$, $\mathbf{M}''_c$, $\mathbf{M}'_a$, $\mathbf{M}''_a$) are required. One non-exact IOSR is sufficient to make the IS method approximate (obviously, the larger p, the more accurate the identification, thanks to the above theorem about IOSR convergence).

Based on these considerations, it makes sense to ask whether other, more favorable, IOSRs can be found. Here two more examples are provided and the resulting representations are referred to as *mixed* since they relate the current state to both past and future input-output data. The remarkable feature of these mixed representations is that each of them needs three interaction matrices, and when used together with a causal or anticausal representation the overall requirement for the IS method to be exact is relaxed to the existence of three interaction matrices only.

6.5.4 Mixed-anticausal IOSR

Start from equation (6.47) and work on its right-hand side replacing $\boldsymbol{x}(k+1)$ with equation (6.49) and $\boldsymbol{x}(k)$ with equation (6.44) to get

$$\begin{aligned}\boldsymbol{x}(k) =&\bar{\mathbf{A}}_a^2\boldsymbol{x}(k+2) + \bar{\mathbf{A}}_a\bar{\mathbf{N}}_a\boldsymbol{x}(k+1)u(k+1) + \bar{\mathbf{A}}_a\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+2)\\ &+ \bar{\mathbf{N}}_a\mathbf{T}_{p_c,c}\boldsymbol{z}_{p_c,c}(k)u(k) + \bar{\mathbf{B}}_a\boldsymbol{v}_a(k+1)\end{aligned} \tag{6.63}$$

Replace $\boldsymbol{x}(k+1)$ in equation (6.63) with equation (6.44) shifted forward by 1 time step, getting

$$\begin{aligned}\boldsymbol{x}(k) =&\bar{\mathbf{A}}_a^2\boldsymbol{x}(k+2) + \bar{\mathbf{A}}_a\bar{\mathbf{N}}_a\mathbf{T}_{p_c,c}\boldsymbol{z}_{p_c,c}(k+1)u(k+1) + \bar{\mathbf{A}}_a\bar{\mathbf{B}}_a\boldsymbol{v}_a(k+2)\\ &+ \bar{\mathbf{N}}_a\mathbf{T}_{p_c,c}\boldsymbol{z}_{p_c,c}(k)u(k) + \bar{\mathbf{B}}_a\boldsymbol{v}_a(k+1)\end{aligned} \tag{6.64}$$

Observe that the only state-dependent term on the right-hand side of equation (6.64) is multiplied by $\bar{\mathbf{A}}_a^2$. If $n = 2$, it is sufficient that $\mathbf{A}^{-1}$ and $\mathbf{C}$ form an observable pair to guarantee the existence of an interaction matrix $\mathbf{M}_a'$ such that $\bar{\mathbf{A}}_a^2$ vanishes. Note that interaction matrix $\mathbf{M}_a''$ becomes unnecessary, therefore $\bar{\mathbf{N}}_a = -\mathbf{A}^{-1}\mathbf{N}$. If $n = 3$, the above derivation can be continued replacing $\boldsymbol{x}(k+2)$ with equation (6.55) and taking care of the resulting terms as done above, getting an equation for $\boldsymbol{x}(k)$ with the only state-dependent term multiplied by $\bar{\mathbf{A}}_a^3$, again guaranteed to be zero if $(\mathbf{A}^{-1}, \mathbf{C})$ is an observable pair. The generalized IOSR is denoted as

$$\boldsymbol{x}(k) = \mathbf{T}_{p_c,p_a,ma}\boldsymbol{z}_{p_c,p_a,ma}(k) \tag{6.65}$$

where, for the case $n = 2$, p_a can be 2 and $\mathbf{T}_{p_c,p_a,ma}$ and $\boldsymbol{z}_{p_c,p_a,ma}(k)$ specialize to

$$\mathbf{T}_{p_c,2,ma} = \begin{bmatrix}\bar{\mathbf{A}}_a\bar{\mathbf{N}}_a\mathbf{T}_{p_c,c} & \bar{\mathbf{N}}_a\mathbf{T}_{p_c,c} & \bar{\mathbf{A}}_a\bar{\mathbf{B}}_a & \bar{\mathbf{B}}_a\end{bmatrix} \tag{6.66a}$$

$$\boldsymbol{z}_{p_c,2,ma}(k) = \begin{bmatrix}\boldsymbol{z}_{p_c,c}(k+1)u(k+1)\\ \boldsymbol{z}_{p_c,c}(k)u(k)\\ \boldsymbol{v}_a(k+2)\\ \boldsymbol{v}_a(k+1)\end{bmatrix} \tag{6.66b}$$

More in general,

$$\boldsymbol{z}_{p_c,p_a,ma}(k) = \begin{bmatrix} \boldsymbol{z}_{p_c,c}(k+p_a-1)u(k+p_a-1) \\ \dots \\ \boldsymbol{z}_{p_c,c}(k)u(k) \\ \boldsymbol{v}_a(k+p_a) \\ \dots \\ \boldsymbol{v}_a(k+1) \end{bmatrix} \tag{6.67}$$

and the subscript *ma* stands for *mixed-anticausal* to distinguish this IOSR from its counterpart derived below. The above IOSR is *mixed* because it depends on both past and future input-output data. The definition of its superstate formally features a heavier contribution from future input-output data, hence the a in the subscript. Also, note that the definition of $\boldsymbol{v}_a(k)$ given in equation (6.48d) leads to some redundancy in the entries of $\boldsymbol{z}_{p_c,p_a,ma}(k)$ in equation (6.67), which is not difficult to eliminate when implementing the desired identification algorithm.

6.5.5 Mixed-causal IOSR

With a similar approach, a relation symmetric to equation (6.65) can be derived

$$\boldsymbol{x}(k) = \mathbf{T}_{p_c,p_a,mc}\boldsymbol{z}_{p_c,p_a,mc}(k) \tag{6.68}$$

where

$$\boldsymbol{z}_{p_c,p_a,mc}(k) = \begin{bmatrix} \boldsymbol{z}_{p_a,a}(k-p_c)u(k-p_c) \\ \dots \\ \boldsymbol{z}_{p_a,a}(k-1)u(k-1) \\ \boldsymbol{v}_c(k-p_c) \\ \dots \\ \boldsymbol{v}_c(k-1) \end{bmatrix} \tag{6.69}$$

and *mc* stands for *mixed-causal*. Again, note that the definition of superstate $\boldsymbol{z}_{p,a}(k)$ of equation (6.61) with $\boldsymbol{v}_a(k)$ given by equation (6.48d) leads to some redundancy in the entries of $\boldsymbol{z}_{p_c,p_a,mc}(k)$, which is suggested to be eliminated when implementing the desired identification algorithm.

6.5.6 Considerations on mixed IOSRs

The mixed-anticausal IOSR can be of great advantage when used in the IS identification method. As already highlighted, for the mixed-anticausal IOSR to be exact, it is necessary for the bilinear system to be ideal only in forward sense, despite its mixed nature. This means that an ideal bilinear system (in forward sense) can be identified exactly with a version of the IS method based on causal and mixed-anticausal IOSRs. The only condition imposed on the backward system, equation (6.47), is that $(\mathbf{A}^{-1},\mathbf{C})$ is an observable pair. $\mathbf{M}''_a$ is no longer required to exist and the overall requirements on interaction matrix existence are significantly relaxed. For the sake of clarity, it is worth remarking that the matrix $\mathbf{M}'_a$ making the mixed-anticausal IOSR exact is in general different from the matrix $\mathbf{M}'_a$ necessary for the anticausal IOSR to be exact. The mixed-causal representation is complementary to the mixed-anticausal IOSR in the sense that, when used together with the anticausal IOSR in the IS identification method, it requires

the bilinear system to be ideal only in backward sense for the identification to be exact. The existence of $\mathbf{M}'_c$, $\mathbf{M}'_a$ and $\mathbf{M}''_a$ only is required.

The advantage does not disappear for more general non-ideal bilinear systems. Obviously, if the system IOSR turns out to converge faster in its forward (backward) formulation, then the IS method will benefit from the choice of causal and mixed-anticausal (anticausal and mixed-causal) IOSRs. To be noted is the fact that the dimension of the superstate in the above mixed IOSRs is larger than in purely causal or anticausal IOSRs. There would be no point in using a mixed representation in the ELM identification method.

6.6 Algorithms

In this section, a step-by-step description of the methods proposed above for discrete-time bilinear system identification is provided. The data necessary to run the algorithms is the time history of the measured input $u(k)$ and output $\boldsymbol{y}(k)$, $k = 0, 1, \ldots, l-1$.

6.6.1 Equivalent linear model (ELM) method

1. *Construction of ELM Input.* Form $\boldsymbol{w}(k)$ time history, equation (6.6), for $k = k_i, k_i + 1, \ldots, k_f$ choosing for $\boldsymbol{z}(k)$ an IOSR among the following

   i Causal IOSR, equation (6.44)

   ii Anticausal IOSR, equation (6.61)

   The order p_c or p_a of the selected IOSR must be chosen as well, typically greater than the assumed order of the system n.

2. *Identification of ELM.* Identify the ELM of equation (6.4) by any linear system identification technique (e.g. the superspace method described in Reference 33 or the intersection subspace method presented in this chapter with modified causal and anticausal IOSRs, which are significantly simplified for a linear system). The order n_{id} of the identified ELM will also be the order of the identified bilinear model.

3. *Reconstruction of State History.* Simulate the identified ELM driven by the input $\boldsymbol{w}(k)$ to reconstruct the state history $\boldsymbol{x}(k)$ for $k = 0, 1, \ldots, l-1$; since the initial state is unknown, discard the reconstructed samples corresponding to the initial transient.

4. *Estimation of Bilinear System Matrices.* Use the reconstructed state history $\boldsymbol{x}(k)$ to form the least-squares problems of equations (6.16) and (6.17), with k_i now equal to a time step after the initial transient is over, and use equations (6.18) and (6.19) to estimate $\mathbf{A}$, $\mathbf{N}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$.

Note that two variants of steps 3 and 4 to complete the estimation of the bilinear model matrices are given in Reference 33.

6.6.2 Intersection subspace (IS) method

1. *Construction of Superspaces.* Form matrices $\mathbf{Z}_a$ and $\mathbf{Z}_b$, equations (6.9) and (6.10), choosing for $\boldsymbol{z}_a(k)$ and $\boldsymbol{z}_b(k)$ two IOSRs among the following

   i Causal IOSR, equation (6.44)

   ii Anticausal IOSR, equation (6.61)

   iii Mixed-causal IOSR, equations (6.65, 6.67)

   iv Mixed-anticausal IOSR, equations (6.68) and (6.69)

The order p_c and p_a of the selected IOSRs must be chosen as well, typically greater than the assumed n (note that p_c can be chosen to be different from p_a).

2. *Intersection of Superspaces.* Take the SVD of $\mathbf{R} = \begin{bmatrix} \mathbf{Z}_a^{\mathrm{T}} & \mathbf{Z}_b^{\mathrm{T}} \end{bmatrix}$ and select the right singular vectors (call them $\boldsymbol{c}_i$, $i = 1, \ldots, m$) corresponding to the zero singular values of R.

3. *Construction of Basis for State History Space.* Extract from vectors $\boldsymbol{c}_i$ the first entries corresponding to vectors $\boldsymbol{a}_i$, equation (6.13b), and use them to compute row vectors $\boldsymbol{h}_i = \boldsymbol{a}_i^{\mathrm{T}} \mathbf{Z}_a$, $i = 1, \ldots, m$.

4. *Reduction of Basis to Linear Independent Vectors.* Stack vectors $\boldsymbol{h}_i$, $i = 1, \ldots, m$ in a matrix $\mathbf{H}$, take the SVD of $\mathbf{H}$ and select its right singular vectors (call them $\boldsymbol{x}_i$, $i = 1, \ldots, n_{id}$) corresponding to the non-zero singular values, whose number n_{id} will then be the order of the identified bilinear model.

5. *Estimation of Bilinear System Matrices.* Transpose column vectors $\boldsymbol{x}_i$ to get row vectors, each of them representing the time history of the i^{th} state variable of the identified model. Use vectors $\boldsymbol{x}_i$ to form the least-squares problems of equations (6.16) and (6.17) and use equations (6.18) and (6.19) to estimate $\mathbf{A}$, $\mathbf{N}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$.

6.7 Examples

Several numerical examples are provided to show the correctness and main features of the different identification algorithms that can be devised based on the IOSRs derived in this chapter. In each example, measured input-output data are simulated by generating a random input sequence $\{u(k)\}$ of $l = 1000$ samples (from a uniform distribution between -0.5 and 0.5) and using it to obtain the output history $\{y(k)\}$ via the state-space model equations of the system to be identified. $\{u(k)\}$ and $\{y(k)\}$ sequences are then used to execute the chosen identification

algorithm. Finally, *another* input sequence $\{u(k)\}$ is generated (independently from the one used for identification) and used to drive both the original system and the identified model. The difference in their outputs, also known as prediction error, is analyzed as a metric to gauge how accurate the identification algorithm was.

For detailed examples and comments on the ELM identification method based on the causal IOSR, see Reference 33. The following bilinear systems are used in the next examples as prototypes of ideal and arbitrary bilinear systems.

System I (ideal in forward sense)

$$\mathbf{A} = \begin{bmatrix} 0 & 0.5 \\ 0.5 & -0.5 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \qquad \mathbf{N} = \begin{bmatrix} 0 & 1 \\ -1 & 1 \end{bmatrix} \tag{6.70a}$$

$$\mathbf{C} = \begin{bmatrix} 0 & 1 \end{bmatrix} \qquad D = 0 \tag{6.70b}$$

It is possible to verify that the following $\mathbf{M}_c'$ and $\mathbf{M}_c''$ satisfy equation (6.30). See Reference 33 for more details.

$$\mathbf{M}_c' = \begin{bmatrix} -0.5 \\ 0.5 \end{bmatrix} \qquad \mathbf{M}_c'' = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \tag{6.71}$$

System II (ideal in backward sense)

$$\mathbf{A} = \begin{bmatrix} 0.5 & 0.5 \\ 0.5 & 0 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \qquad \mathbf{N} = \begin{bmatrix} 0.5 & -1 \\ 0 & -0.5 \end{bmatrix} \tag{6.72a}$$

$$\mathbf{C} = \begin{bmatrix} 0 & 1 \end{bmatrix} \qquad D = 0 \tag{6.72b}$$

No pair $\mathbf{M}_c'$, $\mathbf{M}_c''$ for which equation (6.31) is satisfied exists. However, the following $\mathbf{M}_a'$ and $\mathbf{M}_a''$

$$\mathbf{M}_a' = \begin{bmatrix} -2 \\ 2 \end{bmatrix} \qquad \mathbf{M}_a'' = \begin{bmatrix} -1 \\ -1 \end{bmatrix} \tag{6.73}$$

satisfy equation (6.62) for $p = 2$ or, equivalently, equation (6.51). System II is therefore ideal in backward sense.

System III (arbitrary)

Modifying matrix $\mathbf{N}$ of system I to

$$\mathbf{N} = \begin{bmatrix} 0.3 & 1 \\ -1 & 1 \end{bmatrix} \tag{6.74}$$

is sufficient to lose the above mentioned ideal properties. System III, equal to system I but with $\mathbf{N}$ defined by equation (6.74), is therefore an example of arbitrary bilinear system.

6.7.1 ELM method with anticausal IOSR

Consider system II. Its identification via a version of the ELM method based on an anticausal IOSR of order $p_a = 2$ is exact, as expected thanks to its ideal backward property. The maximum

prediction error is around 10^{-13}, i.e. close to Matlab ® working precision. Also, it is worth remarking that the identified bilinear system has the correct dimension ($n_{id} = 2$). In contrast, using the version of the ELM method based on the causal IOSR leads to an approximate identification algorithm due to the non-ideal nature of system II in forward sense. For $p_c = 2$ the identified model is of (wrong) order $n_{id} = 3$ and the prediction accuracy is quite poor (maximum prediction error of 10^{-2}). However, as guaranteed by the convergence theorem, identification accuracy improves as p_c is increased. With $p_c = 4$, for example, the order of the identified system is correctly estimated to be $n_{id} = 2$, and the maximum prediction error is reduced to 10^{-4}. The anticausal IOSR is therefore a useful complement to the causal IOSR when the ELM method is used, providing more accurate results when the bilinear system is (or at least is closer to being) ideal in backward sense.

6.7.2 IS Method with causal and anticausal IOSRs

Again with reference to system II, the IS identification method can also be applied. For the IS method two IOSRs are needed, and the causal and anticausal IOSRs can serve for the purpose. With system II, which is ideal in backward sense, the causal IOSR is expected to introduce approximation in the identification. In agreement with the convergence theorem, increasing the order of the causal IOSR improves the accuracy of the identified system. While for $p_c = p_a = 2$ the maximum prediction error is around 10^{-2}, with $p_c = 6$, $p_a = 2$ it gets as low as 10^{-6}. In both cases, the order of the identified system turns out to be correct ($n_{id} = 2$). Note that since none of the presented bilinear systems is ideal in both forward and backward sense, the IS algorithm based on causal and anticausal IOSRs cannot be exact and p_c and/or p_a must be increased to improve the identification accuracy.

TABLE 6.1: IS method with causal and mixed-anticausal IOSRs - identification improvement as p_c is increased (p_c = order of causal IOSR, p_a = order of anticausal IOSR, e_{pred} = order of magnitude of maximum prediction error).

p_c	2	3	4	5	6
p_a	3	3	3	3	3
e_{pred}	10^{-1}	10^{-3}	10^{-4}	10^{-5}	10^{-5}

6.7.3 IS Method with causal and mixed-anticausal IOSRs

To appreciate how the IS method can benefit from the use of a mixed IOSR, consider system I. Thanks to its forward ideal property, there exist interaction matrices $\mathbf{M}_c'$ and $\mathbf{M}_c''$ such that equation (6.29) is satisfied for $p = 2$. Also, the use of the mixed-anticausal IOSR instead of a purely anticausal IOSR relaxes the condition for the existence of $\mathbf{M}_a'$ and $\mathbf{M}_a''$ to the existence of a single $\mathbf{M}_a'$, guaranteed since $\mathbf{A}^{-1}$ and $\mathbf{C}$ form an observable pair. Identification is therefore expected to be exact when using these IOSRs both of order as low as 2. This is indeed the case, as confirmed by numerical experiments with $p_c = p_a = 2$, where the maximum prediction error is again of the order of Matlab ® working precision. Relying on a mixed IOSR can make the IS identification method exact, in contrast with the IS version based only on pure IOSRs (causal and anticausal).

In order to give more details about the new IS identification method presented in this chapter and to remark its effectiveness, a thorough example of its application on an arbitrary bilinear system of the IS method based on causal and mixed-anticausal IOSRs is now described. Consider system III and for the sake of generality assume no prior knowledge of its order $n = 2$ is available. The a priori guess for the order of the system is, for example, $n = 3$. Accordingly, p_a is set to be equal to 3 and then p_c can be varied to achieve the desired identification accuracy. Table 6.1 summarizes the obtained results.

Note that in the mixed-anticausal IOSR $p_c < p_a$ is not a consistent choice, since p_a must

TABLE 6.2: IS method with causal and mixed-anticausal IOSRs - identification improvement as p_c is increased (p_c = order of causal IOSR, p_a = order of anticausal IOSR, e_{pred} = order of magnitude of maximum prediction error).

p_c	2	3	4	5	6
p_a	2	2	2	2	2
e_{pred}	10^{-2}	10^{-4}	10^{-5}	10^{-6}	10^{-7}

be chosen to be not less than the assumed order of the system and the requirements for the causal interaction matrices are much tougher than those for the (only) anticausal interaction matrix. This observation is well reflected in the previous table. As p_c is increased, the prediction accuracy of the identified model improves. Also note that for $p_c = 6$ the length of the input-output sequences had to be increased to preserve the overdetermined nature of matrix $\mathbf{R}$ of equation (6.13a). Indeed, as p_c and p_a are increased, the length of the measured dataset should be increased as well to avoid trivial redundancy of columns of $\mathbf{R}$. If the guess on n is improved and the identification algorithm is executed with $p_a = 2$, the accuracy of results improves accordingly (less numerical noise in the SVDs), as shown in Table 6.2.

For the purpose of showing the key steps of the algorithm, the representative case for $p_c = 4$, $p_a = 3$ is now analyzed more in details. Step 1: Indicating with $\mathbf{Z}_a$ the superstate history matrix formed with the causal superstate $\boldsymbol{z}_{4,c}(k)$ of equation (6.44) and with $\mathbf{Z}_b$ the superstate history matrix formed with the mixed-anticausal superstate $\boldsymbol{z}_{4,3,ma}(k)$ of equation (6.67), $\mathbf{Z}_a$ and $\mathbf{Z}_b$ have respectively 60 and 193 rows. Step 2: The initial and final time steps for which both IOSRs are valid are $k_i = p_c = 4$ and $k_f = l - 1 - p_a = 996$, giving rise to a matrix $\mathbf{R} \in \mathbb{R}^{993\text{x}253}$. The singular values of $\mathbf{R}$ (normalized over the largest) are shown in Figure 9.2a. Due to the approximation of the IOSRs, there are no singular values exactly equal to zero and the user has to select the ones that are small enough to suggest some linear dependence of the columns of $\mathbf{R}$. This might be a delicate step, but the proposed algorithm proved to be very robust and the user should not be concerned with minimizing at any cost the number m of singular values chosen at

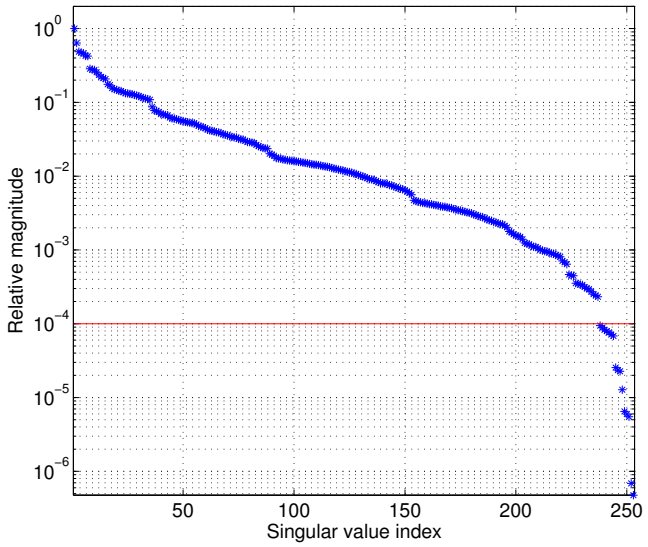


FIGURE 6.1: SVD of step 2 of the IS algorithm (to find the intersection of superspaces Z_a and Z_b).

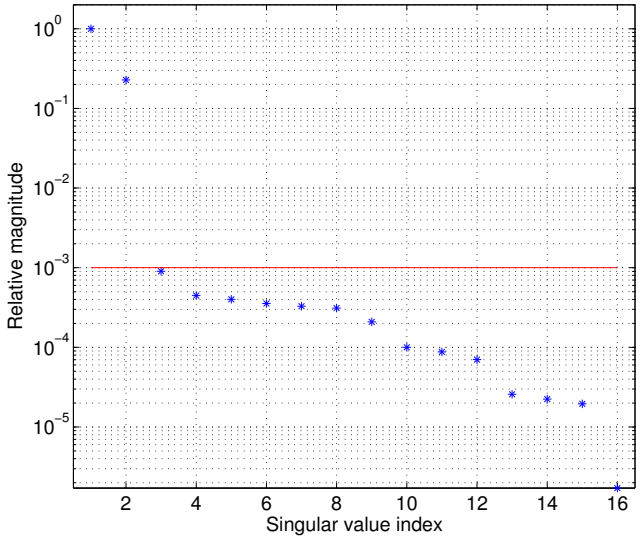


FIGURE 6.2: SVD of step 4 of the IS algorithm (to find a minimum basis for the intersection subspace).

this stage. If necessary, the SVD of Step 4 will automatically perform such further reduction. To remark the point, in this specific example all the singular values 10^4 times smaller than the largest are selected (Figure 9.2a), for a total of $m = 16$ (instead of selecting $m = 2$, which would immediately guarantee $n_{id} = 2$ and lead to similar final results). Step 3: The first 60 entries of each right singular vector $\boldsymbol{c}_i$, $i = 1, ..., 16$ corresponding to the selected zero singular values are

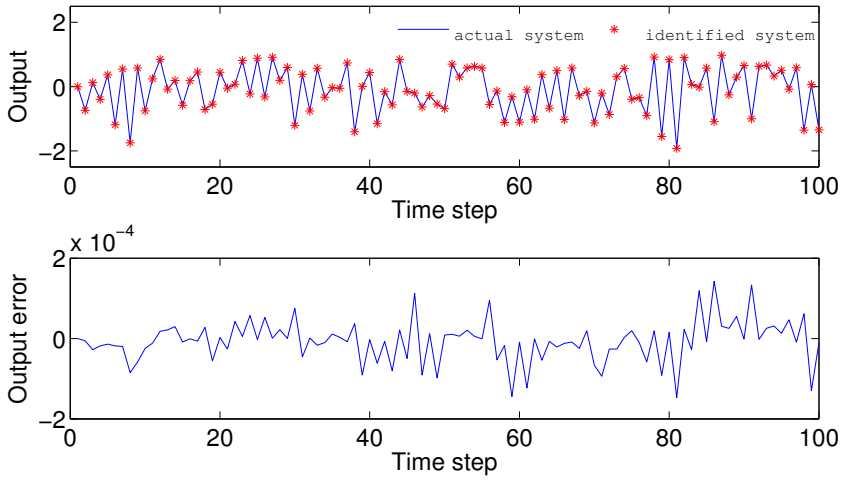


FIGURE 6.3: Comparison of output from actual and identified system III.

TABLE 6.3: IS method with causal and mixed-anticausal IOSRs - eigenvalue comparison between true and identified matrices.

Matrix	Eigenvalue 1	Eigenvalue 2
$\mathbf{A}$	-0.809016994	0.309016994
$\mathbf{A}_{id}$	-0.809012078	0.308957226
Relative error	0.000608%	0.019341%
$\mathbf{N}$	$0.649999999 + i0.936749700$	$0.649999999 - i0.936749700$
$\mathbf{N}_{id}$	$0.650096808 + i0.936686271$	$0.650096808 - i0.936686271$
Relative error	0.000270%	0.000270%

extracted to form column vectors $\boldsymbol{a}_i$, and row vectors $\boldsymbol{h}_i$ are then computed via the first equality of equation (6.12). Step 4: Matrix $\mathbf{H} \in \mathbb{R}^{16\text{x}993}$ with $\boldsymbol{h}_i$ vectors as rows is constructed and its SVD taken; the graph of normalized singular values of $\mathbf{H}$ is shown in Figure 9.2b, from which it is apparent how only two rows of $\mathbf{H}$ are linearly independent. Therefore, only two singular values and corresponding right singular vectors are selected and used as an estimate of the state history $\{\boldsymbol{x}(k)\}$. The order of the identified bilinear model is then $n_{id} = 2$. Step 5: Solving the least-squares problems of equations (6.16, 6.17) leads to the estimate of the following bilinear model matrices (rounded to the 4^{th} significant digit)

$$\mathbf{A}_{id} = \begin{bmatrix} -0.8499 & -0.1322 \\ 0.3587 & 0.3499 \end{bmatrix} \qquad \mathbf{B}_{id} = \begin{bmatrix} 0.04835 \\ 0.08492 \end{bmatrix}$$

$$\mathbf{N}_{id} = \begin{bmatrix} 0.4456 & -0.6513 \\ 1.411 & 0.8546 \end{bmatrix} \qquad \mathbf{C}_{id} = \begin{bmatrix} 20.33 & 11.98 \end{bmatrix}$$

$$D_{id} = -2.486 \times 10^{-5}$$

Figure 6.3 shows the accuracy of the identified model comparing its predicted output with the output from the actual system III, when both are driven by the same random input sequence, generated independently from the one used to simulate input-output data. As another indicator of the effectiveness of the identification, Table 6.3 compares the eigenvalues of the true matrices **A** and **N** of the actual system and the eigenvalues of the corresponding identified matrices.

6.8 Conclusions

This chapter has introduced the problem of deterministic bilinear system identification and the approach chosen to solve it. Additionally, this chapter has rigorously proven the extension of the interaction matrices to bilinear systems. Interaction matrices were used to derive multiple linear relations between the bilinear system state and input-output data (IOSRs), from which several identification algorithms for bilinear systems were developed. The presented intersection subspace method was shown to be very effective for bilinear system identification. The state of the bilinear system is reconstructed by intersecting vector spaces corresponding to two IOSRs. From the state information, a bilinear state-space model of the system is identified by simple least-squares. A new version of the equivalent linear model approach was also introduced, based on an anticausal IOSR. The benefits of having several IOSRs were highlighted and also shown

by numerical examples. In terms of advantages, no specific form of excitation input is required, beyond the standard richness condition for system identification. Also, there is no need to use data from multiple experiments as input-output data from a single experiment is adequate. The identification algorithms developed in this chapter offer a way to develop, from measured input-output data, the mathematical models representing the dynamic behavior of systems that are inherently bilinear. Such models can then be used for system analysis or controller design. Additionally, the algorithms are applicable to input-affine nonlinear systems to find bilinear models approximating their dynamics as established by Carleman linearization, providing a promising approach to handle more general nonlinear control problems.

Although the overall theoretical foundation has been established, a few practical aspects of the identification algorithms presented in this chapter require further work. Accurate identification of high-order bilinear models, such as those arising from Carleman linearization, may require a relatively large value of p which will lead to significant computational burden. Also, the convergence of the presented identification algorithms as p increases is guaranteed at the price of keeping the excitation input within a certain bound. This may lead to two potential difficulties. First, the bound is unknown before identification, hence some trial-and-error procedure might be needed. Second, such a bound might turn out to be very small and challenging for practical applications. A solution to those issues is presented in the chapter 7.

Chapter 7

Deterministic bilinear system identification with specialized input

7.1 Introduction

Drawing inspiration from the technique at the basis of OKID, interaction matrices were used in chapter 6 to derive linear Input-Output-to-State Representations (IOSRs) for bilinear systems, i.e. relationships expressing the state at any time step as a linear function of a superstate defined in terms of only input-output data (typically from the past p time steps or from the future p time steps). The so derived IOSRs were then used to implement identification algorithms for bilinear systems known as equivalent linear model (ELM) and intersection subspace (IS) methods. Interaction matrices giving rise to exact IOSRs exist however only for very specific bilinear models. Theorem 6.1 was proven ensuring for any bilinear system with observable linear part the existence of interaction matrices such that the resulting IOSRs converge to exact relationships as the IOSR order p is increased. Unfortunately, the dimension of the superstate,

and hence the required computational effort, increases exponentially with p. The drawback becomes more relevant when one wants to identify high-order bilinear systems, as is the case in the use of bilinear models to approximate more general nonlinear systems as mentioned above. Another drawback of the methods presented in References 33, 50 is that convergence of IOSRs as p increases is guaranteed only if the input magnitude is kept within a certain bound, which is unknown before system identification is performed.

The content of this chapter was presented at the AAS/AIAA Astrodynamics Specialist Conference in Hilton Head, SC, in 2013 (Reference 61), and addresses the same problem stated in section 6.2. A different approach is now taken to derive small-dimension IOSRs which are exact for any arbitrary bilinear model (with minimum observability requirements) and for any input magnitude, overcoming the above-mentioned difficulties. The new approach exploits the fact that a bilinear system can be seen as a linear-time-varying (LTV) model. Interaction matrices are then used on this LTV formulation to derive exact time-varying IOSRs of minimum order and even minimum dimension for the given order. The computational benefit is twofold. Not only are the time-varying IOSRs exact with minimum order p (less than or equal to the order n of the bilinear model) but also the dimension of the corresponding superstate is increasing linearly with p instead of exponentially. This feature makes the approach very attractive when dealing with high-order bilinear models, as those usually arising from Carleman linearization. Also, the LTV approach requires that the input takes at each time step any value from a finite set whose values must be specified a priori but are not subject to magnitude constraints. For the application of the ELM or IS identification methods, the time-varying IOSRs need to be transformed into time-invariant IOSRs. This operation is crucial since, if performed naively, it generally leads to a fast increase in the dimension of the time-invariant IOSR. We also address this problem and show how appropriate design of the input excitation preserves the algorithm

computational efficiency obtained with the LTV approach. We provide a method to generate an input sequence which is sufficiently rich for identification and at the same time keeps under control the computational requirements of the algorithm.

All these advantages make of the proposed method a crucial step towards the realization of the bridge between linear and nonlinear systems mentioned in chapter 1, where the identification of high-order bilinear models is usually expected. For the purpose of illustrating the potential of the method, detailed examples are given at the end of the chapter on (input-affine) nonlinear dynamic systems of practical interest, such as the mechanical oscillator with cubic spring, also known as Duffing's equation, and the rotation of a rigid body in a reference frame fixed to it, which is typically referred to as Euler's equations and represents a building block of the satellite attitude control problem.

7.2 Time-varying IOSRs

The key concept behind the derivation of IOSRs in chapter 6 is that of *interaction matrix*. In this section, we apply time-varying interaction matrices to derive a new type of IOSRs, which are referred to as Time-Varying (TV) IOSRs as opposed to the Time-Invariant (TI) IOSRs proposed in chapter 6. The TV IOSRs overcome all the limitations of the TI IOSRs, as discussed at the end of the section.

7.2.1 TV causal IOSRs

Any bilinear system of the form of equation (6.1) can be rewritten as a linear system with time-varying system matrix $\mathbf{A}$ or time-varying influence matrix $\mathbf{B}$, depending on how the bilinear term $\mathbf{N}\boldsymbol{x}(k)u(k)$ is lumped with one of the other two terms. In this work we use the time-varying

system matrix formulation

$$\boldsymbol{x}(k+1) = \mathbf{A}(k)\boldsymbol{x}(k) + \mathbf{B}u(k) \tag{7.1}$$

where

$$\mathbf{A}(k) = \mathbf{A} + \mathbf{N}u(k) \tag{7.2}$$

Propagate equation (7.1) forward in time and get

$$\begin{aligned}
\boldsymbol{x}(k+2) &= \mathbf{A}(k+1)\boldsymbol{x}(k+1) + \mathbf{B}u(k+1) \\
&= \mathbf{A}(k+1)\big(\mathbf{A}(k)\boldsymbol{x}(k) + \mathbf{B}u(k)\big) + \mathbf{B}u(k+1) \\
&= \mathbf{A}(k+1)\mathbf{A}(k)\boldsymbol{x}(k) + \begin{bmatrix}\mathbf{A}(k+1)\mathbf{B} & \mathbf{B}\end{bmatrix} \begin{bmatrix} u(k) \\ u(k+1) \end{bmatrix}
\end{aligned} \tag{7.3}$$

From equation (6.1b), we can also write

$$\begin{aligned}
\boldsymbol{y}(k+1) &= \mathbf{C}\boldsymbol{x}(k+1) + \mathbf{D}u(k+1) \\
&= \mathbf{C}\mathbf{A}(k)\boldsymbol{x}(k) + \mathbf{C}\mathbf{B}u(k) + \mathbf{D}u(k+1)
\end{aligned} \tag{7.4}$$

and put equations (6.1b) and (7.4) together in matrix form

$$\begin{bmatrix} \boldsymbol{y}(k) \\ \boldsymbol{y}(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A}(k) \end{bmatrix} \boldsymbol{x}(k) + \begin{bmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{C}\mathbf{B} & \mathbf{D} \end{bmatrix} \begin{bmatrix} u(k) \\ u(k+1) \end{bmatrix} \tag{7.5}$$

Add and subtract $\mathbf{M}_{2,c}(k+2)\begin{bmatrix}\boldsymbol{y}^{\mathrm{T}}(k) & \boldsymbol{y}^{\mathrm{T}}(k+1)\end{bmatrix}$ in equation (7.3) and, plugging equation (7.5) in, obtain

$$\begin{aligned}\boldsymbol{x}(k+2) &= \mathbf{A}(k+1)\mathbf{A}(k)\boldsymbol{x}(k) + \begin{bmatrix}\mathbf{A}(k+1)\mathbf{B} & \mathbf{B}\end{bmatrix}\begin{bmatrix}u(k)\\ u(k+1)\end{bmatrix} \\ &\quad + \mathbf{M}_{2,c}(k+2)\begin{bmatrix}\mathbf{C}\\ \mathbf{C}\mathbf{A}(k)\end{bmatrix}\boldsymbol{x}(k) + \mathbf{M}_{2,c}(k+2)\begin{bmatrix}\mathbf{D} & \mathbf{0}\\ \mathbf{C}\mathbf{B} & \mathbf{D}\end{bmatrix}\begin{bmatrix}u(k)\\ u(k+1)\end{bmatrix} \\ &\quad - \mathbf{M}_{2,c}(k+2)\begin{bmatrix}\boldsymbol{y}(k)\\ \boldsymbol{y}(k+1)\end{bmatrix} \\ &= \big(\tilde{\mathbf{A}}_{2,c}(k+2) + \mathbf{M}_{2,c}(k+2)\tilde{\mathbf{C}}_{2,c}(k+2)\big)\boldsymbol{x}(k) + \tilde{\mathbf{T}}_{2,c}(k+2)\tilde{\boldsymbol{z}}_{2,c}(k+2) \end{aligned} \tag{7.6}$$

where

$$\tilde{\mathbf{A}}_{2,c}(k+2) = \mathbf{A}(k+1)\mathbf{A}(k) \qquad \tilde{\mathbf{C}}_{2,c}(k+2) = \begin{bmatrix}\mathbf{C}\\ \mathbf{C}\mathbf{A}(k)\end{bmatrix} \tag{7.7a}$$

$$\tilde{\boldsymbol{z}}_{2,c}(k+2) = \begin{bmatrix}u(k) & u(k+1) & \boldsymbol{y}^{\mathrm{T}}(k) & \boldsymbol{y}^{\mathrm{T}}(k+1)\end{bmatrix}^{\mathrm{T}} \tag{7.7b}$$

$$\tilde{\mathbf{T}}_{2,c}(k+2) = \begin{bmatrix}\begin{bmatrix}\mathbf{A}(k+1)\mathbf{B} & \mathbf{B}\end{bmatrix} + \mathbf{M}_{2,c}(k+2)\begin{bmatrix}\mathbf{D} & \mathbf{0}\\ \mathbf{C}\mathbf{B} & \mathbf{D}\end{bmatrix} & -\mathbf{M}_{2,c}(k+2)\end{bmatrix} \tag{7.7c}$$

and the subscript c stands for *causal*. Indeed, the superstate $\boldsymbol{z}_{2,c}(k+2)$ is defined solely in terms of *past* input-output data, producing a causal representation. $\mathbf{M}_{2,c}(k+2) \in \mathbb{R}^{n\times 2q}$ is a time-varying interaction matrix.

For equation (7.6) to be an IOSR, the state-dependent term on the right-hand side must vanish for any $\boldsymbol{x}(k)$. As will be proven later, suppose that at every time step k there exists an interaction matrix $\mathbf{M}_{2,c}(k+2)$ such that $\tilde{\mathbf{A}}_{2,c}(k+2) + \mathbf{M}_{2,c}(k+2)\tilde{\mathbf{C}}_{2,c}(k+2) = \mathbf{0}$ and also shift the time

index in equation (7.6) backward by 2 time steps. We can then write the following IOSR of order $p = 2$

$$\boldsymbol{x}(k) = \tilde{\mathbf{T}}_{2,c}(k)\tilde{\boldsymbol{z}}_{2,c}(k) \tag{7.8}$$

By propagating equation (7.3) one step further and adding and subtracting the term $\mathbf{M}_{3,c}(k+3)\begin{bmatrix}\boldsymbol{y}^{\mathrm{T}}(k) & \boldsymbol{y}^{\mathrm{T}}(k+1) & \boldsymbol{y}^{\mathrm{T}}(k+2)\end{bmatrix}$, we obtain

$$\boldsymbol{x}(k+3) = \big(\tilde{\mathbf{A}}_{3,c}(k+3) + \mathbf{M}_{3,c}(k+3)\tilde{\mathbf{C}}_{3,c}(k+3)\big)\boldsymbol{x}(k) + \tilde{\mathbf{T}}_{3,c}(k+3)\tilde{\boldsymbol{z}}_{3,c}(k+3) \tag{7.9}$$

where

$$\tilde{\mathbf{A}}_{3,c}(k+3) = \mathbf{A}(k+2)\mathbf{A}(k+1)\mathbf{A}(k) \qquad \tilde{\mathbf{C}}_{3,c}(k+3) = \begin{bmatrix}\mathbf{C}\\ \mathbf{CA}(k)\\ \mathbf{CA}(k+1)\mathbf{A}(k)\end{bmatrix} \tag{7.10a}$$

$$\tilde{\boldsymbol{z}}_{3,c}(k+3) = \begin{bmatrix}u(k) & u(k+1) & u(k+2) & \boldsymbol{y}^{\mathrm{T}}(k) & \boldsymbol{y}^{\mathrm{T}}(k+1) & \boldsymbol{y}^{\mathrm{T}}(k+2)\end{bmatrix}^{\mathrm{T}} \tag{7.10b}$$

$$\begin{aligned}\tilde{\mathbf{T}}_{3,c}(k+3) = \Big[&\begin{bmatrix}\mathbf{A}(k+2)\mathbf{A}(k+1)\mathbf{B} & \mathbf{A}(k+2)\mathbf{B}\end{bmatrix}\\ &+ \mathbf{M}_{3,c}(k+3)\begin{bmatrix}\mathbf{D} & \mathbf{0} & \mathbf{0}\\ \mathbf{CB} & \mathbf{D} & \mathbf{0}\\ \mathbf{CA}(k+1)\mathbf{B} & \mathbf{CB} & \mathbf{D}\end{bmatrix} - \mathbf{M}_{3,c}(k+3)\Big]\end{aligned} \tag{7.10c}$$

Again, if at every time step k there exists an interaction matrix $\mathbf{M}_{3,c}(k+3)$ such that $\tilde{\mathbf{A}}_{3,c}(k+3) + \mathbf{M}_{3,c}(k+3)\tilde{\mathbf{C}}_{3,c}(k+3) = \mathbf{0}$, equation (7.9) becomes (time index shifted backward by 3 steps)

$$\boldsymbol{x}(k) = \tilde{\mathbf{T}}_{3,c}(k)\tilde{\boldsymbol{z}}_{3,c}(k) \tag{7.11}$$

More in general, propagating equation (7.1) forward in time by $p-1$ time steps and adding and subtracting

$$\mathbf{M}_{p,c}(k+p)\begin{bmatrix} \boldsymbol{y}(k) \\ \boldsymbol{y}(k+1) \\ \vdots \\ \boldsymbol{y}(k+p-1) \end{bmatrix} \tag{7.12}$$

the following TV causal IOSR of order p can be obtained

$$\boldsymbol{x}(k) = \tilde{\mathbf{T}}_{p,c}(k)\tilde{\boldsymbol{z}}_{p,c}(k) \tag{7.13}$$

where the superstate $\tilde{\boldsymbol{z}}_{p,c}(k)$ is defined as

$$\tilde{\boldsymbol{z}}_{p,c}(k) = \begin{bmatrix} u(k-p) & u(k-p+1) & \dots \end{bmatrix} \tag{7.14}$$

$$\begin{bmatrix} \dots & u(k-1) & \boldsymbol{y}^{\mathrm{T}}(k-p) & \boldsymbol{y}^{\mathrm{T}}(k-p+1) & \dots & \boldsymbol{y}^{\mathrm{T}}(k-1) \end{bmatrix}^{\mathrm{T}} \tag{7.15}$$

The TV causal IOSR of general order p is valid provided that at every time step k there exists an interaction matrix $\mathbf{M}_{p,c}(k)$ such that $\tilde{\mathbf{A}}_{p,c}(k)+\mathbf{M}_{p,c}(k)\tilde{\mathbf{C}}_{p,c}(k) = \mathbf{0}$. The necessary and sufficient condition for the existence of such $\mathbf{M}_{p,c}(k)$ is that the matrix $\tilde{\mathbf{C}}_{p,c}(k)$ has rank equal to n at every k. $\tilde{\mathbf{C}}_{p,c}(k)$ is the observability matrix of the LTV system in equation (7.1) and it contains the input values $u(k-p), u(k-p+1), \dots, u(k-2)$. Indeed, the observability of a bilinear system depends on the input as pointed out in Reference 62. The necessary and sufficient condition required for the IOSR in equation (7.13) to hold is then that the bilinear system to be identified is observable under the sequence of input used to excite it. As a particular case, note that when the linear part of the bilinear system is observable, i.e. $\mathbf{A}$ and $\mathbf{C}$ form an observable pair, the above condition is met independently of the input.

As a result, for an observable system driven by a sufficiently rich input, it is necessary that $pq \geq n$ for the IOSR in equation (7.13) to be valid. Note that when $n = q$, p can be as low as 1.

From equations (7.7c) and (7.10c) it is clear that the matrix relating the superstate and the state is not constant. The time dependence of the system matrix $\mathbf{A}(k)$ makes the above IOSR matrix $\tilde{\mathbf{T}}_{p,c}$ time-varying, from which the name *TV IOSR*. It is worth remarking that the interaction matrix $\mathbf{M}_{p,c}(k)$ is time-varying, too, because at each time step the pair $\big(\tilde{\mathbf{A}}_{p,c}(k), \tilde{\mathbf{C}}_{p,c}(k)\big)$ takes a different value and therefore requires a different interaction matrix to make $\tilde{\mathbf{A}}_{p,c}(k) + \mathbf{M}_{p,c}(k)\tilde{\mathbf{C}}_{p,c}(k) = \mathbf{0}$. The time dependence of $\tilde{\mathbf{T}}_{p,c}(k)$ is then due to both $\mathbf{A}(k)$ and $\mathbf{M}_{p,c}(k)$ appearing in its definition and $\tilde{\mathbf{T}}_{p,c}(k)$ turns out to be a function of the ordered sequence $(u(k-p+1), u(k-p+2), \ldots, u(k))$.

7.2.2 TV anticausal IOSRs

The anticausal version of the TV causal IOSR can be derived in a similar fashion. Rewrite equation (7.1) as

$$\boldsymbol{x}(k) = \mathbf{A}^{-1}(k)\boldsymbol{x}(k+1) - \mathbf{A}^{-1}(k)\mathbf{B}u(k) \tag{7.16}$$

and propagate it backward by plugging

$$\boldsymbol{x}(k+1) = \mathbf{A}^{-1}(k+1)\boldsymbol{x}(k+2) - \mathbf{A}^{-1}(k+1)\mathbf{B}u(k+1) \tag{7.17}$$

into equation (7.16)

$$\begin{aligned}\boldsymbol{x}(k) &= \mathbf{A}^{-1}(k)\big(\mathbf{A}^{-1}(k+1)\boldsymbol{x}(k+2) - \mathbf{A}^{-1}(k+1)\mathbf{B}u(k+1)\big) - \mathbf{A}^{-1}(k)\mathbf{B}u(k) \\ &= \mathbf{A}^{-1}(k)\mathbf{A}^{-1}(k+1)\boldsymbol{x}(k+2) - \mathbf{A}^{-1}(k)\mathbf{A}^{-1}(k+1)\mathbf{B}u(k+1) - \mathbf{A}^{-1}(k)\mathbf{B}u(k)\end{aligned} \tag{7.18}$$

Similar to equation (7.5), we can write (backward)

$$\begin{bmatrix} \boldsymbol{y}(k+2) \\ \boldsymbol{y}(k+1) \end{bmatrix} = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A}^{-1}(k+1) \end{bmatrix} \boldsymbol{x}(k+2) + \begin{bmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & -\mathbf{C}\mathbf{A}^{-1}(k+1)\mathbf{B} + \mathbf{D} \end{bmatrix} \begin{bmatrix} u(k+2) \\ u(k+1) \end{bmatrix} \tag{7.19}$$

Adding and subtracting $\mathbf{M}_{2,a}(k) \begin{bmatrix} \boldsymbol{y}(k+2) \\ \boldsymbol{y}(k+1) \end{bmatrix}$ in equation (7.16) and, plugging equation (7.19) in it, get

$$\begin{aligned} \boldsymbol{x}(k) = {} & \mathbf{A}^{-1}(k)\mathbf{A}^{-1}(k+1)\boldsymbol{x}(k+2) - \begin{bmatrix} \mathbf{A}^{-1}(k)\mathbf{A}^{-1}(k+1)\mathbf{B} & \mathbf{A}^{-1}(k)\mathbf{B} \end{bmatrix} \begin{bmatrix} u(k+1) \\ u(k) \end{bmatrix} \\ & + \mathbf{M}_{2,a}(k) \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A}^{-1}(k+1) \end{bmatrix} \boldsymbol{x}(k+2) \\ & + \mathbf{M}_{2,a}(k) \begin{bmatrix} \mathbf{D} & \mathbf{0} \\ \mathbf{0} & -\mathbf{C}\mathbf{A}^{-1}(k+1)\mathbf{B} + \mathbf{D} \end{bmatrix} \begin{bmatrix} u(k+2) \\ u(k+1) \end{bmatrix} - \mathbf{M}_{2,a}(k) \begin{bmatrix} \boldsymbol{y}(k+2) \\ \boldsymbol{y}(k+1) \end{bmatrix} \\ = {} & \Big(\tilde{\mathbf{A}}_{2,a}(k) + \mathbf{M}_{2,a}(k)\tilde{\mathbf{C}}_{2,a}(k) \Big) \boldsymbol{x}(k+2) + \tilde{\mathbf{T}}_{2,a}(k)\tilde{\boldsymbol{z}}_{2,a}(k) \end{aligned} \tag{7.20}$$

where

$$\tilde{\mathbf{A}}_{2,a}(k) = \mathbf{A}^{-1}(k)\mathbf{A}^{-1}(k+1) \qquad \tilde{\mathbf{C}}_{2,a}(k) = \begin{bmatrix} \mathbf{C} \\ \mathbf{C}\mathbf{A}^{-1}(k+1) \end{bmatrix} \tag{7.21a}$$

$$\tilde{\boldsymbol{z}}_{2,a}(k) = \begin{bmatrix} u(k+2) & u(k+1) & u(k) & \boldsymbol{y}^{\mathrm{T}}(k+2) & \boldsymbol{y}^{\mathrm{T}}(k+1) \end{bmatrix}^{\mathrm{T}} \tag{7.21b}$$

$$\begin{aligned} \tilde{\mathbf{T}}_{2,a}(k) = {} & \Big[- \begin{bmatrix} \mathbf{0} & \mathbf{A}^{-1}(k)\mathbf{A}^{-1}(k+1)\mathbf{B} & \mathbf{A}^{-1}(k)B \end{bmatrix} \\ & + \mathbf{M}_{2,a}(k) \begin{bmatrix} \mathbf{D} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & -\mathbf{C}\mathbf{A}^{-1}(k+1)\mathbf{B} + \mathbf{D} & \mathbf{0} \end{bmatrix} - \mathbf{M}_{2,a}(k) \Big] \end{aligned} \tag{7.21c}$$

and the subscript a stands for *anticausal* to indicate that the superstate $\tilde{\boldsymbol{z}}_{2,a}(k)$ depends only on current and future input-output data.

Again, if the first term in equation (7.20) cancels for any $\boldsymbol{x}(k+2)$, a time-varying IOSR analogous to equation (7.8) is obtained.

Equation (7.20) refers to $p = 2$. Propagating equation (7.16) backward by $p-1$ time steps and following the same approach as for the TV causal IOSR derivation allows us to generalize equation (7.20) to any p. We obtain

$$\boldsymbol{x}(k) = \big(\tilde{\mathbf{A}}_{p,a}(k) + \mathbf{M}_{p,a}(k)\tilde{\mathbf{C}}_{p,a}(k)\big)\boldsymbol{x}(k+2) + \tilde{\mathbf{T}}_{p,a}(k)\tilde{\boldsymbol{z}}_{p,a}(k) \tag{7.22}$$

which reduces to

$$\boldsymbol{x}(k) = \tilde{\mathbf{T}}_{p,a}(k)\tilde{\boldsymbol{z}}_{p,a}(k) \tag{7.23}$$

if $\tilde{\mathbf{A}}_{p,a}(k) + \mathbf{M}_{p,a}(k)\tilde{\mathbf{C}}_{p,a}(k) = 0$. The superstate $\tilde{\boldsymbol{z}}_{p,a}(k)$ is defined as

$$\begin{aligned}\tilde{\boldsymbol{z}}_{p,a}(k) = \big[&u(k+p) \quad u(k+p-1) \quad \dots \\ &\dots \quad u(k) \quad \boldsymbol{y}^{\mathrm{T}}(k+p) \quad \boldsymbol{y}^{\mathrm{T}}(k+p-1) \quad \dots \quad \boldsymbol{y}^{\mathrm{T}}(k+1)\big]^{\mathrm{T}}\end{aligned} \tag{7.24}$$

Similar reasoning to the one for the causal case applies to the TV anticausal IOSR. For equation (7.23) to hold, $\tilde{\mathbf{C}}_{p,a}(k)$ has to be of rank n for any k, which for observable bilinear systems is guaranteed by choosing p such that $pq \geq n$. Again, it is worth remarking that the IOSR matrix $\tilde{\mathbf{T}}_{p,a}(k)$ relating the superstate and the state is not constant due to the presence of $\mathbf{A}(k)$ and $\mathbf{M}_{p,a}(k)$ in its definition. It is a function of the ordered sequence $(u(k), u(k+1), \dots, u(k+p-1))$.

7.2.3 Comparison with TI IOSRs

The TI IOSRs presented in Reference 50 have some drawbacks, as summarized below:

1. The TI IOSRs are exact only for a class of bilinear system (referred to as *ideal* in chapter 6) defined by very stringent conditions

2. For *non-ideal* bilinear systems the TI IOSRs converge asymptotically with p to exact IOSRs, provided that $|u(k)| < \gamma$ at every time step k (where γ is a positive constant unknown before identification) and that the linear part of the system is observable for any arbitrary bilinear system

3. The dimension of the superstate of the TI IOSRs grows exponentially with its order p

In contrast, the TV IOSRs are exact for any observable bilinear systems. Note that observability is the minimum requirement to identify any dynamic system. On top of being exact also for *non-ideal* systems, the bilinear observability condition necessary for the TV IOSRs is less stringent than the observability of the linear part and, most importantly, no bound on the magnitude of the excitation input is necessary. Additionally, from equations (7.14) and (7.24), the dimension of the TV IOSR superstates $\tilde{\boldsymbol{z}}_{p,c}$ and $\tilde{\boldsymbol{z}}_{p,a}$ grows linearly with p, avoiding the computational difficulties arising with the method of Reference 50 for systems of large order.

In conclusion, the TV IOSRs developed in this chapter provide a better ground for bilinear system identification. However, in order to be able to use the TV IOSRs in the ELM or IS identification methods (section 6.3), the time dependence of $\tilde{\mathbf{T}}_{p,c}$ (and $\tilde{\mathbf{T}}_{p,a}$) must be eliminated. A form like equation (6.3), with constant $\mathbf{T}$, is needed and in the next section it is shown how to transform the TV IOSRs of equations (7.13) and (7.23) into such form.

7.3 Input design

As already noticed, $\tilde{\mathbf{T}}_{p,c}(k)$ depends on the values that the ordered sequence $\big(u(k-p), u(k-p+1), \ldots, u(k-1)\big)$ takes over time. We can then think of designing the excitation input sequence so that $\tilde{\mathbf{T}}_{p,c}(k)$ changes over time in a *convenient* way. Note that the reasoning and analysis in this section are done with reference to to the TV causal IOSR but they are applicable to the TV anticausal IOSR in the same exact fashion and with analogous benefit.

7.3.1 From time-varying to time-invariant T

The first requirement that the excitation input sequence has to satisfy is to allow us to transform the TV IOSR of equation (7.13) into the constant-$\mathbf{T}$ form of equation (6.3). This can be done by limiting the values that $u(k)$ can take to a finite set $U = \{u_1, u_2, \ldots, u_L\}$. As a consequence $\mathbf{A}(k)$ can take L possible matrix values and the number of possible ordered sequences $\big(\mathbf{A}(k-p), \ldots, \mathbf{A}(k-1)\big)$, which uniquely determine $\tilde{\mathbf{T}}_{p,c}(k)$, is finite. The last observation allows us to construct the following finite-dimensional extended matrix $\boldsymbol{\mathcal{T}}_{p,c}$ and extended superstate $\boldsymbol{\zeta}_{p,c}(k)$

$$\boldsymbol{\mathcal{T}}_{p,c} = \begin{bmatrix} \tilde{\mathbf{T}}_{p,c}^{(1)} & \tilde{\mathbf{T}}_{p,c}^{(2)} & \ldots & \tilde{\mathbf{T}}_{p,c}^{(N)} \end{bmatrix} \qquad \boldsymbol{\zeta}_{p,c}(k) = \begin{bmatrix} \mathbf{0} \\ \vdots \\ \mathbf{0} \\ \tilde{\boldsymbol{z}}_{p,c}(k) \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} \tag{7.25}$$

$\tilde{\boldsymbol{z}}_{p,c}(k)$ is placed in the j^{th} block row of $\boldsymbol{\zeta}_{p,c}(k)$, where j is the index of the actual matrix value that $\tilde{\mathbf{T}}_{p,c}$ takes at time k (among the $\mathbf{N}$ possible values), and all the other block rows are zeros (each block row has as many entries as the dimension of the superstate $\tilde{\boldsymbol{z}}_{p,c}(k)$). The following relationship formally equal to equation (6.3) can now be written

$$\boldsymbol{x}(k) = \boldsymbol{\mathcal{T}}_{p,c}\boldsymbol{\zeta}_{p,c}(k) \tag{7.26}$$

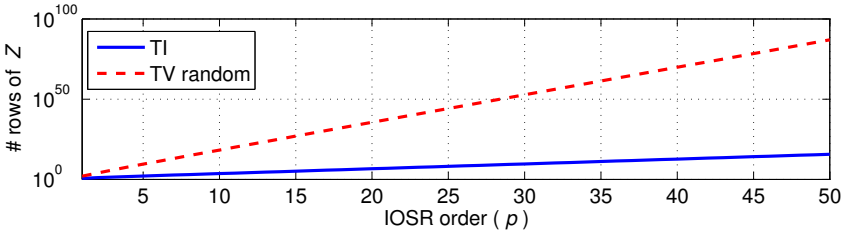
and the same technique can be used to derive the anticausal version of the IOSR of equation (7.26)

$$\boldsymbol{x}(k) = \boldsymbol{\mathcal{T}}_{p,a}\boldsymbol{\zeta}_{p,a}(k) \tag{7.27}$$

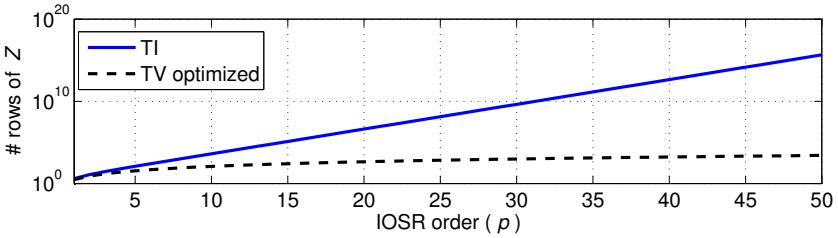
Equations (7.26) and (7.27) are formally two time-invariant IOSRs but in this chapter we continue referring to the them as TV IOSRs to emphasize that they directly stem from the time-varying IOSRs of equations (7.13) and (7.23). $\boldsymbol{\zeta}_{p,c}(k)$ and $\boldsymbol{\zeta}_{p,a}(k)$ can then be used to construct the corresponding input $\boldsymbol{w}(k)$ to the ELM or the superspace matrices $\mathbf{Z}_a$ and $\mathbf{Z}_b$ for the IS method. Since such matrices are highly-structured and contain many zeroes, a preliminary SVD is usually very effective for dimension reduction, i.e.

$$\mathbf{TZ} = \mathbf{T}\begin{bmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{bmatrix}\begin{bmatrix}\mathbf{S} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{V}_1 & \mathbf{V}_2\end{bmatrix}^{\mathrm{T}} = \mathbf{TU}_1\mathbf{SV}_1^{\mathrm{T}} = \mathbf{T}_{red}\mathbf{Z}_{red} \tag{7.28}$$

where $\mathbf{T}_{red} = \mathbf{TU}_1\mathbf{S}$, $\mathbf{Z}_{red} = \mathbf{V}_1^{\mathrm{T}}$, $\mathbf{S}$ is a diagonal matrix with the non-zero singular values of $\mathbf{Z}$ and $\mathbf{U}_1$ and $\mathbf{V}_1$ are the corresponding left and right singular vectors, whereas $\mathbf{U}_2$ and $\mathbf{V}_2$ are the singular vectors associated with the zero singular values of $\mathbf{Z}$. Both $\mathbf{Z}_a$ and $\mathbf{Z}_b$ can be decomposed as in equation (7.28) and replaced by the corresponding $\mathbf{Z}_{red}$ in the IS method. A similar procedure is beneficial in the ELM method, too, in order to reduce the number of inputs to the ELM.



(A) TV causal IOSR derived from 2-level random input.



(B) TV causal IOSR derived from optimal pulsed input.

FIGURE 7.1: Dimension increase for the TI causal IOSR and for the TV causal IOSR with random and optimized input (single-output system).

If the input value at any time step is randomly drawn from the set U, the number of possible matrix values that $\tilde{\mathbf{T}}_{p,c}(k)$ can take is maximum, $N = L^p$. The dimension of the extended state $\boldsymbol{\zeta}_{p,c}$ is then $(q+1)pL^p$. As shown in Figure 7.1a, even choosing the minimum possible L (i.e. $L = 2$), the resulting growth in size of the extended superstate $\boldsymbol{\zeta}_{p,c}$ of the TV causal IOSR is actually faster than the growth of the superstate of the TI causal IOSR. Nevertheless, the exact nature of the superstate $\boldsymbol{\zeta}_{p,c}$ allows the system identification engineer to select a smaller value of p with respect to the one that would be chosen if using TI IOSRs. The following example shows how the proposed algorithm indeed leads to exact identification.

Example 1. Consider the following arbitrary bilinear system from chapter 6

$$\mathbf{A} = \begin{bmatrix} 0 & 0.5 \\ 0.5 & -0.5 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \qquad \mathbf{N} = \begin{bmatrix} 0.3 & 1 \\ -1 & 1 \end{bmatrix} \qquad \mathbf{C} = \begin{bmatrix} 0 & 1 \end{bmatrix} \qquad D = 0 \quad (7.29)$$

and generate an input sequence (and the corresponding output) by randomly drawing at each time step k a value from the set $U = \{0, 0.2\}$, $k = 0, 1, \ldots, 1000$. Applying the ELM method

TABLE 7.1: Identification of the system of example 1 by random 2-level excitation.

	p	**A** eigenvalue 1	**A** eigenvalue 2	**N** eigenvalues
True	-	-0.8090169943749475	0.3090169943749474	0.6499999999999999 ± 0.9367496997597599i
ELM	2	-0.8090169943749478	0.3090169943749524	0.6499999999999947 ± 0.9367496997597534i
	3	-0.8090169943749496	0.3090169943749490	0.6500000000000010 ± 0.9367496997597594i
	4	-0.8090169943749462	0.3090169943749483	0.6499999999999801 ± 0.9367496997597613i
IS	2	-0.8090169943749470	0.3090169943749478	0.6499999999999941 ± 0.9367496997597653i
	3	-0.8090169943749458	0.3090169943749453	0.6500000000000055 ± 0.9367496997597589i
	4	-0.8090169943749492	0.3090169943749483	0.6499999999999946 ± 0.9367496997597634i

based on the TV causal IOSR of equation (7.26) and the IS method based on the TV causal IOSR of equation (7.26) and the TV anticausal IOSR of equation (7.27), the results summarized in Table 7.1 are obtained. In all cases the identified order is correct ($n_{id} = 2$) and the output prediction error, verified on the response to a random unconstrained (i.e. $u(k)$ not restricted to take values from U only) input sequence independently generated from the one used for identification, is close to Matlab® numerical zero. Notice that the identification is exact both when p is chosen to be equal to its minimum theoretical value ($p = 2$, since $n = 2$ and $q = 1$) and when $p > 2$. This is crucial for any system identification algorithm, since the true order n of the system is usually unknown before identification. It is worth mentioning here that if 0 is not included in the allowed input levels, the input richness condition turns out not to be satisfied for the IS method and leads to an identified model of larger order (an additional step then has to be performed to reduce the model and recover the correct bilinear system matrices). It is remarkable how the proposed method can achieve very accurate identification with reduced computational effort. In Reference 50, to achieve an output prediction error of order 10^{-7} with the IS method, it was necessary to know the true order of the system ($n = 2$) and increase the order of the causal IOSR to $p = 6$, with a corresponding number of columns of 253 for $\mathbf{R}$ in equation (6.13b). When using the TV IOSRs, the identification is exact (prediction error of 10^{-14}) and, if $p = 2$, $\mathbf{R}$ has only 36 columns (28 if the preliminary reduction of equation (7.28) is performed).

7.3.2 Input optimization

Having shown the feasibility and correctness of the LTV approach for bilinear system identification, we now propose a technique to minimize,for a given value of p, the extended superstate dimension while preserving the excitation richness needed for the identification. It is worth remarking here how the desired characteristics of the input sequence may vary from application to application. Different strategies can be adopted to address the trade-off between input applicability and computational efficiency, in which the a priori belief of the system order may play a major role. The optimized input sequence proposed below as an example aims to computational efficiency.

The chosen form of input resembles a sequence of pulses and is shown in Figure 7.2, where u_h is the amplitude of the pulses, and $\Delta_h^{(i)}$ and $\Delta_l^{(i)}$ refer to the duration of the nonzero and zero input application at the i^{th} pulse. u_h is a fixed (constant) value, whereas $\Delta_h^{(i)}$ and $\Delta_l^{(i)}$ are in general random discrete variables, whose possible (integer) realizations are constrained as follows

$$h_{min} \leq \Delta_h^{(i)} \leq h_{max} \tag{7.30}$$

$$l_{min} \leq \Delta_l^{(i)} \leq l_{max} \tag{7.31}$$

In other words, the excitation is a sequence of multiple pulses of fixed amplitude and different (random) duration, and its design parameters are u_h, h_{min}, h_{max}, l_{min} and l_{max}. Note that such input form satisfies the conditions established in Reference 63 for continuous-time bilinear system identifiability.

The next step consists in the choice of the input design parameters in order to minimize the dimension of $\boldsymbol{\zeta}_{p,c}$. First notice from equations (7.14) and (7.25) that the dimension of the

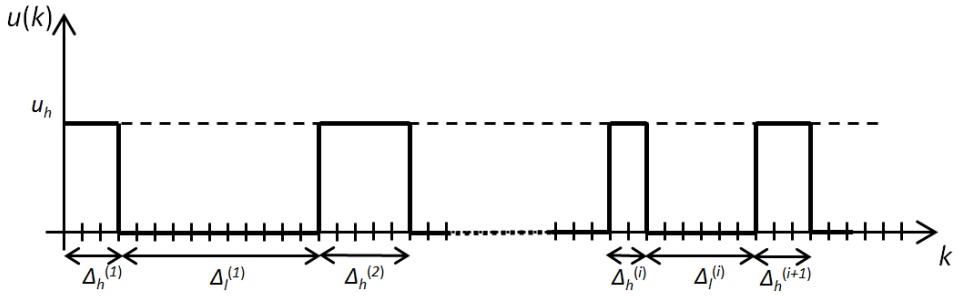


FIGURE 7.2: Input form for optimal excitation input design.

superstate is $(q+1)pN$, where $\mathbf{N}$ is the number of combinations that any sequence of length p $\big(u(p-k), u(p-k+1), \ldots, u(k-1)\big)$ can take. With reference to Figure 7.2, it is then convenient to have $\Delta_l \geq p$ (i.e. fix $l_{min} \geq p$) so that each input sequence of length p embraces at most one pulse, significantly reducing the number of possible combinations. To push the reduction of $\mathbf{N}$ further, one can also impose $\Delta_h = 1$ (i.e. $h_{min} = h_{max} = 1$) and get $N = p+1$, leaving all the variability of the pulses to the duration of the zero-input portions, i.e. Δ_l. Having fixed $l_{min} \geq p$, the only parameter to be arbitrarily chosen is then l_{max}, which does not have an impact on $\mathbf{N}$. The resulting number of rows for the extended superstate $\boldsymbol{\zeta}_{p,c}$ is then quadratic in p

$$n_{rows}(\boldsymbol{\zeta}_{p,c}) = (q+1)(p+1)p \tag{7.32}$$

A closer look at equations (7.25) and (7.14) suggests a further reduction can immediately be obtained, even though the growth rate remains quadratic in p. Due to the low level of the pulse sequence being zero, even the j^{th} block row of $\boldsymbol{\zeta}_{p,c}$ in equation (7.25) has some entries always equal to zero. With $\Delta_h = 1$, among all the possible j^{th} block rows of $\boldsymbol{\zeta}_{p,c}$, p contain $p-1$ entries equal to 0, and one has p entries equal to 0. Hence, equation (7.32) becomes

$$n_{rows}(\boldsymbol{\zeta}_{p,c}) = qp^2 + (q+1)p \tag{7.33}$$

Equation (7.33) represents the maximum number of rows that can remain after performing the

TABLE 7.2: Identification of the system of example 2 by multiple-pulse excitation.

	p	**A** eigenvalue 1	**A** eigenvalue 2	**N** eigenvalues
True	-	-0.8090169943749475	0.3090169943749474	0.6499999999999999 ± 0.9367496997597599i
ELM	2	-0.8090169943749465	0.3090169943749516	0.6499999999999837 ± 0.9367496997597706i
	3	-0.8090169943749496	0.3090169943749462	0.6500000000000312 ± 0.9367496997598526i
	10	-0.8090169943749470	0.3090169943749515	0.6499999993273293 ± 0.9367497010332932i
IS	2	-0.8090169943749468	0.3090169943749468	0.6499999999999113 ± 0.9367496997597923i
	3	-0.8090169943749480	0.3090169943749460	0.6500000000011040 ± 0.9367496997602179i
	10	-0.8090169943749492	0.3090169943749430	0.6499999993267873 ± 0.9367496981421699i

preliminary SVD of equation (7.28), which can result in an even further reduction depending on the actual realization of the input-output sequence.

Figure 7.1b shows how the maximum number of rows of $\boldsymbol{\zeta}_{p,c}$ from equation (7.33) grows with p significantly slower when the optimized input is used. The following example also shows that the identification is still exact.

Example 2. Consider again the bilinear system of equation (7.29) and evaluate its response when subject to an input as in Figure 7.2 with $u_h = 0.2$, $\Delta_h = 1$, $p \leq \Delta_l \leq p + 2$ ($k = 0, 1, \ldots, 1000$). Table 7.2 summarizes the results when applying the ELM method based on the TV causal IOSR of equation (7.26) and the IS method based on the TV causal IOSR of equation (7.26) and the TV anticausal IOSR of equation (7.27). In all cases the identified order is correct and the output prediction error, verified on the response to a completely random input sequence, is close to numerical zero. Notice again that the identification is exact even when the IOSR order p is chosen larger than the true system order n. It is worth mentioning that even for a relatively large value of p, for example $p = 10$, the number of rows in the matrix $\mathbf{Z}$ constructed from $\boldsymbol{\zeta}_{p,c}$ is 220, it goes down to 120 when the all-zero rows are eliminated, and it decreases to 32 after the preliminary SVD is performed. In contrast, the number of rows of the matrix $\mathbf{Z}$ constructed from $\boldsymbol{\zeta}_{p,c}$ if the input was randomly chosen between two levels would be 20,480 and it would reach 4,194,300 if $\mathbf{Z}$ was built from the TI IOSR superstate of Reference 50.

7.4 Discussion

The TV IOSRs are developed in this chapter to overcome the drawbacks of the TI IOSRs presented in chapter 6, as previously explained in details (exact for a class of bilinear systems, approximated for the others, subject to a constraint on the excitation input magnitude, affected by curse of dimensionality). The solution offered by the TV IOSRs comes at the price of the need for a somewhat structured input, which was the main motivation for the development of the identification methods in chapter 6. The latter are indeed applicable on data coming from a single experiment, in contrast for example with Reference 11, and without any restriction on the form that the excitation input can take, marking a significant improvement over the sinusoidal input needed in Reference 49 and the sequence of multiple pulses at the core of Reference 13. With respect to linear system identification, it is interesting to note how the bilinear case seems to require some specialized input to overcome the curse of dimensionality. Even the approach taken in References 11, 13, 14 for continuous-time bilinear system identification is that of a highly-structured input, with some interesting similarities to what proposed in this chapter, for example the use of multiple pulses in Reference 13. It is however important to note how the use of multiple-pulse excitation is a necessary requirement for the algorithm in Reference 13, whereas for the methods proposed in this chapter it is just one possible choice (recommended when computational issues may arise).

It is noteworthy how IOSR-based methods such as ELM and IS provide a unified framework for discrete-time bilinear system identification. Different IOSRs can be used to implement the ELM or IS methods according to the needs that different applications may have. The pros and cons of the various IOSRs have been discussed in a previous section. In what follows, an intuitive interpretation of the role of IOSRs in system identification is provided.

The IOSRs presented in References 33, 50 are derived via interaction matrices. As discussed in detail in chapter 8, the interaction matrices of the TI IOSRs can be interpreted as gains of an observer (state estimator) for the bilinear model to be identified. In the absence of noise, for *ideal* bilinear models the resulting observer is deadbeat and leads to exact IOSRs and in turn exact identification algorithms. For arbitrary bilinear systems the observer is not deadbeat but is convergent (indeed it is the fastest possible bilinear observer) and makes the identification algorithms asymptotically exact. The TV IOSRs presented in this chapter can still be interpreted as state estimators. More specifically, they are time-varying deadbeat observers, providing the exact estimate of the bilinear state in a finite number of steps p, i.e. only based on the values that the input and the output took over the past (in the case of causal TV IOSR) p time steps. The interaction matrix $\mathbf{M}_{p,c}(k)$ plays then the role of a time-varying gain. The underlying approach is therefore the same that led to the development of OKID (Reference 20). Although there are some common ideas (e.g. exploiting the LTV nature of bilinear systems via structured excitation), the identification algorithms proposed in this chapter are then very different from the approach taken in References 11, 13, 14, which is based on the estimation of the observability matrix from a properly constructed data matrix and is therefore closer to subspace methods rather than OKID.

7.5 Identification of Carleman bilinear models of nonlinear systems

The examples provided in this section serve for multiple purposes. First of all, they concern the identification of nonlinear dynamic systems which are well-known in the literature and in applications. The first example on Duffing's equation shows how the proposed IOSRs allow the IS method to identify with great accuracy even bilinear systems of large order ($n = 9$

and $n = 14$). The second examples is on Euler's equations and illustrates how the proposed identification algorithms can be applied to Multiple-Input-Multiple-Output (MIMO) systems and how the condition $q = n$ allows one to choose the excitation input with much more freedom than the multiple-pulse form in Figure 7.2, without incurring in computational issues. Duffing's equations and Eulers' equations are also used as examples of identification of discrete-time bilinear models from data generated by continuous-time (input-affine) nonlinear models. The challenge here is two-fold, since not only are the identified models of large order, but also the data fed to the identification algorithm do not come from discrete-time bilinear systems. The identification algorithm then works on data which are affected by modeling error.

7.5.1 Duffing's equation

Duffing's equation describes the dynamics of a mass-spring oscillator with cubic spring, and is defined as

$$\ddot{y}_D(t) + c\dot{y}_D(t) + by_D^3(t) + ay_D(t) = u(t) \tag{7.34}$$

Equation (7.34) is first bilinearized by the Carleman technique to obtain a continuous-time bilinear model in the form

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}_c\boldsymbol{x}(t) + \mathbf{N}_c\boldsymbol{x}(t)u(t) + \mathbf{B}_c u(t) \tag{7.35a}$$

$$\boldsymbol{y}(t) = \mathbf{C}_c\boldsymbol{x}(t) + \mathbf{D}_c u(t) \tag{7.35b}$$

which is then discretized by one of the methods presented in Reference 64 to obtain a bilinear model in the form of equation (6.1). Defining the primary state variables of equation (7.34) as

$$x_1 = y_D \qquad\qquad x_2 = \dot{y}_D \tag{7.36}$$

Carleman linearization (References 8, 9) introduces a state vector made of progressively higher-order products of the primary state variables. For instance, the second-order Carleman state vector involves quadratic terms

$$\boldsymbol{x} = \begin{bmatrix} x_1 & x_2 & x_1^2 & x_1 x_2 & x_2^2 \end{bmatrix}^{\mathrm{T}} \tag{7.37}$$

and the third-order state vector introduces cubic terms

$$\boldsymbol{x} = \begin{bmatrix} x_1 & x_2 & x_1^2 & x_1 x_2 & x_2^2 & x_1^3 & x_1^2 x_2 & x_1 x_2^2 & x_2^3 \end{bmatrix}^{\mathrm{T}} \tag{7.38}$$

For instance, the state-space matrices of the 3$^{\mathrm{rd}}$-order Carleman model of Duffing's equation are

$$\mathbf{A}_c = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -a & -c & 0 & 0 & 0 & -b & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -a & -c & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2a & -2c & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -a & -c & 2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2a & -2c & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -3a & -3c \end{bmatrix} \tag{7.39a}$$

$$\mathbf{N}_c = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 3 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{B}_c = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{7.39b}$$

$$\mathbf{D}_c = \mathbf{0} \tag{7.39c}$$

Using Euler's discretization method (Reference 64) with step size Δt, the discrete-time bilinear model matrices are

$$\mathbf{A} = \mathbf{I} + \mathbf{A}_c \Delta t \quad \mathbf{N} = \mathbf{N}_c \Delta t \quad \mathbf{B} = \mathbf{B}_c \Delta t \quad \mathbf{D} = \mathbf{D}_c \tag{7.40}$$

In order to simulate a hardening spring, the Duffing coefficients in this example are chosen to be $a = c = 1$ and $b = 0.01$. The sampling time is $\Delta t = 0.002$. The objective is to identify the matrices $\mathbf{A}$, $\mathbf{N}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ of equation (7.40), given input-output data. The excitation input used for identification is of the optimized type presented in Figure 7.2, with $u_h = 2$, $\Delta_h \in \{0.02, 0.04\}$ and $\Delta_l \in \{0.02, 0.04\}$. To verify the accuracy of the identified bilinear model, its predicted output is then compared with the output of the true system when both are driven by the same sequence of input, independently generated from the one used for identification. The input used for verification is plotted in Figure 7.3, whereas Figure 7.4 shows the resulting output of the true Duffing's equation and of its bilinear approximations derived by 3^{rd}- and 4^{th}-order Carleman

linearization and discretized by Euler's method. Figure 7.4 also reports the output of the linear approximation to Duffing's equation, to show how the bilinear approximation is more accurate and therefore attractive.

Assume that the variables x_1, x_1^2, x_1^3 are measured, i.e.

$$\mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{7.41}$$

When the output is generated by the discrete-time Carleman model of equations (7.40) and (7.41), the 3$^{\text{rd}}$-order Carleman bilinear model is identified exactly by the IS method with TV IOSRs with $p = 10$, as demonstrated by the high accuracy in predicting the output of the true Carleman model (Figure 7.5) when both are driven by the input of Figure 7.3. Note that p has been

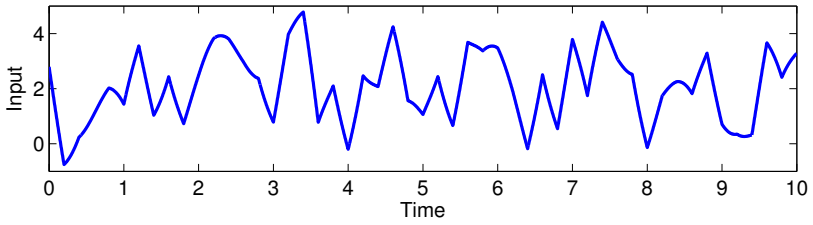


FIGURE 7.3: Input for the verification of the identified bilinear models.

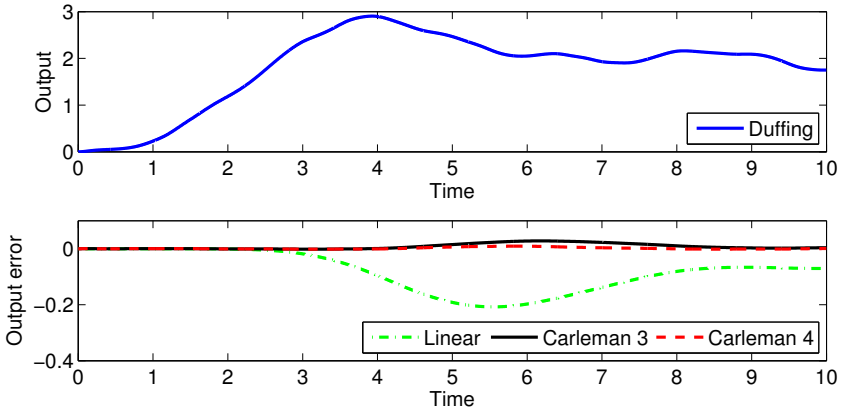


FIGURE 7.4: Output of the true Duffing's equation, its linear approximation and theoretical Carleman bilinear approximations.

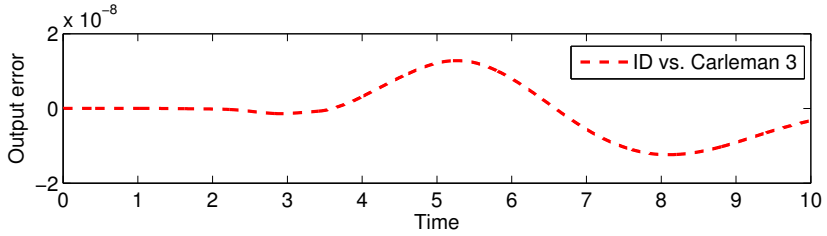


FIGURE 7.5: Prediction accuracy of the bilinear model identified from input-output data generated by the 3rd-order theoretical Carleman bilinear model of Duffing's equation.

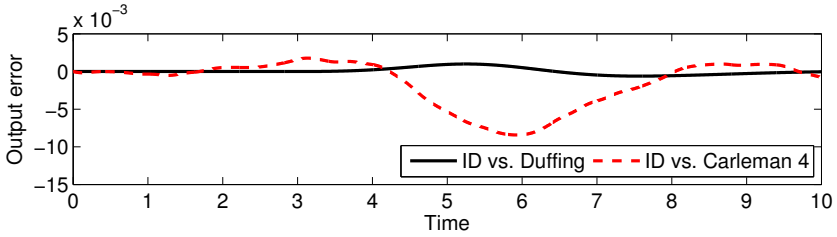


FIGURE 7.6: Prediction accuracy of the bilinear model identified from input-output data generated by the true Duffing's equation.

chosen pretending that the true order of the system is not known, and $p = 10$ is consistent with the assumption that the model can be of up to order 30 ($pq = 30$).

The above example illustrates how the proposed algorithms is capable of identifying relatively large bilinear models ($n = 9$). However, generally one is interested in finding a Carleman bilinear approximation of a nonlinear system, from data measured from the real system or, if the differential governing the system dynamics are known, from data numerically generated from such equations. In the following example, we generate data directly from equation (7.34) driven by and input sequence with $u_h = 2$, $\Delta_h \in \{0.02, 0.04\}$ and $\Delta_l \in \{0.02, 0.04\}$ (Figure 7.2). From the point of view of the identification algorithm, the input-output data are now affected by modeling error, making the identification more challenging. Figure 7.6 shows that the algorithm is still able to provide an excellent bilinear approximation of Duffing's equation. In the example of Figure 7.6, the output data provided to the identification algorithm is the sequence formed by

$$\boldsymbol{y}(k) = \begin{bmatrix} y_D(k\Delta t) & y_D^2(k\Delta t) & y_D^3(k\Delta t) & y_D^4(k\Delta t) \end{bmatrix} \tag{7.42}$$

and the identified bilinear model is therefore similar to a 4^{th}-order Carleman approximation ($n = 14$).

7.5.2 Euler's equations

Consider Euler's equations, which describe the rotation of a rigid body in a reference frame fixed to the rotating body and having axes coincident with the principal axes of inertia and represent a fundamental building block of the satellite attitude control problem

$$\begin{aligned} I_1\dot{\omega}_1(t) + (I_3 - I_2)\omega_2(t)\omega_3(t) &= \tau_1(t) \\ I_2\dot{\omega}_2(t) + (I_1 - I_3)\omega_3(t)\omega_1(t) &= \tau_2(t) \\ I_3\dot{\omega}_3(t) + (I_2 - I_1)\omega_1(t)\omega_2(t) &= \tau_3(t) \end{aligned} \tag{7.43}$$

where ω_i's are the angular velocities along the principal axes, I_i's are the principal moments of inertia and τ_i's are the applied torques, $i = 1, 2, 3$. Letting the driving torques be the sum of a feedback term $-b_i\omega_i$ and a feedforward term f_i

$$\tau_i = -b_i\omega_i + f_i \tag{7.44}$$

and defining

$$a_1 = \frac{I_3 - I_2}{I_1} \quad a_2 = \frac{I_1 - I_3}{I_2} \quad a_3 = \frac{I_2 - I_1}{I_3} \quad c_i = \frac{b_i}{I_i} \quad u_i = \frac{f_i}{I_i} \tag{7.45}$$

equation (7.43) becomes

$$\begin{aligned}\dot{\omega}_1(t) &= -a_1\omega_2(t)\omega_3(t) - c_1\omega_1(t) + u_1(t)\\ \dot{\omega}_2(t) &= -a_2\omega_3(t)\omega_1(t) - c_2\omega_2(t) + u_2(t)\\ \dot{\omega}_3(t) &= -a_3\omega_1(t)\omega_2(t) - c_3\omega_3(t) + u_3(t)\end{aligned} \tag{7.46}$$

Defining the Carleman state as

$$\boldsymbol{x} = \begin{bmatrix}\omega_1 & \omega_2 & \omega_3 & \omega_1^2 & \omega_2^2 & \omega_3^2 & \omega_1\omega_2 & \omega_1\omega_3 & \omega_2\omega_3\end{bmatrix}^{\mathrm{T}} \tag{7.47}$$

a 2^{nd}-order bilinear model is obtained

$$\dot{\boldsymbol{x}}(t) = \mathbf{A}_c\boldsymbol{x}(t) + \sum_{i=1}^{3}\mathbf{N}_{ci}\boldsymbol{x}(t)u_i(t) + \mathbf{B}_c\boldsymbol{u}(t) \tag{7.48a}$$

$$\boldsymbol{y}(t) = \mathbf{C}_c\boldsymbol{x}(t) + \mathbf{D}_c\boldsymbol{u}(t) \tag{7.48b}$$

where $\boldsymbol{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix}$ and

$$\mathbf{A}_c = - \begin{bmatrix} c_1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & a_1 \\ 0 & c_2 & 0 & 0 & 0 & 0 & 0 & a_2 & 0 \\ 0 & 0 & c_3 & 0 & 0 & 0 & a_3 & 0 & 0 \\ 0 & 0 & 0 & 2c_1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2c_2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 2c_3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & c_1 + c_2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_1 + c_3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & c_2 + c_3 \end{bmatrix} \tag{7.49a}$$

$$\mathbf{B}_c = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \mathbf{D}_c = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad \mathbf{N}_{c1} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{7.49b}$$

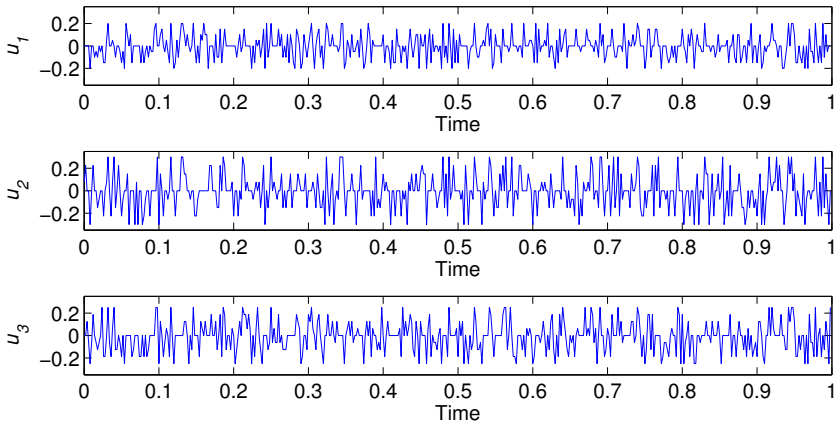


FIGURE 7.7: Excitation input for the identification of Euler's equations.

$$\mathbf{N}_{c2} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \mathbf{N}_{c3} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{7.49c}$$

By equation (7.40) the following discrete-time model with 3 inputs is obtained

$$\boldsymbol{x}(k+1) = \mathbf{A}\boldsymbol{x}(k) + \sum_{i=1}^{3} \mathbf{N}_i \boldsymbol{x}(k) u_i(k) + \mathbf{B}\boldsymbol{u}(k) \tag{7.50a}$$

$$\boldsymbol{y}(k) = \mathbf{C}\boldsymbol{x}(k) + \mathbf{D}\boldsymbol{u}(k) \tag{7.50b}$$

The presence of a vector input can be incorporated in the TV IOSRs just by noting that $\mathbf{A}(k) = \mathbf{A} + \mathbf{N}_1 u_1(k) + \mathbf{N}_2 u_2(k) + \mathbf{N}_3 u_3(k)$. The excitation input of Figure 7.7 generates the

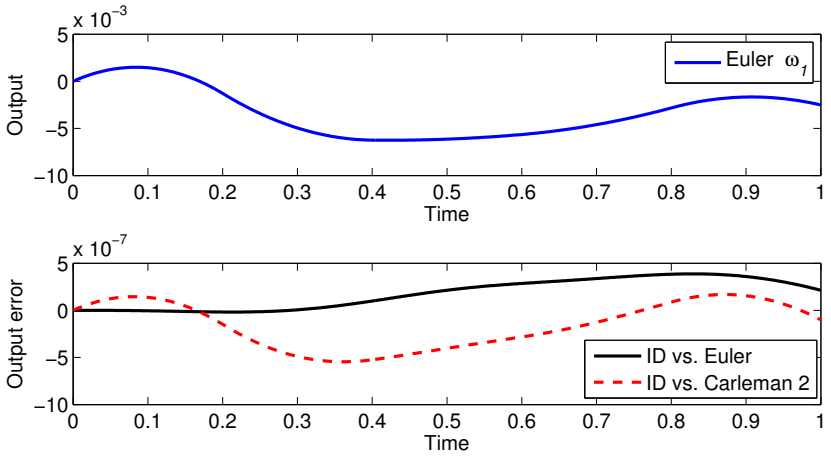


FIGURE 7.8: $\omega_1(t)$ of the true Euler's equations and comparison with its theoretical and identified 2nd-order Carleman bilinear approximation.

output history $\omega_1(t)$, $\omega_2(t)$, $\omega_3(t)$ in accordance with equation (7.46) assuming $I_1 = 30$, $I_2 = 20$, $I_3 = 10$, $b_1 = b_2 = b_3 = 3$ and sampling with $\Delta t = 0.002s$. Suppose that a bilinear model similar to the 2nd-order Carleman approximation is desired. Then the measurements fed to the identification algorithm can be chosen as

$$\begin{aligned}\boldsymbol{y}(k) = \big[&\omega_1(k) \quad \omega_2(k) \quad \omega_3(k) \quad \omega_1^2(k) \quad \omega_2^2(k) \quad \omega_3^2(k) \\ &\omega_1(k)\omega_2(k) \quad \omega_1(k)\omega_3(k) \quad \omega_2(k)\omega_3(k)\big]^{\mathrm{T}}\end{aligned} \tag{7.51}$$

i.e. corresponding to an identity $\mathbf{C}$ matrix in equation (7.50). Knowing that the desired approximated bilinear model is of order $n = 9$ and being $q = 9$, then p can be selected equal to 1. This gives more freedom in choosing the excitation input (Figure 7.7), without worrying about computational problems.

The identified bilinear model has the correct order $n_{id} = 9$ and accurately predicts the output of both the true Euler equations and their Carleman approximation of 2nd order. For verification, all the three models are driven by an input sequence independently generated from the one used for identification and as an example the comparison of their output ω_1 is shown in Figure 7.8.

7.6 Conclusions

This chapter has extended the family of input-output-to-state representations (IOSRs) for bilinear systems, i.e. relationships expressing the bilinear state as a linear combination of input-output data only. The new IOSRs feature crucial benefits over the ones available in the literature and those presented in chapter 6. In particular, they are exact for any arbitrary bilinear system satisfying minimum observability conditions and without imposing restrictions on the magnitude that the excitation input can take. These characteristics have been obtained by exploiting the linear-time-varying nature of bilinear systems. By designing appropriate excitation input, it has been shown how bilinear systems can be correctly identified with reduced computational complexity. The proposed input form is optimized with respect to computational complexity, but the identification method resulting from the new IOSRs is very flexible and can accommodate other input profiles.

The proposed algorithms can be applied to identify systems which are inherently bilinear as well as bilinear models approximating more general nonlinear systems. Numerical examples have been given for both cases, showing how the approach offers benefits over existing methods. In particular, the identification algorithms are exact for arbitrary bilinear systems with minimum observability requirements and the computational effort is significantly reduced to allow one to identify relatively large bilinear models. For these reasons, the proposed approach to bilinear discrete-time system identification can play an important role in the identification of high-order bilinear models to approximate more general nonlinear systems, paving the way for an effective way to handle more general identification and control problems.

Chapter 8

Bilinear observers

8.1 Introduction

Regardless of the technique used to bilinearize a nonlinear system, Carleman linearization or bilinear system identification, the resulting bilinear models have states whose direct measurement is not possible. Even in inherently bilinear phenomena, often not all the states are measured. Like for linear systems, for bilinear systems the use of some kind of state estimator (e.g. an observer) is then indispensable in order to design and implement state-feedback controllers.

References 65 and 66 studied the problem of minimal-order state observers for bilinear systems in continuous time. These results can be shown to apply also to discrete-time systems with obvious modifications (stable matrix when the eigenvalues lie within the unit circle instead of the left-half plane). The observability of bilinear systems is affected by the input (Reference 62), therefore the dynamics of the state estimation error generally depends on the input. Other theoretical contributions to the topic were made in References 67 and 68. Reference 65 formulated time-invariant minimal-order observers with bilinear structure and focused on those whose state

estimation error is independent of the input. Reference 65 found the necessary conditions on the structure of the bilinear system for such observers to exist. Even when they exist, their eigenvalues cannot in general be arbitrarily chosen. If they happen to be stable and sufficiently fast for the desired application, then the design is successful. When the bilinear system meets all the necessary conditions, the design procedure is simple and easily automated. Reference 66 investigated another design method leading to a minimal-order observer with the same bilinear structure as the one formulated in Reference 65 and estimation error possibly dependent on the input but asymptotically convergent to zero irrespective of it. Such a technique was shown to allow one to design stable bilinear observers even in some cases where the method from Reference 65 is not applicable or successful. However, also the method from Reference 66 is applicable only to a certain class of bilinear systems, which is harder to identify a priori than the one addressed in Reference 65. Finally, the design procedure is fairly complicated and more difficult to automate. Both References 65 and 66 dealt with the problem of general dependency on the input of the state estimation error by placing some constraints on the structure of the bilinear systems for which it is possible to design bilinear observers.

In this chapter we approach the problem differently, as presented at the 16$^{\text{th}}$ Yale Workshop on Adaptive and Learning Systems in New Haven, CT, in 2013 (Reference 69). We devise an observer design procedure applicable to any bilinear system with minimum observability requirements (observable linear part), overcoming the limitations of the methods proposed in References 65 and 66. We formulate a time-invariant full-order observer with bilinear structure and develop a design technique to optimize its gains based on system identification. Observers in this form, as opposed to other nonlinear forms, are of particular interest because they are required to develop an extension of OKID for bilinear systems. The connection between the gains of the proposed bilinear observer and the interaction matrices in the context of system

identification is a new development and is central to the chapter. The connection is established and also proposed as an approach to design both deterministic and stochastic bilinear observers. It is here assumed that the bilinear state-space model and the process and measurement noise covariances are known, and only the optimal bilinear observer gains are to be designed. The link between observer gains and interaction matrices is also exploited in the reverse direction. The properties of the proposed bilinear observer are proven to be, under the sole additional assumption of stationary white input, equivalent to the properties of the steady-state Kalman filter that led to OKID in the linear case. In short, not only do the presented results lead to a new bilinear observer design technique, but they also form the first fundamental step of working out a bilinear version of OKID, where a bilinear system model and an associated optimal bilinear observer are identified directly from noisy input-output measurements.

8.2 Problem statement

Consider the bilinear system in equation (6.1) with additive process and measurement noise

$$\boldsymbol{x}(k+1) = \mathbf{A}\boldsymbol{x}(k) + \mathbf{N}\boldsymbol{x}(k)u(k) + \mathbf{B}u(k) + \boldsymbol{w}_p(k) \tag{8.1a}$$

$$\boldsymbol{y}(k) = \mathbf{C}\boldsymbol{x}(k) + \mathbf{D}u(k) + \boldsymbol{w}_m(k) \tag{8.1b}$$

where $\boldsymbol{w}_p \in \mathbb{R}^{n\times 1}$ and $\boldsymbol{w}_m \in \mathbb{R}^{q\times 1}$ are stationary zero-mean white processes with covariance matrices $\mathbf{Q} \in \mathbb{R}^{n\times n}$ and $\mathbf{R} \in \mathbb{R}^{q\times q}$, respectively. Both $\boldsymbol{w}_p$ and $\boldsymbol{w}_m$ are assumed to be uncorrelated with u and $\boldsymbol{y}$ as well as, for simplicity, mutually uncorrelated.

We introduce a full-order time-invariant bilinear observer of the form

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{A}\hat{\boldsymbol{x}}(k) + \mathbf{B}u(k) + \mathbf{M}_1\left(\boldsymbol{y}(k) - \hat{\boldsymbol{y}}(k)\right) + \mathbf{N}\hat{\boldsymbol{x}}(k)u(k) + \mathbf{M}_2\left(\boldsymbol{y}(k) - \hat{\boldsymbol{y}}(k)\right)u(k) \tag{8.2a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \mathbf{D}u(k) \tag{8.2b}$$

where $\hat{\boldsymbol{y}}(k) \in \mathbb{R}^{q\times 1}$ is the observer output based on the estimated state $\hat{\boldsymbol{x}}(k) \in \mathbb{R}^{n\times 1}$, and $\mathbf{M}_1, \mathbf{M}_2 \in \mathbb{R}^{n\times q}$ are the observer gains. Defining

$$\bar{\mathbf{A}} = \mathbf{A} - \mathbf{M}_1\mathbf{C} \quad \bar{\mathbf{N}} = \mathbf{N} - \mathbf{M}_2\mathbf{C} \tag{8.3a}$$

$$\bar{\mathbf{B}} = \begin{bmatrix} \mathbf{B} - \mathbf{M}_1\mathbf{D} & \mathbf{M}_1 & -\mathbf{M}_2\mathbf{D} & \mathbf{M}_2 \end{bmatrix} \tag{8.3b}$$

$$\boldsymbol{v}(k) = \begin{bmatrix} u(k) & \boldsymbol{y}^{\mathrm{T}}(k) & u^2(k) & u(k)\boldsymbol{y}^{\mathrm{T}}(k) \end{bmatrix}^{\mathrm{T}} \tag{8.3c}$$

the bilinear observer in equation (8.2) can be written in the following equivalent form

$$\hat{\boldsymbol{x}}(k+1) = \bar{\mathbf{A}}\hat{\boldsymbol{x}}(k) + \bar{\mathbf{N}}\hat{\boldsymbol{x}}(k)u(k) + \bar{\mathbf{B}}\boldsymbol{v}(k) \tag{8.4a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \mathbf{D}u(k) \tag{8.4b}$$

It can be shown that the dynamics of the state estimation error $\boldsymbol{e}(k) = \boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k)$ is governed by

$$\boldsymbol{e}(k+1) = \bar{\mathbf{A}}\boldsymbol{e}(k) + \bar{\mathbf{N}}\boldsymbol{e}(k)u(k) + \boldsymbol{w}_p(k) - \mathbf{M}_1\boldsymbol{w}_m(k) - \mathbf{M}_2\boldsymbol{w}_m(k)u(k) \tag{8.5}$$

The presence of input-dependent terms in equation (8.5) makes the bilinear state estimation problem more complex than the well-known linear case.

This chapter is concerned with the observer design problem. In the deterministic case ($\mathbf{Q} = \mathbf{0}$, $\mathbf{R} = \mathbf{0}$), given $\mathbf{A}$, $\mathbf{N}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, the objective is to design observer gains $\mathbf{M}_1$ and $\mathbf{M}_2$ so that in the limit as k tends to infinity, the state estimation error $\boldsymbol{e}(k)$ converges to zero. In addition,

we also seek to answer the question if a deadbeat bilinear observer exists that causes the state estimation error to converge to zero identically in a minimum number of time steps. In the stochastic case, we assume further that the process and measurement noise covariances $\mathbf{Q}$ and $\mathbf{R}$ are known, and the design problem is to find the appropriate bilinear observer gains $\mathbf{M}_1$ and $\mathbf{M}_2$ so that the time average of the Euclidean norm squared of the steady-state state estimation error is minimized. This requirement is similar to that of the Kalman filter in the linear case. Indeed, when the problem is linear ($\mathbf{N} = \mathbf{0}$) the solution presented in this chapter produces the well-known steady-state Kalman filter gain $\mathbf{K}$ ($\mathbf{M}_1 = \mathbf{K}$, $\mathbf{M}_2 = \mathbf{0}$).

8.3 Deterministic bilinear observer

In the absence of noise, the dynamics of the state estimation error is described by

$$\boldsymbol{e}(k+1) = \bar{\mathbf{A}}\boldsymbol{e}(k) + \bar{\mathbf{N}}\boldsymbol{e}(k)u(k) \tag{8.6}$$

To guarantee the stability of the deterministic bilinear observer, $\boldsymbol{e}(k)$ in equation (8.6) must converge to zero regardless of its initial condition $\boldsymbol{e}(0)$ and the applied input sequence. From (8.6), $\boldsymbol{e}(k)$ can be interpreted as being governed by a linear-time-varying difference equation with a time-varying dynamic matrix $\bar{\mathbf{A}}+\bar{\mathbf{N}}u(k)$. Convergence of $\boldsymbol{e}(k)$ to zero is in general dependent on

$u(k)$, which is exogenous to the observer. Propagating equation (8.6) forward in time produces

$$
\begin{aligned}
\boldsymbol{e}(1) &= \bar{\mathbf{A}}\boldsymbol{e}(0) + \bar{\mathbf{N}}\boldsymbol{e}(0)u(0) \\
\boldsymbol{e}(2) &= \bar{\mathbf{A}}^2\boldsymbol{e}(0) + \bar{\mathbf{A}}\bar{\mathbf{N}}\boldsymbol{e}(0)u(0) + \bar{\mathbf{N}}\bar{\mathbf{A}}\boldsymbol{e}(0)u(1) + \bar{\mathbf{N}}^2\boldsymbol{e}(0)u(0)u(1) \\
\boldsymbol{e}(3) &= \bar{\mathbf{A}}^3\boldsymbol{e}(0) + \bar{\mathbf{A}}^2\bar{\mathbf{N}}\boldsymbol{e}(0)u(0) + \bar{\mathbf{A}}\bar{\mathbf{N}}\bar{\mathbf{A}}\boldsymbol{e}(0)u(1) + \bar{\mathbf{A}}\bar{\mathbf{N}}^2\boldsymbol{e}(0)u(0)u(1) \\
&\quad + \bar{\mathbf{N}}\bar{\mathbf{A}}^2\boldsymbol{e}(0)u(2) + \bar{\mathbf{N}}\bar{\mathbf{A}}\bar{\mathbf{N}}\boldsymbol{e}(0)u(0)u(2) + \bar{\mathbf{N}}^2\bar{\mathbf{A}}\boldsymbol{e}(0)u(1)u(2) + \bar{\mathbf{N}}^3\boldsymbol{e}(0)u(0)u(1)u(2) \\
&\vdots \\
\boldsymbol{e}(k) &= \boldsymbol{\mathcal{S}}_k(k)\boldsymbol{e}(0)
\end{aligned}
\tag{8.7}
$$

Observe that the relationship that expresses $\boldsymbol{e}(k)$ in terms of the previous inputs $u(k-1), ..., u(0)$ and the initial error $\boldsymbol{e}(0)$ contains all possible products of $\bar{\mathbf{A}}$ and $\bar{\mathbf{N}}$ whose combined power is k. The sum of such terms (multiplied by the appropriate input values) is compactly denoted by $\boldsymbol{\mathcal{S}}_k(k)$ similar to what done in equation (6.29). As seen from equation (8.3a), the two observer gains $\mathbf{M}_1$ and $\mathbf{M}_2$ provide the design freedom for $\boldsymbol{\mathcal{S}}_k(k)$ to converge to zero. In mathematics literature the topic is well-known under the name of *Infinite Product of Matrices*. To our knowledge, no general result is available that guarantees the convergence to zero of the above mentioned products as k tends to infinity regardless of the magnitude of u. However, when the magnitude of u is bounded, it is possible to guarantee the existence of the observer gains $\mathbf{M}_1$ and $\mathbf{M}_2$ to ensure that the state estimation error $\boldsymbol{e}(k)$ converges to zero as k tends to infinity. This result can be explained in the context of state-space bilinear identification using interaction matrices.

8.3.1 Observer gains and interaction matrices

The interaction matrices were introduced in chapter 6 to derive input-output-to-state representations (IOSRs) for deterministic bilinear system identification. These IOSRs allow one to express the current state, $\boldsymbol{x}(k)$, in terms of a fixed number of past input and output values, $u(k-1), ..., u(k-p), \boldsymbol{y}(k-1), ..., \boldsymbol{y}(k-p)$. Such relationships are proven to be asymptotically exact for a general bilinear system as p increases (Reference 51). It turns out that the approximation error of the IOSRs in system identification has exactly the same structure as the state estimation error of equation (8.7). The interaction matrices $\mathbf{M}'$ and $\mathbf{M}''$ in chapter 6 can then be interpreted as bilinear observer gains. Additionally, the same theorem that ensures the validity of the IOSRs in the system identification problem (theorem 6.1), also guarantees the existence of the observer gains $\mathbf{M}_1$ and $\mathbf{M}_2$ to cause the state estimation error $\boldsymbol{e}(k)$ to converge to zero as k tends to infinity, provided that $(\mathbf{A}, \mathbf{C})$ is an observable pair[1], and the magnitude of the input $u(k)$ is upper bounded. This result is restated below for the bilinear observer problem.

Theorem 8.1. *If* $(\mathbf{A}, \mathbf{C})$ *is an observable pair, then there exist observer gains* $\mathbf{M}_1$ *and* $\mathbf{M}_2$ *and a value* γ *such that, for* $|u(k)| < \gamma$ *for all* k, $\boldsymbol{e}(k)$ *in equation (8.7) converges to* $\mathbf{0}$ *as* k *tends to infinity.*

The proof is the same as for theorem 6.1.

The interaction matrix formulation offers a fundamental connection between the system identification problem and the state estimation problem. One can indeed think of system identification in the context of a larger problem of observer identification where an optimal observer gain is identified simultaneously with the system model, hence the term *observer identification* might be more appropriate. In the bilinear system identification methods in chapter 6 the interaction

[1] The theorem in Reference 51 requires $(\mathbf{A}, \mathbf{C})$ to be a detectable pair, instead of observable. Observability is a more stringent requirement than detectability, therefore the proof in Reference 51 is valid for theorem 8.1, too.

matrices are used to derive the desired IOSRs, but they do not need to be found explicitly for the identification of the discrete-time bilinear state-space model matrices $\mathbf{A}$, $\mathbf{N}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$. This same formulation can now be exploited in reverse order to identify the observer gains $\mathbf{M}_1$ and $\mathbf{M}_2$ given the bilinear state-space model $\mathbf{A}$, $\mathbf{N}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$. Here one has the additional advantage of knowing the system state because the model is given. State information is not available in the system identification problem where only input-output data is known.

8.3.2 Observer gain identification

A special consideration in the deterministic case is to determine if suitable observer gains exist that would cause the state estimation error $\boldsymbol{e}(k)$ to become identically zero after a finite number of time steps. This is analogous to the case of a deadbeat observer for a linear system. Note that Theorem 8.1 does not imply that all the possible products of $\bar{\mathbf{A}}$ and $\bar{\mathbf{N}}$ whose powers add up to a certain value vanish identically. In other words, unlike the linear case, no deadbeat observer is guaranteed to exist for a general bilinear system. Deadbeat observers only exist for a very limited class of bilinear systems where $\mathbf{A}$ and $\mathbf{N}$ satisfy certain restrictive conditions as illustrated via an example in Reference 33. These systems are referred to as *ideal* in the present work, following the same terminology used in the context of bilinear system identification (chapter 6). For *non-ideal* systems, although deadbeat observers do not exist to cause the state estimation error to converge to zero identically in a finite number of time steps, observer gains can still be found to cause the state estimation error to converge to zero asymptotically. The design of these observer gains is described below.

Propagating equation (8.1) forward in time by p time steps, we obtain

$$\hat{\boldsymbol{x}}(k) = \mathbf{T}_p \boldsymbol{z}_p(k) + \boldsymbol{\mathcal{S}}_p(k)\hat{\boldsymbol{x}}(k-p) \tag{8.8}$$

where $\mathbf{T}_p$ is a matrix containing products of $\bar{\mathbf{A}}$, $\bar{\mathbf{N}}$ and $\bar{\mathbf{B}}$, $\boldsymbol{\mathcal{S}}_p(k)$ is the sum of all products of $\bar{\mathbf{A}}$ and $\bar{\mathbf{N}}$ of combined power p multiplied by appropriate products of past input values $u(k-1)$, ..., $u(k-p)$, and $\boldsymbol{z}_p(k)$ is a column vector defined solely in terms of input-output data exactly as $\boldsymbol{z}_{p,c}$ in section 6.5 below equation (6.44).

For example, for $p = 2$ we have

$$\mathbf{T}_2 = \begin{bmatrix} \bar{\mathbf{A}}\bar{\mathbf{B}} & \bar{\mathbf{N}}\bar{\mathbf{B}} & \bar{\mathbf{B}} \end{bmatrix} \tag{8.9a}$$

$$\boldsymbol{z}_2(k) = \begin{bmatrix} \boldsymbol{v}(k-2) \\ \boldsymbol{v}(k-2)u(k-1) \\ \boldsymbol{v}(k-1) \end{bmatrix} \tag{8.9b}$$

$$\boldsymbol{\mathcal{S}}_2(k) = \bar{\mathbf{A}}^2 + \bar{\mathbf{A}}\bar{\mathbf{N}}u(k-2) + \bar{\mathbf{N}}\bar{\mathbf{A}}u(k-1) + \bar{\mathbf{N}}^2u(k-2)u(k-1) \tag{8.9c}$$

From Theorem 8.1, if p is chosen to be sufficiently large then $\boldsymbol{\mathcal{S}}_p(k) \to \mathbf{0}$. For $k \geq p$, (8.8) becomes

$$\hat{\boldsymbol{x}}(k) = \mathbf{T}_p\boldsymbol{z}_p(k) \tag{8.10}$$

Equation (8.10) expresses the estimated state $\hat{\boldsymbol{x}}(k)$ in terms of p past input and p past output measurements, $u(k-1), ..., u(k-p), \boldsymbol{y}(k-1), ..., \boldsymbol{y}(k-p)$. Following the same terminology used in chapter 6, equation (8.10) is an observer IOSR. Since $\boldsymbol{e}(k) = \boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k)$, (8.10) can be rewritten as

$$\boldsymbol{x}(k) = \mathbf{T}_p\boldsymbol{z}_p(k) + \boldsymbol{e}(k) \tag{8.11}$$

leading to the following matrix relationship

$$\mathbf{X} = \mathbf{T}_p\mathbf{Z}_p + \mathbf{E} \tag{8.12}$$

where

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{x}(p) & \boldsymbol{x}(p+1) & \boldsymbol{x}(p+2) & \dots & \boldsymbol{x}(l) \end{bmatrix} \tag{8.13a}$$

$$\boldsymbol{z}_p = \begin{bmatrix} \boldsymbol{z}_p(p) & \boldsymbol{z}_p(p+1) & \boldsymbol{z}_p(p+2) & \dots & \boldsymbol{z}_p(l) \end{bmatrix} \tag{8.13b}$$

$$\mathbf{E} = \begin{bmatrix} \boldsymbol{e}(p) & \boldsymbol{e}(p+1) & \boldsymbol{e}(p+2) & \dots & \boldsymbol{e}(l) \end{bmatrix} \tag{8.13c}$$

where l is the final time step in a data record. This relationship is used to develop a numerical data-based approach to design the bilinear observer gains from a given model of the system as described below.

Using the system state-space model, state and output data, denoted by $\{\boldsymbol{x}(k)\}$ and $\{\boldsymbol{y}(k)\}$, can be generated from one or more sufficiently long and rich input data sequences $\{u(k)\}$, and the matrices $\mathbf{X}$ and $\mathbf{Z}_p$ are then formed. The matrix $\mathbf{T}_p$ can then be solved according to

$$\tilde{\mathbf{T}}_p = \mathbf{X} \left(\mathbf{Z}_p\right)^{\dagger} \tag{8.14}$$

where the superscript † denotes the pseudo-inverse operation. The observer gains $\mathbf{M}_1$ and $\mathbf{M}_2$ are extracted directly from $\tilde{\mathbf{T}}_p$ because they appear explicitly in $\bar{\mathbf{B}}$ which is in $\tilde{\mathbf{T}}_p$. For an ideal bilinear system, $\mathbf{T}_p$ can be found by equation (8.14) to satisfy (8.12) exactly with $\mathbf{E} = \mathbf{0}$. The observer gains extracted from $\mathbf{T}_p$ when p is minimum are the deadbeat observer gains that would cause the state estimation error to converge to zero identically in p time steps. For the more general case of a non-ideal system, $\mathbf{E}$ cannot be made exactly $\mathbf{0}$ and the pseudo-inverse solution corresponds to an observer that minimizes the Frobenius norm of the state estimation error matrix $\mathbf{E}$ (least-squares solution).

8.4 Stochastic bilinear observer

In the stochastic case, the dynamics of the state estimation error is given by equation (8.5). For the purpose of stability analysis, we are interested in studying its response to initial conditions, which leads again to the analysis of equations (8.6) and (8.7). Therefore, Theorem 8.1 is applicable to the stochastic case, too, and guarantees the existence of stable bilinear observers even in the presence of process and measurement noise.

For a linear system, the optimal linear observer in the presence of noise is the Kalman filter (in one-step-ahead predictor form). Similar to OKID (see chapter 2), in this chapter we are concerned with time-invariant observers, therefore the optimal observer we are after is the steady-state Kalman filter. In the following development, we work out a solution for the linear case, and then extend the result to the bilinear case. In the linear case, the solution produces the steady-state Kalman filter gain.

8.4.1 Linear case

In the linear case, $\mathbf{N} = \mathbf{0}$ and $\mathbf{M}_2 = \mathbf{0}$. The matrices $\mathbf{T}_p$, $\boldsymbol{z}_p(k)$ and $\boldsymbol{\mathcal{S}}_p$ then become

$$\mathbf{T}_p = \begin{bmatrix} \bar{\mathbf{A}}^{p-1}\bar{\mathbf{B}} & \dots & \bar{\mathbf{A}}\bar{\mathbf{B}} & \bar{\mathbf{B}} \end{bmatrix} \tag{8.15a}$$

$$\boldsymbol{z}_p(k) = \begin{bmatrix} \boldsymbol{v}(k-p) \\ \vdots \\ \boldsymbol{v}(k-2) \\ \boldsymbol{v}(k-1) \end{bmatrix} \tag{8.15b}$$

$$\boldsymbol{\mathcal{S}}_p = \bar{\mathbf{A}}^p \tag{8.15c}$$

where also the input to the observer simplifies to

$$\boldsymbol{v}(k) = \begin{bmatrix} u(k) \\ \\ \boldsymbol{y}(k) \end{bmatrix} \tag{8.16}$$

If p is chosen to be sufficiently large so that $\bar{\mathbf{A}}^p$ can be neglected, the least-squares solution $\tilde{\mathbf{T}}_p = \mathbf{X}\left(\mathbf{Z}_p\right)^\dagger$ minimizes the Frobenius norm of $\mathbf{E}$ which is

$$\gamma = \left(\sum_{k=p}^{l} \sum_{i=1}^{n} \boldsymbol{e}_i(k)^2 \right)^{1/2} \tag{8.17}$$

We now argue that $\tilde{\mathbf{T}}_p$ contains the steady-state Kalman filter gain. As shown in chapter 2, the Kalman filter is the unique linear observer of the form given by equation (8.2), with $\mathbf{N} = \mathbf{0}$ and $\mathbf{M}_2 = \mathbf{0}$ (linear case), which minimizes $\mathbb{E}\left[\boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k)\right]$. The corresponding observer gain $\mathbf{M}_1$ can be computed by solving the algebraic Riccati equation in (2.7d) together with equation (2.7c) and will be denoted by $\mathbf{K}$. The stationarity of the noise makes it possible to estimate the expected value of the norm squared of the state error as its time average (ergodic property)

$$\begin{aligned} \mathbb{E}\left[\boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k)\right] &= \lim_{l\to\infty} \frac{1}{l-p+1} \sum_{k=p}^{l} \boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k) \\ &= \lim_{l\to\infty} \frac{1}{l-p+1} \sum_{k=p}^{l} \sum_{i=1}^{n} \boldsymbol{e}_i^2(k) \\ &= \lim_{l\to\infty} \frac{\gamma^2}{l-p+1} \end{aligned} \tag{8.18}$$

where the last expression is proportional to the Frobenius norm, minimized by $\tilde{\mathbf{T}}_p = \mathbf{X}(\mathbf{Z}_p)^\dagger$. Therefore, the solution $\tilde{\mathbf{T}}_p = \mathbf{X}(\mathbf{Z}_p)^\dagger$, in the limit as the data record length tends to infinity, minimizes the same objective function minimized by $\mathbf{K}$. Since the Kalman filter is the unique linear observer minimizing $\mathbb{E}\left[\boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k)\right]$, it follows that $\tilde{\mathbf{T}}_p = \mathbf{X}(\mathbf{Z}_p)^\dagger$ must contain the state

Markov parameters of the Kalman filter, i.e. the products $\bar{\mathbf{A}}^j\bar{\mathbf{B}}$, $j = 0, 1, ..., p-1$, where $\bar{\mathbf{A}}$ and $\bar{\mathbf{B}}$ are defined in equation (8.3) with $\mathbf{M}_1 = \mathbf{K}$ and $\mathbf{M}_2 = \mathbf{0}$. Also note that in the linear case $\bar{\mathbf{B}} = \begin{bmatrix} \mathbf{B} - \mathbf{KD} & \mathbf{K} \end{bmatrix}$. Hence, $\mathbf{M}_1$ extracted from the last column of $\tilde{\mathbf{T}}_p$ converges to $\mathbf{K}$, completing the argument.

8.4.2 Bilinear case

Although (8.14) can be immediately used in the stochastic case, for a bilinear model the high dimensionality of $\mathbf{Z}_p$ may become impractical. In the stochastic case the locations of the eigenvalues of $\bar{\mathbf{A}}$ and $\bar{\mathbf{N}}$ via $\mathbf{M}_1$ and $\mathbf{M}_2$ are determined by the statistical structure of the noise, i.e. by $\mathbf{Q}$ and $\mathbf{R}$. Therefore, $\boldsymbol{\mathcal{S}}_p(k)$ will generally converge to zero with p more slowly than in the deterministic case, where the absence of noise makes the eigenvalue placement optimized for the convergence of $\boldsymbol{\mathcal{S}}_p(k)$. In the following we develop an alternative technique to overcome the high dimensionality associated with equation (8.14). Starting with

$$\boldsymbol{x}(k+1) = \hat{\boldsymbol{x}}(k+1) + \boldsymbol{e}(k+1) \tag{8.19}$$

and substituting (8.4a) for $\hat{\boldsymbol{x}}(k+1)$ produces

$$\boldsymbol{x}(k+1) = \bar{\mathbf{A}}\hat{\boldsymbol{x}}(k) + \bar{\mathbf{N}}\hat{\boldsymbol{x}}(k)u(k) + \bar{\mathbf{B}}\boldsymbol{v}(k) + \boldsymbol{e}(k+1) \tag{8.20}$$

Since $\hat{\boldsymbol{x}}(k) = \boldsymbol{x}(k) - \boldsymbol{e}(k)$, equation (8.20) becomes

$$\boldsymbol{x}(k+1) = \bar{\mathbf{A}}\left[\boldsymbol{x}(k) - \boldsymbol{e}(k)\right] + \bar{\mathbf{B}}\boldsymbol{v}(k) + \bar{\mathbf{N}}\left[\boldsymbol{x}(k) - \boldsymbol{e}(k)\right]u(k) + \boldsymbol{e}(k+1) \tag{8.21}$$

Because we are interested in the batch-form solution, it is more convenient to rewrite equation (8.21) as

$$\boldsymbol{x}(k+1) = \mathbf{P}\mathbf{V}(k) + \boldsymbol{e}(k+1) \tag{8.22}$$

where the bilinear observer gains $\mathbf{M}_1$ and $\mathbf{M}_2$ are explicitly present in $\bar{\mathbf{B}}$ in $\mathbf{P}$

$$\mathbf{P} = \begin{bmatrix} \bar{\mathbf{A}} & \bar{\mathbf{N}} & \bar{\mathbf{B}} \end{bmatrix} \tag{8.23a}$$

$$\mathbf{V}(k) = \begin{bmatrix} \boldsymbol{x}(k) \\ \boldsymbol{x}(k)u(k) \\ \boldsymbol{v}(k) \end{bmatrix} - \begin{bmatrix} \boldsymbol{e}(k) \\ \boldsymbol{e}(k)u(k) \\ \mathbf{0} \end{bmatrix} \tag{8.23b}$$

Equation (8.22) can be written using all available data as

$$\mathbf{X} = \mathbf{P}\left(\mathbf{V}_X - \mathbf{V}_E\right) + \mathbf{E} \tag{8.24}$$

where

$$\mathbf{X} = \begin{bmatrix} \boldsymbol{x}(p) & \boldsymbol{x}(p+1) & \cdots & \boldsymbol{x}(l) \end{bmatrix} \tag{8.25a}$$

$$\mathbf{V}_X = \begin{bmatrix} \boldsymbol{x}(p-1) & \cdots & \boldsymbol{x}(l-1) \\ \boldsymbol{x}(p-1)u(p-1) & \cdots & \boldsymbol{x}(l-1)u(l-1) \\ \boldsymbol{v}(p-1) & \cdots & \boldsymbol{v}(l-1) \end{bmatrix} \tag{8.25b}$$

$$\mathbf{V}_E = \begin{bmatrix} \boldsymbol{e}(p-1) & \cdots & \boldsymbol{e}(l-1) \\ \boldsymbol{e}(p-1)u(p-1) & \cdots & \boldsymbol{e}(l-1)u(l-1) \\ \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix} \tag{8.25c}$$

$$\mathbf{E} = \begin{bmatrix} \boldsymbol{e}(p) & \boldsymbol{e}(p+1) & \cdots & \boldsymbol{e}(l) \end{bmatrix} \tag{8.25d}$$

Equation (8.24) is now in a convenient form because it isolates in a single term ($\mathbf{E}$) the state estimation error sequence whose Frobenius norm has to be minimized. Such term can be viewed as the residuals of the least-squares (LS) problem (8.24), with $\mathbf{V}_X - \mathbf{V}_E$ as data matrix and $\bar{\mathbf{A}}, \bar{\mathbf{N}}$, and $\bar{\mathbf{B}}$ as coefficients to be estimated. As a result, the LS solution to equation (8.24) yields the matrices $\bar{\mathbf{A}}$, $\bar{\mathbf{N}}$, and $\bar{\mathbf{B}}$ containing the optimal gain $\mathbf{M}_1$ and $\mathbf{M}_2$ (the concept of optimality for the proposed bilinear observer is defined more rigorously in section 8.5). However, ordinary LS techniques cannot be directly applied to equation (8.24) since $\mathbf{V}_E$ is not known before the LS solution is found and the corresponding residuals $\mathbf{E}$ are computed. Nevertheless, equation (8.24) is in a form that existing standard generalized (or extended) LS methods can be adapted to find the solution that minimizes the Frobenius norm of $\mathbf{E}$ as desired. Notice how, in system identification, generalized least-squares (GLS) methods usually refer to techniques to eliminate the bias in ordinary LS estimates caused by correlated residuals (see for example Reference 48). Although pursuing a different goal, one of such techniques, proposed in Reference 47, addresses the LS estimation bias by rewriting the model to be identified in the form of autoregressive-moving-average model and solving the same type of problem where direct application of the ordinary LS technique is not possible. The iterative method presented in Reference 47 is described below with reference to equation (8.24). In section 8.5 it will then be proven how to guarantee that the matrices $\bar{\mathbf{A}}$, $\bar{\mathbf{N}}$ and $\bar{\mathbf{B}}$ found via (8.14) or via the iterative method contain indeed the optimal bilinear observer gains $\mathbf{M}_1$ and $\mathbf{M}_2$.

8.4.3 Optimal bilinear observer gains by iterative method

Let the superscript j denote the iteration number, starting from $j = 1$. Using $\{u(k)\}$, $\{\boldsymbol{x}(k)\}$, $\{\boldsymbol{y}(k)\}$ data that are generated from the given bilinear model and the specified process and

measurement noise covariances, we form $\mathbf{X}$ and $\mathbf{V}_X$, then solve for $\mathbf{P}^{(1)}$

$$\mathbf{P}^{(1)} = \mathbf{X}\,(\mathbf{V}_X)^{\dagger} \tag{8.26}$$

and compute the corresponding error matrix $\mathbf{E}^{(1)}$ associated with this solution

$$\mathbf{E}^{(1)} = \mathbf{X} - \mathbf{P}^{(1)}\mathbf{V}_X \tag{8.27}$$

The entries in $\mathbf{E}^{(1)}$ are denoted by the superscript (1)

$$\mathbf{E}^{(1)} = \begin{bmatrix} \boldsymbol{e}^{(1)}(p) & \boldsymbol{e}^{(1)}(p+1) & \cdots & \boldsymbol{e}^{(1)}(l) \end{bmatrix} \tag{8.28}$$

and used to update the parameter estimate. Since $\mathbf{V}_E$ calls for $\boldsymbol{e}(p-1)$, but $\boldsymbol{e}(p-1)$ is not available in $\mathbf{E}$ which starts with $\boldsymbol{e}(p)$, the first column of $\mathbf{X}$ must now start from $p+1$ instead of p

$$\mathbf{X}^{(1)} = \begin{bmatrix} \boldsymbol{x}(p+1) & \boldsymbol{x}(p+2) & \cdots & \boldsymbol{x}(l) \end{bmatrix} \tag{8.29}$$

$\mathbf{V}_X$ associated with $\mathbf{X}^{(1)}$ is then adjusted accordingly so that $\mathbf{V}_E$ can start with $\boldsymbol{e}^{(1)}(p)$ which is available in $\mathbf{E}^{(1)}$

$$\mathbf{V}_X^{(1)} = \begin{bmatrix} \boldsymbol{x}(p) & \cdots & \boldsymbol{x}(l-1) \\ \boldsymbol{x}(p)u(p) & \cdots & \boldsymbol{x}(l-1)u(l-1) \\ \boldsymbol{v}(p) & \cdots & \boldsymbol{v}(l-1) \end{bmatrix} \tag{8.30}$$

$$\mathbf{V}_E^{(1)} = \begin{bmatrix} \boldsymbol{e}^{(1)}(p) & \cdots & \boldsymbol{e}^{(1)}(l-1) \\ \boldsymbol{e}^{(1)}(p)u(p) & \cdots & \boldsymbol{e}^{(1)}(l-1)u(l-1) \\ \mathbf{0} & \cdots & \mathbf{0} \end{bmatrix} \tag{8.31}$$

The next update $\mathbf{P}^{(2)}$ and $\mathbf{E}^{(2)}$ can be computed from

$$\mathbf{P}^{(2)} = \mathbf{X}^{(1)} \left(\mathbf{V}_X^{(1)} - \mathbf{V}_E^{(1)} \right)^{\dagger} \tag{8.32}$$

$$\mathbf{E}^{(2)} = \mathbf{X}^{(1)} - \mathbf{P}^{(2)} \left(\mathbf{V}_X^{(1)} - \mathbf{V}_E^{(1)} \right) \tag{8.33}$$

$\mathbf{E}^{(2)}$ is then used to update the parameter estimate. The first entry in $\mathbf{E}^{(2)}$ is $\boldsymbol{e}^{(2)}(p+1)$ which is consistent with $\mathbf{X}^{(1)}$. For the next iteration, $\mathbf{X}^{(2)}$ must start from $\boldsymbol{x}(p+2)$, $\mathbf{V}_X^{(2)}$ from $\boldsymbol{x}(p+1), u(p+1), \boldsymbol{v}(p+1)$ so that $\mathbf{V}_E^{(2)}$ can start from $\boldsymbol{e}^{(2)}(p+1)$, etc. To avoid losing one data sample at each iteration, an alternative strategy is inserting $\boldsymbol{e}^{(1)}(p-1) = \mathbf{0}$ to equation (8.28), so that $\mathbf{V}_X^{(1)}$ can remain the same as $\mathbf{V}_X$. The first column of $\mathbf{V}_E^{(1)}$ which now starts with $\boldsymbol{e}^{(1)}(p-1) = \mathbf{0}$ will be zero. Once $\mathbf{P}$ is identified, the bilinear observer gains $\mathbf{M}_1$ and $\mathbf{M}_2$ can be directly extracted from it.

8.5 Stochastic properties of the optimal bilinear observer

In this section the theoretical aspects behind the proposed stochastic observer are analyzed, for two main purposes. The first two lemmas justify the design procedure illustrated above and highlight the properties of the resulting state observer. Additionally, the properties of the observer output residuals are proven to show how the proposed observer possesses properties similar to those of the Kalman filter that led to OKID in the linear case. Notice the similarity with the approach taken in section 2.4 to prove the theoretical foundation of OKID in the linear case.

Consider the system in equation (8.1) and the proposed bilinear observer in (8.4). The most general form for such a time-invariant observer is

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{F}\hat{\boldsymbol{x}}(k) + \mathbf{G}\hat{\boldsymbol{x}}(k)u(k) + \mathbf{H}u(k) + \mathbf{L}u^2(k) + \mathbf{M}_1 y(k) + \mathbf{M}_2 \boldsymbol{y}(k)u(k) \tag{8.34}$$

where $\mathbf{F}, \mathbf{G} \in \mathbb{R}^{n\times n}$, $\mathbf{H}, \mathbf{L} \in \mathbb{R}^{n\times 1}$, and $\mathbf{M}_1, \mathbf{M}_2 \in \mathbb{R}^{n\times q}$. From the definition of state estimation error and equations (8.1) and (8.34), the dynamics of the state estimation error is

$$\begin{aligned}
\boldsymbol{e}(k+1) &= \boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k) \\
&= \mathbf{A}\boldsymbol{x}(k) + \mathbf{N}\boldsymbol{x}(k)u(k) + \mathbf{B}u(k) + \boldsymbol{w}_p(k) - \mathbf{F}\hat{\boldsymbol{x}}(k) - \mathbf{G}\hat{\boldsymbol{x}}(k)u(k) - \mathbf{H}u(k) \\
&\quad - \mathbf{L}u^2(k) - \mathbf{M}_1\boldsymbol{y}(k) - \mathbf{M}_2\boldsymbol{y}(k)u(k) \\
&= (\mathbf{A} - \mathbf{M}_1\mathbf{C})\,\boldsymbol{e}(k) + (\mathbf{N} - \mathbf{M}_2\mathbf{C})\,\boldsymbol{e}(k)u(k) + (\mathbf{A} - \mathbf{F} - \mathbf{M}_1\mathbf{C})\,\hat{\boldsymbol{x}}(k) \\
&\quad + (\mathbf{N} - \mathbf{G} - \mathbf{M}_2\mathbf{C})\,\hat{\boldsymbol{x}}(k)u(k) + (\mathbf{B} - \mathbf{H} - \mathbf{M}_1\mathbf{D})\,u(k) - (\mathbf{L} + \mathbf{M}_2\mathbf{D})\,u^2(k) \\
&\quad + \boldsymbol{w}_p(k) - \mathbf{M}_1\boldsymbol{w}_m(k) - \mathbf{M}_2\boldsymbol{w}_m(k)u(k)
\end{aligned} \tag{8.35}$$

We would like to analyze the properties of the observer in (8.34) and in particular the conditions under which the observer state $\hat{\boldsymbol{x}}(k)$ is an unbiased estimate for the system state $\boldsymbol{x}$ and the estimation error is minimized. The assumptions on the noises $\boldsymbol{w}_p$ and $\boldsymbol{w}_m$ are the same as stated in section 6.2.

8.5.1 Unbiased observer

Lemma 8.2. *The state $\hat{\boldsymbol{x}}(k)$ of the observer in (8.34) is an unbiased estimate for the state $\boldsymbol{x}(k)$ of the system in (8.1) for all $k \geq p$ and for any arbitrary input sequence if and only if the*

following conditions are satisfied

$$\mathbf{F} = \mathbf{A} - \mathbf{M}_1\mathbf{C} \tag{8.36a}$$

$$\mathbf{G} = \mathbf{N} - \mathbf{M}_2\mathbf{C} \tag{8.36b}$$

$$\mathbf{H} = \mathbf{B} - \mathbf{M}_1\mathbf{D} \tag{8.36c}$$

$$\mathbf{L} = -\mathbf{M}_2\mathbf{D} \tag{8.36d}$$

$$\boldsymbol{\mathcal{S}}_p = \mathbf{0} \tag{8.36e}$$

In general, condition (8.36e) cannot be satisfied exactly, but for an appropriately bounded input it can be approximately met by a sufficiently large value of p (Theorem 8.1). Condition (8.36e) corresponds then to the observer in (8.34) being stable and having reached its steady state.

Proof. Define for convenience

$$\bar{\mathbf{A}} = \mathbf{A} - \mathbf{M}_1\mathbf{C} \tag{8.37a}$$

$$\bar{\mathbf{N}} = \mathbf{N} - \mathbf{M}_2\mathbf{C} \tag{8.37b}$$

$$\bar{\mathbf{F}} = \mathbf{A} - \mathbf{F} - \mathbf{M}_1\mathbf{C} \tag{8.37c}$$

$$\bar{\mathbf{G}} = \mathbf{N} - \mathbf{G} - \mathbf{M}_2\mathbf{C} \tag{8.37d}$$

$$\bar{\mathbf{H}} = \mathbf{B} - \mathbf{H} - \mathbf{M}_1\mathbf{D} \tag{8.37e}$$

$$\bar{\mathbf{L}} = \mathbf{L} + \mathbf{M}_2\mathbf{D} \tag{8.37f}$$

Equation (8.35) can be rewritten as

$$\begin{aligned}\boldsymbol{e}(k+1) = \bar{\mathbf{A}}\boldsymbol{e}(k) + \bar{\mathbf{N}}\boldsymbol{e}(k)u(k) + \bar{\mathbf{F}}\hat{\boldsymbol{x}}(k) + \bar{\mathbf{G}}\hat{\boldsymbol{x}}(k)u(k) + \bar{\mathbf{H}}u(k) \\ - \bar{\mathbf{L}}u^2(k) + \boldsymbol{w}_p(k) - \mathbf{M}_1\boldsymbol{w}_m(k) - \mathbf{M}_2\boldsymbol{w}_m(k)u(k)\end{aligned} \tag{8.38}$$

and the expected value of the state estimation error is

$$\begin{aligned}\mathbb{E}\left[\boldsymbol{e}(k)\right] = \bar{\mathbf{A}}\mathbb{E}\left[\boldsymbol{e}(k-1)\right] + \bar{\mathbf{N}}\mathbb{E}\left[\boldsymbol{e}(k-1)\right]u(k-1) + \bar{\mathbf{F}}\mathbb{E}\left[\hat{\boldsymbol{x}}(k-1)\right] \\ + \bar{\mathbf{G}}\mathbb{E}\left[\hat{\boldsymbol{x}}(k-1)\right]u(k-1) + \bar{\mathbf{H}}u(k-1) - \bar{\mathbf{L}}u^2(k-1)\end{aligned} \tag{8.39}$$

To obtain equation (8.39), the noise terms vanish thanks to the assumption of zero-mean noises and the input is pulled out of the expectation operator since it is a known, deterministic variable. The expected values in (8.39) are ensemble averages, i.e. they can be interpreted as the expected values obtained over different realizations of the process and measurement noise while driving the system with the same input sequence. Propagating equation (8.39) one time step backwards yields

$$\begin{aligned}\mathbb{E}\left[\boldsymbol{e}(k)\right] = \left(\bar{\mathbf{A}}^2 + \bar{\mathbf{A}}\bar{\mathbf{N}}u(k-2) + \bar{\mathbf{N}}\bar{\mathbf{A}}u(k-1) + \bar{\mathbf{N}}^2u(k-1)u(k-2)\right)\mathbb{E}\left[\boldsymbol{e}(k-2)\right] \\ + \bar{\mathbf{A}}\bar{\mathbf{F}}\mathbb{E}\left[\hat{\boldsymbol{x}}(k-2)\right] + \bar{\mathbf{A}}\bar{\mathbf{G}}\mathbb{E}\left[\hat{\boldsymbol{x}}(k-2)\right]u(k-2) + \bar{\mathbf{A}}\bar{\mathbf{H}}u(k-2) + \bar{\mathbf{A}}\bar{\mathbf{L}}u^2(k-2) \\ + \bar{\mathbf{N}}\bar{\mathbf{F}}\mathbb{E}\left[\hat{\boldsymbol{x}}(k-2)\right]u(k-1) + \bar{\mathbf{N}}\bar{\mathbf{G}}\mathbb{E}\left[\hat{\boldsymbol{x}}(k-2)\right]u(k-1)u(k-2) \\ + \bar{\mathbf{N}}\bar{\mathbf{H}}u(k-1)u(k-2) - \bar{\mathbf{N}}\bar{\mathbf{L}}u(k-1)u^2(k-2) \\ + \bar{\mathbf{F}}\mathbb{E}\left[\hat{\boldsymbol{x}}(k-1)\right] + \bar{\mathbf{G}}\mathbb{E}\left[\hat{\boldsymbol{x}}(k-1)\right]u(k-1) + \bar{\mathbf{H}}u(k-1) - \bar{\mathbf{L}}u^2(k-1)\end{aligned} \tag{8.40}$$

and, by propagating back to the initial time step, we can write the expected state estimation error as

$$\mathbb{E}\left[\boldsymbol{e}(k)\right] = \boldsymbol{\mathcal{S}}_k(k)\mathbb{E}\left[\boldsymbol{e}(0)\right] + f\left(\bar{\mathbf{A}}, \bar{\mathbf{N}}, \bar{\mathbf{F}}, \bar{\mathbf{G}}, \bar{\mathbf{H}}, \bar{\mathbf{L}}, u(0), u(1), ..., u(k-1), \hat{\boldsymbol{x}}(0)\right) \tag{8.41}$$

where the second term on the right-hand side of (8.41) represents all the terms not depending on the initial error and is identically zero for any input history and initial observer state if and only if $\bar{\mathbf{F}}, \bar{\mathbf{G}}, \bar{\mathbf{H}}, \bar{\mathbf{L}}$ are null matrices, i.e. if and only if $\mathbf{F}, \mathbf{G}, \mathbf{H}, \mathbf{L}$ in equation (8.34) are chosen in accordance with conditions (8.36a)-(8.36d). The first term is guaranteed to vanish for any choice of $\hat{\boldsymbol{x}}(0)$ and for any possible $\boldsymbol{x}(0)$ if and only if the observer is stable (Theorem 8.1) and k is sufficiently large for the observer to have reached its steady state ($k \geq p$). □

8.5.2 Optimal observer

Let us start by choosing for the bilinear observer the same optimality criterion as for the steady-state linear Kalman filter, i.e. minimum expected value of the norm squared of the state estimation error $\mathbb{E}\left[\boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k)\right]$ for all k. The criterion is equivalent to minimizing the trace of the covariance matrix of the state estimation error $\boldsymbol{\Pi}(k) = \mathbb{E}\left[\boldsymbol{e}(k)\boldsymbol{e}^{\mathrm{T}}(k)\right]$. Recall that the second moment for a random variable is greater than or equal to the corresponding central moment (variance). Equality is achieved only in the case of zero mean. It follows that in order to minimize the sum of the second moments of each component of the estimation error $\mathbb{E}\left[\boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k)\right]$, the observer needs to be unbiased so that $\mathbb{E}\left[\boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k)\right]$ is the sum of the variances. Conditions (8.36a)-(8.36e) needs then to be satisfied and $\boldsymbol{\Pi}(k+1)$ can be expressed

as

$$\begin{aligned}\mathbf{\Pi}(k+1) = \bar{\mathbf{A}}\mathbf{\Pi}(k)\bar{\mathbf{A}}^{\mathrm{T}} + \mathbf{M}_1\mathbf{R}\mathbf{M}_1^{\mathrm{T}} + \mathbf{Q} \\ + \left(\bar{\mathbf{A}}\mathbf{\Pi}(k)\bar{\mathbf{N}}^{\mathrm{T}} + \bar{\mathbf{N}}\mathbf{\Pi}(k)\bar{\mathbf{A}}^{\mathrm{T}} + \mathbf{M}_1\mathbf{R}\mathbf{M}_2^{\mathrm{T}} + \mathbf{M}_2\mathbf{R}\mathbf{M}_1^{\mathrm{T}}\right) u(k) \\ + \left(\bar{\mathbf{N}}\mathbf{\Pi}(k)\bar{\mathbf{N}}^{\mathrm{T}} + \mathbf{M}_2\mathbf{R}\mathbf{M}_2^{\mathrm{T}}\right) u^2(k) \end{aligned} \tag{8.42}$$

From equation (8.42) we notice that, even after the transient has vanished, the covariance of the state estimation error changes in time due to its dependence on the input. In contrast with the linear case, to minimize the trace of $\mathbf{\Pi}(k)$ at every $k \geq p$, time-varying gains $\mathbf{M}_1(k)$ and $\mathbf{M}_2(k)$ would be necessary also in steady state. Since the focus of this work is on time-invariant bilinear observers, we relegate the analysis of the case with time-varying gains to the appendix. In what follows we seek to answer the question of what optimality can be achieved with constant gains.

Lemma 8.3. *The observer in (8.34) minimizes the time average* $\bar{\mathbf{\Pi}}$ *of the expected value of the norm squared of the state estimation error* $\mathbf{\Pi}(k)$ *if and only if the observer is unbiased (Lemma 8.2), the input is stationary and white, and* $\mathbf{M}_1$ *and* $\mathbf{M}_2$ *satisfy the following conditions*

$$\bar{\mathbf{A}}\bar{\mathbf{\Pi}}\mathbf{C}^{\mathrm{T}} - \mathbf{M}_1\mathbf{R} = \mathbf{0} \tag{8.43a}$$

$$\bar{\mathbf{N}}\bar{\mathbf{\Pi}}\mathbf{C}^{\mathrm{T}} - \mathbf{M}_2\mathbf{R} = \mathbf{0} \tag{8.43b}$$

Proof. We observed from equation (8.42) that $\mathbf{\Pi}(k)$ changes with time but if the input is white and stationary (constant mean μ and second moment β^2), then $\mathbf{\Pi}(k)$ is a stationary process

and its steady-state time-average $\bar{\mathbf{\Pi}}$ satisfies the following equation

$$\begin{aligned}\bar{\mathbf{\Pi}} = \bar{\mathbf{A}}\bar{\mathbf{\Pi}}\bar{\mathbf{A}}^{\mathrm{T}} + \mathbf{M}_1\mathbf{R}\mathbf{M}_1^{\mathrm{T}} + \mathbf{Q} \\ + \left(\bar{\mathbf{A}}\bar{\mathbf{\Pi}}\bar{\mathbf{N}}^{\mathrm{T}} + \bar{\mathbf{N}}\bar{\mathbf{\Pi}}\bar{\mathbf{A}}^{\mathrm{T}} + \mathbf{M}_1\mathbf{R}\mathbf{M}_2^{\mathrm{T}} + \mathbf{M}_2\mathbf{R}\mathbf{M}_1^{\mathrm{T}}\right)\mu \\ + \left(\bar{\mathbf{N}}\bar{\mathbf{\Pi}}\bar{\mathbf{N}}^{\mathrm{T}} + \mathbf{M}_2\mathbf{R}\mathbf{M}_2^{\mathrm{T}}\right)\beta^2 \end{aligned} \tag{8.44}$$

We can now think of choosing $\mathbf{M}_1$ and $\mathbf{M}_2$ such that the trace of $\bar{\mathbf{\Pi}}$ is minimized. Imposing the first-order conditions, from equation (8.44) we obtain

$$\frac{\partial \operatorname{trace} \bar{\mathbf{\Pi}}}{\partial \mathbf{M}_1} = -2\left(\bar{\mathbf{A}}\bar{\mathbf{\Pi}}\mathbf{C}^{\mathrm{T}} - \mathbf{M}_1\mathbf{R}\right) - 2\left(\bar{\mathbf{N}}\bar{\mathbf{\Pi}}\mathbf{C}^{\mathrm{T}} - \mathbf{M}_2\mathbf{R}\right)\mu = \mathbf{0} \tag{8.45a}$$

$$\frac{\partial \operatorname{trace} \bar{\mathbf{\Pi}}}{\partial \mathbf{M}_2} = -2\left(\bar{\mathbf{N}}\bar{\mathbf{\Pi}}\mathbf{C}^{\mathrm{T}} - \mathbf{M}_2\mathbf{R}\right)\beta^2 - 2\left(\bar{\mathbf{A}}\bar{\mathbf{\Pi}}\mathbf{C}^{\mathrm{T}} - \mathbf{M}_1\mathbf{R}\right)\mu = \mathbf{0} \tag{8.45b}$$

which, to hold for arbitrary μ and β^2, lead to the two optimality conditions (8.43). □

8.5.3 Properties of the residuals

Lemma 8.4. *The expected value of the output residuals of the observer in (8.34) is zero at all $k \geq p$ if and only if the observer state is an unbiased estimate of the system state (Lemma 8.2).*

Proof. From the definition of observer residual, we can write

$$\boldsymbol{\epsilon}(k) = \boldsymbol{y}(k) - \hat{\boldsymbol{y}}(k) = \mathbf{C}\boldsymbol{e}(k) + \boldsymbol{w}_m(k) \tag{8.46}$$

Taking the expectation (ensemble average) of both sides of (8.46) we obtain

$$\mathbb{E}\left[\boldsymbol{\epsilon}(k)\right] = \mathbf{C}\mathbb{E}\left[\boldsymbol{e}(k)\right] \tag{8.47}$$

and for the same conditions that guarantee that the observer in (8.34) is unbiased (Lemma 8.2), $\mathbb{E}[\boldsymbol{\epsilon}(k)] = \mathbf{0}$ at all $k \geq p$, i.e. for all k in steady state. □

Lemma 8.5. *The output residuals of the observer in (8.34) in steady state form a white sequence, i.e. the time average* $\bar{\mathbf{W}}_\epsilon(j)$ *of* $\mathbf{W}_\epsilon(k,j) = \mathbb{E}\left[\boldsymbol{\epsilon}(k+j)\boldsymbol{\epsilon}^{\mathrm{T}}(k)\right]$ *for* $k \geq p$ *is zero for* $j \geq 1$*, if and only if the observer is optimal (Lemma 8.3).*

Proof. We prove here Lemma 8.5 for $j = 1$, the proof for other values of j follows the same lines. From equations (8.46) and (8.5), we can write $\mathbf{W}_\epsilon(k,1)$ as

$$\begin{aligned}
\mathbb{E}\left[\boldsymbol{\epsilon}(k+1)\boldsymbol{\epsilon}^{\mathrm{T}}(k)\right] &= \mathbb{E}\left[\left(\mathbf{C}\boldsymbol{e}(k+1) + \boldsymbol{w}_m(k+1)\right)\left(\mathbf{C}\boldsymbol{e}(k) + \boldsymbol{w}_m(k)\right)\right] \\
&= \mathbb{E}\Big[\left(\mathbf{C}\bar{\mathbf{A}}\boldsymbol{e}(k) + \mathbf{C}\bar{\mathbf{N}}\boldsymbol{e}(k)u(k) + \mathbf{C}\boldsymbol{w}_p(k) - \mathbf{C}\mathbf{M}_1\mathbf{w}_m(k)\right. \\
&\qquad \left. - \mathbf{C}\mathbf{M}_2\boldsymbol{w}_m(k)u(k) + \boldsymbol{w}_m(k+1)\right)\left(\mathbf{C}\boldsymbol{e}(k) + \boldsymbol{w}_m(k)\right)\Big] \\
&= \mathbf{C}\left(\bar{\mathbf{A}}\boldsymbol{\Pi}(k)\mathbf{C}^{\mathrm{T}} - \mathbf{M}_1\mathbf{R}\right) + \mathbf{C}\left(\bar{\mathbf{N}}\boldsymbol{\Pi}(k)\mathbf{C}^{\mathrm{T}} - \mathbf{M}_2\mathbf{R}\right)u(k) \qquad (8.48)
\end{aligned}$$

The assumption of stationary white excitation input allows us to write the time average of $\mathbf{W}_\epsilon(k,1)$ as

$$\bar{\mathbf{W}}_\epsilon(1) = \mathbf{C}\left(\bar{\mathbf{A}}\bar{\boldsymbol{\Pi}}\mathbf{C}^{\mathrm{T}} - \mathbf{M}_1\mathbf{R}\right) + \mathbf{C}\left(\bar{\mathbf{N}}\bar{\boldsymbol{\Pi}}\mathbf{C}^{\mathrm{T}} - \mathbf{M}_2\mathbf{R}\right)\mu \qquad (8.49)$$

which vanishes under the optimality conditions (8.43). □

Lemma 8.6. *The output residuals of the observer in (8.34) in steady state are orthogonal to the current and past input values, i.e. the ensemble average* $\mathbf{W}_u(k,j) = \mathbb{E}[\boldsymbol{\epsilon}(k)u(k-j)]$ *for all* $k \geq p$ *is zero for* $j \geq 0$*, if and only if the observer is unbiased (Lemma 8.2).*

Proof. For $j = 0$, starting from equation (8.46), we can write $\mathbf{W}_u(k, 0)$ as

$$\mathbb{E}\left[\boldsymbol{\epsilon}(k)u(k)\right] = \mathbf{C}\mathbb{E}\left[\boldsymbol{\epsilon}(k)\right]u(k) \tag{8.50}$$

which vanishes for zero-mean output residuals, i.e. if and only if the observer state is an unbiased estimate for the system state (Lemma 8.4). Similarly, for $j > 0$ we obtain

$$\mathbb{E}\left[\boldsymbol{\epsilon}(k)u(k-j)\right] = \mathbf{C}\mathbb{E}\left[\boldsymbol{\epsilon}(k)\right]u(k-j) \tag{8.51}$$

with the same conclusion. □

Lemma 8.7. *The output residuals of the observer in (8.34) in steady state are orthogonal to the past output values, i.e. the time average* $\bar{\mathbf{W}}_y(j)$ *of* $\mathbf{W}_y(k, j) = \mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{y}^{\mathrm{T}}(k-j)\right]$ *for* $k \geq p$ *is zero for* $j \geq 1$*, if and only if the observer is optimal (Lemma 8.3).*

Proof. We prove Lemma 8.7 below for $j = 1$, the proof for other values of j follows the same lines, therefore it is omitted. From equation (8.46) and considering an unbiased observer (Lemma 8.2), we can write $\mathbf{W}_y(k, 1)$ as

$$\begin{aligned}
\mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{y}^{\mathrm{T}}(k-1)\right] &= \mathbb{E}\left[\left(\mathbf{C}\boldsymbol{e}(k) + \boldsymbol{w}_m(k)\right)\boldsymbol{y}^{\mathrm{T}}(k-1)\right] = \mathbb{E}\left[\mathbf{C}\boldsymbol{e}(k)\boldsymbol{y}^{\mathrm{T}}(k-1)\right] \\
&= \mathbf{C}\mathbb{E}\Big[\big(\bar{\mathbf{A}}\boldsymbol{e}(k-1) + \bar{\mathbf{N}}\boldsymbol{e}(k-1)u(k-1) + \boldsymbol{w}_p(k-1) - \mathbf{M}_1\boldsymbol{w}_m(k-1) \\
&\qquad - \mathbf{M}_2\boldsymbol{w}_m(k-1)u(k-1)\big)\big(\mathbf{C}\hat{\boldsymbol{x}}(k-1) + C\boldsymbol{e}(k-1) \\
&\qquad + \mathbf{D}u(k-1) + \boldsymbol{w}_m(k-1)\big)^{\mathrm{T}}\Big] \\
&= \mathbf{C}\bar{\mathbf{A}}\mathbb{E}\left[\boldsymbol{e}(k-1)\hat{\boldsymbol{x}}^{\mathrm{T}}(k-1)\right]\mathbf{C}^{\mathrm{T}} + \mathbf{C}\bar{\mathbf{N}}\mathbb{E}\left[\boldsymbol{e}(k-1)\hat{\boldsymbol{x}}^{\mathrm{T}}(k-1)\right]\mathbf{C}^{\mathrm{T}}u(k-1) \\
&\quad + \mathbf{C}\left(\bar{\mathbf{A}}\boldsymbol{\Pi}(k-1)\mathbf{C}^{\mathrm{T}} - \mathbf{M}_1\mathbf{R}\right) + \mathbf{C}\left(\bar{\mathbf{N}}\boldsymbol{\Pi}(k-1)\mathbf{C}^{\mathrm{T}} - \mathbf{M}_2\mathbf{R}\right)u(k-1)
\end{aligned} \tag{8.52}$$

Assuming stationary white input, defining $\bar{\mathbf{W}}_{\hat{x}}$ as the time average of $\mathbf{W}_{\hat{x}} = \mathbb{E}\left[\boldsymbol{e}(k)\hat{\boldsymbol{x}}^{\mathrm{T}}(k)\right]$ and taking the time average of equation (8.52), we obtain

$$\begin{aligned}\bar{\mathbf{W}}_y(1) = \mathbf{C}\bar{\mathbf{A}}\bar{\mathbf{W}}_{\hat{x}}\mathbf{C}^{\mathrm{T}} + \mathbf{C}\bar{\mathbf{N}}\bar{\mathbf{W}}_{\hat{x}}\mathbf{C}^{\mathrm{T}}\mu \\ + \mathbf{C}\left(\bar{\mathbf{A}}\bar{\mathbf{\Pi}}\mathbf{C}^{\mathrm{T}} - \mathbf{M}_1\mathbf{R}\right) + \mathbf{C}\left(\bar{\mathbf{N}}\bar{\mathbf{\Pi}}\mathbf{C}^{\mathrm{T}} - \mathbf{M}_2\mathbf{R}\right)\mu\end{aligned} \tag{8.53}$$

which vanishes for the optimal observer of Lemma 8.3. The first two terms on the right-hand side of equation (8.53) are null because so is $\bar{\mathbf{W}}_{\hat{x}}$ by the orthogonality argument coming from the least-squares optimality criterion, the last two terms vanish thanks to the optimality conditions (8.43). □

Lemma 8.8. *The output residuals of the observer in (8.34) in steady state are orthogonal to the current $\boldsymbol{z}_p$ defined below equation (8.8), i.e. the time average $\bar{\mathbf{W}}_z$ of $\mathbf{W}_z(k) = \mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{z}_p^{\mathrm{T}}(k)\right]$ for $k \geq p$ is zero, if and only if the observer is optimal (Lemma 8.3).*

Proof. Lemma 8.8 can be proven entry by entry of $\boldsymbol{z}_p(k)$. Consider the remaining entries of $\boldsymbol{v}(k)$ not yet taken into account in Lemma 8.6 and Lemma 8.7. In steady state

$$\mathbb{E}\left[\boldsymbol{\epsilon}(k)u^2(k-j)\right] = \mathbb{E}\left[\boldsymbol{\epsilon}(k)\right]u^2(k-j) \tag{8.54}$$

which is null for any $j > 0$ under the same conditions that guarantee the output residuals to be a zero-mean process (Lemma 8.4), and so is its time average. Similarly, the time average of

$$\mathbb{E}\left[\boldsymbol{\epsilon}(k)u(k-j)\boldsymbol{y}^{\mathrm{T}}(k-j)\right] = \mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{y}^{\mathrm{T}}(k-j)\right]u(k-j) \tag{8.55}$$

is zero because so is the time average $\bar{\mathbf{W}}_y(j)$ of $\mathbf{W}_y(k,j) = \mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{y}^{\mathrm{T}}(k-j)\right]$ for $j \geq 1$ (Lemma 8.7). Together with Lemma 8.6 and Lemma 8.7 themselves, this proves that the time

average $\bar{\mathbf{W}}_v(j)$ of $\mathbf{W}_v(k,j) = \mathbb{E}\left[\boldsymbol{\epsilon}(k)\boldsymbol{v}(k-j)\right]$ is null for any $j > 0$ if and only if the observer is optimal (Lemma 8.3). All the other entries of $\boldsymbol{z}_p(k)$ involves products of $\boldsymbol{v}(j)$, $j = 1, 2, ..., p$, with input values. It is then straightforward to show that $\bar{\mathbf{W}}_z = \mathbf{0}$. □

8.6 Examples

Numerical examples are provided to support the theoretical findings and to provide additional insights. State and output data $\{\boldsymbol{x}(k)\}$, $\{\boldsymbol{y}(k)\}$ are generated from random input sequences $\{u(k)\}$ that are uniformly distributed between -0.5 and 0.5. For the stochastic case, zero-mean Gaussian process and measurement noises are added with the following covariances $\mathbf{Q}$ and R, respectively

$$\mathbf{Q} = \begin{bmatrix} 0.01 & 0.005 \\ 0.005 & 0.0025 \end{bmatrix} \qquad R = 0.04$$

Note that the generated input and noise satisfy the assumptions made in section 8.5 to derive the properties of the optimal bilinear observer in the stochastic case.

Three systems are used in the illustration. System I (ideal bilinear) is defined by

$$\mathbf{A} = \begin{bmatrix} -0.5 & 0.5 \\ 0.5 & 0 \end{bmatrix} \quad \mathbf{N} = \begin{bmatrix} 1 & -1 \\ 1 & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} 2 \\ 1 \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad \mathbf{D} = 0$$

System II (non-ideal bilinear) is modified from System I with the $(2,2)$ element of $\mathbf{N}$ set to 0.3, which is sufficient to turn it into a non-ideal bilinear model. Systems I and II are the same illustrated in the examples in chapter 6. System III (linear) is modified from System I by setting $\mathbf{N} = \mathbf{0}$.

8.6.1 Deterministic bilinear observer

For System I, $p = 2$ is the smallest value for which $\mathbf{T}_p$ can be found to satisfy equation (8.12) exactly ($\mathbf{E} = \mathbf{0}$ when $\mathbf{T}_2$ is solved for). This is an indication that a deadbeat observer exists for this system. The identified gains associated with this minimum value of $p = 2$ are extracted from $\mathbf{T}_2$ as

$$\mathbf{M}_1 = \begin{bmatrix} -0.5 & 0.5 \end{bmatrix}^{\mathrm{T}} \qquad \mathbf{M}_2 = \begin{bmatrix} 1 & 1 \end{bmatrix}^{\mathrm{T}}$$

These bilinear observer gains can be verified to cause $\bar{\mathbf{A}}^2 = \bar{\mathbf{A}}\bar{\mathbf{N}} = \bar{\mathbf{N}}\bar{\mathbf{A}} = \bar{\mathbf{N}}^2 = \mathbf{0}$. The state estimation error converges to zero in 2 time steps for any input as seen in Figure 8.1. For System II, the smaller singular values of $\mathbf{Z}_p$ decrease gradually as p increases, and a solution for $\mathbf{T}_p$ that satisfies equation (8.12) with $\mathbf{E} = \mathbf{0}$ does not exist. This is a certain indication that a deadbeat observer does not exist for this system. Indeed, the gains identified by equation (8.14) do not make the products of $\bar{\mathbf{A}}$ and $\bar{\mathbf{N}}$ whose sum of powers added up to p vanish. Instead, these gains minimize the Frobenius norm of the state estimation error matrix $\mathbf{E}$ in equation (8.12).

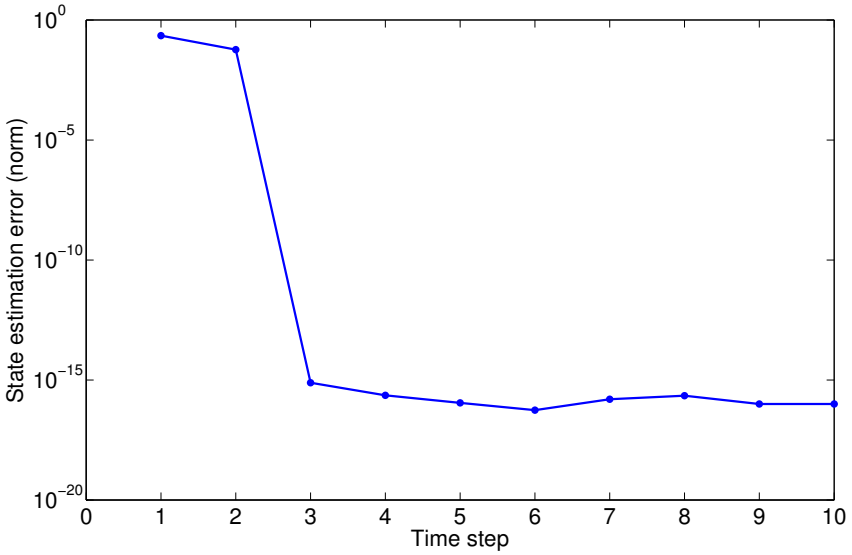


FIGURE 8.1: The state estimation error of the observer designed for System I (deterministic case) converges to zero in exactly 2 time steps. The designed observer is deadbeat.

Although the identified observer is not deadbeat, numerical results confirm that it is the fastest observer when compared to observers where the identified gains are perturbed. The case for $p = 6$ is illustrated in Figure 8.2, where for every perturbed observer a random value is added to each entry of the gains in order to randomly generate perturbation in all directions in the gain space. Note that both Figure 8.1 and 8.2 are obtained by driving the bilinear system with an input sequence generated independently from the one used for gain design.

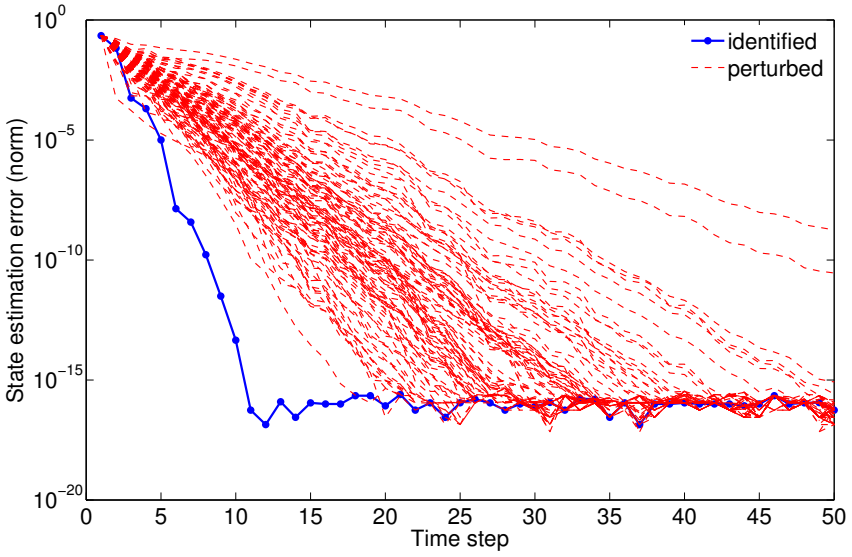


FIGURE 8.2: The observer designed for System II (deterministic case) has the fastest convergence compared to those with perturbed gains.

TABLE 8.1: Comparison of observer gains and mean-norm error obtained from Kalman theory, IOSR-based method and iterative technique (System III).

	Kalman	Non-iterative (IOSR)	Iterative
$\mathbf{M}_1(1)$	-0.085757111	-0.086211775	-0.086133466
$\mathbf{M}_1(2)$	0.117193716	0.117111690	0.117122168
$\|\|\mathbf{M}_1\|\|$	0.145219314	0.145422206	0.145384236
e_{MN}	0.113100785	0.113100750	0.113100749

8.6.2 Stochastic bilinear observer

In the stochastic case, we first confirm that the proposed observer identification techniques indeed reproduce the well-known steady-state Kalman filter gain for a linear system (System III). Both the non-iterative (IOSR-based) and the iterative techniques identify the Kalman filter gain correctly. The two elements of $\mathbf{M}_1$ ($\mathbf{M}_2 = 0$) are shown in Table 8.1. Since the design objective is to minimize the expected value of the norm of the state estimation error, an appropriate performance measure is the mean-norm state estimation error $e_{MN} = (l+1)^{-1}\sum_{k=0}^{l}\left(\boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k)\right)^{1/2}$, which in the limit as $l \to \infty$ converges to the expected value of the norm of the state estimation error. We use $p = 40$ for the IOSR-based method. To identify the Kalman filter gain exactly, an infinitely long data set would be necessary. To avoid the obvious computational issues, the design is performed by averaging the identified gains from 100 independent data sets of 10^4 samples each. For evaluation, both in this case and later in the bilinear case, another 100 independent data sets of 10^4 samples are used, and the averaged e_{MN} values are reported. Note that the relatively large value of $p = 40$ used in the IOSR-based approach guarantees that $\mathbf{S}_p$ is negligible. With the identified observer gain, $\mathbf{S}_{40} = \bar{\mathbf{A}}^{40}$ has entries whose magnitudes are of the order of 10^{-7}.

We evaluate the effectiveness of the proposed design methods by analyzing the actual performance of the identified bilinear observers. Table 8.2 compares the e_{MN} of the observers designed by the IOSR-based technique (again from another 100 independent data sets of 10^4 samples each) for different values of p. As expected, increasing p leads to better identification,

TABLE 8.2: Improvement of mean-norm error for the bilinear observer designed by non-iterative (IOSR-based) method, as p increases (System II).

p	2	3	4	5	6
# Z_p rows	12	28	60	124	252
e_{MN}	0.15292	0.12592	0.12164	0.12022	0.11988

since a larger p improves the approximation of equation (8.10). For $p = 7$, the identified gains are

$$\mathbf{M}_{1,IOSR} = \begin{bmatrix} -0.09792 \\ 0.12945 \end{bmatrix} \qquad \mathbf{M}_{2,IOSR} = \begin{bmatrix} 0.19703 \\ 0.27786 \end{bmatrix}$$

In the linear case, the number of rows of $\mathbf{Z}_p$ grows linearly with p. In the bilinear case, the growth is exponential, hence the iterative solution is called for. The bilinear observer gains based on the iterative solution are found to be

$$\mathbf{M}_{1,ITER} = \begin{bmatrix} -0.09137 \\ 0.12509 \end{bmatrix} \qquad \mathbf{M}_{2,ITER} = \begin{bmatrix} 0.18319 \\ 0.27114 \end{bmatrix}$$

The e_{MN} of the observer with gains designed by the iterative technique is 0.11975. For the IOSR-based design, it is 0.11978. The slightly larger error for the latter can be explained by the fact that $p = 7$ is not sufficiently large to make $\mathbf{S}_7(k)$ negligible. For instance, the entries of $(\mathbf{A} - \mathbf{M}_1\mathbf{C})^7$ and $(\mathbf{N} - \mathbf{M}_2\mathbf{C})^7$ are of the order of 10^{-2}. A larger p would improve the approximation of equation (8.10), but computational and ill-conditioning issues might in practice limit p. As a confirmation we compare the state estimation error of the observer found via the iterative method to the observers whose gains are perturbed from the values given above (again, random and independent perturbation for each entry of the gain vectors, with standard deviation equal to 30% of the corresponding gain entry). The resulting e_{MN} values, obtained by averaging 100 tests of 10^4 samples each, are shown in Figure 8.3. The observer designed via the iterative technique performs better than all 100 observers with randomly perturbed gains. The state estimation error is minimized. In fact, this identified bilinear observer for System II can legitimately be considered as being optimal or very close to it. Finally, fast convergence is observed for the proposed iterative technique. In the above examples, in fewer than 100 iterations, the gains in successive iterations converge with a relative difference of about 10^{-16}

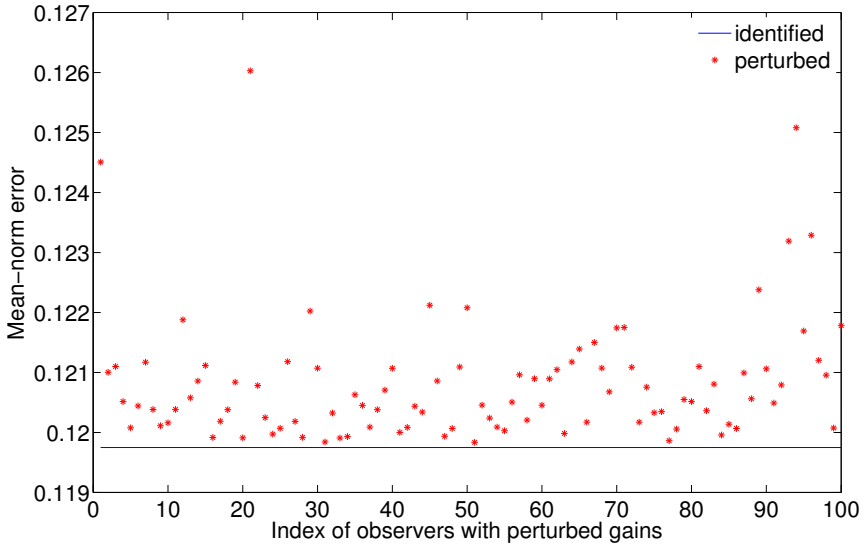


FIGURE 8.3: The observer designed for System II (stochastic case) has minimum state estimation error compared to those with perturbed gains.

in order of magnitude, which is numerically zero by Matlab® double precision calculation.

Additionally, Fig. 8.4 shows how the residuals of the bilinear observer designed via the iterative method are white, as expected since the gains and the excitation input satisfy the requirements of Lemma 8.5. The state estimation error is not expected to be white. These results exactly parallel those of the optimal Kalman filter in the linear case, and the only required additional assumption is to have stationary and white input.

8.6.3 Numerical validation of the stochastic properties of the optimal bilinear observer

In order to verify the theory developed in section 8.5, the validity of equations (8.43) and (8.44) is illustrated below via the same numerical example on system II presented above (section 8.6.2). Generate a sequence of input u made of 10^4 values independently drawn from a uniform distribution between -0.5 and 0.5 and follow the steps below. Note again that the generated input

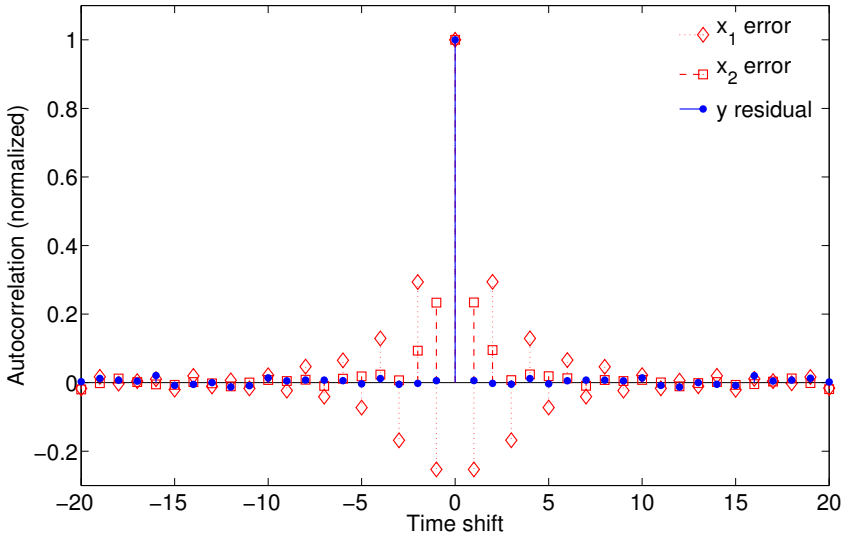


FIGURE 8.4: The autocorrelation of state estimation error and output residual for System II (stochastic case). The output residual is white.

satisfies the assumptions made in section 8.5 to derive the properties of the optimal bilinear observer.

8.6.3.1 Gain design on single dataset

Run for 100 times the following steps (the superscript (i) denotes the index of the run)

1. Generate the sequences of $\boldsymbol{w}_p^{(i)}$ and $w_m^{(i)}$ as gaussian noise with covariance matrices in (8.56)

2. Simulate the system dynamics according to equation (8.1) to get the sequences of the system state $\boldsymbol{x}^{(i)}$ and output $y^{(i)}$

3. Use u, $\boldsymbol{x}^{(i)}$, $y^{(i)}$ to design via the iterative method the gains $\mathbf{M}_1^{(i)}$ and $\mathbf{M}_2^{(i)}$ of the bilinear observer in equation (8.2)

4. Run the designed observer on the same dataset used for its design to obtain the sequence of $\hat{\boldsymbol{x}}^{(i)}$

5. Estimate the time average of the expected state error covariance as the covariance $\bar{\mathbf{\Pi}}^{(i)}$ of the sequence of $\boldsymbol{e}^{(i)} = \boldsymbol{x}^{(i)} - \hat{\boldsymbol{x}}^{(i)}$. This is an estimate of the time-average of the expected error covariance matrix, computed by the ergodic property as a simple time average.

6. Compute the gains $\mathbf{M}_{1,the}^{(i)}$ and $\mathbf{M}_{2,the}^{(i)}$ that one would obtain from the optimality conditions (8.43) using the estimated $\bar{\mathbf{\Pi}}^{(i)}$

$$\mathbf{M}_{1,the}^{(i)} = \mathbf{A}\bar{\mathbf{\Pi}}^{(i)}\mathbf{C}^{\mathrm{T}}\left(\mathbf{C}\bar{\mathbf{\Pi}}^{(i)}\mathbf{C}^{\mathrm{T}} + R\right)^{-1} \tag{8.56a}$$

$$\mathbf{M}_{2,the}^{(i)} = \mathbf{N}\bar{\mathbf{\Pi}}^{(i)}\mathbf{C}^{\mathrm{T}}\left(\mathbf{C}\bar{\mathbf{\Pi}}^{(i)}\mathbf{C}^{\mathrm{T}} + R\right)^{-1} \tag{8.56b}$$

   and compare them with the designed gains (Figures 8.5 and 8.6)

7. Evaluate the right-hand side of Eq. (8.44) using $\bar{\mathbf{\Pi}}^{(i)}$, $\mathbf{M}_1^{(i)}$ and $\mathbf{M}_2^{(i)}$ in the place of $\bar{\mathbf{\Pi}}$, $\mathbf{M}_1$ and $\mathbf{M}_2$ to obtain $\bar{\mathbf{\Pi}}_{RHS}^{(i)}$ and compare it with $\bar{\mathbf{\Pi}}^{(i)}$ (Figure 8.7)

Figures 8.5, 8.6, and 8.7 show indeed good agreement. Exact match of the displayed variables and invariance across different runs cannot be achieved due to the stochastic nature of the problem and the finiteness of each run dataset.

8.6.3.2 Gain design on multiple datasets

1. Improve the design of the gains averaging the values obtained over the 100 runs in section 8.6.3.1, to get $\mathbf{M}_1$ and $\mathbf{M}_2$

2. Estimate $\mathbf{\Pi}(k)$ according to its definition, i.e. as the average of $\boldsymbol{e}^{(i)}(k)\left(\boldsymbol{e}^{(i)}(k)\right)^{\mathrm{T}}$ over i

3. Estimate $\bar{\mathbf{\Pi}}$ as the average of $\mathbf{\Pi}(k)$ over k

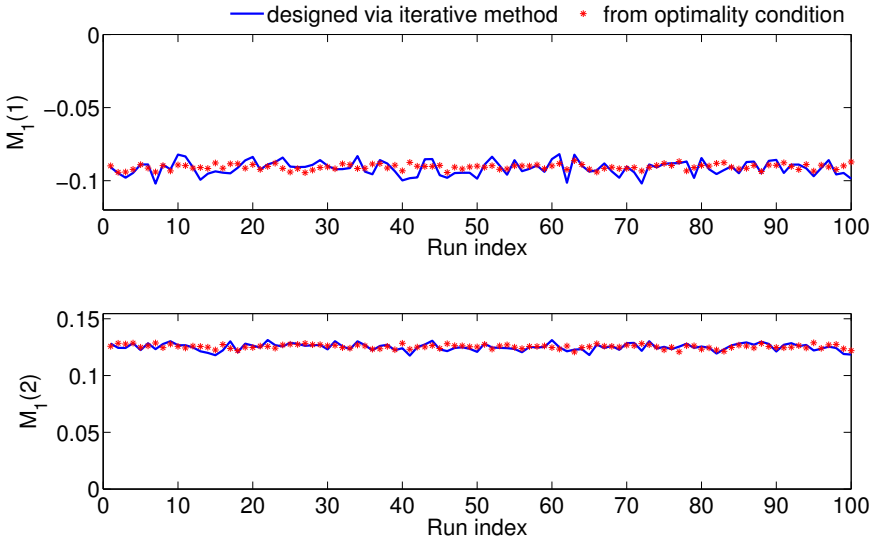


FIGURE 8.5: Comparison between $\mathbf{M}_1^{(i)}$ and $\mathbf{M}_{1,the}^{(i)}$ at each run.

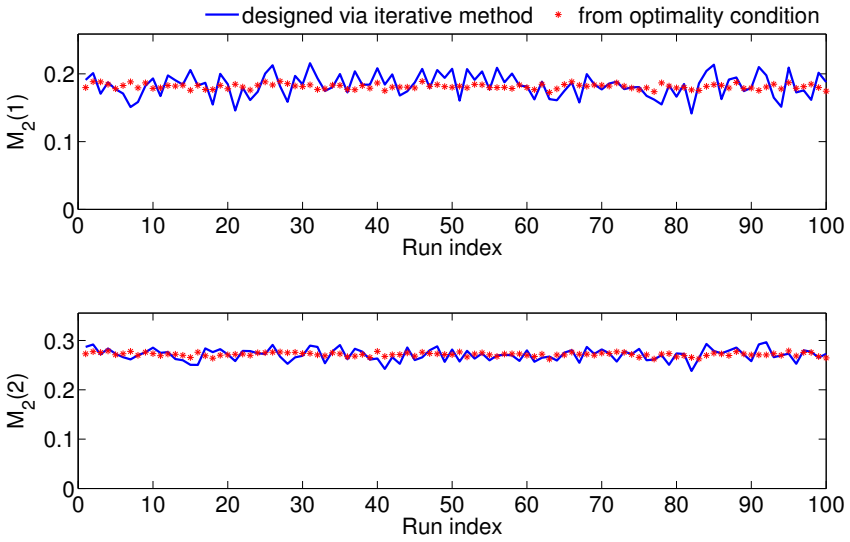


FIGURE 8.6: Comparison between $\mathbf{M}_2^{(i)}$ and $\mathbf{M}_{2,the}^{(i)}$ at each run.

4. Compute the gains $\mathbf{M}_{1,the}$ and $\mathbf{M}_{2,the}$ that one would obtain from the optimality conditions in equation (8.43) using the estimated $\bar{\mathbf{\Pi}}$

$$\mathbf{M}_{1,the} = \mathbf{A}\bar{\mathbf{\Pi}}\mathbf{C}^{\mathrm{T}}\left(\mathbf{C}\bar{\mathbf{\Pi}}\mathbf{C}^{\mathrm{T}} + R\right)^{-1} \tag{8.57a}$$

$$\mathbf{M}_{2,the} = \mathbf{N}\bar{\mathbf{\Pi}}\mathbf{C}^{\mathrm{T}}\left(\mathbf{C}\bar{\mathbf{\Pi}}\mathbf{C}^{\mathrm{T}} + R\right)^{-1} \tag{8.57b}$$

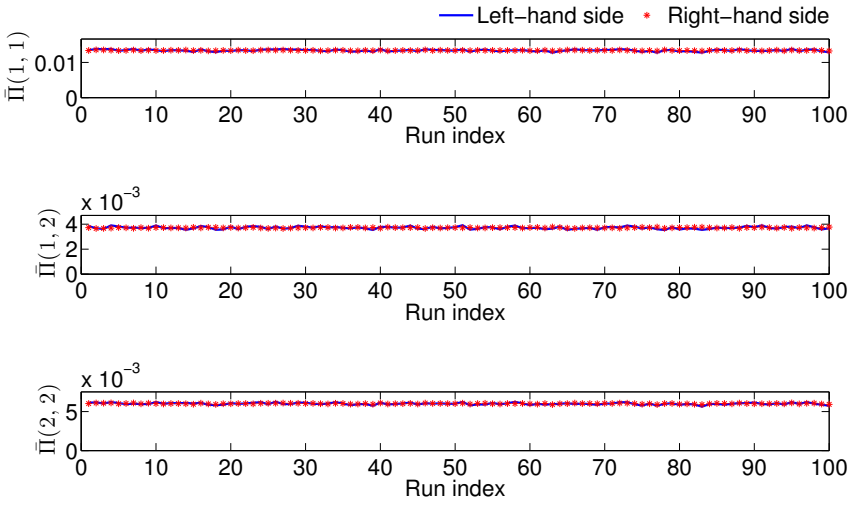


FIGURE 8.7: Comparison between left-hand and right-hand side of equation (8.44) at each run.

and compare them with the designed gains

$$\mathbf{M}_1 = \begin{bmatrix} -0.09137 \\ 0.12509 \end{bmatrix} \qquad \mathbf{M}_2 = \begin{bmatrix} 0.18319 \\ 0.27114 \end{bmatrix} \tag{8.58a}$$

$$\mathbf{M}_{1,the} = \begin{bmatrix} -0.09087 \\ 0.12559 \end{bmatrix} \qquad \mathbf{M}_{2,the} = \begin{bmatrix} 0.18174 \\ 0.27202 \end{bmatrix} \tag{8.58b}$$

5. Evaluate the right-hand side of equation (8.44) using $\bar{\mathbf{\Pi}}$, $\mathbf{M}_1$ and $\mathbf{M}_2$ to obtain $\bar{\mathbf{\Pi}}_{RHS}$ and compare it with $\bar{\mathbf{\Pi}}$

$$\bar{\mathbf{\Pi}} = \begin{bmatrix} 0.01342 & 0.00371 \\ 0.00371 & 0.00601 \end{bmatrix} \qquad \bar{\mathbf{\Pi}}_{RHS} = \begin{bmatrix} 0.01341 & 0.00371 \\ 0.00371 & 0.00603 \end{bmatrix} \tag{8.59}$$

The numerical results in (8.58) and (8.59) confirm the validity of the approach and in particular the correctness of equations (8.43) and (8.44).

8.7 Discussion

The presented observer design strategy can be applied to any bilinear system with minimum observability requirements (observable linear part) and to both the deterministic and stochastic case. The magnitude of the input to ensure stability of the observer in operations is usually subject to a bound, which goes to infinity for deterministic ideal bilinear systems. Note that instability under large input is an intrinsic characteristic of bilinear models, therefore it affects both the system and its observer. The introduction of the gains $\mathbf{M}_1$ and $\mathbf{M}_2$ generally allows one to enhance the observer stability over the system stability.

In the deterministic case, when the bilinear system is ideal the resulting observer is of deadbeat type. It converges to zero error in the state estimate in exactly n time steps, independently of the input. It is worth making a connection with the work in Reference 65 on reduced-order observers. If their method was applied to the example given on System I, it would turn out that a state-independent first-order observer would exist and have the eigenvalue fixed at the origin (deadbeat). Reference 65 also gave a general sufficient condition for the existence of a reduced-order bilinear observer whose eigenvalues can be arbitrarily chosen. The condition requires the existence of a similarity transformation $\boldsymbol{x}' = \mathbf{M}\boldsymbol{x}$ such that the given bilinear system can be written in the form

$$\boldsymbol{x}'(k+1) = \begin{bmatrix} \mathbf{A}'_{11} & \mathbf{A}'_{12} \\ \mathbf{A}'_{21} & \mathbf{A}'_{22} \end{bmatrix} \boldsymbol{x}'(k) + \begin{bmatrix} \mathbf{N}'_{11} & \mathbf{0}_{q,n-q} \\ \mathbf{N}'_{21} & \mathbf{0}_{n-q,n-q} \end{bmatrix} \boldsymbol{x}'(k)u(k) + \begin{bmatrix} \mathbf{B}'_1 \\ \mathbf{B}'_2 \end{bmatrix} u(k) \tag{8.60a}$$

$$\boldsymbol{y}(k) = \begin{bmatrix} \mathbf{I}_q & \mathbf{0}_{q,n-q} \end{bmatrix} \boldsymbol{x}'(k) + \mathbf{D}u(k) \tag{8.60b}$$

where $\mathbf{0}_{r,s}$ indicates a null matrix of dimension $r \times s$ and $\mathbf{I}_r$ is an identity matrix of dimension $r \times r$. It is straightforward to verify that the design technique proposed in this chapter would

yield a deadbeat gain for a system described by equation (8.60). Indeed, an interaction matrix (or observer gain) $\mathbf{M}_2$ would exist to make $\bar{\mathbf{N}}$ null, so that all the matrix products in $\mathbf{S}_p$ would be identically $\mathbf{0}$ except from $\bar{\mathbf{A}}^p$. The observability of the pair $(\mathbf{A}, \mathbf{C})$ ensures the existence of $\mathbf{M}_1$ to make $\bar{\mathbf{A}}^p = \mathbf{0}$. Reference 65 only studied bilinear observers with error dynamics independent of the input and indeed $\bar{\mathbf{N}} = \mathbf{0}$ in equation (8.6) eliminates any effect of u on $\boldsymbol{e}$. In summary, if a reduced-order deadbeat observer exists, the proposed design method will accordingly yield a full-order deadbeat observer.

When the bilinear system is not ideal, the proposed design approach can still be applied even though a deadbeat observer does not exist. The result is the fastest possible converging bilinear observer. Since the design procedure is done on a specific input sequence, the observer will preserve the property when the system is driven by an input with the same characteristics of the input used to design the gains. For other input profiles (e.g. different amplitude or different frequency spectrum), the observer can still be used but will in general be suboptimal in the sense that it might not be the fastest possible observer. Stability is guaranteed as long as the magnitude of the input is kept smaller than the maximum value satisfying equation (6.41). With reference to the work in Reference 60, the observer proposed in the present chapter can be interpreted as an optimization of the bound to the input, achieved by introducing a second gain ($\mathbf{M}_2$). Note that the maximum input satisfying equation (6.41) provides a sufficient but not necessary bound for stability.

The design strategy proposed in this chapter can also be applied to bilinear systems whose dynamics is affected by process and measurement noise. Noise makes it impossible for the bilinear observer to converge to the exact state, eliminating the conceptual difference between ideal and non-ideal bilinear systems. When the input is stationary and white, Lemma 8.3 guarantees that the observer minimizing the time average of the state estimation error variance is

unbiased and has the structure proposed in equation (8.2). This provides a rigorous justification for the design procedures presented, which then require stationary white excitation. If the designed observer is then used in operation conditions with input (or noise) characteristics different from the ones used to design the gains, the observer can still be used, it works and is unbiased but generally suboptimal, i.e. in steady state the estimated state is the true system state in expectation but the average estimation error variance is not guaranteed to be minimum. For optimality under arbitrary input profiles, time-varying gains are required and can be found as illustrated in the appendix.

The importance of the link established between the bilinear observer gains and the interaction matrices in bilinear system identification goes beyond the development of a numerical technique for bilinear observer design. The bilinear system identification methods presented in Reference 51 are formulated for noise-free bilinear systems and can now be interpreted as based on deterministic bilinear observers. The development of an optimal bilinear observer for bilinear systems affected by process and measurement noise paves the way for a bilinear version of OKID (see section 8.1) able to optimally handle noise in the measured data. This would allow one to simultaneously estimate both the model matrices $(\mathbf{A}, \mathbf{N}, \mathbf{B}, \mathbf{C}, \mathbf{D})$ and the optimal observer gains $(\mathbf{M}_1, \mathbf{M}_2)$ directly from data measured from the real bilinear system, solving a broader problem than the one addressed in this chapter.

The proposed optimal bilinear observer is the right candidate for bilinear OKID. In the linear case, the steady-state Kalman filter is the unique linear-time-invariant observer with output residuals orthogonal to the current input and past input-output values. This guarantees that the observer Markov parameters (unit pulse response) estimated in OKID from input-output data are those of the optimal linear-time-invariant observer (steady-state Kalman filter). The structure of the Kalman filter makes it possible to relate the state-space model of the Kalman

filter to that of the system and complete the identification. The following theorem states a similar property for the bilinear case. Note that when passing from linear to bilinear OKID, the additional assumption of stationary white excitation is required.

Theorem 8.9. *The output residuals $\boldsymbol{\epsilon}$ of the observer in (8.34) in steady state are orthogonal to the input at the same time step and to input-output data at previous steps (i.e. the sequence $\{\boldsymbol{\epsilon}(k)\}$ is orthogonal to the sequences $\{u(k)\}$ and $\{\boldsymbol{z}_p(k)\}$ for $k \geq p$) only if the observer takes the form of (8.2) and if the input is stationary and white.*

Proof. Lemma 8.6 and Lemma 8.8 require the observer to be optimal in the sense of Lemma 8.3 for the output residuals to be orthogonal to current input and past input-output data. Lemma 8.3 requires stationary white input and unbiased observer for the observer to be optimal. Lemma 8.2 shows how the observer is unbiased if and only if it takes the form of equation (8.2). □

8.8 Conclusions

In this chapter we have formulated a full-order bilinear time-invariant observer for a bilinear state-space model. The key feature of the proposed bilinear observer is its link to the interaction matrices at the core of the methods for bilinear system identification in the absence of noise presented in chapter 6. In particular, we have established a connection between the interaction matrices in bilinear system identification and the observer gains in bilinear state estimation. This connection has been exploited in both directions, to devise a technique to design bilinear observer gains as well as to lay the ground for the development of a bilinear version of OKID for bilinear system identification in the presence of noise.

With regard to the observer design problem, we have taken the approach of identifying the observer gains with data generated from a known model and the noise covariances (an observer

identification problem), instead of finding their closed-form solution. In the absence of noise, the fastest convergent observer is obtained. It is also shown how for certain bilinear systems a deadbeat observer exists, where the estimated state estimation error converges to zero identically in a finite number of time steps. For such systems, the proposed design technique finds the deadbeat gains. In the presence of noise, the resulting observer minimizes the state estimation error in a manner similar to how the Kalman filter does in the linear case. Provided the input is stationary and white, the designed time-invariant bilinear observer minimizes the time average of the expected value of the norm squared of the state estimation error. Additionally, we have shown how time-varying gains are needed for the observer to minimize at each time step the expected value of the norm squared of the state estimation error. Such time-varying bilinear observer is the fastest in the presence of noise and its optimality holds for arbitrary input sequences. Numerical examples successfully illustrated both the theoretical and computational aspects of the new results.

The properties of the bilinear stochastic observer have also been investigated and proven to be, under the sole additional assumption of stationary white input, the bilinear counterpart of the properties of the Kalman filter that led to OKID in the linear case. This forms the first fundamental step of working out a bilinear version of OKID, where not only is the bilinear system model identified directly from noisy input-output measurements, but also the associated bilinear observer is simultaneously identified without a priori knowledge of the noise covariance or the system model. Bilinear OKID would address at once both the system identification and the state estimation problem.

Chapter 9

Bilinear OKID

9.1 Introduction

In chapters 6 and 7 we developed several methods for the identification of bilinear state-space models from noise-free input-output measurements. In chapter 8 the connection between the interaction matrices used in chapter 6 and the gain of an optimal bilinear time-invariant observer provides an observer with, under the sole additional assumption of stationary white excitation input, similar structure and properties to the ones of the Kalman filter that led to OKID in the linear case. Such a result paves the way for the development of a bilinear counterpart to the OKID equation in (2.44). Finally, the novel OKID approach based on the estimation of the observer output residuals presented in chapter 3 provides us with a way to get around the lack of ERA for bilinear systems, and complete the identification process via the construction of a new deterministic identification problem to be solved by the methods developed in chapter 6. The solution to said deterministic identification problem yields the state-space model of the optimal bilinear observer from which the matrices of the bilinear system to be identified can easily be recovered.

The outcome of the chapter is a bilinear system identification method optimally taking into account the noise corrupting the measured data, which is a significant improvement over the deterministic methods of chapter 6 and represents the first extension of OKID to nonlinear systems.

The content of this chapter was presented at the 24$^{\text{th}}$ AAS/AIAA Space Flight Mechanics Meeting in Santa Fe, NM, in 2014 (Reference 70).

9.2 Problem statement

Consider the n-state, single-input, q-output discrete-time bilinear system in state-space form introduced in equation (8.1). A single set of length l of input-output data that starts from some unknown initial state $\boldsymbol{x}(0)$ is given

$$\{u(k)\} = \{u(0),\ \ u(1),\ \ u(2),\ \ ...,\ \ u(l-1)\} \tag{9.1a}$$

$$\{\boldsymbol{y}(k)\} = \{\boldsymbol{y}(0),\ \ \boldsymbol{y}(1),\ \ \boldsymbol{y}(2),\ \ ...,\ \ \boldsymbol{y}(l-1)\} \tag{9.1b}$$

The objective is to identify the system of equation (8.1), i.e. the matrices $\mathbf{A}$, $\mathbf{N}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, with the input-output data provided in equation (9.1). The process and measurement noises are unknown, as well as their covariance matrices. The data of equation (9.1) is assumed to be of sufficient length and richness so that the system of equation (8.1) can be correctly identified. For simplicity, we focus on a single-input model in this work. Extension to the multi-input case can be made without conceptual difficulties.

9.3 Approach overview

As mentioned in the introduction, OKID/ERA (chapter 3) has proven to be a very successful technique for the identification of linear systems, solving the linear counterpart ($\mathbf{N} = \mathbf{0}$) of the problem addressed in this chapter. OKID/ERA is based on a relationship between the measured input and output, equation (2.44), which was originally derived via the interaction matrix technique (Reference 20). In the presence of noise in the data, the least-squares (LS) solution of the corresponding set of equations is proven to yield the Markov parameters (or unit pulse response) of the optimal linear observer (Kalman filter) for the system to be identified and the noise statistics embedded in the data. From the Markov parameters of the observer, those of the system can be recovered and fed to ERA or ERA-DC (appendix A) to find a realization of the system (matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$) and the corresponding Kalman gain. The use of ERA (or ERA/DC) to complete the OKID process is not the only possible choice as proven in chapter 3, and in this chapter we exploit such finding to overcome the lack of a bilinear version of ERA and develop the first OKID-based identification method for bilinear systems. The method is articulated around the main steps described below.

9.3.1 Input-output relationship via interaction matrices

In system identification, the measured input and output are the only known signals. No knowledge of the evolution of the state over time is generally available. It is then useful to derive an equation relating the output of the system directly to the input, without the state appearing explicitly. In the same fashion as in the original OKID work (Reference 20), interaction matrices are used to derive such a relation, which can be classified as a bilinear autoregressive model with exogenous input (ARX). The proof for the existence of interaction matrices for bilinear

systems is given in chapter 6. Due to its central role in this paper, the corresponding theorem is restated and briefly discussed in the next section.

9.3.2 Estimation of the observer residuals

The bilinear ARX gives rise to a set of algebraic equations, which represents the core of bilinear OKID. The LS solution of the bilinear ARX equations are proven to be related to the optimal bilinear observer corresponding to the system to be identified and to the noise statistics embedded in the data. In particular, the LS residuals of the bilinear OKID core equation are the residuals $\boldsymbol{\epsilon}$ of the optimal bilinear observer.

9.3.3 Construction of a noise-free identification problem

Along the same lines of chapter 3, the observer residuals are used to construct the following new identification problem. Consider the optimal observer in equation (8.2). Recalling the definition of observer output residual, equation (2.8), and indicating the gains as $\mathbf{K}'$ and $\mathbf{K}''$ to remark their optimality, we can write the optimal bilinear observer as

$$\hat{\boldsymbol{x}}(k+1) = \mathbf{A}\hat{\boldsymbol{x}}(k) + \mathbf{N}\hat{\boldsymbol{x}}(k)u(k) + \mathbf{B}u(k) + \mathbf{K}'\boldsymbol{\epsilon}(k) + \mathbf{K}''\boldsymbol{\epsilon}(k)u(k) \tag{9.2a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \mathbf{D}u(k) \tag{9.2b}$$

Since, from the definition of observer residual we can compute $\hat{\boldsymbol{y}}(k) = \boldsymbol{y}(k) - \boldsymbol{\epsilon}(k)$, equation (9.2) can be looked at as the state-space model of a dynamic system whose inputs, $u(k), \boldsymbol{\epsilon}(k), \boldsymbol{\epsilon}(k)u(k)$, and output, $\hat{\boldsymbol{y}}(k)$, are known. Additionally, notice that no (unknown) noise term is present in equation (9.2). We can then think of identifying the matrices of such system, namely $\mathbf{A}$, $\mathbf{N}$, $\mathbf{B}$, $\mathbf{K}'$, $\mathbf{K}''$, $\mathbf{C}$ and $\mathbf{D}$, with a deterministic bilinear system identification method.

9.3.4 Observer identification

The methods developed in chapter 6 are used to solve the noise-free system identification problem in equation (9.2). The reader will notice similarities in the technique used to derive the input-output-to-state relationships at the core of the methods in chapter 6 and the way the bilinear ARX model is derived in the present work. Indeed, the proposed bilinear OKID method can be interpreted as the extension of the methods of chapter 6 to bilinear identification problems where the dynamic process and the measurements are corrupted by noise. The crucial difference is that in the latter case the optimal bilinear observer is identified instead of the original dynamical system. Note that the identification of the observer of equation (9.2) solves the original problem of identifying the system of equation (8.1). Due to the peculiarities of the identification problem of equation (9.2), namely different input in the state equation and the observation equation, the deterministic identification algorithms in chapter 6 are briefly reviewed and accordingly modified in a dedicated section.

9.4 Interaction matrices and optimal observer

The existence of interaction matrices for bilinear systems is guaranteed by theorem 6.1, proven and thoroughly discussed in chapter 6. Theorem 6.1 is invoked at several points in this paper, therefore it is conveniently restated here below.

Theorem 9.1. *Given the matrices* $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{N} \in \mathbb{R}^{n\times n}$, $\mathbf{C} \in \mathbb{R}^{q\times n}$, *define the matrices* $\bar{\mathbf{A}}_M = \mathbf{A} - \mathbf{M}'\mathbf{C}$ *and* $\bar{\mathbf{N}}_M = \mathbf{N} - \mathbf{M}''\mathbf{C}$ *and the product*

$$\boldsymbol{\mathcal{S}}_{M,p}(k) = \left(\bar{\mathbf{A}}_M + \bar{\mathbf{N}}_M u(k-1)\right)\left(\bar{\mathbf{A}}_M + \bar{\mathbf{N}}_M u(k-2)\right) \ldots \left(\bar{\mathbf{A}}_M + \bar{\mathbf{N}}_M u(k-p)\right)$$

where $\mathbf{M}'$, $\mathbf{M}'' \in \mathbb{R}^{n\times q}$ *are called interaction matrices and* $\{u(k-p), u(k-p+1), ..., u(k-1)\}$ *is a scalar sequence. If* (**A**,**C**) *is an observable pair, then there exist some interaction matrices* $\mathbf{M}'$, $\mathbf{M}''$ *and a positive scalar* γ *such that* $\boldsymbol{\mathcal{S}}_{M,p}(k)$ *converges asymptotically to* $\mathbf{0}$ *with* p *provided that* $|u(i)| < \gamma$ *for all* $i = k-p, k-p+1..., k-1$.

As discussed in details in chapter 6, it is worth noting that there are bilinear systems, referred to as *ideal*, for which there exist $\mathbf{M}'$ and $\mathbf{M}''$ such that $\boldsymbol{\mathcal{S}}_{M,p}(k)$ is identically equal to 0 for $p \geq n$ and $\gamma \to \infty$ (i.e. no bound on $|u|$). The class of ideal bilinear systems comprises, among others, all those with $\operatorname{rank}\mathbf{C} = n$. For arbitrary (*non-ideal*) bilinear systems, $\boldsymbol{\mathcal{S}}_{M,p}(k)$ cannot be identically equal to $\mathbf{0}$ but it is guaranteed to converge to $\mathbf{0}$ asymptotically with p thanks to $\mathbf{M}_1$, $\mathbf{M}_2$ and the finite bound γ on the magnitude of the values that u can take.

At the core of the linear OKID method are steady-state linear state observers, in particular linear-time-invariant observers minimizing $\mathbb{E}\left[\boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k)\right]$ for all k after the initial transient, where $\boldsymbol{e}(k) = \boldsymbol{x}(k) - \hat{\boldsymbol{x}}(k)$ is the state estimation error. Such observers are the deadbeat observer (in the absence of noise) and the Kalman filter (in the presence of noise). In the bilinear version of OKID we rely on the bilinear observer defined by a similar optimality criterion and presented in detail in chapter 8.

Its key properties are summarized here below, where, exploiting the ergodic property ensured by the stationary drive to the system, the properties are written purely in terms of time averages. Under the assumption made to derive the optimal gain conditions of equation (8.43), i.e. stationary white excitation input u, the residuals $\boldsymbol{\epsilon}(k)$ of the bilinear observer at steady state ($k \geq p$) can be shown to have the following properties if and only if the observer is optimal:

1. $\{\boldsymbol{\epsilon}(k)\}$ is a zero-mean sequence, i.e. $\lim_{l\to+\infty} \frac{1}{l-p} \sum_{k=p}^{l-1} \boldsymbol{\epsilon}(k) = \mathbf{0}$
2. $\{\boldsymbol{\epsilon}(k)\}$ is a white sequence, i.e. $\lim_{l\to+\infty} \frac{1}{l-p-j} \sum_{k=p+j}^{l-1} \boldsymbol{\epsilon}^{\mathrm{T}}(k)\boldsymbol{\epsilon}(k-j) = \mathbf{0}$ for $j > 0$

3. $\boldsymbol{\epsilon}(k)$ is orthogonal to the past outputs and to the current and past inputs, and some products of them, i.e. $\lim_{l\to+\infty} \frac{1}{l-p} \sum_{k=p}^{l-1} \boldsymbol{\epsilon}(k)u(k) = \mathbf{0}$ and $\lim_{l\to+\infty} \frac{1}{l-p} \sum_{k=p}^{l-1} \boldsymbol{\epsilon}^{\mathrm{T}}(k)\boldsymbol{z}_p(k) = \mathbf{0}$

where $\boldsymbol{z}_p(k)$ contains past inputs and outputs and is defined more precisely later in the paper, below equation (9.9). Note that such properties are similar to those of the Kalman filter that led to the development of OKID in the linear case.

9.5 Estimation of the observer residuals

The proposed bilinear OKID method consists of two main steps, the first of which is the estimation of the optimal observer residuals. The task is accomplished by solving by LS a set of equations arising from an input-output relationship in which the state does not appear explicitly. The derivation in this section is done following the same strategy as in the original OKID paper (Reference 20). Accordingly, the presence of noise in equation (8.1) is temporarily ignored and its effect is discussed when solving the above mentioned LS problem. Additionally, connections with previous work on bilinear discrete-time system identification are made along the derivation to provide a comprehensive framework for the method proposed in this paper. Add and subtract the terms $\mathbf{H}'\boldsymbol{y}(k)$ and $\mathbf{H}''\boldsymbol{y}(k)u(k)$ to equation (6.1a) where $\mathbf{H}'$, $\mathbf{H}'' \in \mathbb{R}^{n\times q}$ are interaction matrices, getting

$$\begin{aligned}
\boldsymbol{x}(k+1) &= \mathbf{A}\boldsymbol{x}(k) + \mathbf{N}\boldsymbol{x}(k)u(k) + \mathbf{B}u(k) + \mathbf{H}'\boldsymbol{y}(k) - \mathbf{H}'\boldsymbol{y}(k) + \mathbf{H}''\boldsymbol{y}(k)u(k) - \mathbf{H}''\boldsymbol{y}(k)u(k) \\
&= \mathbf{A}\boldsymbol{x}(k) + \mathbf{N}\boldsymbol{x}(k)u(k) + \mathbf{B}u(k) + \mathbf{H}'\boldsymbol{y}(k) - \mathbf{H}'\mathbf{C}\boldsymbol{x}(k) - \mathbf{H}'\mathbf{D}u(k) \\
&\quad + \mathbf{H}''\boldsymbol{y}(k)u(k) - \mathbf{H}''\mathbf{C}\boldsymbol{x}(k)u(k) - \mathbf{H}''Du^2(k) \\
&= \bar{\mathbf{A}}_H\boldsymbol{x}(k) + \bar{\mathbf{N}}_H\boldsymbol{x}(k)u(k) + \bar{\mathbf{B}}_H\boldsymbol{v}(k)
\end{aligned} \tag{9.3}$$

where

$$\bar{\mathbf{A}}_H = \mathbf{A} - \mathbf{H}'\mathbf{C} \qquad \bar{\mathbf{B}}_H = \begin{bmatrix} \mathbf{B} - \mathbf{H}'\mathbf{D} & \mathbf{H}' & -\mathbf{H}''\mathbf{D} & \mathbf{H}'' \end{bmatrix} \tag{9.4a}$$

$$\bar{\mathbf{N}}_H = \mathbf{N} - \mathbf{H}''\mathbf{C} \qquad \boldsymbol{v}(k) = \begin{bmatrix} u(k) & \boldsymbol{y}^{\mathrm{T}}(k) & u^2(k) & \boldsymbol{y}^{\mathrm{T}}(k)u(k) \end{bmatrix}^{\mathrm{T}} \tag{9.4b}$$

The interaction matrices convert the bilinear model $\mathbf{A}$, $\mathbf{B}$, $\mathbf{N}$ of equation (6.1a) into the equivalent bilinear model of equation (9.3) with matrices $\bar{\mathbf{A}}_H$, $\bar{\mathbf{N}}_H$, $\bar{\mathbf{B}}_H$. The observation equation, equation (6.1b), does not change. The freedom introduced by $\mathbf{H}'$ and $\mathbf{H}''$ will be used to impose conditions to express the state at the current time step k solely in terms of past input and output data. Equation (9.3) has the same form as equation (9.23), anticipating the connection between interaction matrices and observer gains. Propagating equation (9.3) one step forward, we get

$$\begin{aligned} \boldsymbol{x}(k+2) &= \left(\bar{\mathbf{A}}_H + \bar{\mathbf{N}}_H u(k+1)\right)\boldsymbol{x}(k+1) + \bar{\mathbf{B}}_H \boldsymbol{v}(k+1) \\ &= \left(\bar{\mathbf{A}}_H + \bar{\mathbf{N}}_H u(k+1)\right)\left(\left(\bar{\mathbf{A}}_H + \bar{\mathbf{N}}_H u(k)\right)\boldsymbol{x}(k) + \bar{\mathbf{B}}_H \boldsymbol{v}(k)\right) + \bar{\mathbf{B}}_H \boldsymbol{v}(k+1) \\ &= \boldsymbol{\mathcal{S}}_{H,2}(k)\boldsymbol{x}(k) + \mathbf{T}_{H,2}\boldsymbol{z}_2(k+2) \end{aligned} \tag{9.5}$$

where

$$\boldsymbol{\mathcal{S}}_{H,2}(k+2) = \left(\bar{\mathbf{A}}_H + \bar{\mathbf{N}}_H u(k+1)\right)\left(\bar{\mathbf{A}}_H + \bar{\mathbf{N}}_H u(k)\right) \tag{9.6a}$$

$$\mathbf{T}_{H,2} = \begin{bmatrix} \bar{\mathbf{A}}_H\bar{\mathbf{B}}_H & \bar{\mathbf{N}}_H\bar{\mathbf{B}}_H & \bar{\mathbf{B}}_H \end{bmatrix} \qquad \boldsymbol{z}_2(k+2) = \begin{bmatrix} \boldsymbol{v}^{\mathrm{T}}(k) & \boldsymbol{v}^{\mathrm{T}}(k)u(k+1) & \boldsymbol{v}^{\mathrm{T}}(k+1) \end{bmatrix}^{\mathrm{T}} \tag{9.6b}$$

Propagating equation (9.5) further in time, we obtain the more general expression

$$\boldsymbol{x}(k+p) = \boldsymbol{\mathcal{S}}_{H,p}(k+p)\boldsymbol{x}(k) + \mathbf{T}_{H,p}\boldsymbol{z}_p(k+p) \tag{9.7}$$

which, by shifting the time index backward by p time steps, can be more conveniently written as

$$\boldsymbol{x}(k) = \boldsymbol{\mathcal{S}}_{H,p}(k)\boldsymbol{x}(k-p) + \mathbf{T}_{H,p}\boldsymbol{z}_p(k) \tag{9.8}$$

where

$$\boldsymbol{\mathcal{S}}_{H,p}(k) = \left(\bar{\mathbf{A}}_H + \bar{\mathbf{N}}_H u(k-1)\right)\left(\bar{\mathbf{A}}_H + \bar{\mathbf{N}}_H u(k-2)\right) \ldots \left(\bar{\mathbf{A}}_H + \bar{\mathbf{N}}_H u(k-p)\right) \tag{9.9}$$

$\mathbf{T}_{H,p}$ contains products of $\bar{\mathbf{A}}_H$, $\bar{\mathbf{N}}_H$ and $\bar{\mathbf{B}}_H$, and $\boldsymbol{z}_p(k)$ is constructed exactly like $\boldsymbol{z}_{p,c}(k)$ in equation (6.44).

By choosing p sufficiently large, theorem 9.1 ensures that $\boldsymbol{\mathcal{S}}_{H,p}(k)$ vanishes and we can write

$$\boldsymbol{x}(k) = \mathbf{T}_{H,p}\boldsymbol{z}_p(k) \tag{9.10}$$

So far the development is the same as in chapter 6, where equation (9.10) is referred to as an input-output-to-state relationship (IOSR) and used as the starting point to develop the equivalent linear model (ELM) and intersection subspace (IS) methods for bilinear system identification in the absence of noise.

Putting together equations (9.10) and (6.1b), always assuming no noise in the measured data, we realize that

$$\boldsymbol{x}(k) = \mathbf{T}_{H,p}\boldsymbol{z}_p(k) \tag{9.11a}$$

$$\boldsymbol{y}(k) = \mathbf{C}\boldsymbol{x}(k) + \mathbf{D}u(k) \tag{9.11b}$$

are the equations of an observer in non-recursive form, whose state is determined solely by past input-output values. Indeed, the observation that the matrices $\mathbf{H}'$ and $\mathbf{H}''$ embedded in $\mathbf{T}_{H,p}$ can be interpreted as observer gains was exploited in chapter 8 to devise a technique to design the fastest bilinear observers in the absence of noise. For ideal bilinear systems, such observers are deadbeat in a strict sense, i.e. the observer state is exactly equal to the system state after $p = n$ steps. For arbitrary bilinear systems, the fastest deterministic observer state converges asymptotically to the system state, i.e. one needs to wait a larger number p of steps before the observer state approaches the system state. By plugging equation (9.11a) into equation (9.11b) we obtain the following relationship between the input and output

$$\boldsymbol{y}(k) = \mathbf{C}\mathbf{T}_{H,p}\boldsymbol{z}_p(k) + \mathbf{D}u(k) \tag{9.12}$$

which is the equation used in Reference 52 to directly identify bilinear input-output maps in the absence of noise. By defining

$$\boldsymbol{\Phi}_{H,p} = \begin{bmatrix} \mathbf{C}\mathbf{T}_{H,p} & \mathbf{D} \end{bmatrix} \qquad \boldsymbol{v}_p(k) = \begin{bmatrix} \boldsymbol{z}_p^{\mathrm{T}}(k) & u(k) \end{bmatrix}^{\mathrm{T}} \tag{9.13}$$

equation (9.12) takes the form of the classic OKID equation

$$\boldsymbol{y}(k) = \boldsymbol{\Phi}_{H,p}\boldsymbol{v}_p(k) \tag{9.14}$$

which can be written at all time steps $p \leq k \leq l-1$ to obtain the following set of equations

$$\mathbf{Y} = \mathbf{\Phi}_{H,p}\mathbf{V}_p \tag{9.15}$$

where

$$\mathbf{Y} = \begin{bmatrix} \boldsymbol{y}(p) & \boldsymbol{y}(p+1) & \dots & \boldsymbol{y}(l-1) \end{bmatrix} \tag{9.16a}$$

$$\mathbf{V}_p = \begin{bmatrix} \boldsymbol{v}_p(p) & \boldsymbol{v}_p(p+1) & \dots & \boldsymbol{v}_p(l-1) \end{bmatrix} \tag{9.16b}$$

The presence of noise in equation (8.1) makes equation (9.15) inconsistent. The inconsistency can be expressed by an error term E

$$\mathbf{Y} = \mathbf{\Phi}_{H,p}\mathbf{V}_p + \mathbf{E} \tag{9.17}$$

with

$$\mathbf{E} = \begin{bmatrix} \boldsymbol{\epsilon}(p) & \boldsymbol{\epsilon}(p+1) & \dots & \boldsymbol{\epsilon}(l-1) \end{bmatrix} \tag{9.18}$$

The same symbol $\boldsymbol{\epsilon}$ indicating the observer residuals is used for the error terms of equation (9.17). As shown later in this section, the two turns out to correspond.

By having a sufficiently long record, it is possible to find the LS solution to equation (9.17)

$$\tilde{\mathbf{\Phi}}_{H,p} = \mathbf{Y}\mathbf{V}_p^{\mathrm{T}} \left(\mathbf{V}_p\mathbf{V}_p^{\mathrm{T}}\right)^{-1} = \mathbf{Y}\mathbf{V}_p^{\dagger} \tag{9.19}$$

where † denotes the Moore-Penrose pseudoinverse of $\mathbf{V}_p$. Right-multiplying equation (9.17) by $\mathbf{V}_p^{\mathrm{T}}$ and replacing $\mathbf{\Phi}_{H,p}$ with its LS estimate $\tilde{\mathbf{\Phi}}_{H,p}$, we obtain

$$\mathbf{Y}\mathbf{V}_p^{\mathrm{T}} = \mathbf{Y}\mathbf{V}_p^{\mathrm{T}} \left(\mathbf{V}_p\mathbf{V}_p^{\mathrm{T}}\right)^{-1} \mathbf{V}_p\mathbf{V}_p^{\mathrm{T}} + \mathbf{E}\mathbf{V}_p^{\mathrm{T}} = \mathbf{Y}\mathbf{V}_p^{\mathrm{T}} + \mathbf{E}\mathbf{V}_p^{\mathrm{T}} \tag{9.20}$$

which implies that $\mathbf{E}\mathbf{V}_p^{\mathrm{T}} = \mathbf{0}$, i.e. that

$$\sum_{k=p}^{l-1} \boldsymbol{\epsilon}(k)\boldsymbol{v}_p^{\mathrm{T}}(k) = \mathbf{0} \tag{9.21}$$

Assuming l is large and dividing equation (9.21) by $l-p$, we recognize its left-hand side as the time average of each entry of the product $\epsilon\boldsymbol{v}_p^{\mathrm{T}}$. In other words, the LS residuals $\boldsymbol{\epsilon}(k)$ are orthogonal to all the entries of $\boldsymbol{v}_p(k)$, i.e. $u(k)$ and all the entries of $\boldsymbol{z}_p(k)$. The latter are products of past input-output data. This is the same property characterizing the optimal bilinear observer in equation (9.2), provided u is a stationary white process. Hence, under the stated assumptions, for a long record (large l) and for a sufficiently large p, the residuals corresponding to the LS solution of equation (9.17) are an estimate of the optimal bilinear observer residuals in the presence of noise, and they can be computed by

$$\tilde{\mathbf{E}} = \mathbf{Y} - \tilde{\mathbf{\Phi}}_{H,p}\mathbf{V}_p \tag{9.22}$$

In deriving equation (9.15), theorem 9.1 was invoked to make $\boldsymbol{\mathcal{S}}_{H,p}(k)$ vanish in equation (9.8). In the absence of noise, as considered in chapter 6, the interaction matrices $\mathbf{H}'$ and $\mathbf{H}''$ will attempt to place the eigenvalues of $\bar{\mathbf{A}}_H$ and $\bar{\mathbf{N}}_H$ to make $\boldsymbol{\mathcal{S}}_{H,p}(k)$ decay as fast as possible under the given input history. For ideal bilinear systems, this results in placing the eigenvalues of $\bar{\mathbf{A}}_H$ and $\bar{\mathbf{N}}_H$ at the origin to obtain a decay of $\boldsymbol{\mathcal{S}}_{H,p}(k)$ in n time steps (deadbeat) under any input history. Such a deadbeat decay corresponds to $\mathbb{E}\left[\boldsymbol{e}^{\mathrm{T}}(k)\boldsymbol{e}(k)\right] = 0$ for all $k \geq p \geq n$.

The presence of noise in equation (8.1) makes the latter equality impossible and the interaction matrices attempt to place the eigenvalues of $\bar{\mathbf{A}}_H$ and $\bar{\mathbf{N}}_H$ so that $\bar{\mathbf{\Pi}}$ in lemma 8.3 is minimized, i.e. $\mathbf{H}' = \mathbf{K}'$ and $\mathbf{H}'' = \mathbf{K}''$. Theorem 9.1 still guarantees that $\boldsymbol{\mathcal{S}}_{H,p}(k)$ vanishes, possibly for a lower value of γ.

To better clarify the result, we briefly provide an alternative derivation, similar to how the core OKID equation was derived for the linear case in chapter 2. One could have started from the optimal bilinear observer in equation (9.2) and transformed it into

$$\hat{\boldsymbol{x}}(k+1) = \bar{\mathbf{A}}_K \hat{\boldsymbol{x}}(k) + \bar{\mathbf{N}}_K \hat{\boldsymbol{x}}(k) u(k) + \bar{\mathbf{B}}_K \boldsymbol{v}(k) \tag{9.23a}$$

$$\hat{\boldsymbol{y}}(k) = \mathbf{C}\hat{\boldsymbol{x}}(k) + \mathbf{D}u(k) \tag{9.23b}$$

where

$$\bar{\mathbf{A}}_K = \mathbf{A} - \mathbf{K}'\mathbf{C} \qquad \bar{\mathbf{B}}_K = \begin{bmatrix} \mathbf{B} - \mathbf{K}'\mathbf{D} & \mathbf{K}' & -\mathbf{K}''\mathbf{D} & \mathbf{K}'' \end{bmatrix} \tag{9.24a}$$

$$\bar{\mathbf{N}}_K = \mathbf{N} - \mathbf{K}''\mathbf{C} \qquad \boldsymbol{v}(k) = \begin{bmatrix} u(k) & \boldsymbol{y}^{\mathrm{T}}(k) & u^2(k) & \boldsymbol{y}^{\mathrm{T}}(k) u(k) \end{bmatrix}^{\mathrm{T}} \tag{9.24b}$$

similar to how equation (3.1) was transformed into equation (3.7).

Propagate equation (9.23a) forward in time like equation (9.3) to obtain

$$\hat{\boldsymbol{x}}(k) = \boldsymbol{\mathcal{S}}_{K,p}(k) \hat{\boldsymbol{x}}(k-p) + \mathbf{T}_{K,p} \boldsymbol{z}_p(k) \tag{9.25}$$

where $\boldsymbol{\mathcal{S}}_{K,p}(k)$ and $\mathbf{T}_{K,p}$ are defined as in equation (9.9), except for $\bar{\mathbf{A}}_H$ and $\bar{\mathbf{N}}_H$ being repleaced by $\bar{\mathbf{A}}_K$ and $\bar{\mathbf{N}}_K$. Plugging equation (9.25) into equation (9.23b) and recalling the definition of

observer residual, we get

$$\boldsymbol{y}(k) = \mathbf{C}\boldsymbol{\mathcal{S}}_{K,p}(k)\hat{\boldsymbol{x}}(k) + \mathbf{C}\mathbf{T}_{K,p}\boldsymbol{z}_p(k) + \boldsymbol{\epsilon}(k) \tag{9.26}$$

From theorem 9.1, we know that there is a bound γ to the input magnitude for which $\boldsymbol{\mathcal{S}}_{K,p}(k)$ converges to $\mathbf{0}$ for all k, allowing us to write

$$\boldsymbol{y}(k) = \boldsymbol{\Phi}_{K,p}\boldsymbol{z}_p(k) + \boldsymbol{\epsilon}(k) \tag{9.27}$$

Note that $\boldsymbol{z}_p(k)$ contains past outputs and current and past inputs, hence equation (9.27) is a bilinear ARX model (with noise $\boldsymbol{\epsilon}$ in the output). We can rewrite equation (9.26), in matrix form, to obtain equation (9.17). By the above orthogonality argument, the LS solution to equation (9.17) produces LS residuals which are indeed the residuals of the optimal bilinear observer.

9.6 Observer identification

The second main step of the proposed bilinear OKID method consists in the identification of the optimal observer. Once the residuals $\boldsymbol{\epsilon}(k)$ of the optimal bilinear observer have been estimated, they can be used to construct the noise-free identification problem of equation (9.2). Equation (9.2) describes the dynamics of the optimal bilinear observer, it has no noise terms, its inputs $u(k)$, $\boldsymbol{\epsilon}(k)$ and $\boldsymbol{\epsilon}(k)u(k)$ are known for $k = p, p+1, ..., l-1$ as well as its output $\hat{\boldsymbol{y}}(k)$. Among the matrices to be identified in equation (9.2), there are those of the system of equation (8.1). Identifying the observer of equation (9.2) solves then the problem addressed in this paper. In this step lies the essence of the proposed bilinear OKID method. The identification problem of equation (8.1), affected by (unmeasured) noise, is transformed into the identification

problem of equation (9.2), with no noise term in it. Note that the identification of the optimal observer could also be performed in the form of equation (9.23), as discussed in chapter 3 for the linear case.

To identify the noise-free system of equation (9.2), one can use either of the approaches illustrated in chapter 6, i.e. the ELM or the IS approach. Both of them are based on the IOSR in equation (6.3), which in the case of the identification of the observer in equation (9.2) becomes

$$\hat{\boldsymbol{x}}(k) = \mathbf{T}\boldsymbol{z}(k) \tag{9.28}$$

Indeed, the ELM and the IS methods are applied exactly as in chapter 6 but replacing $\boldsymbol{x}$ with $\hat{\boldsymbol{x}}$. In the example in this chapter, we use the IS method. The last step to estimate the model in equation (9.2) alos needs to be modified with respect to equations (6.16) and (6.17) as follows. Once the observer state history has been reconstructed, from a certain time step k_i to k_f, we can write

$$\begin{bmatrix} \hat{\boldsymbol{x}}(k_i+1) & \dots & \hat{\boldsymbol{x}}(k_f) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} & \mathbf{N} & \mathbf{K}' & \mathbf{K}'' \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}}(k_i) & \dots & \hat{\boldsymbol{x}}(k_f-1) \\ u(k_i) & \dots & u(k_f-1) \\ \hat{\boldsymbol{x}}(k_i)u(k_i) & \dots & \hat{\boldsymbol{x}}(k_f-1)u(k_f-1) \\ \boldsymbol{\epsilon}(k_i) & \dots & \boldsymbol{\epsilon}(k_f-1) \\ \boldsymbol{\epsilon}(k_i)u(k_i) & \dots & \boldsymbol{\epsilon}(k_f-1)u(k_f-1) \end{bmatrix} \tag{9.29a}$$

$$\begin{bmatrix} \hat{\boldsymbol{y}}(k_i) & \dots & \hat{\boldsymbol{y}}(k_f) \end{bmatrix} = \begin{bmatrix} \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{x}}(k_i) & \dots & \hat{\boldsymbol{x}}(k_f) \\ u(k_i) & \dots & u(k_f) \end{bmatrix} \tag{9.29b}$$

and recover $\mathbf{A}$, $\mathbf{B}$, $\mathbf{N}$, $\mathbf{K}'$, $\mathbf{K}''$ and $\mathbf{C}$, $\mathbf{D}$ via pseudo-inversion (Moore-Penrose). Note that the reconstructed state is not necessarily in the original coordinate system, and so will be

the identified bilinear observer matrices. As usual with state-space formulation, the change in coordinate system does not affect the identified model validity.

To reconstruct the state, the superspaces $\mathbf{Z}_a$ and $\mathbf{Z}_b$ for the IS method are built for example with the causal and mixed-anticausal IOSRs, which need to be modified as follows to take into account the extra additive input components $\boldsymbol{\epsilon}(k)$ and $\boldsymbol{\epsilon}(k)u(k)$ in equation (9.2a).

By introducing two interaction matrices $\mathbf{M}'_c$ and $\mathbf{M}''_c$ and adding and subtracting the terms $\mathbf{M}'_c\hat{\boldsymbol{y}}(k)$ and $\mathbf{M}''_c\hat{\boldsymbol{y}}(k)u(k)$, equation (9.2a) can be written as

$$\hat{\boldsymbol{x}}(k+1) = \bar{\mathbf{A}}_c\hat{\boldsymbol{x}}(k) + \bar{\mathbf{N}}_c\hat{\boldsymbol{x}}(k)u(k) + \bar{\mathbf{B}}_c\boldsymbol{v}_c(k) \tag{9.30}$$

where

$$\bar{\mathbf{A}}_c = \mathbf{A} - \mathbf{M}'_c\mathbf{C} \qquad \bar{\mathbf{N}}_c = \mathbf{N} - \mathbf{M}''_c\mathbf{C} \tag{9.31a}$$

$$\bar{\mathbf{B}}_c = \begin{bmatrix} \mathbf{B} - \mathbf{M}'_c\mathbf{D} & \mathbf{M}'_c & -\mathbf{M}''_c\mathbf{D} & \mathbf{M}''_c & \mathbf{K}' & \mathbf{K}'' \end{bmatrix} \qquad \boldsymbol{v}_c(k) = \begin{bmatrix} u(k) \\ \hat{\boldsymbol{y}}(k) \\ u^2(k) \\ \hat{\boldsymbol{y}}(k)u(k) \\ \boldsymbol{\epsilon}(k) \\ \boldsymbol{\epsilon}(k)u(k) \end{bmatrix} \tag{9.31b}$$

Propagating equation (9.30) forward in time by $p_c - 1$ steps, we obtain

$$\hat{\boldsymbol{x}}(k+p_c) = \boldsymbol{\mathcal{S}}_{c,p_c}\hat{\boldsymbol{x}}(k) + \mathbf{T}_{c,p_c}\boldsymbol{z}_{c,p_c}(k+p_c) \tag{9.32}$$

where $\boldsymbol{\mathcal{S}}_{c,p_c}$ is defined as in equation (9.9), except for $\bar{\mathbf{A}}_M$ and $\bar{\mathbf{N}}_M$ being replaced by $\bar{\mathbf{A}}_c$ and $\bar{\mathbf{N}}_c$,

and the causal superstate $\boldsymbol{z}_{c,p_c}(k)$, made of input-output data at steps $k-1, k-2, ..., k-p_c$ only, is defined like $\boldsymbol{z}_p(k)$ in equation (9.10) except for $\boldsymbol{v}(k)$ replaced by $\boldsymbol{v}_c(k)$ in equation (9.31b). By invoking theorem 9.1 and shifting the time index backward by p_c steps, equation (9.32) yields the causal IOSR

$$\hat{\boldsymbol{x}}(k) = \mathbf{T}_{c,p_c}\boldsymbol{z}_{c,p_c}(k) \tag{9.33}$$

The mixed-anticausal IOSR for the optimal bilinear observer can similarly be derived, following the steps in section 6.5.4, to obtain

$$\hat{\boldsymbol{x}}(k) = \mathbf{T}_{ma,p_c,p_a}\boldsymbol{z}_{ma,p_c,p_a}(k) \tag{9.34}$$

where the mixed-anticausal superstate is defined as

$$\boldsymbol{z}_{ma,p_c,p_a}(k) = \begin{bmatrix} \boldsymbol{z}_{c,p_c}(k+p_a-1)u(k+p_a-1) \\ \vdots \\ \boldsymbol{z}_{c,p_c}(k)u(k) \\ \boldsymbol{v}_a(k+p_a) \\ \vdots \\ \boldsymbol{v}_a(k+1) \end{bmatrix} \tag{9.35}$$

with

$$\boldsymbol{v}_a(k+1) = \begin{bmatrix} u(k) & u(k+1) & \hat{\boldsymbol{y}}^{\mathrm{T}}(k+1) & u^2(k) & \hat{\boldsymbol{y}}^{\mathrm{T}}(k)u(k) & \boldsymbol{\epsilon}^{\mathrm{T}}(k) & \boldsymbol{\epsilon}^{\mathrm{T}}(k)u(k) \end{bmatrix}^{\mathrm{T}} \tag{9.36}$$

9.7 Examples

Numerical examples are provided to demonstrate bilinear OKID and show more details about its implementation. Since the proposed method can be interpreted as the stochastic extension of the deterministic bilinear identification method described in chapter 6, the examples refer to the same bilinear systems utilized in the latter and highlight how the new method can achieve accurate identification even in cases where the direct application of the deterministic algorithms of chapter 6 fail because of the process and measurement noise affecting the data.

In each example, measured input-output data are simulated as follows. First we generate a random input sequence $\{u(k)\}$ of 10,000 samples (from a uniform distribution between -0.5 and 0.5) and two zero-mean gaussian sequences $\{\boldsymbol{w}_p(k)\}$ and $\{\boldsymbol{w}_m(k)\}$, respectively with covariance

$$\mathbf{Q} = \begin{bmatrix} 1 & 2 \\ 2 & 4 \end{bmatrix} \times 10^{-4} \qquad R = 10^{-4} \tag{9.37}$$

Said sequences are used to generate the measured output via equation (8.1).

9.7.1 Ideal bilinear system

The following system is used as a prototype of ideal bilinear system (chapter 6)

$$\mathbf{A} = \begin{bmatrix} 0 & 0.5 \\ 0.5 & -0.5 \end{bmatrix} \qquad \mathbf{B} = \begin{bmatrix} 1 \\ 2 \end{bmatrix} \qquad \mathbf{N} = \begin{bmatrix} 0 & 1 \\ -1 & 1 \end{bmatrix} \qquad \mathbf{C} = \begin{bmatrix} 0 & 1 \end{bmatrix} \qquad D = 0 \tag{9.38}$$

The first step of the bilinear OKID method is the estimation of the residuals of the optimal observer of equation (9.2). This can be done by LS on the set of equations arising from the bilinear ARX model. However notice how the ideal property of the system of equation (9.38) is

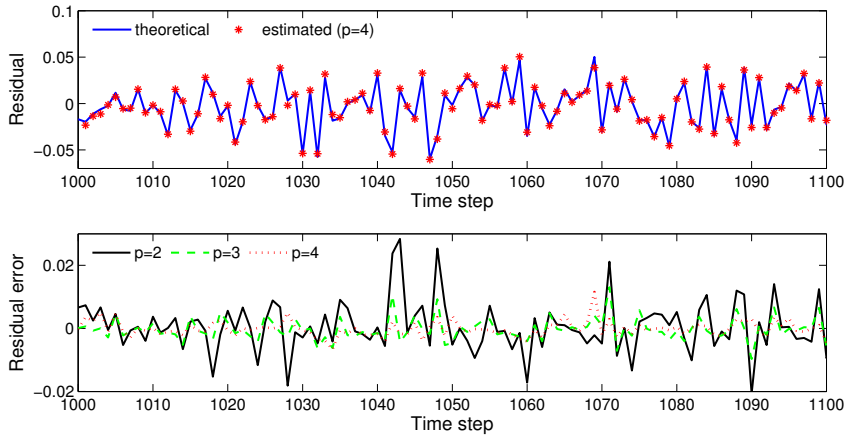
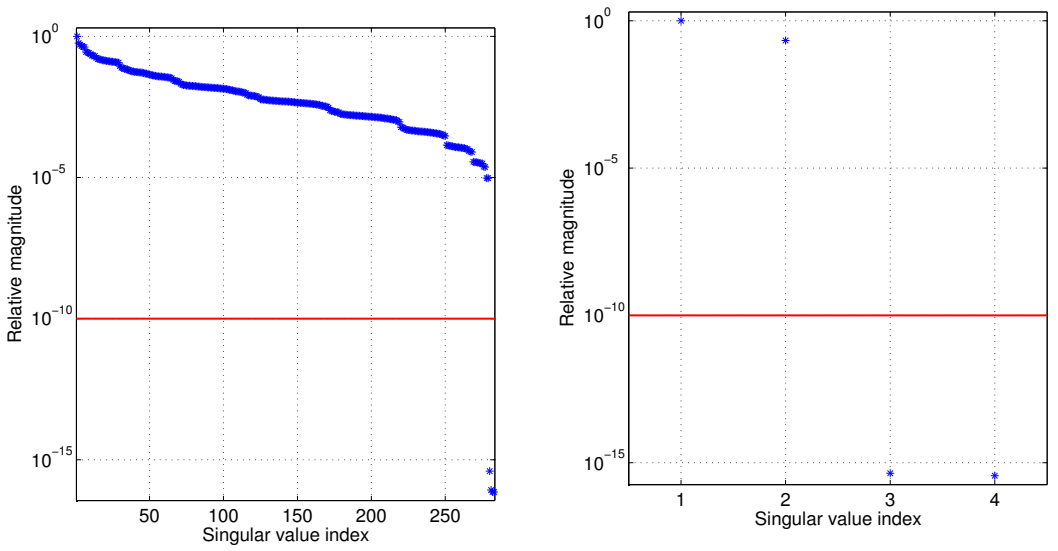


FIGURE 9.1: Estimation of observer residuals for ideal system: comparison for different values of p.

irrelevant, since the poles of $\bar{\mathbf{A}}_H$ and $\bar{\mathbf{N}}_H$ are dictated by the statistics of the noise embedded in the input-output data. As a consequence, the noise structure determines how $\boldsymbol{\mathcal{S}}_{L,p}$ converges to $\mathbf{0}$ with p. Figure 9.1 shows the error in the estimation of the observer residuals over a portion of the dataset. As expected, for $p = 2$ the estimates are poor because $\boldsymbol{\mathcal{S}}_{L,2}$ is not negligible. The estimates improve as p increases. It is remarkable how it is possible to accurately estimate a white random process such as the observer residual sequence.

Having now estimated $\boldsymbol{\epsilon}(k)$ for $k \geq p$, say with $p = 4$, we can proceed with the second step of bilinear OKID, i.e. the identification of the observer of equation (9.2), using the IS method with causal and mixed-anticausal IOSRs with $p_c = 4$ and $p_a = 2$. The choice of the values for p, p_c and p_a is consistent with the assumption that the true order of the system is unknown and believed to be $n \leq 4$. When intersecting the two IOSR superspaces to reconstruct the sequence of the observer state $\hat{\boldsymbol{x}}$, two SVDs have to be performed and they reveal the order of the observer in equation (9.2), which is the same as the order of the system of equation (8.1). In order to reduce the SVD computational effort, the identification of the observer can be done on a reduced portion of the record. In this example, we select the last 3000 samples. Figure 9.2a shows the first SVD, to find vectors spanning the intersection subspace of the two superspaces. At this



(A) Intersection of superspaces $\mathbf{Z}_a$ and $\mathbf{Z}_b$ (zero singular values are selected).

(B) Minimum basis for the intersection subspace (non-zero singular values are selected).

FIGURE 9.2: Observer identification for ideal system: SVDs of the IS method.

stage we need to split the zero and non-zero singular values, and the cut is indeed very clear. Figure 9.2b shows the second SVD, performed to find a basis for the intersection subspace. Here we need to retain the non-zero singular values, and again the difference with respect to the zero singular values is of 10 order of magnitudes, making the selection unquestionable. Since two singular values are retained, the order of the identified model will be $n_{id} = 2$, which is indeed correct. It is worth adding here that applying directly the IS method to the identification of the bilinear system (as prescribed by the deterministic identification method in chapter 6), without passing through the identification of the associated observer, makes the singular values of both SVDs decrease in a continuous fashion, making it impossible to identify the order of the system and reconstruct correctly the state sequence. In other words, the noise embedded in the data can heavily affect the outcome of the identification methods presented in chapter 6, where it is assumed the measured data are noise-free. The bilinear OKID method proposed in this paper represents a significant improvement over chapter 6 for any practical application, where noise cannot be eliminated.

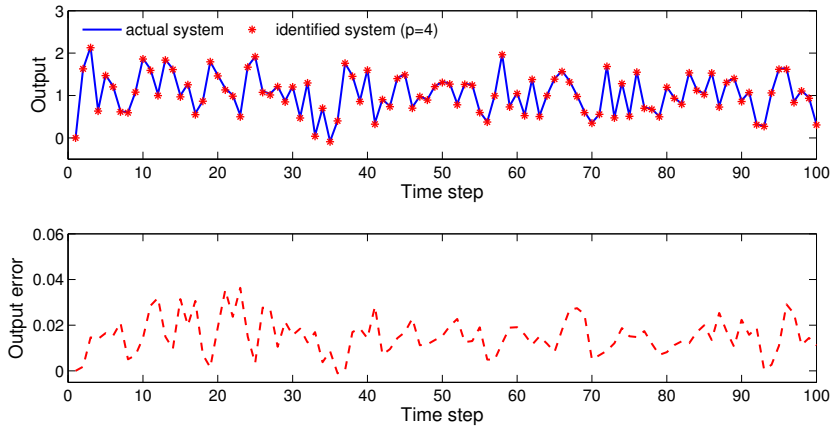


FIGURE 9.3: Comparison of output from actual ideal system and identified models.

The right singular vectors from the second SVD provide the observer state sequence that allows one to construct the LS problem of equation (9.29) and solve it for the observer matrices. The estimated matrices **A**, **B**, **N**, **C**, **D** yield the state-space model for the bilinear system. Its accuracy can be verified by driving both the true and identified models with the same input sequence (without noise), generated independently from the one used for the identification, and comparing the corresponding output, as shown in Figure 9.3.

9.7.2 Arbitrary bilinear system

Modifying matrix **N** in equation (9.38) to

$$\mathbf{N} = \begin{bmatrix} 0.3 & 1 \\ -1 & 1 \end{bmatrix} \tag{9.39}$$

is sufficient to lose the above mentioned ideal properties (chapter 6). The resulting system is therefore an example of arbitrary bilinear system. In the deterministic case addressed in chapter 6, the identification process is affected by whether the underlying system is ideal or arbitrary, the former leading to an exact identified model even for minimum values of p ($p = n$)

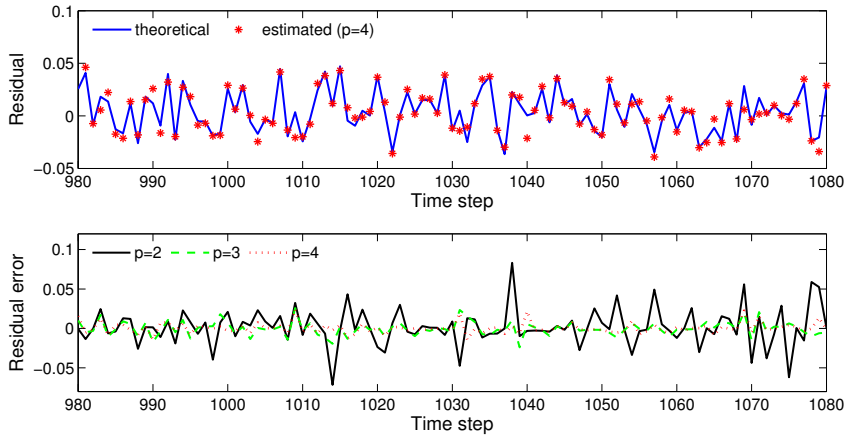


FIGURE 9.4: Estimation of observer residuals for arbitrary system: comparison for different values of p.

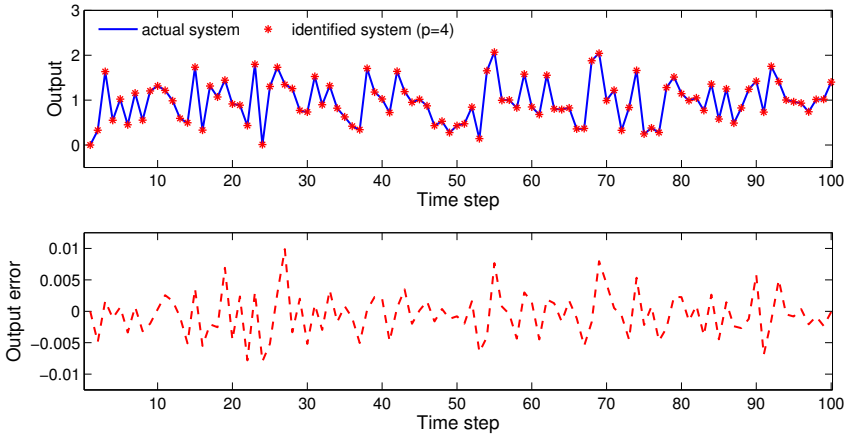


FIGURE 9.5: Comparison of output from actual arbitrary system and identified models.

and the latter providing comparable accuracy only for significantly larger values of p. In contrast, when noise is present in the input-output data, any identification method can only estimate an approximate model. Therefore, the difference between ideal and arbitrary bilinear system identification is less significant in the presence of noise. As shown in Figure 9.4, the estimates of the residuals improve as p is increased.

Using the residuals estimated with $p = 4$, the following bilinear model is identified (matrices rounded to the 4^{th} significant digit) via the IS method with causal and mixed-anticausal IOSRs $(p_c = 4, p_a = 2)$

$$\mathbf{A}_{id} = \begin{bmatrix} -0.8525 & -0.1467 \\ 0.3453 & 0.3529 \end{bmatrix} \qquad \mathbf{B}_{id} = \begin{bmatrix} -0.02807 \\ -0.04807 \end{bmatrix} \tag{9.40a}$$

$$\mathbf{N}_{id} = \begin{bmatrix} 0.4323 & -0.6828 \\ 1.352 & 0.8648 \end{bmatrix} \qquad \mathbf{C}_{id} = \begin{bmatrix} -34.50 & -21.40 \end{bmatrix} \qquad D_{id} = 6.805 \times 10^{-3} \tag{9.40b}$$

Again, the system order is obtained by singular value plots with a clear difference between zero and non-zero singular values, similar to the ones in Figures 9.2a and 9.2b. The accuracy of the identified model can be assessed by comparing the predicted output with the true output when both the identified model and the true system are driven by the same input sequence (without noise), as shown in Figure 9.5. Note that the LS solution to equation (9.29a) also yields

$$\mathbf{K}_{1,id} = -0.01314 \qquad \mathbf{K}_{2,id} = -0.008434 \tag{9.41}$$

Such gains can be used to construct a bilinear observer for the system of equation (9.40). In operation conditions where the input is white, the identification experiment can be performed on line and the proposed bilinear OKID method provides simultaneously both the system and observer mathematical models, which can then be used in similar operation conditions. The key advantage with respect to separate system identification and observer design lies with the difficulties in estimating the process noise covariance $\mathbf{Q}$ (necessary, together with $\mathbf{R}$, to design $\mathbf{K}'$ and $\mathbf{K}''$ as shown in Reference 69). The difficulties are completely overcome by bilinear OKID, which yields the optimal observer gains without requiring knowledge of $\mathbf{Q}$ or $\mathbf{R}$.

9.8 Conclusions

This chapter has presented the first extension to nonlinear systems of the well-established OKID approach for linear system identification, under the sole additional assumption that the excitation input is stationary and white. The result is a new method for bilinear system identification in the presence of process and measurement noise. The method relies on a bilinear steady-state observer, which is proven to have properties similar to the well-known linear Kalman filter. The estimation of the residuals of such bilinear observer allows one to construct a new noise-free identification problem that can be solved via the methods developed in chapter 6. The resulting bilinear OKID method represents indeed a significant improvement over such methods, being able to accurately identify the bilinear system under consideration even in cases where the noise prevents the direct application of the previous methods from identifying even the correct order. In other words, the presented bilinear OKID method can be interpreted as the extension of the methods presented in chapter 6 to the case where the measured input-output data are corrupted by noise. The need for white excitation input parallels the result in Reference 71, which presents a subspace method for the identification of bilinear systems in the presence of noise. OKID is indeed an alternative approach to subspace methods and has proven to be very successful in linear system identification, in particular in cases where one is interested in the system model as well as in the corresponding optimal observer. Therefore this paper represents a significant step towards the realization of the bridge between linear and nonlinear systems in the areas of system identification and controls, as mentioned in chapter 1. Further research will aim to remove the requirements on the excitation input to be used in the identification process, in particular its whiteness, and to reduce the curse of dimensionality as the parameter p is increased. The latter problem has already been successfully addressed in the deterministic case (chapter 7).

Chapter 10

Conclusions and future work

10.1 Conclusions

This work significantly extended OKID/ERA (observer/Kalman filter identification followed by eigensystem realization algorithm), a popular method for simultaneous identification of a linear state-space model and the associated Kalman filter from noisy input-output measurements.

From a single method, OKID has been raised to the level of a general and unified approach to system identification. OKID has been shown to be unifying as any linear state-space model identification algorithm formulated from noise-free data can be turned into an identification algorithm capable of optimally handling noise by the preliminary application of the OKID core equation. OKID has also been demonstrated to be a general approach to system identification as its framework has been applied to develop novel algorithms for output-only linear system identification (i.e. identification from output measurements only, as is typical in structural health monitoring) as well as bilinear system identification. The extension to the bilinear case required

the development of an optimal observer in bilinear form and, following the same approach, OKID might be extended to other classes of nonlinear systems, too.

From a more practical point of view, this work has provided the engineering community with:

- New algorithms for linear state-space model identification both for the input-output and the output-only case, one of them (OKID/DPi) with particularly promising performance as illustrated via numerical and experimental examples.
- Novel methods for bilinear system identification both in the absence and in the presence of noise in the data.
- A potential new approach to nonlinear system identification, not only because OKID has proven to be successful for bilinear systems (which represent a specific class of nonlinear systems), but also because bilinear models are claimed to be universal approximators of more general nonlinear systems.
- Thorough understanding and intuitive interpretation of the general and unifying framework offered by OKID, which is expected to make OKID more accessible to students, researchers and practitioners.

10.2 Future work

The new results in OKID for linear systems and the successful extension of OKID to bilinear models open several new possibilities for future work.

First of all, a thorough comparison of the presented OKID methods with subspace methods (Reference 2) could be undertaken. The latter represent another general framework for linear system identification and the question of which approach and which specific algorithm should be

preferred, maybe in different applications, is legitimate and of practical interest. However, due to the stochastic nature of the problem (noisy data), it seems difficult to address the question from a theoretical point of view. The use of Monte Carlo simulations is viable but the conclusions might be difficult to generalize.

Another path to investigate is what could be called OKID/OKID or OKID^2, i.e. the repeated application OKID when solving the new identification problem constructed from the output residuals estimated by the OKID core equation. Since such estimates are not exact, the new identification problem is generally not perfectly deterministic and using OKID instead of a deterministic identification method might be beneficial.

Also, the concept that the Kalman filter is the bridge (interface) between the noisy measurements and the identification of the system model suggests what follows. Techniques aiming at the estimation of the model matrices based on the optimization of a cost function minimizing, in a least-squares sense, the error between the measurements and the output predicted by the system model should replace the latter with the Kalman filter model to obtain asymptotically unbiased estimates.

Finally, some preliminary results shown in chapter 8 suggest that some of the methods presented for bilinear system identification can be applied to find bilinear models approximating more general nonlinear systems. More work in this direction might be needed to make said methods robust to noise both from the measurements and from the bilinear approximation. As mentioned above, the framework developed for OKID might be exploited to develop methods directly identifying other classes of nonlinear systems as well, following the same strategy adopted for bilinear models.

In summary, it is expected that new interesting results in the field of system identification will be found following the ideas illustrated and demonstrated in this work.

Bibliography

[1] J.-N. Juang, *Applied System Identification*. New Jersey: Prentice Hall, 1994.

[2] P. Van Overschee and B. Moor, *Subspace Identification for Linear Systems: Theory, Implementation, Applications*. Dordrecht: Kluwer Academic Publishers, 1996.

[3] C. Bruni, G. Di Pillo, and G. Koch, "Bilinear Systems: An Appealing Class of "Nearly Linear" Systems in Theory and Applications," *IEEE Transactions on Automatic Control*, Vol. AC-19, No. 4, 1974, pp. 334–348.

[4] R. R. Mohler, *Nonlinear Systems. Applications to Bilinear Control*, Vol. II. New Jersey: Prentice-Hall, 1991.

[5] D. L. Elliott, "Bilinear Systems," *Encyclopedia of Electrical Engineering* (J. Webster, ed.), Vol. II, pp. 308–323, New York: John Wiley and Sons, 1999.

[6] J. T.-H. Lo, "Global Bilinearization of Systems with Control Appearing Linearly," *SIAM Journal on Control*, Vol. 13, No. 4, 1975, pp. 879–885.

[7] S. Svoronos, G. Stephanopoulos, and A. Rutherford, "Bilinear Approximation of General Non-Linear Dynamic Systems with Linear Inputs," *International Journal of Control*, Vol. 31, No. 1, 1980, pp. 109–126.

[8] K. Kowalski and W.-H. Steeb, *Nonlinear Dynamical Systems and Carleman Linearization*. Singapore: World Scientific, 1991.

[9] C.-H. Lee and J.-N. Juang, "Nonlinear System Identification - A Continuous-Time Bilinear State Space Approach," *Advances in the Astronautical Sciences*, Vol. 139, 2011, pp. 421–444.

[10] N. B. Hizir, *Identification of Discrete-Time Bilinear Systems through Equivalent Linear Models*. PhD thesis, Columbia University, 2010.

[11] J.-N. Juang, "Continuous-Time Bilinear System Identification," *Nonlinear Dynamics*, Vol. 39, No. 1-2, 2005, pp. 79–94.

[12] J.-N. Juang, "Generalized Bilinear System Identification," *The Journal of the Astronautical Sciences*, Vol. 57, No. 1/2, 2009, pp. 261–273.

[13] J.-N. Juang and C.-H. Lee, "Continuous-Time Bilinear System Identification Using Single Experiment with Multiple Pulses," *Nonlinear Dynamics*, Vol. 69, No. 3, 2012, pp. 1009–1021.

[14] C.-H. Lee and J.-N. Juang, "System Identification for a General Class of Observable and Reachable Bilinear Systems," *Journal of Vibration and Control, doi:10.1177/1077546312473768*, in press.

[15] T. Clarke and X.-D. Sun, "Minimal State-Space Model Realisation of a Nonlinear Helicopter," *Control Theory and Applications, IEE Proceedings*, Vol. 145, No. 4, 1998, pp. 415–422.

[16] J. Valasek and W. Chen, "Observer/Kalman Filter Identification for Online System Identification of Aircraft," *Journal of Guidance, Control, and Dynamics*, Vol. 26, No. 2, 2003, pp. 347–353.

[17] A. Tiano, R. Sutton, A. Lozowicki, and W. Naeem, "Observer Kalman Filter Identification of an Autonomous Underwater Vehicle," *Control Engineering Practice*, Vol. 15, 2007, pp. 727–739.

[18] G. Heredia and A. Ollero, "Detection of Sensor Faults in Small Helicopter UAVs using Observer/Kalman Filter Identification," *Mathematical Problems in Engineering*, Vol. 2011, 2011.

[19] W. Gawronski, H. G. Ahlstrom, and A. M. Bernardo, "Control Systems of the NASA 70-Meter Antennas," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Austin, TX, 2013.

[20] J.-N. Juang, M. Q. Phan, L. G. Horta, and R. W. Longman, "Identification of Observer/Kalman Filter Markov Parameters: Theory and Experiments," *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 2, 1993, pp. 320–329.

[21] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediction and Control*. Englewood Cliffs: Prentice-Hall, 1984.

[22] H. Lus, R. Betti, and R. W. Longman, "Identification of Linear Structural Systems Using Earthquake-Induced Vibration Data," *Earthquake Engineering and Structural Dynamics*, Vol. 28, 1999, pp. 1449–1467.

[23] H. Lus, R. Betti, and R. W. Longman, "Obtaining Refined First-Order Predictive Models of Linear Structural Systems," *Earthquake Engineering and Structural Dynamics*, Vol. 31, 2002, pp. 1413–1440.

[24] R. E. Kalman, "A New Approach to Linear Filtering and Prediction Problems," *Transactions of the ASME–Journal of Basic Engineering*, Vol. 82, No. Series D, 1960, pp. 35–45.

[25] F. Vicario, M. Q. Phan, R. Betti, and R. W. Longman, "OKID as a Unified Approach to System Identification," *Advances in the Astronautical Sciences*, Vol. 152, 2014, pp. 3443–3460.

[26] J.-N. Juang, L. G. Horta, and M. Q. Phan, "System/Observer/Controller/Identification Toolbox," *NASA Technical Memorandum 107566*, 1992.

[27] M. Q. Phan, J.-N. Juang, and R. W. Longman, "Identification of Linear Multivariable Systems by Identification of Observers with Assigned Real Eigenvalues," *Journal of the Astronautical Sciences*, Vol. 40, No. 2, 1992, pp. 261–27.

[28] M. Q. Phan, L. G. Horta, J.-N. Juang, and R. W. Longman, "Linear System Identification via an Asymptotically Stable Observer," *Journal of Optimization Theory and Applications*, Vol. 79, No. 1, 1993, pp. 59–86.

[29] M. Moonen, B. De Moor, L. Vandenberghe, and J. Vanderwalle, "On- and Off-Line Identification of Linear State Space Models," *International Journal of Control*, Vol. 49, No. 1, 1989, pp. 219–232.

[30] B. De Moor and J. Vanderwalle, "A Geometrical Strategy for the Identification of State Space Models of Linear Multivariable Systems with Singular Value Decomposition," *Proceedings of the 3rd International Symposium on Applications of Multivariable System Techniques*, Plymouth, UK, 1987, pp. 59–69.

[31] M. Q. Phan, J. A. Solbeck, and L. R. Ray, "A Direct Method For State-Space Model and Observer/Kalman Filter Gain Identification," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Providence, Rhode Island, 2004.

[32] P. Lin, M. Q. Phan, and S. A. Ketcham, "State-Space Model and Kalman Filter Gain Identification by a Superspace Method," *Proceedings of the 5th International Conference on Modeling, Simulation and Optimization of Complex Processes*, Hanoi, Vietnam, 2012.

[33] M. Q. Phan and H. Celik, "A Superspace Method for Discrete-Time Bilinear Model Identification by Interaction Matrices," *Advances in the Astronautical Sciences*, Vol. 139, 2011, pp. 445–464.

[34] M. Q. Phan, L. G. Horta, J.-N. Juang, and R. W. Longman, "Improvement of Observer/Kalman Filter Identification (OKID) by Residual Whitening," *Journal of Vibrations and Acoustics*, Vol. 117, 1995, pp. 232–238.

[35] P. Van Overschee and B. De Moor, "N4SID: Subspace Algorithms for the Identification of Combined Deterministic-Stochastic Systems," *Automatica*, Vol. 30, No. 1, 1994, pp. 75–93.

[36] E. Figueiredo, G. Park, J. Figueiras, C. Farrar, and K. Worden, "Structural Health Monitoring Algorithm Comparisons Using Standard Data Sets," *Los Alamos National Laboratory Internal Report*, 2009, LA-14393.

[37] J. M. W. Brownjohn, A. A. Dumanoglu, and S. R. T., "Ambient Vibration Survey of the Fathih Sultan Mehmet (Second Bosporus) Suspension Bridge," *Earthquake Engineering and Structural Dynamics*, Vol. 21, 1992, pp. 907–924.

[38] B. Peeters and G. De Roeck, "Stochastic System Identification for Operational Modal Analysis: a Review," *Journal of Dynamic Systems, Measurement and Control*, Vol. 123, 2001, pp. 659–667.

[39] R. Brincker, P. Andersen, and N. J. Jacobsen, "Modal Identification of Output-Only Systems Using Frequency Domain Decomposition," *Smart Materials and Structures*, Vol. 10, No. 3, 2001, pp. 441–445.

[40] L. Hermans, P. Guillaume, and H. V. d. Auweraer, "A Frequency Domain Maximum Likelihood Approach for the Extraction of Modal Parameters from Output-Only Data," *Proceedings of ISMA 23, Noise and Vibration Engineering*, K. U. Leuven, Belgium, 1998.

[41] J. Schoukens and R. Pintelon, *Identification of Linear Systems: a Practical Guideline to Accurate Modelling*. Pergamon Press, 1991.

[42] G. H. I. James, T. G. Carne, and J. P. Lauffer, "The Natural Excitation Technique (NExT) for Modal Parameter Extraction from Operating Wind Turbines," *Sandia Report*, Vol. SAND92-1666, No. UC-261, 1993.

[43] F. Vicario, M. Q. Phan, R. Betti, and R. W. Longman, "Extension of OKID to Output-Only System Identification," *Sixth World Conference on Structural Control and Monitoring*, Barcelona, 2014.

[44] F. Vicario, M. Q. Phan, R. Betti, and R. W. Longman, "Output-Only Observer/Kalman Filter Identification (O^3KID)," *Structural Control and Health Monitoring*, (accepted for publication).

[45] H. Lus, M. De Angelis, R. Betti, and R. W. Longman, "Constructing Second-Order Models of Mechanical Systems from Identified State Space Realizations. Part I: Theoretical Discussions," *Journal of Engineering Mechanics*, Vol. 129, No. 5, 2003, pp. 477–488.

[46] J.-N. Juang and R. W. Longman, "Optimized System Identification," *NASA Technical Memorandum 209711*, 1999.

[47] T. C. Hsia, "On Least Squares Algorithms for System Parameter Identification," *IEEE Transactions on Automatic Control*, Vol. 21, 1976, pp. 104–108.

[48] K. J. Astrom and P. Eykhoff, "System Identification - A Survey," *Automatica*, Vol. 7, 1971, pp. 123–162.

[49] N. B. Hizir, M. Q. Phan, R. Betti, and R. W. Longman, "Identification of Discrete-Time Bilinear Systems through Equivalent Linear Models," *Nonlinear Dynamics*, Vol. 69, No. 4, 2012, pp. 2065–2078.

[50] F. Vicario, M. Q. Phan, R. Betti, and R. W. Longman, "Linear State Representations for Identification of Bilinear Discrete-Time Models by Interaction Matrices," *Advances in the Astronautical Sciences*, Vol. 148, 2013, pp. 2039–2058.

[51] F. Vicario, M. Q. Phan, R. Betti, and R. W. Longman, "Linear State Representations for Identification of Bilinear Discrete-Time Models by Interaction Matrices," *Nonlinear Dynamics*, Vol. 77, No. 4, 2014, pp. 1561–1576.

[52] H. Celik and M. Q. Phan, "Identification of Input-Output Maps for Bilinear Discrete-Time State-Space Models," *Advances in the Astronautical Sciences*, Vol. 142, 2012, pp. 393–412.

[53] B. De Moor, M. Moonen, L. Vandenberghe, and J. Vandewalle, "The Application of the Canonical Correlation Concept to the Identification of Linear State Space Models," *Analysis and Optimization of Systems* (A. Bensoussan and J. Lions, eds.), Vol. 111 of *Lecture Notes in Control and Information Sciences*, pp. 1103–1114, Springer Berlin Heidelberg, 1988.

[54] N. Goodzeit and M. Q. Phan, "System Identification in the Presence of Completely Unknown Periodic Disturbances," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 2, 2000, pp. 251–259.

[55] N. Goodzeit and M. Q. Phan, "System and Disturbance Identification for Feedforward Feedback Control Applications," *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 2, 2000, pp. 260–268.

[56] N. Goodzeit and M. Q. Phan, "A Clear-Box Adaptive System for Flexible Spacecraft Identification and Disturbance Rejection," *Journal of Vibration and Control*, Vol. 6, 2000, pp. 757–780.

[57] B. H. Koh, Z. Li, P. Dharap, S. Nagarajaiah, and M. Q. Phan, "Actuator Failure Detection Through Interaction Matrix Formulation," *Journal of Guidance, Control, and Dynamics*, Vol. 28, No. 5, 2005, pp. 895–901.

[58] R. S. Darling and M. Q. Phan, "Decentralized Predictive Control for Tracking and Disturbance Rejection," *Proceedings of the AIAA Guidance, Navigation, and Control Conference and Exhibit*, Keystone, CO, 2006.

[59] M. Q. Phan, "Interaction Matrices in System Identification and Control," *Proceedings of the 15th Yale Workshop on Adaptive and Learning Systems*, Center for Systems Science, Yale University, New Haven, CT, 2011.

[60] K. E. Bouazza, M. Boutayeb, and M. Darouach, "State and Output Feedback Stabilization of a Class of Discrete-Time Nonlinear Systems," *Proceedings of the 2004 American Control Conference*, Vol. 4, 2004, pp. 3023–3028.

[61] F. Vicario, M. Q. Phan, R. W. Longman, and R. Betti, "A Linear-Time-Varying Approach for Exact Identification of Bilinear Discrete-Time Systems by Interaction Matrices," *Advances in the Astronautical Sciences*, Vol. 150, 2014, pp. 1057–1076.

[62] S. Hara and K. Furuta, "Observability for Bilinear Systems," *International Journal of Control*, Vol. 26, No. 4, 1977, pp. 559–572.

[63] E. Sontag, Y. Wang, and A. Megretski, "Input Classes for Identifiability of Bilinear Systems," *IEEE Transactions on Automatic Control*, Vol. 54, No. 2, 2009, pp. 195–207.

[64] M. Q. Phan, Y. Shi, R. Betti, and R. W. Longman, "Discrete-Time Bilinear Representation of Continuous-Time Bilinear State-Space Models," *Advances in the Astronautical Sciences*, Vol. 143, 2012, pp. 571–589.

[65] S. Hara and K. Furuta, "Minimal Order State Observers for Bilinear Systems," *International Journal of Control*, Vol. 24, No. 5, 1976, pp. 705–718.

[66] Y. Funahashi, "Stable State Estimator for Bilinear Systems," *International Journal of Control*, Vol. 29, No. 2, 1979, pp. 181–188.

[67] O. M. Grasselli and A. Isidori, "An Existence Theorem for Observers of Bilinear Systems," *IEEE Transactions on Automatic Control*, Vol. AC-26, No. 6, 1981, pp. 1299–1300.

[68] I. A. Derese and E. J. Noldus, "Existence of Bilinear State Observers for Bilinear Systems," *IEEE Transactions on Automatic Control*, Vol. AC-26, No. 2, 1981, pp. 590–592.

[69] M. Q. Phan, F. Vicario, R. W. Longman, and R. Betti, "Observers for Bilinear State-Space Models by Interaction Matrices," *Proceedings of the 16th Yale Workshop on Adaptive and Learning Systems*, Center for Systems Science, Yale University, New Haven, CT, 2013.

[70] F. Vicario, M. Q. Phan, R. Betti, and R. W. Longman, "Bilinear Observer/Kalman Filter Identification," *Advances in the Astronautical Sciences*, Vol. 152, 2014, pp. 1517–1536.

[71] W. Favoreel, B. De Moor, and P. Van Overschee, "Subspace Identification of Bilinear Systems Subject to White Inputs," *Automatic Control, IEEE Transactions on*, Vol. 44, No. 6, 1999, pp. 1157–1165.

[72] J.-N. Juang and R. S. Pappa, "An Eigensystem Realization Algorithm for Modal Parameter Identification and Model Reduction," *Journal of Guidance, Control, and Dynamics*, Vol. 8, No. 5, 1985, pp. 620–627.

[73] J.-N. Juang, J. E. Cooper, and J. R. Wright, "An Eigensystem Realization Algorithm Using Data Correlations (ERA/DC) for Modal Parameter Identification," *Journal of Control Theory and Advanced Technology*, Vol. 4, No. 1, 1988, pp. 5–14.

Appendix A

Algorithms for deterministic system identification

As mentioned in section 1.1, many algorithms have been developed to identify linear state-space models, some of them of deterministic nature, i.e. without considering noise in the measured data, and others combined, i.e. with formulations minimizing the noise uncertainty in the identification. The OKID-based algorithms described in chapters 2-5 belong to the latter category. Nevertheless, the interpretation of the OKID core equation as a converter from combined to deterministic system identification makes it possible to develop combined system identification algorithms from any deterministic state-space model identification method, as discussed in illustrated in chapter 5 and illustrated in chapter 3.

In this appendix, the four deterministic identification methods used to demonstrate and illustrate OKID as a general approach to linear system identification are reviewed. For more details, see References 1, 2, 29, 30, 72, 73.

A.1 Eigensystem realization algorithm (ERA)

ERA (References 1, 72) is based on the fact that arranging the Markov parameters of the system in equation (1.1) in the following matrix with Hankel structure

$$\mathbf{H}_0 = \begin{bmatrix} \mathbf{CB} & \mathbf{CAB} & \dots & \mathbf{CA}^{N/2-1}\mathbf{B} \\ \mathbf{CAB} & \mathbf{CA}^2\mathbf{B} & \dots & \mathbf{CA}^{N/2}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{N/2-1}\mathbf{B} & \mathbf{CA}^{N/2}\mathbf{B} & \dots & \mathbf{CA}^{N-2}\mathbf{B} \end{bmatrix} \tag{A.1}$$

gives rise to the following relationship between the unit pulse response of the system and its extended observability and controllability matrices $\mathbf{\Gamma}_{N/2}$ and $\mathbf{\Theta}_{N/2}$

$$\mathbf{H}_0 = \mathbf{\Gamma}_{N/2}\mathbf{\Theta}_{N/2} \tag{A.2}$$

where

$$\mathbf{\Gamma}_{N/2} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{N/2-1} \end{bmatrix} \qquad \mathbf{\Theta}_{N/2} = \begin{bmatrix} \mathbf{K} & \mathbf{AK} & \dots & \mathbf{A}^{N/2-1}\mathbf{K} \end{bmatrix} \tag{A.3}$$

and N is an even integer. From control theory, provided that $qN \geq n$, both $\mathbf{\Gamma}_{N/2}$ and $\mathbf{\Theta}_{N/2}$ have rank n. Therefore $\mathbf{H}_0$ has rank n, too, and the SVD of $\mathbf{H}_0$

$$\mathbf{H}_0 = \begin{bmatrix} \mathbf{U}_1 & \mathbf{U}_2 \end{bmatrix} \begin{bmatrix} \mathbf{S}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}_1^{\mathrm{T}} \\ \mathbf{V}_2^{\mathrm{T}} \end{bmatrix} \tag{A.4}$$

will then have n non-zero singular values in the diagonal matrix $\mathbf{S}_1 \in \mathbb{R}^{n\times n}$, whereas the corresponding n left and right singular vectors are the columns of the matrices $\mathbf{U}_1 \in \mathbb{R}^{qN/2\times n}$ and $\mathbf{V}_1 \in \mathbb{R}^{qN/2\times n}$, respectively. The SVD of $\mathbf{H}_0$ does not only reveal the order n of $\mathbf{A}$, but also the observability and controllability matrices of the system. Hence, $\mathbf{H}_0$ can be written as

$$\mathbf{H}_0 = \mathbf{U}_1\mathbf{S}_1\mathbf{V}_1^{\mathrm{T}} \tag{A.5}$$

Comparison of (A.2) and (A.5) suggests that $\mathbf{\Gamma}_{N/2}$ is related to $\mathbf{U}_1$ and $\mathbf{\Theta}_{N/2}$ is related to $\mathbf{V}_1^{\mathrm{T}}$. Indeed, a possible choice is

$$\mathbf{\Gamma}_{N/2} = \mathbf{U}_1\mathbf{S}_1^{1/2} \qquad \mathbf{\Theta}_{N/2} = \mathbf{S}_1^{1/2}\mathbf{V}_1^{\mathrm{T}} \tag{A.6}$$

Other choices are possible, reflecting the fact that a change of coordinate system for the state variables gives a different but equivalent state-space model. In the literature, the choice in (A.6) is referred to as balanced (Reference 1). The first q rows of $\mathbf{\Gamma}_{N/2}$ and the first q columns of $\mathbf{\Theta}_{N/2}$ yield estimates for $\mathbf{C}$ and $\mathbf{K}$, respectively. To find an estimate for $\mathbf{A}$, another block Hankel matrix needs to be constructed via the Markov parameters of the system

$$\mathbf{H}_1 = \begin{bmatrix} \mathbf{CAB} & \mathbf{CA}^2\mathbf{B} & \dots & \mathbf{CA}^{N/2}\mathbf{B} \\ \mathbf{CA}^2\mathbf{B} & \mathbf{CA}^3\mathbf{B} & \dots & \mathbf{CA}^{N/2+1}\mathbf{B} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{N/2}\mathbf{B} & \mathbf{CA}^{N/2+1}\mathbf{B} & \dots & \mathbf{CA}^{N-1}\mathbf{B} \end{bmatrix} \tag{A.7}$$

which can be shown to satisfy

$$\mathbf{H}_1 = \mathbf{\Gamma}_{N/2}\mathbf{A}\mathbf{\Theta}_{N/2} = \mathbf{U}_1\mathbf{S}_1^{1/2}\mathbf{A}\mathbf{S}_1^{1/2}\mathbf{V}_1^{\mathrm{T}} \tag{A.8}$$

By recalling the orthogonality properties of $\mathbf{U}_1$ and $\mathbf{V}_1$, i.e. $\mathbf{U}_1^{\mathrm{T}}\mathbf{U}_1 = \mathbf{I}$ and $\mathbf{V}_1^{\mathrm{T}}\mathbf{V}_1 = \mathbf{I}$, (A.8) can be solved for $\mathbf{A}$ as proven in Reference 1

$$\mathbf{A} = \mathbf{S}_1^{-1/2}\mathbf{U}_1^{\mathrm{T}}\mathbf{H}_1\mathbf{V}_1\mathbf{S}_1^{-1/2} \tag{A.9}$$

ERA is summarized in the following steps.

ERA

1. construct the matrices $\mathbf{H}_0$ and $\mathbf{H}_1$ as in (A.1) and (A.7)
2. calculate $\mathbf{S}_1$, $\mathbf{U}_1$ and $\mathbf{V}_1$ from the SVD of $\mathbf{H}_0$ in (A.4)
3. read $\mathbf{C}$ as the first q rows of $\mathbf{U}_1\mathbf{S}_1^{1/2}$
4. read $\mathbf{K}$ as the first q columns of $\mathbf{S}_1^{1/2}\mathbf{V}_1^{\mathrm{T}}$
5. calculate $\mathbf{A} = \mathbf{S}_1^{-1/2}\mathbf{U}_1^{\mathrm{T}}\mathbf{H}_1\mathbf{V}_1\mathbf{S}_1^{-1/2}$

A.2 ERA with data correlation (ERA-DC)

An improved version of ERA can be devised (References 1, 73) defining the following square matrices of order $qN/2$

$$\boldsymbol{\mathcal{H}}_0 = \mathbf{H}_0\mathbf{H}_0^{\mathrm{T}} \tag{A.10a}$$

$$\boldsymbol{\mathcal{H}}_1 = \mathbf{H}_1\mathbf{H}_0^{\mathrm{T}} \tag{A.10b}$$

By recalling (A.1) and (A.7), $\boldsymbol{\mathcal{H}_0}$ and $\boldsymbol{\mathcal{H}_1}$ can be shown to consist of auto- and cross-correlations of the system Markov parameters. If noise[1] in the Markov parameters is not correlated, then $\boldsymbol{\mathcal{H}}_0$ and $\boldsymbol{\mathcal{H}}_1$ will contain less noise than $\mathbf{H}_0$ and $\mathbf{H}_1$. Hence the name and the benefit of the algorithm.

From (A.2) and (A.10a), we obtain

$$\boldsymbol{\mathcal{H}}_0 = \boldsymbol{\Gamma}_{N/2}\boldsymbol{\mathcal{Q}}_c \tag{A.11}$$

where

$$\boldsymbol{\mathcal{Q}}_c = \boldsymbol{\Theta}_{N/2}\boldsymbol{\Theta}_{N/2}^{\mathrm{T}}\boldsymbol{\Gamma}_{N/2}^{\mathrm{T}} \tag{A.12}$$

With an approach similar to basic ERA, $\boldsymbol{\mathcal{H}}_0$ can be decomposed by SVD into

$$\boldsymbol{\mathcal{H}}_0 = \begin{bmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{bmatrix}\begin{bmatrix}\mathbf{S}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{V}_1^{\mathrm{T}}\\ \mathbf{V}_2^{\mathrm{T}}\end{bmatrix} = \mathbf{U}_1\mathbf{S}_1\mathbf{V}_1^{\mathrm{T}} \tag{A.13}$$

where $\mathbf{S}_1 \in \mathbb{R}^{n\times n}$ contains the n non-zero singular values of $\boldsymbol{\mathcal{H}}_0$ and $\mathbf{U}_1, \mathbf{V}_1 \in \mathbb{R}^{qN/2\times n}$ contain the corresponding left and right singular vectors, respectively. Comparing (A.11) with (A.13), we can choose

$$\boldsymbol{\Gamma}_{\mathbf{N/2}} = \mathbf{U}_1\mathbf{S}_1^{1/2} \qquad \boldsymbol{\mathcal{Q}}_c = \mathbf{S}_1^{1/2}\mathbf{V}_1^{\mathrm{T}} \tag{A.14}$$

$\boldsymbol{\Theta}_{N/2}$ can be estimated from $\boldsymbol{\Gamma}_{N/2}$ obtained in (A.14) via (A.2)

$$\boldsymbol{\Theta}_{N/2} = \boldsymbol{\Gamma}_{N/2}^{\dagger}\mathbf{H}_0 = \mathbf{S}_1^{-1/2}\mathbf{U}_1^{\mathrm{T}}\mathbf{H}_0 \tag{A.15}$$

[1]Even though the system in equation (1.1) is formally not affected by noise, recall that all the deterministic identification algorithms described in this appendix are applied in the presented OKID framework with inputs that are estimated from originally noisy data, hence the estimation error is a source of noise, although attenuated by the OKID core equation (see also chapter 5)

The first q rows of $\mathbf{\Gamma}_{N/2}$ and the first q columns of $\mathbf{\Theta}_{N/2}$ yield estimates for $\mathbf{C}$ and $\mathbf{K}$, respectively.

To find an estimate for $\mathbf{A}$, the fact that

$$\boldsymbol{\mathcal{H}}_1 = \mathbf{\Gamma}_{N/2}\mathbf{A}\boldsymbol{\mathcal{Q}}_c \tag{A.16}$$

needs to be exploited to obtain, with the help of (A.14),

$$\mathbf{A} = \mathbf{S}_1^{-1/2}\mathbf{U}_1^{\mathrm{T}}\boldsymbol{\mathcal{H}}_1\mathbf{V}_1\mathbf{S}_1^{-1/2} \tag{A.17}$$

ERA-DC is summarized in the following steps.

ERA-DC

1. construct the matrices $\mathbf{H}_0$ and $\mathbf{H}_1$ as in (A.1) and (A.7)
2. construct the matrices $\boldsymbol{\mathcal{H}}_0$ and $\boldsymbol{\mathcal{H}}_1$ as in (A.10)
3. calculate $\mathbf{S}_1$, $\mathbf{U}_1$ and $\mathbf{V}_1$ from the SVD of $\mathbf{H}_0$ in (A.13)
4. read $\mathbf{C}$ as the first q rows of $\mathbf{U}_1\mathbf{S}_1^{1/2}$
5. read $\mathbf{K}$ as the first q columns of $\mathbf{S}_1^{-1/2}\mathbf{U}_1^{\mathrm{T}}\mathbf{H}_0$
6. calculate $\mathbf{A} = \mathbf{S}_1^{-1/2}\mathbf{U}_1^{\mathrm{T}}\boldsymbol{\mathcal{H}}_1\mathbf{V}_1\mathbf{S}_1^{-1/2}$

A.3 Deterministic intersection (DI) algorithm

For a more detailed derivation and rigorous justification of the DI method, see Reference 2, 29.

The property at the core of the DI algorithm for deterministic system identification is that the

state history of the system in equation (1.1) can be expressed via the following two matrix relations

$$\mathbf{X} = \mathbf{L}_p \mathbf{H}_p \qquad \mathbf{X} = \mathbf{L}_f \mathbf{H}_f \tag{A.18}$$

where

$$\mathbf{X} = \begin{bmatrix} \boldsymbol{x}(i) & \boldsymbol{x}(i+1) & \dots & \boldsymbol{x}(l-i) \end{bmatrix} \tag{A.19a}$$

$$\mathbf{H}_p = \begin{bmatrix} \boldsymbol{u}(0) & \boldsymbol{u}1) & \dots & \boldsymbol{u}(l-2i) \\ \boldsymbol{y}(0) & \boldsymbol{y}(1) & \dots & \boldsymbol{y}(l-2i) \\ \boldsymbol{u}(1) & \boldsymbol{u}(2) & \dots & \boldsymbol{u}(l-2i+1) \\ \boldsymbol{y}(1) & \boldsymbol{y}(2) & \dots & \boldsymbol{y}(l-2i+1) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{u}(i-1) & \boldsymbol{u}(i) & \dots & \boldsymbol{u}(l-i-1) \\ \boldsymbol{y}(i-1) & \boldsymbol{y}(i) & \dots & \boldsymbol{y}(l-i-1) \end{bmatrix} \tag{A.19b}$$

$$\mathbf{H}_f = \begin{bmatrix} \boldsymbol{u}(i) & \boldsymbol{u}(i+1) & \dots & \boldsymbol{u}(l-i) \\ \boldsymbol{y}(i) & \boldsymbol{y}(i+1) & \dots & \boldsymbol{y}(l-i) \\ \boldsymbol{u}(i+1) & \boldsymbol{u}(i+2) & \dots & \boldsymbol{u}(l-i+1) \\ \boldsymbol{y}(i+1) & \boldsymbol{y}(i+2) & \dots & \boldsymbol{y}(l-i+1) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{u}(2i-1) & \boldsymbol{u}(2i) & \dots & \boldsymbol{u}(l-1) \\ \boldsymbol{y}(2i-1) & \boldsymbol{y}(2i) & \dots & \boldsymbol{y}(l-1) \end{bmatrix} \tag{A.19c}$$

(A.19d)

and $\mathbf{L}_p$, $\mathbf{L}_f \in \mathbb{R}^{n \times (m+q)i}$ are two constant matrices. For (A.18) to hold, the parameter i must be such that $qi \geq n$. From a linear algebra perspective, (A.18) shows how the Kalman

state sequence lies both in the row space of $\mathbf{H}_p$ and in the row space of $\mathbf{H}_f$. $\mathbf{X}$ can then be reconstructed by intersection of the row spaces of $\mathbf{H}_p$ and $\mathbf{H}_f$, which can be accomplished by two singular value decompositions (SVDs). The most intuitive way to compute said intersection is to note that (A.18) implies that

$$\mathbf{L}_p\mathbf{H}_p - \mathbf{L}_f\mathbf{H}_f = \mathbf{0} \tag{A.20}$$

which can be rewritten as

$$\begin{bmatrix}\mathbf{L}_p & -\mathbf{L}_f\end{bmatrix}\begin{bmatrix}\mathbf{H}_p \\ \mathbf{H}_f\end{bmatrix} = \mathbf{0} \tag{A.21}$$

$\mathbf{L}_p$ and $\mathbf{L}_f$ can then be computed as the left singular vectors associated with the zero singular values of the comprehensive data matrix $\mathbf{H}$, defined as the concatenation of the past data matrix $\mathbf{H}_p$ and the future data matrix $\mathbf{H}_f$

$$\mathbf{H} = \begin{bmatrix}\mathbf{H}_p \\ \mathbf{H}_f\end{bmatrix} \tag{A.22}$$

The SVD of $\mathbf{H}$ is then needed

$$\mathbf{H} = \begin{bmatrix}\mathbf{U}_{11} & \mathbf{U}_{12} \\ \mathbf{U}_{21} & \mathbf{U}_{22}\end{bmatrix}\begin{bmatrix}\mathbf{S}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}\mathbf{V}^{\mathrm{T}} \tag{A.23}$$

where $\mathbf{S}_{11} \in \mathbb{R}^{(mi+qi+n)\times(mi+qi+n)}$ is a diagonal matrix with the non-zero singular values of $\mathbf{H}$, $\mathbf{V} \in \mathbb{R}^{(l-2i+1)\times(l-2i+1)}$ is the matrix with its right singular vectors, and the matrix $\mathbf{U}$ with the left singular vectors is partitioned into $\mathbf{U}_{11}, \mathbf{U}_{21} \in \mathbb{R}^{(m+q)i\times(mi+qi+n)}$ and $\mathbf{U}_{12}, \mathbf{U}_{22} \in \mathbb{R}^{(m+q)i\times(mi+qi-n)}$. By left-multiplying $\mathbf{H}$ by its left singular vectors corresponding to the zero singular values and recalling the orthogonality property of the singular vector matrix $\mathbf{U}$, we

obtain

$$\begin{bmatrix} \mathbf{U}_{12}^{\mathrm{T}} & \mathbf{U}_{22}^{\mathrm{T}} \end{bmatrix} \begin{bmatrix} \mathbf{H}_p \\ \\ \mathbf{H}_f \end{bmatrix} = \mathbf{0} \tag{A.24}$$

or equivalently

$$\mathbf{U}_{12}^{\mathrm{T}} \mathbf{H}_p = -\mathbf{U}_{22}^{\mathrm{T}} \mathbf{H}_f \tag{A.25}$$

Equation (A.25) shows that $\mathbf{U}_{12}^{\mathrm{T}} \mathbf{H}_p$ (or $-\mathbf{U}_{22}^{\mathrm{T}} \mathbf{H}_f$) represents the required intersection of the row spaces of $\mathbf{H}_p$ and $\mathbf{H}_f$. However, $\mathbf{U}_{12}^{\mathrm{T}} \mathbf{H}_p$ contains $mi + qi - n$ row vectors, only n of which are linearly independent. Another SVD can be taken to compute a basis of n linearly independent vectors for its row space. Such basis provides the sequence of the Kalman state[2] $\mathbf{X}$ and the identification of the system in equation (1.1) can then be completed by solving the following sets of linear equations

$$\begin{bmatrix} \boldsymbol{x}(p+i+1) & \dots & \boldsymbol{x}(l-i) \\ \\ \boldsymbol{y}(p+i) & \dots & \boldsymbol{y}(l-i-1) \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(p+i) & \dots & \boldsymbol{x}(l-i-1) \\ \\ \hat{\boldsymbol{u}}(p+i) & \dots & \hat{\boldsymbol{u}}(l-i-1) \end{bmatrix} \tag{A.26}$$

Since the true $\boldsymbol{u}$ and $\boldsymbol{y}$ are not known and only their estimates are available from the OKID equation (2.44), the above equations should be solved in a LS sense

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \\ \mathbf{C} & \mathbf{D} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}(p+i+1) & \dots & \boldsymbol{x}(l-i) \\ \\ \boldsymbol{y}(p+i) & \dots & \boldsymbol{y}(l-i-1) \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(p+i) & \dots & \boldsymbol{x}(l-i-1) \\ \\ \hat{\boldsymbol{u}}(p+i) & \dots & \hat{\boldsymbol{u}}(l-i-1) \end{bmatrix}^{\dagger} \tag{A.27}$$

Note that what described so far is the traditional solution scheme of DI. In the OKID approach, however, DI is used to identify the Kalman filter of equation (3.1) or equation (3.5), i.e a state-space model in which only part of the input to the state equation is also an input to the observation equation. Obviously, when solving equation (A.27) $\mathbf{D}$ is expected to be found equal

[2] As usual in state-space model identification, equivalent sets of system matrices can be found by changing the coordinate system. The basis found for $\mathbf{X}$ in the DI algorithm fixes the coordinate system in which $\mathbf{A}$, $\mathbf{C}$ and $\mathbf{K}$ are identified

to **0**. As verified by numerical and experimental examples, **D** turns out to be negligible and existing DI codes such as the one provided with Reference 2 can be used without modification. Additionally, note that the LS estimate of **A** in equation (A.27) is independent from the presence of **D** and if one is only interested in the eigenvalues of the system, **A** can be obtained via the following reduced equation

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \end{bmatrix} = \begin{bmatrix} \boldsymbol{x}(p+i+1) & \dots & \boldsymbol{x}(l-i) \end{bmatrix} \begin{bmatrix} \boldsymbol{x}(p+i) & \dots & \boldsymbol{x}(l-i-1) \\ \hat{\boldsymbol{u}}(p+i) & \dots & \hat{\boldsymbol{u}}(l-i-1) \end{bmatrix}^{\dagger} \tag{A.28}$$

In Reference 29 a more robust and computationally efficient way to compute the intersection of $\mathbf{H}_p$ and $\mathbf{H}_f$ and construct a LS problem equivalent to (A.26) is provided. The algorithm in Reference 29 can be summarized as follows.

DI algorithm

1. construct the matrices $\mathbf{H}_p$ and $\mathbf{H}_f$ in (A.19b) and (A.19c) choosing i greater than q times the assumed order of the system to be identified, and concatenate them into a single matrix $\mathbf{H}$ as defined in equation (A.22)

2. take the SVD of $\mathbf{H}$

$$\mathbf{H} = \mathbf{U}\mathbf{S}\mathbf{V}^{\mathrm{T}} = \begin{bmatrix} \mathbf{U}_{11} & \mathbf{U}_{12} \\ \mathbf{U}_{21} & \mathbf{U}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{S}_{11} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} V^{\mathrm{T}}$$

3. calculate $\mathbf{U}_q$ from the SVD of $\mathbf{U}_{12}^{\mathrm{T}}\mathbf{U}_{11}\mathbf{S}_{11}$

$$\mathbf{U}_{12}^{\mathrm{T}}\mathbf{U}_{11}\mathbf{S}_{11} = \begin{bmatrix} \mathbf{U}_q & \mathbf{U}_r \end{bmatrix} \begin{bmatrix} \mathbf{S}_q & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{V}_q^{\mathrm{T}} \\ \mathbf{V}_r^{\mathrm{T}} \end{bmatrix}$$

4. compute the LS solution to the following set of equations to obtain **A**, **B**, **C**, **D**

$$\begin{bmatrix} \mathbf{U}_q^{\mathrm{T}}\mathbf{U}_{12}^{\mathrm{T}}\mathbf{U}(m+q+1:(m+q)(i+1),1:2mi+n)\mathbf{S}_{11} \\ \mathbf{U}(mi+qi+m+1:(m+q)(i+1),1:2mi+n)\mathbf{S}_{11} \end{bmatrix} = \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix} \begin{bmatrix} \mathbf{U}_q^{\mathrm{T}}\mathbf{U}_{12}^{\mathrm{T}}\mathbf{U}(1:mi+qi,1:2mi+n)\mathbf{S}_{11} \\ \mathbf{U}(mi+qi+1:mi+qi+m),1:2mi+n)\mathbf{S}_{11} \end{bmatrix} \tag{A.29}$$

where the standard Matlab® notation has been used to indicate specific submatrices of **U**. For example, $\mathbf{U}(a:b,c:d)$ indicates the submatrix of U at the intersection of rows **A** to **B** and columns **C** to **D**.

A.4 Deterministic projection (DP) algorithm

The Deterministic Projection (DP) algorithm to identify the system in equation (1.1) is described below. For a more detailed presentation, the reader is referred to References 2, 30. The equation at the core of the method is the following

$$\mathbf{Y}_h = \mathbf{\Gamma X} + \mathbf{H}_t\mathbf{U}_h \tag{A.30}$$

where $\mathbf{Y}_h$ and $\mathbf{U}_h$ are matrices with system output and input data, respectively, arranged according to a block Hankel structure, $\mathbf{H}_t$ is a block Toeplitz lower-triangular matrix that contains the first i Markov parameters of the system, $\mathbf{\Gamma}$ is the extended observability matrix

and $\mathbf{X}$ is the state time history

$$\mathbf{Y}_h = \begin{bmatrix} \boldsymbol{y}(0) & \boldsymbol{y}(1) & \dots & \boldsymbol{y}(l-i) \\ \boldsymbol{y}(1) & \boldsymbol{y}(2) & \dots & \boldsymbol{y}(l-i+1) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{y}(i-1) & \boldsymbol{y}(i) & \dots & \boldsymbol{y}(l-1) \end{bmatrix} \tag{A.31a}$$

$$\mathbf{U}_h = \begin{bmatrix} \boldsymbol{u}(0) & \boldsymbol{u}(1) & \dots & \boldsymbol{u}(l-i) \\ \boldsymbol{u}(1) & \boldsymbol{u}(2) & \dots & \boldsymbol{u}(l-i+1) \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{u}(i-1) & \boldsymbol{u}(i) & \dots & \boldsymbol{u}(l-1) \end{bmatrix} \tag{A.31b}$$

$$\mathbf{H}_t = \begin{bmatrix} \mathbf{D} & \mathbf{0} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{CB} & \mathbf{D} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{CAB} & \mathbf{CB} & \mathbf{D} & \dots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{CA}^{i-2}\mathbf{B} & \mathbf{CA}^{i-3}\mathbf{B} & \mathbf{CA}^{i-4}\mathbf{B} & \dots & \mathbf{D} \end{bmatrix} \tag{A.31c}$$

$$\mathbf{X} = \begin{bmatrix} \boldsymbol{x}(0) & \boldsymbol{x}(1) & \dots & \boldsymbol{x}(l-i) \end{bmatrix} \tag{A.31d}$$

$$\mathbf{\Gamma} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \mathbf{CA}^2 \\ \vdots \\ \mathbf{CA}^{i-1} \end{bmatrix} \tag{A.31e}$$

The key observation in the DP method is of geometrical nature. Under general conditions (Reference 30), the dimension of the projection of the row space of $\mathbf{Y}_h$ on the orthogonal complement of the row space of $\mathbf{U}_h$ provides the order of the system and its column space yields

an estimate of $\boldsymbol{\Gamma}$. Indeed the estimation of the observability matrix is at the core of the DP method, similar to how the reconstruction of the state history is the keystone of the DI method. Once $\boldsymbol{\Gamma}$ is estimated, $\mathbf{C}$ can be read as the first q rows of $\boldsymbol{\Gamma}$ whereas $\mathbf{A}$ can be computed thanks to the following relationship with $\boldsymbol{\Gamma}$

$$\underline{\boldsymbol{\Gamma}}\mathbf{A} = \bar{\boldsymbol{\Gamma}} \tag{A.32}$$

where $\underline{\boldsymbol{\Gamma}}$ and $\bar{\boldsymbol{\Gamma}}$ are obtained from $\boldsymbol{\Gamma}$ by deleting the last and first q rows, respectively.

Let $\boldsymbol{\Pi}_{\mathbf{U}_h^\perp}$ be the matrix that projects the row space of a matrix onto the row space of the orthogonal complement to the row space of $\mathbf{U}_h$. The projection of $\mathbf{Y}_h$ onto $\mathbf{U}_h^\perp$ is denoted by $\mathbf{Y}_h/\mathbf{U}_h^\perp$ and can be expressed as $\mathbf{Y}_h/\mathbf{U}_h^\perp = \mathbf{Y}_h\boldsymbol{\Pi}_{\mathbf{U}_h^\perp}$. Post-multiplying both sides of (A.30) by $\boldsymbol{\Pi}_{\mathbf{U}_h^\perp}$, we obtain

$$\mathbf{Y}_h/\mathbf{U}_h^\perp = \boldsymbol{\Gamma}\mathbf{X}\boldsymbol{\Pi}_{\mathbf{U}_h^\perp} \tag{A.33}$$

because $\mathbf{U}_h\boldsymbol{\Pi}_{\mathbf{U}_h^\perp}$ is null by definition. Take the SVD of $\mathbf{Y}_h/\mathbf{U}_h^\perp$

$$\mathbf{Y}_h/\mathbf{U}_h^\perp = \begin{bmatrix}\mathbf{U}_1 & \mathbf{U}_2\end{bmatrix}\begin{bmatrix}\mathbf{S}_1 & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{V}_1^{\mathrm{T}}\\ \mathbf{V}_2^{\mathrm{T}}\end{bmatrix} \tag{A.34}$$

where $\mathbf{S}_1 \in \mathbb{R}^{n\times n}$ is a diagonal matrix containing the non-zero singular values, $\mathbf{U}_1 \in \mathbb{R}^{qi\times n}$ and $\mathbf{U}_2 \in \mathbb{R}^{qi\times(qi-n)}$ contain the left singular vectors, and $\mathbf{V}_1 \in \mathbb{R}^{(l-i+1)\times n}$ and $\mathbf{V}_2 \in \mathbb{R}^{(l-i+1)\times(l-i+1-n)}$ contain the right singular vectors. $\mathbf{U}_1$ is then an orthonormal basis for the column space of $\mathbf{Y}_h/\mathbf{U}_h^\perp$ and can be taken as an estimate of the extended observability matrix $\boldsymbol{\Gamma}$. The matrix $\mathbf{C}$ can be read as the first q rows of $\mathbf{U}_1$ and $\mathbf{A}$ can be estimated as

$$\mathbf{A} = \underline{\mathbf{U}}_1^\dagger\bar{\mathbf{U}}_1 \tag{A.35}$$

where $\underline{\mathbf{U}}_1$ and $\bar{\mathbf{U}}_1$ are defined similarly to $\underline{\boldsymbol{\Gamma}}$ and $\bar{\boldsymbol{\Gamma}}$.

$\mathbf{A}$ and $\mathbf{C}$ are now identified. To find $\mathbf{B}$ and $\mathbf{D}$, an extra step is necessary. By the orthonormality properties of the SVD, the columns of $\mathbf{U}_2$ are orthonormal to the columns of $\mathbf{U}_1$, i.e. to the columns of $\mathbf{\Gamma}$. Pre-multiplying (A.30) by $\mathbf{U}_2^{\mathrm{T}}$ and post-multiplying by $\mathbf{U}_h^{\dagger}$, we obtain

$$\mathbf{U}_2^{\mathrm{T}}\mathbf{Y}_h\mathbf{U}_h^{\dagger} = \mathbf{U}_2^{\mathrm{T}}\mathbf{H}_t \tag{A.36}$$

which can be rewritten as an overdetermined set of linear equations in the unknown $\mathbf{B}$ as follows. Denote by $\mathbf{M}_j \in \mathbb{R}^{(qi-n)\times m}$ and $\mathbf{L}_j \in \mathbb{R}^{(qi-n)\times q}$ the block columns of $\mathbf{U}_2^{\mathrm{T}}\mathbf{Y}_h\mathbf{U}_h^{\dagger}$ and $\mathbf{U}_2^{\mathrm{T}}$, respectively, i.e.

$$\mathbf{U}_2^{\mathrm{T}}\mathbf{Y}_h\mathbf{U}_h^{\dagger} = \begin{bmatrix}\mathbf{M}_1 & \mathbf{M}_2 & \dots & \mathbf{M}_i\end{bmatrix} \tag{A.37a}$$

$$\mathbf{U}_2^{\mathrm{T}} = \begin{bmatrix}\mathbf{L}_1 & \mathbf{L}_2 & \dots & \mathbf{L}_i\end{bmatrix} \tag{A.37b}$$

Equation (A.36) can then be written as

$$\begin{bmatrix}\mathbf{M}_1\\ \mathbf{M}_2\\ \vdots\\ \mathbf{M}_i\end{bmatrix} = \begin{bmatrix}\mathbf{L}_1 & \mathbf{L}_2 & \dots & \mathbf{L}_i\\ \mathbf{L}_2 & \mathbf{L}_3 & \dots & \mathbf{0}\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{L}_i & \mathbf{0} & \dots & \mathbf{0}\end{bmatrix}\begin{bmatrix}\mathbf{I}_q & \mathbf{0}\\ \mathbf{0} & \mathbf{U}_1\end{bmatrix}\begin{bmatrix}\mathbf{D}\\ \mathbf{B}\end{bmatrix} \tag{A.38}$$

where $\mathbf{I}_q \in \mathbb{R}^{q\times q}$ is the identity matrix, and solved by LS for $\mathbf{B}$ and $\mathbf{D}$

$$\begin{bmatrix}\mathbf{D}\\ \mathbf{B}\end{bmatrix} = \left(\begin{bmatrix}\mathbf{L}_1 & \mathbf{L}_2 & \dots & \mathbf{L}_i\\ \mathbf{L}_2 & \mathbf{L}_3 & \dots & \mathbf{0}\\ \vdots & \vdots & \ddots & \vdots\\ \mathbf{L}_i & \mathbf{0} & \dots & 0\end{bmatrix}\begin{bmatrix}\mathbf{I}_q & \mathbf{0}\\ \mathbf{0} & \mathbf{U}_1\end{bmatrix}\right)^{\dagger}\begin{bmatrix}\mathbf{M}_1\\ \mathbf{M}_2\\ \vdots\\ \mathbf{M}_i\end{bmatrix} \tag{A.39}$$

Note that, similar to the DI algorithm, in OKID the DP algorithm is used to identify the Kalman filter of equation (3.1) or equation (3.5), i.e a state-space model in which only part of the input to the state equation is also an input to the observation equation. The DP method described above would attempt to identify also a complete direct influence matrix $\mathbf{D}$. Existing codes for the classic DP algorithm, based on (A.38), can nevertheless be used without modification and yield $\mathbf{D} \approx \mathbf{0}$, as verified by numerical examples. The DP algorithm is summarized below step by step for ease of implementation.

DP algorithm

1. construct the matrices $\mathbf{Y}_h$ and $\mathbf{U}_h$ as in (A.31)
2. project[3] $\mathbf{Y}_h$ on $\mathbf{U}_h^\perp$ to obtain $\mathbf{Y}_h/\mathbf{U}_h^\perp$
3. calculate $\mathbf{U}_1$ and $\mathbf{U}_2$ from the SVD of $\mathbf{Y}_h/\mathbf{U}_h^\perp$ as in (A.34)
4. read $\mathbf{C}$ as the first q rows of $\mathbf{U}_1$
5. compute $\mathbf{A}$ via equation (A.35)
6. compute $\mathbf{B}$ and $\mathbf{D}$ via equation (A.39)

[3]The projection of $\mathbf{Y}_h$ on $\mathbf{U}_h^\perp$ can be computed in closed form via the projection operator $\Pi_{\mathbf{U}_h^\perp} = I - \mathbf{U}_h^{\mathrm{T}} \left(\mathbf{U}_h \mathbf{U}_h^{\mathrm{T}}\right)^\dagger \mathbf{U}_h$ or via numerically more robust techniques such as QR decomposition. The code for DP available at http://homes.esat.kuleuven.be/~smc/sysid/software/ relies on the latter, as illustrated in chapter 6 of Reference 2. In the examples illustrated in this work, the same technique based on QR decomposition is used.

Appendix B

Bilinear observers with time-varying gains

We derive below the relations that one should use to find the time-varying gains $\mathbf{M}_1(k)$ and $\mathbf{M}_2(k)$ that at each time step minimize the current expected value of the norm squared of the state estimation error. In the case of time-varying gains, the matrices defined in equation (8.37) become time-varying

$$\bar{\mathbf{A}}(k) = \mathbf{A} - \mathbf{M}_1(k)\mathbf{C} \tag{B.1a}$$

$$\bar{\mathbf{N}}(k) = \mathbf{N} - \mathbf{M}_2(k)\mathbf{C} \tag{B.1b}$$

$$\bar{\mathbf{F}}(k) = \mathbf{A} - \mathbf{F} - \mathbf{M}_1(k)\mathbf{C} \tag{B.1c}$$

$$\bar{\mathbf{G}}(k) = \mathbf{N} - \mathbf{G} - \mathbf{M}_2(k)\mathbf{C} \tag{B.1d}$$

$$\bar{\mathbf{H}}(k) = \mathbf{B} - \mathbf{H} - \mathbf{M}_1(k)\mathbf{D} \tag{B.1e}$$

$$\bar{\mathbf{L}}(k) = \mathbf{L} + \mathbf{M}_2(k)\mathbf{D} \tag{B.1f}$$

and the conditions they have to satisfy for the observer to be unbiased are similar to (8.36)

$$\mathbf{F}(k) = \mathbf{A} - \mathbf{M}_1(k)\mathbf{C} \tag{B.2a}$$

$$\mathbf{G}(k) = \mathbf{N} - \mathbf{M}_2(k)\mathbf{C} \tag{B.2b}$$

$$\mathbf{H}(k) = \mathbf{B} - \mathbf{M}_1(k)\mathbf{D} \tag{B.2c}$$

$$\mathbf{L}(k) = -\mathbf{M}_2(k)\mathbf{D} \tag{B.2d}$$

Equation (8.42) then becomes

$$\begin{aligned}\mathbf{\Pi}(k+1) = {} & \bar{\mathbf{A}}(k)\mathbf{\Pi}(k)\bar{\mathbf{A}}^{\mathrm{T}}(k) + \mathbf{M}_1(k)\mathbf{R}\mathbf{M}_1^{\mathrm{T}}(k) + \mathbf{Q} \\ & + \left(\bar{\mathbf{A}}(k)\mathbf{\Pi}(k)\bar{\mathbf{N}}^{\mathrm{T}}(k) + \bar{\mathbf{N}}(k)\mathbf{\Pi}(k)\bar{\mathbf{A}}^{\mathrm{T}}(k) + \mathbf{M}_1(k)\mathbf{R}\mathbf{M}_2^{\mathrm{T}}(k) + \mathbf{M}_2(k)\mathbf{R}\mathbf{M}_1^{\mathrm{T}}(k)\right) u(k) \\ & + \left(\bar{\mathbf{N}}(k)\mathbf{\Pi}(k)\bar{\mathbf{N}}^{\mathrm{T}}(k) + \mathbf{M}_2(k)\mathbf{R}\mathbf{M}_2^{\mathrm{T}}(k)\right) u^2(k)\end{aligned} \tag{B.3}$$

The first-order conditions to minimize $\mathbf{\Pi}(k+1)$ are

$$\frac{\partial \operatorname{trace} \mathbf{\Pi}(k+1)}{\partial \mathbf{M}_1(k)} = -2\left(\bar{\mathbf{A}}(k)\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}} - \mathbf{M}_1(k)\mathbf{R}\right) - 2\left(\bar{\mathbf{N}}(k)\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}} - \mathbf{M}_2(k)\mathbf{R}\right) u(k) = \mathbf{0} \tag{B.4a}$$

$$\frac{\partial \operatorname{trace} \mathbf{\Pi}(k+1)}{\partial \mathbf{M}_2(k)} = -2\left(\bar{\mathbf{N}}(k)\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}} - \mathbf{M}_2(k)\mathbf{R}\right) u^2(k) - 2\left(\bar{\mathbf{A}}(k)\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}} - \mathbf{M}_1(k)\mathbf{R}\right) u(k) = \mathbf{0} \tag{B.4b}$$

and lead to optimality conditions similar to (8.43)

$$\bar{\mathbf{A}}(k)\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}} - \mathbf{M}_1(k)\mathbf{R} = \mathbf{0} \tag{B.5a}$$

$$\bar{\mathbf{N}}(k)\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}} - \mathbf{M}_2(k)\mathbf{R} = \mathbf{0} \tag{B.5b}$$

By solving equation (B.5) for the time-varying gains

$$\mathbf{M}_1(k) = \mathbf{A}\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}}\left(\mathbf{C}\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}} + \mathbf{R}\right)^{-1} \tag{B.6a}$$

$$\mathbf{M}_2(k) = \mathbf{N}\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}}\left(\mathbf{C}\mathbf{\Pi}(k)\mathbf{C}^{\mathrm{T}} + \mathbf{R}\right)^{-1} \tag{B.6b}$$

at each time step k, from $\mathbf{\Pi}(k)$ we can find the gains $\mathbf{M}_1(k)$ and $\mathbf{M}_2(k)$ to be used in the observer to get the estimate $\hat{\boldsymbol{x}}(k+1)$ of the next system state and to be plugged in equation (B.3) to provide $\mathbf{\Pi}(k+1)$ and repeat the process at the next time step $k+1$. The scheme is the same as for the linear Kalman filter.

Note that no requirement of whiteness or stationarity of the input is necessary to derive the above optimality conditions. The bilinear observer with time-varying gains is then optimal for arbitrary input sequences and at every time step. Since it minimizes at every time step the expected value of the norm squared of the state estimation error, the observer with gains given by equation (B.6) is the fastest converging bilinear observer in the presence of noise.

On the same numerical example in sections 8.6.2 and 8.6.3, the time-varying observers have gains that, after the initial transients, oscillate around the gains of the optimal time-invariant observer. The corresponding improvement of the state estimation error offered by the time-varying observer is not significant and therefore not shown.