

M.Sc. THESIS

Subspace-Based Identification Techniques for a ‘Smart’ Wind Turbine Rotor Blade:

A Study Towards Adaptive Data-Driven Control

Robert B. C. de Korte

January 8, 2009



Subspace-Based Identification Techniques
for a **‘Smart’ Wind turbine Rotor Blade**
A Study Towards Adaptive Data-driven Control

By
Robert Bastiaan Clark de Korte

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Systems and Control
at Delft University of Technology

Delft Center for Systems and Control
Faculty of Mechanical, Maritime, and Materials Engineering
Delft University of Technology

Delft
January 8, 2009



Delft University of Technology

Copyright © Delft Center for Systems and Control
All rights reserved.

DELFT UNIVERSITY OF TECHNOLOGY
DELFT CENTER FOR SYSTEMS AND CONTROL

The undersigned hereby certify that they have read and recommend to the Faculty of Department of Mechanical Engineering and Marine Technology for acceptance a thesis entitled:

Subspace-Based Identification Techniques
for a **‘Smart’ Wind turbine Rotor Blade**
A Study Towards Adaptive Data-driven Control

by **Robert Bastiaan Clark de Korte** in partial fulfillment of the requirements for the degree of **Master of Science**.

Dated: January 8, 2009

Supervisors:

prof.dr.ir. M. Verhaegen

dr.ir. J.W. van Wingerden

Readers:

prof.dr.ir. G.A.M. van Kuik

dr.ir. D.P. Molenaar

Abstract

More cost-effective wind turbines with higher rated power, are desired to emerge in the near future. The current trend in wind energy is for this reason the increase of the wind turbine dimensions. Increasing their size is however strongly limited by the rotor blades structural integrity towards occurring aeroelastic loads, especially with respect to blade fatigue. Advanced (re)active load reduction methods and advanced control techniques for wind turbines are therefore considered important research areas. The thesis provides a study towards subspace-based adaptive data-driven control for a ‘smart rotor blade. In an on-line setting closed-loop subspace identification can be of most importance to obtain adequate data-driven controllers for wind turbine installations, that are self-learning/adaptable to changing site/wind conditions. Moreover, the procedure of closed-loop subspace identification, followed by controller design in an off-line (automated) controller synthesis is also most desirable to find an adequate controller for a certain site location.

The contents of the thesis are two-fold. It provides a simulation model of a 2-dimensional ‘smart rotor blade. This model effectively describes the unsteady aeroelastic behavior of a wind turbine rotor blade section for changing wind conditions, equipped with a trailing edge flap, and suitable to be interfaced with a self-learning adaptive controller. The thesis contributes furthermore by proposing several novel closed-loop Subspace Model Identification (SMI) techniques, which are presented in a unifying framework that also includes some recent state-of-the-art closed-loop SMI techniques from literature. Included is a comparative simulation study of these closed-loop SMI procedures based on several models, like the constructed ‘smart rotor blade model. In addition to obtaining insight in identification of such a controlled smart rotor blade for wind turbine purposes, the identified open-loop system model can also be used to predict the systems classical flutter instability point, especially in an on-line scheme.

Acknowledgments

This report concludes my M.Sc. thesis at Delft Center for Systems and Control (DCSC). Its main directive is a study on closed-loop subspace identification techniques in prospect of adaptive data-driven control for wind turbine load alleviation. The report only gives a view, as well as advances, in two areas: aeroelastic modeling and subspace identification techniques, but discusses them with a perspective on adaptive data-driven control. I sincerely hope that the report succeeds in presenting an easy to grasp description about subspace model identification, as well as aeroelastic modeling, readable for people with different backgrounds, and equally as important, I hope it will make an interesting read.

I like to express my gratitude to my supervisors, Michel Verhaegen and Jan-Willem van Wingerden, for giving me the opportunity to start this project that very nicely combines system identification and control with aerodynamics. Also for making it possible to broaden the project towards closed-loop subspace model identification for which I started to form a special interest. Probably because of the enthusiasm you both provide towards this field. Special thanks goes to Jan-Willem for his continues efforts, the MATLAB debugging lessons, and being there whenever I had troubles or doubts. Additionally, much gratitude goes to Thanasis Barlas for the discussions, advice, and proofreading with regard to the aeroelastic modeling, and also for providing the UpWind 5MW verification data. Also many thanks to Ivo Houtzager for some of the proofreading, his enthusiasm for subspace model identification and showing interest enthusiasm in this thesis.

Furthermore, I would like to thank my parents and sister for supporting me in my work no matter what, and Jordy van Kalken for his support and keeping me interested in aerodynamics even though I changed studies. This provided me enough reasons to start this thesis project. Also many thanks to Joe George, Luisa Pires, Aysegul Danaci and Thomas Latour for making my weekends working at DCSC and staying in Delft worth while. I would also like to thank Aram Ghieratmand, Joren Alberts, Asbjørn Klomp, Jasper Witte, Yiquan Chen, Benedicta Purba, Joost de Bruin, and Ben van der Meer for their support and kindness during the thesis.

Delft, January 8, 2009
Robert de Korte

Table of Contents

Acronyms	xi
Nomenclature	xii
1 Introduction	1
1.1 Background and motivation	1
1.2 Problem formulation	5
1.3 Outline of the M.Sc thesis report	6
2 The aeroelastic plant	9
2.1 An introduction to aeroelasticity	9
2.2 Mechanical sub-model	10
2.3 Aerodynamic sub-model	12
2.3.1 Remarks on the wind turbine flow characteristics	13
2.3.2 Linearized incompressible potential flow equations	14
2.3.3 Two-dimensional thin airfoil theory	16
2.3.4 Solution to the harmonically oscillating thin-airfoil problem	18
2.4 The aeroelastic coupling	22
2.5 Verification and UpWind 5MW turbine implementation	25
2.5.1 Verification using aerospace models from literature	26
2.5.2 Verification with gust responses from BLADED software package	26
2.5.3 Frequency responses UpWind 5MW aeroelastic plant	28
2.6 Summary and final remarks	31

3	Subspace Identification	33
3.1	The closed-loop identification problem	33
3.1.1	Formulations and the Markov state	35
3.1.2	Subspace model identification	37
3.2	A combined framework: presenting the methods	40
3.2.1	The traditional open-loop methods described	41
3.2.2	The closed-loop methods presented	41
3.3	Unifying Closed-loop identification description	43
3.3.1	HOARX estimate: single-stage procedures	43
3.3.2	Pre-estimate based SMI: double-stage procedures	53
3.3.3	Multi-stage row-wise least squares estimates	63
3.4	Summary and concluding remarks	66
4	Closed-loop SMI on UpWind 5MW	71
4.1	The controlled aeroelastic plant	71
	Defining the wind gust disturbance signal	75
4.2	closed-loop UpWind 5MW simulations	76
4.2.1	UpWind 5MW identification at 75% annuli	76
4.2.2	Pole estimates for marginally stable model	82
4.2.3	Pole estimates for unstable model	82
4.3	Summary and concluding remarks	83
5	Conclusion and contributions	85
	Closed-loop Subspace Model Identification	85
	Bibliography	89
A	Theodorsen's geometrical functions	95
B	Derivation predictor terms	97

Acronyms

ARX	Auto-Regressive with eXogenous inputs model
ARXISS	ARX and Innovation-based SubSpace method
BEM	Blade-Element-Momentum theory
CCA	Canonical Correlation Analysis (see CVA)
CVA	Canonical Variate Analysis
CVA-MOESP	CVA Multivariable Output-Error State sSpace method
DARE	Discrete Algebraic Riccati Equation
DCSC	Delft Center for Systems and Control
DOF	Degrees Of Freedom
DUT	Delft University of Technology
EA	Elastic Axis
GSVD	Generalized Singular Value Decomposition
HOARX	High-Order Auto-Regressive with eXogenous inputs model
IEM	Innovation Estimate Method (i.e. PARSIM-E method)
IE-MOESP	Innovation Estimate Multivariable Output-Error State sSpace method
ISSCVA	Innovation-based SubSpace Canonical Variate Analysis method
LPV	Linear Parameter Varying
LQG	Linear Quadratic Gaussian
MW	MegaWatt
MIMO	Multiple-Input Multiple-Output
N4SID	Numerical algorithms for Subspace State Space System IDentification
PARSIM	PARsimonious Subspace Identification Method
PEM	Prediction Error Methods
PBSID	Predictor Based Subspace IDentification method
PO-MOESP	Past-Output Multivariable Output-Error State sSpace method
SIM	Subspace Identification Methods
SMI	Subspace Model Identification
SPC	Subspace Predictive Control
SVD	Singular Value Decomposition
WFA	Whitening Filter Algorithm (i.e. PBSID)

Nomenclature

Alphabetical

A	First of the quadruple of state space matrices.
$\bar{A} = (A - KC)$	Observer A-matrix.
B	Second of the quadruple of state space matrices.
$\bar{B} = (B - KD)$	Observer B-matrix.
$b = c/2$	Half chord length in [m].
C	Third of the quadruple of state space matrices.
$C(q)$	Transfer function of the controller.
c	Chord length of the airfoil.
D	Fourth of the quadruple of state space matrices.
E_i	Block Hankel matrix containing ‘future’ innovation noise terms, defined as $E_i = E_{p,i,N}$ (<i>e.g.</i> $i = f$) in (3.3).
$e_k \in \mathbb{R}^{n_y}$	Innovation: $e_k = y_k - Cx_k - Du_k$.
h	Plunge motion in [m], <i>i.e.</i> one of the mechanical states.
I_1, I_2	Sectional mass moment of inertia, with the subscripts ₁ for the airfoil and ₂ for the flap.
K	Kalman filter gain.
L	Aerodynamic lift.
M	Aerodynamic moment.
m_1, m_2	Sectional mass, with the subscripts ₁ for the airfoil and ₂ for the flap.
$k = \omega b/V_\infty$	Reduced (<i>i.e.</i> dimensionless) frequency in [rad].
n	System order.
n_u	Number of inputs.
n_y	Number of outputs.
q	Shift operator, <i>i.e.</i> $x_k \cdot q = x_{k+1}$ or $x_k \cdot q^{-1} = x_{k-1}$.
q	Mechanical state in the aeroelastic problem.
R	Radius of the wind turbine rotor disc.
Re	Reynolds number.
$r_k \in \mathbb{R}^{n_y}$	Reference signal.
T	Aerodynamic flap hinge moment.
U_i	Block Hankel matrix containing ‘future’ inputs, defined as $U_i = U_{p,i,N}$ (<i>e.g.</i> $i = f$) in (3.3), or the ‘past’ inputs for $i = p$: $U_p = U_{0,p,N}$.

U_n	Reduced order pre-multiplicative matrix in SVD problem.
u	Flow speed in x -direction [ms^{-1}].
$u_k \in \mathbb{R}^{n_u}$	System input signal.
V_∞	Free-stream velocity in [ms^{-1}].
$V_{flutter}$	Flutter speed in [ms^{-1}], <i>i.e.</i> free-stream velocity that yields a marginally stable system.
V_n	Reduced order post-multiplicative matrix in SVD problem.
\mathbf{V}	Velocity vector, $\mathbf{V} = \begin{bmatrix} u & v \end{bmatrix}^T$.
w	Flow speed in y -direction [ms^{-1}].
$v_k \in \mathbb{R}^{n_y}$	Measurement noise signal.
W_1, W_2	GSVD weightings.
w_G	Wind gust disturbance in [ms^{-1}].
$w_k \in \mathbb{R}^n$	Process noise signal.
$X_{p,N} \approx \bar{\mathcal{K}} Z_p$	Markov State Sequence.
$x_k \in \mathbb{R}^n$	State signal.
Y_i	Block Hankel matrix containing ‘future’ outputs, defined as $Y_i = Y_{p,i,N}$ (<i>e.g.</i> ($i = f$)) in (3.3), or the ‘past’ outputs for $i = p$: $Y_p = Y_{0,p,N}$.
$y_k \in \mathbb{R}^{n_y}$	System output signal.
Z_p	Past data block Hankel matrix containing the past inputs and outputs, <i>i.e.</i> $Z_p = \begin{bmatrix} U_p^T & Y_p^T \end{bmatrix}^T$.

Greek, Latin, mathematical symbols

δ	Flap deflection angle in [rad].
$\Gamma(t)$	Circulation in the aerodynamic problem.
Γ_f	Extended observability matrix in innovation form.
$\bar{\Gamma}_f$	Extended observability matrix in predictor form.
$\Gamma_{f \times n} \in \mathbb{R}^{f \times n}$	reduced-order observability matrix in innovation form.
$\Lambda_i^B = CA^{i-1}B$	Innovation form parameters used in the identification problem (entries of H_f Toeplitz matrix).
$\Lambda_i^K = CA^{i-1}K$	Innovation form parameters used in the identification problem (entries of G_f Toeplitz matrix).
ω	Frequency or angular velocity in [rad/s].
$\Phi_i^B = C\bar{A}^{i-1}\bar{B}$	Markov parameters (predictor form) related to the input (entries of \bar{G}_f Toeplitz matrix).
$\Phi_i^K = C\bar{A}^{i-1}K$	Markov parameters (predictor form) related to the output (entries of \bar{G}_f Toeplitz matrix).
ϕ	Velocity potential (scalar) function.
ρ	Air density.
Σ_n	Reduced-order square matrix with the singular values on its diagonal.
θ	Pitch angle in [rad], <i>i.e.</i> one of the mechanical states.
$\xi = \nabla \times \mathbf{V}$	Vorticity in [rad/s].

Chapter 1

Introduction

1.1 Background and motivation

With the worldwide demand on electricity, which is expected to double by the year 2030, and concerning the accompanying substantial increase in unsustainable CO₂ emissions, wind energy is regarded as a very favorable source of renewable energy for now and in the future [1]. Since the kinetic energy in wind is a free and unlimited sustainable source of energy that is widely available, however wind turbine economics associated with assembly¹ and maintenance per kilowatt-hour are still considered too expensive [2]. Wind turbines with higher rated-power are therefore desired to emerge in the near future. Moreover, a certain minimum wind speed is required to produce the rated power, so optimal energy production highly depends on the occurring wind speed, which of course makes some locations more suitable than others. Wind turbines are for this reason build more regularly at off-shore locations, since the average wind speed and thus power-production potential is higher. The available wind field at offshore locations is also more favorable (*i.e.* more steady, so less turbulence), and equally as important, wind turbines can grow in size and numbers without much obstruction compared to the onshore possibilities. It should be noted that the wind turbine's power production capabilities, and cost effectiveness keeps improving as technology propagates into the future. Some of the main trends in wind energy technology to make wind-energy conversion installations economically more competitive are presented next.

Development of wind turbines with higher rated power

To make wind-energy conversion systems more cost effective, the current trend is the development of wind turbines with higher rated power. Blade Element Momentum (BEM) theory states that the power produced by a wind turbine is proportional to its rotor area, so bigger wind turbines can produce more power [3]. A wind-energy conversion 'farm' needs for this reason less large wind turbines to produce the same amount of energy than with their smaller counterparts. Needing less wind turbines consequently results in less costs associated with building and maintaining each wind farm per Watt. Current 5 MegaWatt (MW) sized wind turbines have a rotor diameter reaching as much as 126m [4]. Increasing their size even more is however strongly limited by the rotor blades' weight and structural integrity towards

¹Like foundation and erecting operation costs.

occurring (aeroelastic) loads. Advanced (re)active load/fatigue reduction methods and advanced control techniques for wind turbines are therefore considered important research areas [5, 6, 7, 8, 9].

Aerodynamic load alleviation

Increasing the rotor blade length consequently results in operating conditions where large and swift aerodynamic load variations along the blade become more dominant due to unsteady wind speeds, like turbulence, tower shadow, and wind shears. The longer the blades become, the more prone they are to elastic deformations by occurrence of locally acting unsteady aerodynamic loads. Since elastic deformations consequently alter the orientation and shape of the blades with respect to the flow, thus also changing the already time-varying aerodynamic loads, the result is an aeroelastic load distribution that significantly varies along the blade. This, of course, effects the blade's torsional and bending moments with respect to blade fatigue, and henceforth has generated a particular interest in more advanced load control techniques to control the aerodynamic loads. Nowadays, MW sized wind turbines pitch their rotor blades for power output regulation above rated wind speeds, but also to actively reduce load increments. Compared to collective pitch control, the use of cyclic pitch control or the current state-of-the-art individual pitch control suit the purpose of load alleviation more appropriately [10, 11]. Individual pitch is however only able to control the average load on a blade, so the technique becomes less efficient as rotor blades get longer due to more dominant local aeroelastic load effects. The 'smart' rotor blade concept provides a solution by application of locally placed control devices (*i.e.* trailing edge flaps), with a faster response than blade pitch, to compensate for the swift local fluctuations in the aeroelastic loads. These local control devices, when interfaced with appropriate sensors, in combination with an advanced feedback control algorithm, represent a sort of built-in intelligence to effectively alleviate occurring loads (*i.e.* strains). Efficient load alleviation on wind turbines serves the purposes:

- Reduction of blade fatigue and the control of extreme load events to improve the wind turbine's life span and maintenance/operational costs. Load alleviation also gives the possibility of lighter blade constructions, paving the way to even larger wind turbines, and pushing the current trend of an increase in wind turbine dimensions into the future.
- Since increasing the rotor area is most beneficial for the wind turbine's cost-effectiveness, the radial velocities of future wind turbine rotor blades will eventually increase as well. This puts the velocities closer to the blade's flutter speed, *i.e.* the aeroelastic point of marginal stability. Therefore, load control research on future wind turbines is also directed to improving the aeroelastic stability of the wind turbine [12].

More efficient and adequate control techniques

The 56th topical expert meeting of the International Energy Agency (IEA) [13] presented that a major development in adequate control techniques for wind-energy conversion systems is required. These developments, and corresponding research, are supported by both universities and companies. The DUWIND² group at the Delft University of Technology

²<http://www.duwind.tudelft.nl>

(DUT), performs extensive research to effectively control the aerodynamic loads using the ‘smart’ rotor blade concept described earlier [8, 9]. Comparable research is performed at Risø in the ADAPWING project [14]. A feasibility study of the ‘smart’ rotor blade concept for wind turbines has been performed by Basualdo [15] and Buhl et al. [16]. They showed the potential of using a trailing edge flap, with feedback controller, to successfully alleviate the loads on a simulated 2-dimensional blade element (*i.e.* airfoil). Wind tunnel tests that provided a proof of concept for a non-rotating blade are performed by Bak et al. in [7] and Van Wingerden et al. in [8]. The latter, in contrast to [7], used a feedback controller to effectively alleviate the aerodynamic loads.

For the synthesis of adequate control laws, it is important to have a model that describes the (most important) dynamics of the system accurately. Controller design consequently starts with a step of model determination, either from using first principles modeling techniques or experimentally obtained models. A first principles based model of a ‘smart’ rotor blade depends on a significant number of parameters that determines the dynamic behavior of the system. These parameters are normally difficult to calculate or estimate without inducing uncertainties on the model. Therefore, for effective advanced control of the ‘smart rotor’ blade, an ‘identification for controller synthesis’ approach is highly applicable. This methodology also refers to a data-driven (automated) control law synthesis. This approach is illustrated by [8], where Van Wingerden et al. use measurements of a practical ‘smart’ blade wind tunnel set-up to compute a data-based model using subspace identification techniques, and compare it a model obtained by using first-principles. There were some significant differences reported between both models, like unmodeled dynamics in the lower frequencies for the first principles model. Moreover, the experimentally obtained model had only 10 states, whereas the analytical model had a system order of 28. For advanced control techniques (e.g. LQG and \mathcal{H}_∞), this large order will give rise to computational problems, since the controller will be of (at least) equivalent order. The comparison in [8] is applicable on a 2-dimensional model of the ‘smart’ rotor blade. When the whole rotating wind turbine is considered, this gives rise to even more modeling uncertainties, especially in real-life conditions instead of a controlled wind tunnel environment. The difficulty of modeling a full scale wind turbine is supported by the findings of the NREL blind comparison test, where the prediction accuracy of wind turbine simulation codes compared to measured behavior was investigated [17]. Henceforth, initial system identification on a wind turbine for the synthesis of an advanced, and effective controller is considered a requisite. Furthermore, due to operational constraints on wind turbines, only specialized closed-loop identification techniques that measure the wind turbine during controlled conditions are appropriate [18, 19].

The 56th IEA topical expert meeting allocated the following key areas of advances in control for wind-energy conversion systems³ [13]:

- Real-time load identification;
- A predictive nature;
- The controller must be fault-tolerant;
- The controller must be adaptive to site and operational conditions (self learning).

³This information is gathered from a presentation that showed the main conclusions of IEA’s 56th topical expert meeting.

These topics are perfectly applicable to an adaptive data-driven control approach, like the closed-loop Subspace Predictive Control (SPC) methods developed at the Delft Center for Systems and Control (DCSC) [20, 21, 22]. Figure 1.1 gives a basic illustration of the subspace-based data-driven control technique, which essentially performs automated subspace identification followed by an automated controller synthesis based on some mathematical expression of the plant. For more information on the two approaches illustrated in the Figure, we refer the interested reader to [23, 18] for SMI-LQG) and [20, 21, 18] for SPC. A subspace-based adaptive data-driven controller was therefore considered for the thesis. However, providing an aeroelastic simulation model of the ‘smart’ rotor blade, and giving a fresh look at the state-of-the art in closed-loop Subspace Model Identification (SMI), turned out to be an important first step towards a feasibility study of adaptive data-driven control for a ‘smart’ wind turbine rotor blade. The thesis contributes by presenting an ‘unifying description’ of the most recent closed-loop Subspace Model Identification techniques, which resulted in some novel methods. Closed-loop SMI is usually unnecessarily presented such that it is difficult to understand for people initiated into this subject. The ‘unifying description’ tries to establish a novel structured closed-loop SMI framework that combines existing methods with some alternate identification procedures in an attempt to make it accessible to a broader public.

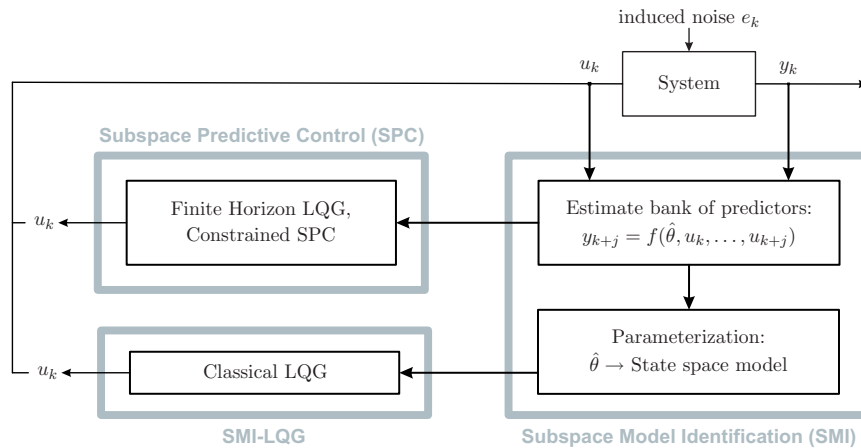


Figure 1.1: Illustration of two subspace-based adaptive data-driven control methods.

Additionally, the thesis provides a database that can be used for future ‘smart’ rotor blade feasibility studies, or further research into closed-loop subspace identification methods and the data-driven control methodology. This report reflects a start of a thorough investigation into the possibilities of subspace-based adaptive data-driven control methods for wind turbines. It ends therefore not only with conclusions about the thesis contents, but also gives a clear perspective on research topics that can form a sequel or extension to this thesis, like the feasibility study of adaptive data-driven control for a smart rotor blade that initially started the project. Although the previously mentioned topics are all as important with regard to a future feasibility study, the main contribution of this thesis to the academic world is regarded the novel investigation into subspace model identification methods, the new state-of-the-art identification techniques, and a thorough comparative simulation study between methods from literature, combined with the new SMI algorithms.

1.2 Problem formulation

The flow around a 2-dimensional rotor blade element (*e.g.* at 75% rotor anulli) explicitly depends on the blade's sectional free-stream velocity as a time-varying parameter [24]. This particular model structure of a nonlinear time-varying system is referred to as a Linear-Parameter-Varying (LPV) system [25] and the references therein. In a state space format this model structure applies to state space matrices that are a function of this time-varying parameter. It should be noted that this LPV structure can be used to model only quasi-steady changes in free-stream velocity. Interpretation is thus as follows: the LPV model varies the aerodynamic operating point, and can, as such, be regarded an interpolation of local linear models. This makes LPV control suitable, however since the dynamics are considered to be slowly varying, an adaptive data-driven control methodology seems also very appropriate for a 'smart' rotor blade. For wind turbines, the self-learning behavior of the adaptive data-driven controller appears a truly desired advantage. An LPV representation of the aeroelastic plant model enables the possibility for an extension towards future adaptive joint LPV-SPC controller. Furthermore, imposing this LPV behavior in a simulation model makes an ideal testbed to study adaptive data-driven control and subspace identification techniques, under static and varying dynamics. The main directives of the study are twofold:

- Build a mathematical model that is suitable to simulate the 'smart' rotor blade, equipped with a trailing edge flap, and interfaced with a self-learning adaptive controller;
- Perform a study on closed-loop subspace identification, and present an easy to grasp 'unifying' description that explains this special field of identification techniques. Use the acquired knowledge to perform a preparatory study for subspace-based adaptive data-driven control with respect to suitability of the methods. The suitability of the methods is considered with respect to the method's complexity (*e.g.* recursive implementation or computational effort), sensitivity to colored noise and input-noise correlation, and the biasedness and variance of the estimated models.

Comparable to a feasibility study by Basualdo [15], the thesis only focuses on a 2-dimensional rotor blade section (*i.e.* airfoil) of a wind turbine rotor blade to investigate the adaptive data-driven control methodology for active load alleviation on wind turbines.

M.Sc. thesis subjects and objectives

The M.Sc. thesis provides a focus on three topics, based on which the reports follow the following step-wise approach.

1. Providing a simulation model of the unsteady aeroelastic behavior of a 2-dimensional wind turbine rotor blade section, equipped with a trailing edge flap. The aeroelastic plant is presented in a novel LPV structured state space model, where the varying parameter represents the aerodynamic operating condition. This parameter can be used to make quasi-steady changes in the rotor disk's rotational speed. The novelty of this model in the field of aeroelasticity resides in the fact that this representation makes it especially suitable for filtering, control and identification techniques. On another

note, flutter analysis can also be performed very easily with this model and presents an attractive alternative to the classical p-k method (see [26]) used in aeroelasticity. The open-loop system model of a controlled ‘smart’ rotor blade set-up, for example using a data-driven control approach, can be identified using subspace model identification techniques, and can be used to determine the system’s classical flutter instability point. In an on-line scheme this methodology can be used to predict the flutter boundary. Recursive identification techniques are therefore very useful for this purpose.

2. The necessity of closed-loop subspace identification techniques for wind turbine applications is considered with respect to multiple purposes. First, a experimentally obtained model from data can be formed such that it suits a certain control methodology better, and it can find a relatively low-order model that accurately describes the dynamics at whatever site location or operational condition the data is obtained in. Secondly, closed-loop subspace identification methods are very suitable for on line recursive identification of a model, which enables for example an adaptive data-driven control approach, like SPC [27, 20]. Therefore, closed-loop subspace identification is of most importance to obtain adequate self-learning/adaptable controllers for wind turbine installations. The procedure of identification followed by (automated) controller design can be implemented statically to find an adequate controller for a certain site location, or dynamically (*i.e.* on-line) to make it adaptive to changing operating conditions. This thesis focuses on the most recent closed-loop SMI techniques, that distant themselves by the characteristic that they do not require explicit controller knowledge. The report presents a ‘unifying’ description of five closed-loop subspace model identification methods from literature, and tries to establish an easy to grasp general step-wise methodology for closed-loop SMI. As a consequence of this study, some alternate novel subspace identification procedures were developed in this thesis, and presented as a part of closed-loop SMI framework. A partial objective is to present a MATLAB toolbox for closed-loop subspace identification that consists of blocks that can be interconnected to form a particular closed-loop identification method. This can be very useful for a comparative research to efficiently perform closed-loop SMI. It can also be used as an interactive introductory tool into closed-loop subspace identification for students or beginners.
3. The study of subspace identification techniques combined with an accurate description of the aeroelastic plant, results in a simulation based comparison between the closed-loop Subspace Identification Methods (SIM) that are allocated under the ‘unifying’ description on the ‘smart’ rotor blade model. The model identification methods are thus implemented on the obtained aeroelastic simulation code. This simulation study can be especially useful to get better insight in the automated (partial) subspace identification step in adaptive data-driven controller, like the batch-wise data-driven controller presented in [23].

1.3 Outline of the M.Sc thesis report

Figure 1.2 gives a schematic representation of the report structure, which follows a similar step-wise approach as the M.Sc. thesis subjects and objectives previously described. First, in Chapter 2, the aeroelastic plant is described. It provides a basic understanding in the field of aeroelasticity, formed by fluid-structure interactions, and builds an LPV

structured simulation model of the UpWind 5MW reference wind turbine. Chapter 3 provides a ‘unifying’ description of the most recent closed-loop subspace model identification techniques, as well as the new subspace model identification procedures developed in this thesis. Throughout Chapter 3, small simulation examples from the literature will be given, so some comments/observations can be made for the methods. Thereafter, Chapter 4 shows some comparative closed-loop simulations of the methods presented under the closed-loop SMI framework, based on the aeroelastic plant model from Chapter 2. Finally, Chapter 5 presents a summary of the most important results obtained in this M.Sc. thesis project, as well as a conclusion on the findings of each subproblem. This chapter ends with a perspective on extensions and future work: a feasibility study of adaptive data-driven control for a ‘smart’ wind turbine rotor blade.

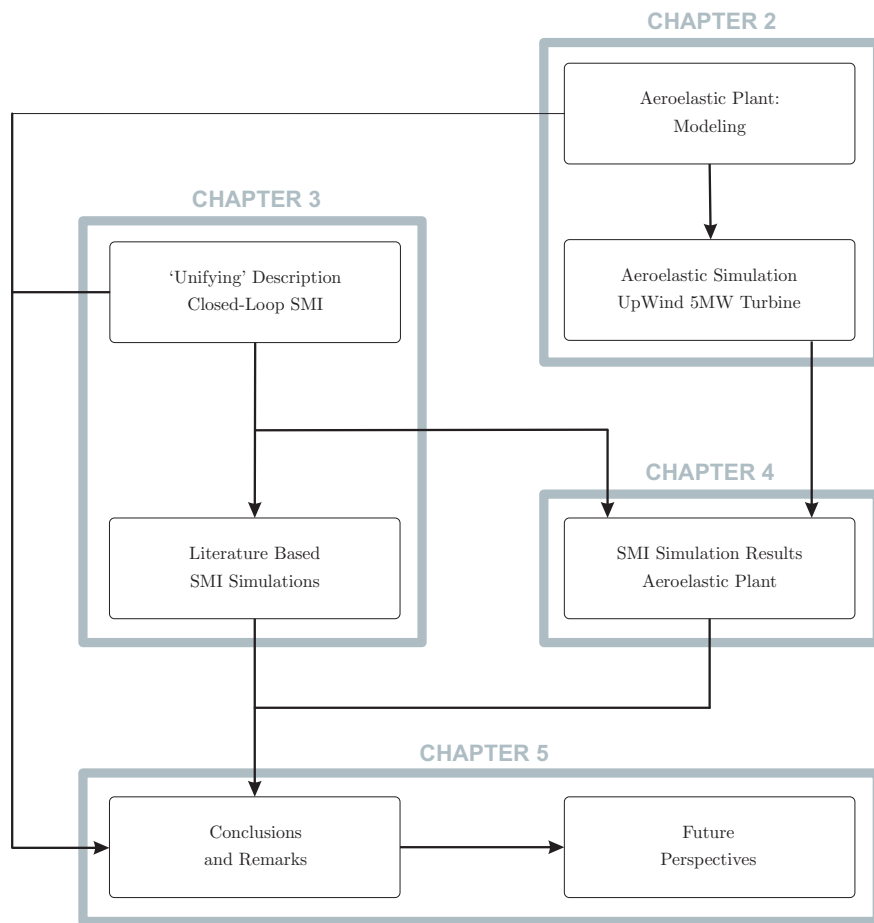


Figure 1.2: Schematic of the report structure with regard to the feasibility study.

Chapter 2

The aeroelastic plant

Aeroelasticity describes the fluid-structure interaction that models a ‘smart’ rotor blade. This chapter therefore presents a model that incorporates the aeroelastic coupling between an aerodynamic sub-model and a mechanical sub-model, and present it in a form suitable for control, identification, and aeroelastic analysis. The model is time-varying, and can be parameterized using a Linear Parameter-Varying (LPV) state space model. This structure is not exploited in this study, but can be of much interest for LPV control, or even data-driven LPV control.

The aeroelastic study only includes 2-dimensional modeling techniques, hence the aerodynamic and mechanical subproblems of a rotating wind turbine blade are reduced to a blade section (i.e. an airfoil), equipped with trailing edge flap, at a particular rotor annuli.

A basic introduction to aeroelasticity, describing the fluid-structure interaction, is given in Section 2.1. Both a mechanical and an aerodynamic sub-model are given thereafter in Section 2.2 and Section 2.3. These sub-models are coupled to form the aeroelastic plant in Section 2.4. Section 2.5 evaluates the aeroelastic plant with respect to literature and implements the 5MW UpWind reference wind turbine, which forms the simulation model for the remainder of the thesis. Finally, this chapter will end by giving a short summary and some final remarks.

2.1 An introduction to aeroelasticity

As presented in the introduction in Chapter 1, the current trend is an increase of wind turbine dimensions to make wind-power conversion systems more cost effective. 5MW wind turbines already have rotor blades with a length of as much as 63m, with larger rotor discs most desirable in the future [2, 4]. Enlarging the rotor blades, combined with the correspondent increase in fatigue loads, imposes the need for flexible blade structures. This consequently makes the rotor blades more vulnerable to irregular elastic deformations due to fluctuating external aerodynamic loads. A wind turbine rotor blade produces lift when located in a wind

field, which consequently makes the wind turbine rotate, and therefore produce power. The lift that is produced by a rotor blade is dependent on its shape and dimension, as well as its orientation in a flow. An aerodynamic sub-model is therefore needed to describe the occurring external loads on a blade. The aerodynamic lift, and corresponding moments, however result in structural bending, twist, and other elastic deformations of the blades, which consequently yields another aerodynamic load distribution along the blades. Henceforth, to describe the elastic behavior of a rotor blade, an additional mechanical sub-model is also needed. The time-varying interaction between the structure's elastic forces and deformations, and the external aerodynamic forces, is known as the field of aeroelasticity. Since aeroelasticity describes the interaction between the airflow and the structure, the mathematical expression of an aeroelastic plant is simply a model that incorporates, and couples, the mechanical sub-model with an aerodynamic sub-model. Therefore, in constructing the aeroelastic plant, a mechanical and aerodynamic sub-model are presented first in Section 2.2 and Section 2.3, respectively.

2.2 Mechanical sub-model

This section forms the mechanical sub-model in a state space model representation, based on first-principle equations of motion. There exists a wide range of approaches that can be used to obtain the mechanical model; Finite Element Methods (FEM), the Rayleigh-Ritz approach (with assumed modes), or the lumped parameter (*i.e.* multi body) approach [28, 8]. The structural properties of the 5MW UpWind reference wind turbine, that forms the basis of the feasibility study, are readily adaptable to Lumped parameters, hence this particular modeling technique is considered. The lumped parameter approach needs a minimal amount of equations of motion, which makes it very suitable for controller design. Moreover, the lumped parameter approach is used extensively for similar studies in the literature [29, 30, 31, 24, 32, 33].

Figure 2.1 shows the free-body-diagram of an airfoil with its corresponding dimensional and structural parameters specified. The mechanical model that can be formed according to Figure 2.1 requires 3 generalized coordinates, a plunge motion h , a pitch motion θ , and a flap deflection δ . This corresponds to a system with 3 Degrees-Of-Freedom (DOF). Note that, comparable to the literature, the figure assumes a small pitch angle θ , as well as a small flap deflection δ .

Waszak presented in [29] a derivation of the structural model using Lagrange's method. This section gives a small summary of this model derivation. Lagrange's equation is given as:

$$\frac{d}{dt} \left(\frac{\partial T_e}{\partial \dot{q}_i} \right) - \frac{\partial T_e}{\partial q_i} + \frac{\partial U}{\partial q_i} = Q_{q_i}, \quad (2.1)$$

with the kinetic energy T_e , the potential energy U and the generalized coordinates q_i . The generalized coordinates are illustrated in Figure 2.1, and yield the plunge motion h , the pitch angle θ , and the flap deflection angle δ . Using Figure 2.1, the kinetic energy is formulated as [29]:

$$T_e = \frac{1}{2} m_1 \left(\dot{h} + x_{cg} \dot{\theta} \right)^2 + \frac{1}{2} I_1 \dot{\theta}^2 + \frac{1}{2} m_2 \left(\dot{h} + x_{cg_f} \dot{\theta} + x_{\delta} \dot{\delta} \right)^2 + \frac{1}{2} I_2 \left(\dot{\theta} + \dot{\delta} \right)^2. \quad (2.2)$$

Assuming very large control flap stiffness, *i.e.* the structural deformations due to the hinge load are insignificant [29], the 3-DOF system can be simplified by a 2-DOF system. Thus assuming K_δ is very large, (2.4) can be approximated by [31]:

$$\begin{bmatrix} m & s_{h\theta} \\ s_{h\theta} & I_\theta \end{bmatrix} \begin{bmatrix} \ddot{h} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} D_h & 0 \\ 0 & D_\theta \end{bmatrix} \begin{bmatrix} \dot{h} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} K_h & 0 \\ 0 & K_\theta \end{bmatrix} \begin{bmatrix} h \\ \theta \end{bmatrix} = \begin{bmatrix} m \\ s_{h\theta} \end{bmatrix} g - \begin{bmatrix} L \\ M \end{bmatrix}. \quad (2.6)$$

The approximated mechanical system in (2.6), in fact, represents an airfoil without flap, hence the trailing edge related structural and dimensional properties, *i.e.* m_2 and I_2 need to be set to zero with regard to the structural parameters. To simulate operation of the trailing edge flap for the 2-DOF system, the system input of the aeroelastic plant is not T_{ctrl} , but a quasi-steady flap deflection δ that will be included in the aerodynamic sub-model (Sec. 2.3). In Section 2.4, the linear model in (2.4) or (2.6) will be interconnected with an aerodynamic sub-model through the aerodynamic load vector \mathbf{Q}_{aero} . Due to this coupling, the aerodynamic forces and moments modify the mass, damping, and stiffness properties of the mechanical system, which essentially defines the aeroelastic system [26, 29]. The aeroelastic behavior of a system, in our case a ‘smart’ rotor blade, is therefore dependent on the velocity of the flow. At a certain flow velocity, the interaction with the aerodynamic loads can make the coupled aeroelastic system unstable. This instability is called flutter, with the flow velocity that yields a marginal stability defined as the so-called flutter speed $V_{flutter}$.

Structural parameters of the UpWind 5MW wind turbine

Section 2.2 presented a mechanical model, either as a 3-DOF system or a simplified 2-DOF system. The aeroelastic simulation code has the option to select the appropriate Degrees Of Freedom. The dimensional and structural properties of the UpWind 5MW reference wind turbine are only available as a 2-DOF mechanical system. Since the structural design of a trailing edge flap on a wind turbine rotor blade is no part of the thesis assignment, a 2-DOF mechanical system is used. The system input of the 2-dimensional aeroelastic model of 5MW reference wind turbine is thus the flap deflection δ . This however leaves one flap design variable open to be chosen, namely the distance from the mid-chord to the flap hinge. This variable was chosen as $x_f = 0.2$, *i.e.* the flap spans 10% of the chord length. Table 2.1 contains the dimensional and structural parameters, illustrated in Figure 2.1, that were obtained for the UpWind 5MW reference wind turbine. A sheet with the parameters was readily available for the UpWind project, and only some additional coordinate and unit transformations needed to be performed. The obtained parameters in Table 2.1 are given for a 75% rotor annuli (*i.e.* 47.7 m radius), since a future feasibility study is most likely to focus on the aeroelastic behavior at this particular local position. The spring constants K_h and K_θ are computed using the flap-wise and torsional natural frequencies of the 3-dimensional wind turbine rotor blade, respectively [34]:

$$K_h = m \cdot (2\pi \cdot \omega_{n,1^{st}flap})^2 \quad K_\theta = I \cdot (2\pi \cdot \omega_{n,pitch})^2. \quad (2.7)$$

2.3 Aerodynamic sub-model

Aerodynamic theory is described by the three fundamental laws of physics applied on a fluid (*e.g.* an airflow), namely the conservation of mass, momentum and energy. There exist

Table 2.1: UpWind 5MW reference wind turbine parameters for 75% rotor annuli ($R=47.7\text{m}$).

Parameter	Description	Value (+ unit)
$c = 2b$	chord length	2.8152 m
x_a	distance mid-chord to EA [-]	-0.1820
x_f	distance mid-chord to flap hinge [-]	0.2
x_{cg}	distance EA to center of gravity (c.o.g.)	0.1263 m
$\theta_{struc.twist}$	structural twist	2.5740°
m_1	sectional mass	146.3190 kg
I_1	sectional mass moment of inertia (in c.o.g.)	76.0132 kgm
$\omega_{n,1^{st}flap}$	1 st flap-wise natural frequency	0.6896 Hz
$\omega_{n,pitch}$	torsional natural frequency	5.3311 Hz
K_h	spring constant plunge motion	2747 N/m
K_θ	spring constant pitch motion	85287 Nm/rad
D_h	damping term plunge motion	0.04 N/ms ⁻¹
D_θ	damping term pitch motion	0.02 Nm/rads ⁻¹

however multiple possible aerodynamic governing equations, with their appended accuracy depending on the assumptions and simplifications applicable to the flow characteristics. In this section the analytical solution of thin airfoil theory, governed by the aerodynamic equations for an incompressible, irrotational, inviscid, and linearized flow, is considered. Only the governing equations that are part of the aeroelastic plant model to be constructed are presented in this section. The interested reader is however referred to [35, 18] for more detailed information. The derivation of the analytical aerodynamic model of a ‘smart’ wind turbine rotor blade follows the step-wise approach:

- Description of the wind turbine flow characteristics;
- Linearized potential flow theory for an incompressible flow;
- Thin airfoil theory, based on the above aerodynamic theory, by Theodorsen and Sears.

We start Section 2.3 with an evaluation of the assumptions applicable to wind turbine aerodynamics and the ‘smart’ rotor blade.

2.3.1 Remarks on the wind turbine flow characteristics

In aerodynamics, the Mach number and the Reynolds number give a useful indication of the flow characteristics. The former is a grade for the flow compressibility, whereas the latter is related to viscous effects.

Incompressibility assumption

Even one of the largest practical wind-turbine installations at the moment, the REpower 5M with 126m rotor diameter, has a rated tip speed of approximately 80m/s [4], *i.e.* a Mach number well below 0.3. The flow around a wind turbine rotor blade can thus be considered incompressible [35, 36, 37]. This indication is well found for aerodynamics, however for

the aeroelastic problem, where the aerodynamics are coupled with a structural model, the reduced frequency of the flap motion also plays an important role. According to Leishman in [38], incompressible flow is no longer an accurate assumption when the local effective reduced frequencies based on flap motion exceed unity. The reduced frequency k is defined as [26]:

$$k = \frac{\omega b}{V_\infty}, \quad (2.8)$$

with b the half chord length (*i.e.* $c = 2b$) in meters, ω the frequency of the unsteady disturbances in rad/s and V_∞ the flow velocity in m/s. In [8], Van Wingerden et al. experimentally estimate the eigenfrequencies of a ‘smart’ rotor blade wind tunnel set-up with subspace model identification techniques (Chapter 3). The largest eigenfrequency, for a flow velocity of 45 ms^{-1} , yields a reduced frequency k below unity, hence the incompressibility condition is maintained for this practical example. Since wind turbines can be assumed to operate in the incompressible flow regions, the air density ρ can be considered constant, and the energy conservation law decouples from the set of equations [36]. This simplifies the governing set of equations considerably.

High Reynold numbers

For large wind turbines (*e.g.* $>2\text{MW}$), the Reynolds number has values in the order $Re > 6 \cdot 10^6$, meaning that the viscous forces can be neglected in the main flow, and the main velocity field can be considered irrotational [36]. However, the assumption does not hold for:

1. flow regions directly adjacent to solid surfaces (*e.g.* the airfoil), where a thin boundary of viscous flow exists, and
2. the viscous/vortex wakes shed by the rotor blades.

Henceforth, with the simplification of an irrotational flow, the above needs to be modeled explicitly. The inviscid and irrotational flow assumption also simplifies the governing equations considerably, as will be presented in Section 2.3.2.

2.3.2 Linearized incompressible potential flow equations

This section presents the aerodynamic governing equations that form the basis of the analytically solvable thin-airfoil theory, applicable to incompressible irrotational flows. Figure 2.2 presents a schematic overview of aerodynamic governing equations, based on [35, 26, 28], and shows the simplifications that correspond to each set of equations. The Navier Stokes equations are the most accurate, and revolve around the conservation of momentum, mass, and energy for a viscous, unsteady, compressible fluid. Solving the Navier-Stokes equations for engineering problems of interest however requires the use of sophisticated numerical techniques that are computationally time consuming and costly. This section presents the linearized potential flow equations for incompressible flows.

As can be seen from Figure 2.2, the full potential flow equations can be derived from the Euler equations by assuming the flow to be irrotational and isentropic¹. This set of equations

¹From [35] - an isentropic process is a condition that is adiabatic (*i.e.* no heat is added to or taken away from the system) and reversible (*i.e.* no dissipative phenomena, hence no viscosity effects, no thermal conductivity, and no mass diffusion). Note that an incompressible flow satisfies the isentropic flow conditions.

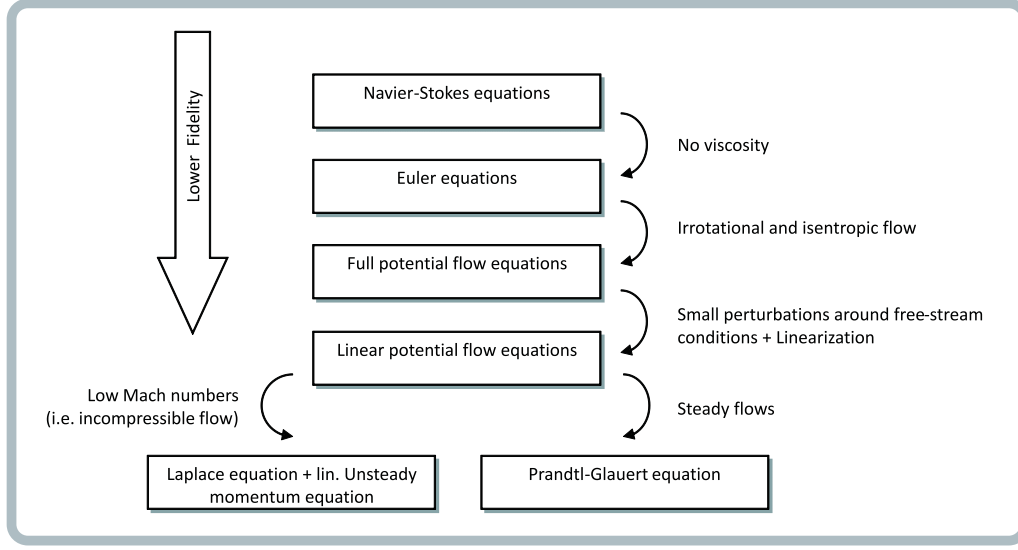


Figure 2.2: Schematic overview of assumptions in aerodynamic models.

obtains its name from the velocity potential ϕ that results from the condition of irrotational flows. The property of irrotational flows is described in the following.

Condition of irrotational flows

We start this subsection with presenting the so-called Nabla operator, used to denote for example the divergence of the velocity field with the in-product: $\nabla \cdot \mathbf{V}$, and the curl of the velocity field with the cross-product: $\nabla \times \mathbf{V}$. For a 2-dimensional problem, the velocity vector $\mathbf{V} = \begin{bmatrix} u & v \end{bmatrix}^T$ contains the flow speed in x and y direction (Fig. 2.1). The definition of the Nabla operator is best shown by considering the divergence of the velocity field:

$$\nabla \cdot \mathbf{V} = \begin{bmatrix} \frac{\partial}{\partial x} & \frac{\partial}{\partial y} \end{bmatrix} \cdot \begin{bmatrix} u \\ v \end{bmatrix} = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y}.$$

The description of irrotational flow is based on [35, 36]. A fluid particle may, besides having a translating motion, also rotate with an angular velocity ω . The angular velocity of such a fluid element relates to the so-called vorticity, defined by the curl of the velocity field: $\xi = 2\omega = \nabla \times \mathbf{V}$. Since pressure acts normal to a fluid particle it can not cause a rotation, therefore vorticity can only be created by viscous forces. A flow is said to be irrotational if:

$$\xi = \nabla \times \mathbf{V} = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y} = 0. \quad (2.9)$$

Subsonic flows over airfoils, applicable in this study, essentially have an irrotational flow field. There exist however, a thin boundary of viscous flow, which is highly rotational, at the airfoil surface. These effects, including the shed vortex wake, have to be modeled explicitly when assuming the flow to be inviscid and irrotational. The condition of irrotational flows in (2.9) is analogous to:

$$\nabla \times (\nabla \phi) = 0,$$

with ϕ a scalar function, the so-defined velocity potential. The analogy gives: $\mathbf{V} = \nabla\phi$, and leads to the remarkable simplification where the flow speeds at any position are defined with one scalar function:

$$\mathbf{V} = \nabla\phi \quad \text{equals:} \quad \begin{cases} u &= \frac{\partial\phi}{\partial x} \\ v &= \frac{\partial\phi}{\partial y} \end{cases} \quad (2.10)$$

Basically, the derivation of the full potential flow equations involves around a substitution of (2.10) in the Euler equations. The conservation of energy decouples from the set of equations as described earlier. This leaves the continuity and momentum equation as the governing set.

Continuity equation for incompressible flows: Laplace's equation

The continuity equation deals with the conservation of mass and yields for an unsteady, 2-dimensional flow [35]:

$$\frac{\partial\rho}{\partial t} + \nabla \cdot (\rho\mathbf{V}) = 0. \quad (2.11)$$

For incompressible flows, where ρ is constant, (2.11) simplifies to $\nabla \cdot \mathbf{V} = 0$. Combining this condition with (2.10) leads to the well known Laplace equation, the governing continuity equation for incompressible irrotational flows:

$$\nabla^2\phi = \frac{\partial^2\phi}{\partial x^2} + \frac{\partial^2\phi}{\partial y^2} = 0. \quad (2.12)$$

The Unsteady Bernoulli equation

The full potential flow equations are often too difficult to solve, hence a simplification is made based on the small perturbation theory. The small perturbation theory considers a disturbance around a uniform equilibrium state for all variables, *e.g.* $\phi = V_\infty x + \tilde{\phi}$. The ∞ subscript denotes free-stream conditions (Fig. 2.1), *i.e.* the aerodynamic operating point. To connect local pressure to the potential perturbation $\tilde{\phi}$, (2.12) is coupled with the unsteady Bernoulli equation (discarding the tilde):

$$p - p_\infty = -\rho_\infty \left(\frac{\partial\phi}{\partial t} + V_\infty \frac{\partial\phi}{\partial x} \right), \quad (2.13)$$

obtainable from the momentum equation in the Navier-Stokes equations by applying the assumptions in the linear potential flow theory. Hence, in conclusion, the governing equations for an incompressible, inviscid, irrotational flow can be given by the unsteady linearized potential flow equations (2.12) and (2.13). This set of equations has the advantage that they can be solved analytically when combined with 2-dimensional thin-airfoil theory.

2.3.3 Two-dimensional thin airfoil theory

This section, based on [35], gives a small overview of thin airfoil theory, a method to calculate the aerodynamic loads on a 2-dimensional wing profile. Most important, it shows the assumptions that are applicable to the methodology. It uses the linearized potential flow equations from previous section as the governing equations.

The problem of a 2-dimensional flow around an airfoil uses the concept of replacing the airfoil surface with a vortex sheet, as shown in Figure 2.3.

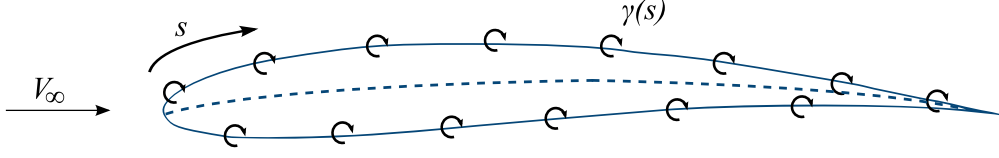


Figure 2.3: Airfoil with of arbitrary size and shape distributed by a vortex sheet.

Aerodynamic theory provides that the lift produced by such an airfoil is given by the circulation Γ around the airfoil. This theory uses the Kutta-Joukowski theorem [35]:

$$L'(t) = \rho_\infty V_\infty \Gamma(t) = \rho_\infty V_\infty \int \gamma(t, s) ds, \quad (2.14)$$

where $\gamma(t, s)$ presents the strength of the modeled vortices and s describes the contour of the airfoil, as illustrated in Figure 2.3. Basically, the Kutta-Joukowski theorem implies that the flow around a 2-dimensional lift producing airfoil, can be modeled by a uniform parallel flow plus a flow that circulates the airfoil. The vortices are solved in such a way that the airfoil surface is a streamline of the flow. Besides being a mathematical tool to solve the aerodynamic problem, the vortex sheet also serves another purpose. In real life, there is a viscous boundary layer on the airfoil surface. As mentioned in the previous section, this highly viscous region has substantial vorticity, which by effect is modeled with the vortex sheet [35]. In this manner the vortices and inviscid behavior is modeled explicitly allowing the use of the velocity potential. The same applies to the shed vortex wake, which will also be modeled explicitly.

With an arbitrary airfoil, as shown in Figure 2.3, the aerodynamic problem can only be solved numerically, the so-called panel method. In this section we explore an analytical solution to the problem by assuming the airfoil to be very thin. With this assumption, the aerodynamic problem is approximated by a vortex sheet distributed over the camber line, or propagating the assumption even further, the chord line, as shown in Figure 2.4. The vortices $\gamma(t, x)$, which are now located on the chord line (*i.e.* x -axis), are solved such that the camber line (*i.e.* s) is a streamline of the flow and such that Kelvin's theorem, *i.e.* an unsteady version of the Kutta condition, is satisfied. The Kutta condition takes care of the fact that in real life, the lower and upper flow leave the trailing edge smoothly. Mathematically this yields that the upper flow velocity that leaves the trailing edge is equal to the flow velocity under the airfoil leaving the trailing edge: $\gamma(TE) = 0$. Kelvin's theorem is based on the requirement that the Kutta condition must be satisfied at each time instant, and states that the shed vorticity is equal, but opposite in sign, to the change in bound circulation around the airfoil. Mathematically this reads [28]:

$$\gamma(b, t) = -\frac{1}{V_\infty} \frac{d\Gamma(t)}{dt}. \quad (2.15)$$

The shed vortices that make up the wake, from b to ∞ , behind the airfoil are modeled explicitly (Fig. 2.4). The vortices on the airfoil and in the wake are represented by γ_a and γ_w , respectively. Note that the induced velocities for the thin airfoil theory satisfy

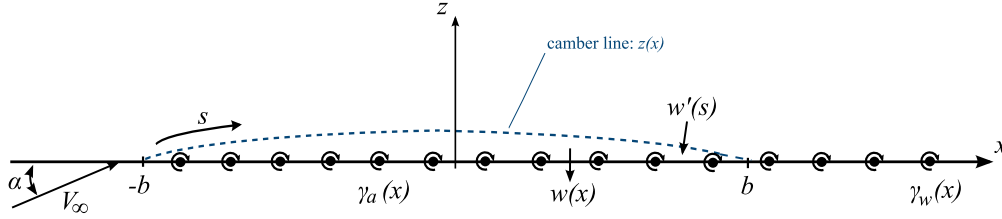


Figure 2.4: Thin-airfoil aerodynamic problem definition.

the relation: $w(x) \approx w'(s)$, where $w(x)$ is the velocity induced by the vortex sheet, and $w'(s)$ is the induced velocity due to modeling the camber as a streamline. Note that the vortex sheet (airfoil and wake) must satisfy the governing aerodynamic equations. Using the characteristics made in thin airfoil theory, the induced velocities can be written as [26, 28]:

$$w'(s) = w(x),$$

$$\frac{\partial z(x, t)}{\partial t} + V_\infty \frac{\partial z(x, t)}{\partial x} = -\frac{1}{2\pi} \int_{-b}^b \frac{\gamma_a(\xi, t)}{x - \xi} d\xi - \frac{1}{2\pi} \int_b^\infty \frac{\gamma_w(\xi, t)}{x - \xi} d\xi, \quad (2.16)$$

where ξ is just a dummy variable used for integration. In (2.16), the left side terms equal the boundary condition: the camber is defined a stream-line² of the flow, *i.e.* the flow tangency requirement on the airfoil surface. This ensures that no flow can pass through the airfoil, and is known as the Neumann boundary condition [39]. The right side of (2.16) is obtained as $w(x) = \frac{\partial \phi(x, z)}{\partial z} \Big|_{z=0}$, from the potential perturbation induced by the vortex sheet (see [26, 28] for a derivation). The right side terms also satisfy the far-field condition, *i.e.* the influence of the vortex sheet tends to zero in the far-field of the wake ($x \rightarrow \infty$).

An analytical solution for the thin-airfoil problem in (2.16) originated from Theodorsen [40], a topic reserved for the next section.

2.3.4 Solution to the harmonically oscillating thin-airfoil problem

Before giving the unsteady aerodynamic solution of Theodorsen, let us first look back at (2.16). Assuming harmonic responses, like $\gamma_a(x, t) = \bar{\gamma}_a(x)e^{j\omega t}$, etc., and assuming downstream convection of the shed vorticity with V_∞ , an expression of (2.16) for a harmonically oscillating airfoil can be found [28]:

$$j\omega \bar{z}(x) + V_\infty \frac{\partial \bar{z}(x)}{\partial x} = -\frac{1}{2\pi} \int_{-b}^b \frac{\bar{\gamma}_a(\xi)}{x - \xi} d\xi + \frac{j\omega \bar{\Gamma}}{2\pi V_\infty} \int_b^\infty \frac{e^{j\omega \left(\frac{b-\xi}{V_\infty}\right)}}{x - \xi} d\xi. \quad (2.17)$$

In (2.17), the second integral clearly shows that the vortex wake sheet ($x = b \rightarrow x = \infty$) is formed by an exponentially decaying shed vortex at $x = b$. After all, very far away from the airfoil, the shed vortices do not have any effect on the airfoil anymore. The harmonic shed vortex is defined by Kelvin's theorem in (2.15) transformed to the frequency domain: $\gamma(b, \omega) = \frac{-j\omega}{V_\infty} \Gamma(\omega)$. For the camber line, $\bar{z}(x)$, defined as a harmonically oscillating flat plate, Theodorsen solved this integral equation (2.17) analytically for $\gamma_a(x)$, which relates to the

²The path that the particles in a flow follow.

lift and moment on the airfoil with the Kutta-Joukowski theorem in (2.14). The solutions, normally expressed in terms of Theodorsen's function $C(k)$, are parameterized in terms of the reduced frequency $k = \frac{\omega b}{V_\infty}$. The reduced frequency, defined in (2.8), is clearly visible in (2.17), and has to do with the wake convection.

The solution of the thin airfoil problem, governed by the incompressible linearized potential flow equations, is presented in a equivalent manner as the mechanical model in Section 2.2, hence with so-called mass-damping-stiffness matrices. This format is more beneficial for the coupling of the mechanical sub-model with the aerodynamic sub-model in Section 2.4. The solution of the aerodynamic loads, based on the set of equations in [38, 41], can be written as the following matrix equation:

$$\begin{aligned} \mathbf{Q}_{\text{aero}} = & \underbrace{\mathbf{M}_{nc} \cdot \ddot{\mathbf{q}} + V_\infty \mathbf{D}_{nc} \cdot \dot{\mathbf{q}} + V_\infty^2 \mathbf{K}_{nc} \cdot \mathbf{q}}_{\text{non-circulatory term}} + \\ & \underbrace{\mathbf{C}_{se} \cdot \left(\theta_{qs}(\mathbf{q}, V_\infty) \cdot C(k) + \delta_{qs}(\mathbf{q}, V_\infty) \cdot C(k) + V_\infty w_G \cdot S(k) \right)}_{\text{circulatory term}}. \end{aligned} \quad (2.18)$$

In (2.18), the aerodynamic matrices are given by:

$$\begin{aligned} \mathbf{M}_{nc} &= \rho_\infty b^2 \cdot \begin{bmatrix} \pi & -\pi x_a b & -T_1 b \\ -\pi x_a b & \pi \left(\frac{1}{8} + x_a^2 \right) b^2 & -(T_7 + x_{f-a} T_1) b^2 \\ -T_1 b & 2T_{13} b^2 & -\frac{1}{\pi} T_3 b^2 \end{bmatrix}, \\ \mathbf{D}_{nc} &= \rho_\infty b^2 \cdot \begin{bmatrix} 0 & \pi & -T_4 \\ 0 & \pi \left(\frac{1}{2} - x_a \right) b & (T_1 - T_8 - x_{f-a} T_4 + \frac{1}{2} T_{11}) b \\ 0 & -\left(2T_9 + T_1 + \left(\frac{1}{2} - x_a \right) T_4 \right) b & -\frac{1}{2\pi} T_4 T_{11} b \end{bmatrix}, \\ \mathbf{K}_{nc} &= \rho_\infty b^2 \cdot \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & T_4 + T_{10} \\ 0 & 0 & \frac{1}{\pi} (T_5 - T_4 T_{10}) \end{bmatrix}, \\ \mathbf{C}_{se} &= \rho_\infty \cdot \begin{bmatrix} b \cdot C_{l_\theta} \\ -b^2 \cdot C_{m_\theta} \\ b^2 \cdot T_{12} \end{bmatrix}, \end{aligned}$$

with $x_{f-a} = x_f - x_a$ (see Table 2.1), $C_{l_\theta} = 2\pi$, $C_{m_\theta} = C_{l_\theta}(\frac{1}{2} + x_a)$, and the geometrical functions T_1, \dots, T_{14} are given in Appendix A.

The aerodynamic loads in (2.18) consist of two parts, namely a non-circulatory part independent of the reduced frequency k , and self-excited loads that dependent on k . The latter, *i.e.* the circulatory term, basically represents the vortex wake sheet. In (2.18), $\theta_{qs}(\mathbf{q}, V_\infty)$ represents the quasi steady pitch angle, and $\delta_{qs}(\mathbf{q}, V_\infty)$ the quasi steady pitch angle due to imposed flap deflection [38]. To enable easy coupling with the mechanical model in Section 2.4, we define these terms as:

$$\begin{bmatrix} \theta_{qs}(\mathbf{q}, V_\infty) \\ \delta_{qs}(\mathbf{q}, V_\infty) \end{bmatrix} = V_\infty \underbrace{\begin{bmatrix} 1 & \left(\frac{1}{2} - x_a \right) b & 0 \\ 0 & 0 & \frac{1}{2\pi} T_{11} b \end{bmatrix}}_{\mathbf{D}_{qs}} \dot{\mathbf{q}} + V_\infty^2 \underbrace{\begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & \frac{1}{\pi} T_{10} \end{bmatrix}}_{\mathbf{K}_{qs}} \mathbf{q}. \quad (2.19)$$

For a 2-DOF structural system the mechanical state is given by $\mathbf{q} = [h \ \theta]^T$, hence the non-circulatory and quasi-steady angle of attack matrices need to be changed accordingly. This

is easily performed by discarding the last row and column from \mathbf{M}_{nc} , \mathbf{D}_{nc} , \mathbf{K}_{nc} , \mathbf{D}_{qs} , \mathbf{K}_{qs} , and adding δ as a system input term to (2.18):

$$\mathbf{Q}_{aero}^{DOF} = \mathbf{Q}_{aero} + V_{\infty}^2 \mathbf{K}_{nc}^{(3)} \delta + \mathbf{C}_{se} C(k) V_{\infty}^2 \mathbf{K}_{qs}^{(3)} \delta. \quad (2.20)$$

with the superscript $M^{(i_1:i_2)}$ representing the matrix M that consists only of the columns contained in the superscript. The circulatory terms in (2.18) and (2.20) are given in Theodorsen's function $C(k)$ and Sears' function $S(k)$. These frequency domain functions are considered in the next subsection.

Theodorsen's and Sears' function

Theodorsen's function $C(k)$ describes the aerodynamic loads as a reaction to sinusoidal oscillatory (*i.e.* harmonic) motions of an airfoil within a uniform flow field. Moreover, the so-called Sears' function $S(k)$ describes the lift due to encountering a sinusoidal/harmonic gust traveling in the direction of flow [39, 42]. The gust is induced as a vertical (w.r.t. chord) velocity added at a certain instant to the free-stream velocity vector V_{∞} under the assumption that the wind gust disturbance has past at a certain time. The magnitude, in m/s, of the gust disturbance is presented with the symbol w_G . For the case of the wind turbine this leaves [43]:

$$V_{\infty} = \sqrt{V_t^2 + w_G^2},$$

where $V_t = R \cdot \omega_{rotor}$ is the tangential velocity, with for a 75% rotor annuli $R = 47.4$ m (see Table 2.1). The angular velocity ω_{rotor} of the wind turbine rotor is proportional to the wind speed that enters the rotor disk, as shown in Figure 2.5.

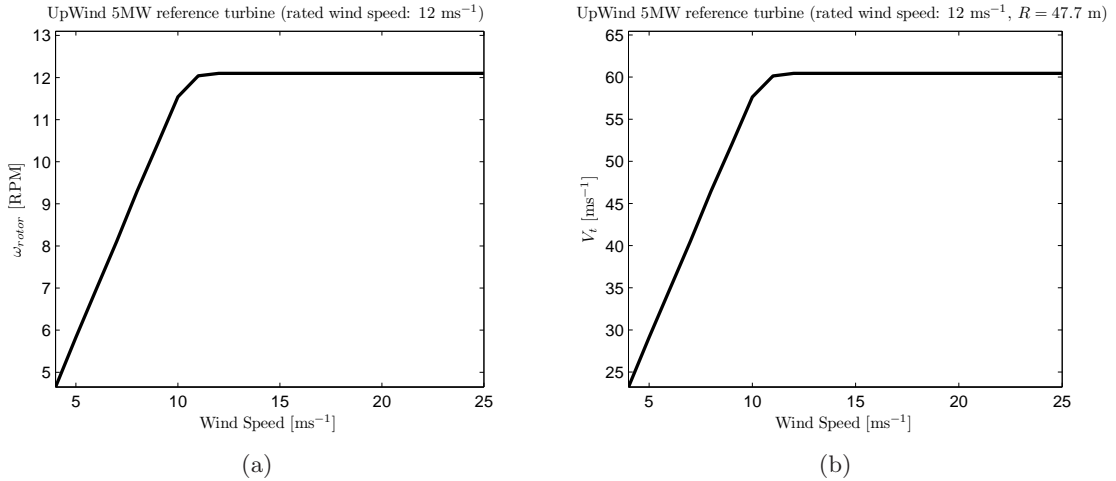


Figure 2.5: Wind Speed versus rotor angular velocity ω and tangential velocity V_t at 75% rotor annuli. Above the rated wind speed the wind turbine's pitch control acts to remain a steady angular velocity.

Additionally, small disturbances in the wind speed are modeled by w_G . Since V_t^2 has a much higher value than w_G^2 , especially for large wind turbines, we can assume, for the aeroelastic simulation plant, that $V_{\infty} \approx V_t$. This condition gives the possibility to model small swift fluctuations in wind speed (*e.g.* turbulence) with w_G , and large slowly varying wind speeds

with V_∞ . However, as remarked in [44], V_∞ is assumed constant in Theodorsen's original derivations, since the harmonic wake assumption fails for a time-varying free-stream velocity and arbitrary airfoil motions. Nevertheless, varying V_∞ (*i.e.* V_t) in a quasi-steady manner is reasonably accurate in particular circumstances, like for slow variations [45]. In effect, the quasi-steady velocity changes can be seen as varying the aerodynamic operating point of the linear aerodynamic model. To enable this option, the aeroelastic plant will be represented in a Linear Parameter-Varying (LPV) system format in Section 2.4. This also enables the possibility to incorporate (data-driven) LPV control methods in future projects.

Theodorsen's, and Sears's functions are only valid for harmonic responses, and given in the frequency domain. For arbitrary excitations, indicial (*i.e.* step) responses are more conveniently used in the field of aeroelasticity. It uses the idea that each arbitrary input response can be approximated numerically with either indicial/step responses or impulse responses via convolution. The time domain functions, known as Wagner's and Küssner's indicial step response function, relate to the earlier mentioned Theodorsen's and Sears' function through a Fourier transform pair [38]. In this report only the frequency domain functions are considered. Theodorsen's and Sears' function can be approximated by [38]:

$$C(k) = 1 - \frac{a_1 jk}{b_1 + jk} - \frac{a_2 jk}{b_2 + jk} \quad C(s) = 1 - \frac{a_1 s}{s + b_1 \frac{V_\infty}{b}} - \frac{a_2 s}{s + b_2 \frac{V_\infty}{b}}, \quad (2.21a)$$

$$S(k) = 1 - \frac{a_3 jk}{b_3 + jk} - \frac{a_4 jk}{b_4 + jk} \quad S(s) = 1 - \frac{a_3 s}{s + b_3 \frac{V_\infty}{b}} - \frac{a_4 s}{s + b_4 \frac{V_\infty}{b}}, \quad (2.21b)$$

with the parameters defined according to Table 2.2, and $j = \sqrt{-1}$. In (2.21), the $C(k)$ and $S(k)$ functions are also written in the Laplace domain using the definition of the reduced frequency in (2.8): $jk = \frac{j\omega b}{V_\infty} = \frac{sb}{V_\infty}$. Considering (2.21), a quasi-steady aerodynamic model can be represented by $C(k) = S(k) = 1$. Meaning that the decaying exponent $e^{-j\omega t}$ of assumed harmonic responses goes to zero for $t \rightarrow \infty$, which corresponds to the condition that $s \rightarrow 0$.

For arbitrary airfoil motion and/or arbitrary flap deflection, the unsteady circulatory load responses can be modeled into a state space model, like [38, 46], using Theodorsen's and Sears' function. As an example, we can rewrite (2.21a) into:

$$C(s) = \frac{(1 - a_1 - a_2)s^2 + (b_1 + b_2 - a_1 b_2 - a_2 b_1) \frac{V_\infty}{b} s + b_1 b_2 \frac{V_\infty^2}{b^2}}{s^2 + (b_1 + b_2) \frac{V_\infty}{b} s + b_1 b_2 \frac{V_\infty^2}{b^2}} = \frac{d_0 s^2 + d_1 s + d_2}{s^2 + c_1 s + c_2},$$

and transform it into the quadruple of state space matrices in controller canonical form [47]:

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array} \right] = \left[\begin{array}{cc|c} 0 & 1 & 0 \\ -c_1 & -c_2 & 1 \\ \hline d_2 - d_0 a_2 & d_2 - d_0 a_1 & d_0 \end{array} \right]. \quad (2.22)$$

Using (2.22), the circulatory (*i.e.* self-excited) terms in (2.18) can be converted to a state space system representation:

$$\begin{bmatrix} \frac{dx(t)}{dt} \\ \mathbf{Q}_{se}(t) \end{bmatrix} = \begin{bmatrix} A_{aero}(V_\infty) \\ \mathbf{C}_{se} \cdot C_{aero}(V_\infty) \end{bmatrix} \cdot x(t) + \begin{bmatrix} B_{aero}(V_\infty) \\ \mathbf{C}_{se} \cdot D_{aero} \end{bmatrix} \cdot \begin{bmatrix} \theta_{qs}(q, V_\infty) \\ \delta_{qs}(q, V_\infty) \\ w_G \end{bmatrix}. \quad (2.23)$$

Table 2.2: Parameters of Leishman's approximation of Theodorsen's and Sears' function [38].

Theodorsen's function		Sears' function	
a_1	0.2048	a_3	0.5792
b_1	0.0557	b_3	0.1393
a_2	0.2952	a_4	0.4208
b_2	0.2952	b_4	1.802

In (2.23), $x(t) \in \mathbb{R}^4$ is the aerodynamic lag states, and the quadruple of aerodynamic circulatory state space matrices given by:

$$\begin{aligned}
 A_{aero} &= \begin{bmatrix} 0 & 1 & 0 & 0 \\ -b_1 b_2 \frac{V_\infty^2}{b^2} & -(b_1 + b_2) \frac{V_\infty}{b} & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -b_3 b_4 \frac{V_\infty^2}{b^2} & -(b_3 + b_4) \frac{V_\infty}{b} \end{bmatrix}, \\
 B_{aero} &= \begin{bmatrix} 0 & 0 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & V_\infty \end{bmatrix}, \\
 C_{aero} &= \begin{bmatrix} b_1 b_2 (a_1 + a_2) \frac{V_\infty^2}{2b^2} & (a_1 b_1 + a_2 b_2) \frac{V_\infty}{b} & b_3 b_4 (a_3 + a_4) \frac{V_\infty^2}{2b^2} & (a_3 b_3 + a_4 b_4) \frac{V_\infty}{b} \end{bmatrix}, \\
 D_{aero} &= \begin{bmatrix} 1 - a_1 - a_2 & 1 - a_1 - a_2 & 1 - a_3 - a_4 \end{bmatrix}.
 \end{aligned} \tag{2.24}$$

To show the direct dependency of the free-stream velocity V_∞ , the state space matrices in (2.23) are structured as an LPV system:

$$\begin{aligned}
 A_{aero} &= A_{a_0} + A_{a_1} V_\infty + A_{a_2} V_\infty^2, \\
 B_{aero} &= B_{a_0} + B_{a_1} V_\infty + B_{a_2} V_\infty^2, \\
 C_{aero} &= C_{a_0} + C_{a_1} V_\infty + C_{a_2} V_\infty^2.
 \end{aligned} \tag{2.25}$$

2.4 The aeroelastic coupling

In the previous section we obtained a mechanical and aerodynamic sub-model. The aeroelastic plant is constructed by coupling these two models with by means of the $(Q)_{aero}$ term. We start this section with a small summary of the equations obtained so far:

- The mechanical sub-model in (2.4), which yields:

$$\mathbf{M}_{mech} \cdot \ddot{\mathbf{q}} + \mathbf{D}_{mech} \cdot \dot{\mathbf{q}} + \mathbf{K}_{mech} \cdot \mathbf{q} = \mathbf{C} + \mathbf{I}_{T_c} \cdot T_{ctrl} - \mathbf{Q}_{aero}.$$

- The aerodynamic sub-model from (2.18) and (2.23):

$$\mathbf{Q}_{aero} = \mathbf{M}_{nc} \cdot \ddot{\mathbf{q}} + V_\infty \mathbf{D}_{nc} \cdot \dot{\mathbf{q}} + V_\infty^2 \mathbf{K}_{nc} \cdot \mathbf{q} + \mathbf{C}_{se} \cdot \left(C_{aero}(V_\infty) \cdot x + D_{aero} \cdot \begin{bmatrix} \begin{bmatrix} \theta_{qs}(q, V_\infty) \\ \delta_{qs}(q, V_\infty) \end{bmatrix} \\ w_G \end{bmatrix} \right).$$

- The quasi-steady pitch motion due to pitch angle and flap deflection in (2.19), which yields:

$$\begin{bmatrix} \theta_{qs}(q, V_\infty) \\ \delta_{qs}(q, V_\infty) \end{bmatrix} = V_\infty \mathbf{D}_{qs} \dot{q} + V_\infty^2 \mathbf{K}_{qs} q.$$

The mechanical state is denoted with $q = [h \ \theta \ \delta]^T$, the aerodynamic lag state with x , and the input of the aeroelastic plant is formed by T_{ctrl} for the 3-DOF case. It should be noted that for a 2-DOF mechanical system, q , as well as its multiplicative terms, have one generalized coordinate less (Sec 2.2), hence T_{ctrl} disappears from the equations, and δ becomes the input, subsequently changing the aerodynamic system to (2.20). In this section, the aeroelastic model is worked out for both Degrees-Of-Freedom (DOF) and for unsteady aerodynamics, thus using (2.23). We form the aeroelastic plant in the following state space system:

$$\begin{bmatrix} \ddot{q} \\ \dot{q} \\ \dot{x} \\ y \end{bmatrix} = \begin{bmatrix} A_{AEPL} \\ C_{AEPL} \end{bmatrix} \cdot \begin{bmatrix} \dot{q} \\ q \\ x \end{bmatrix} + \begin{bmatrix} B_{AEPL} \\ D_{AEPL} \end{bmatrix} \cdot \begin{bmatrix} 1 \\ T_{ctrl} \text{ or } \delta \\ w_G \end{bmatrix}, \quad (2.26)$$

where the first input is added to incorporate constant terms, like for example the gravitational effects contained in \mathbf{C} in (2.4). We start this section by working out the A_{AEPL} and B_{AEPL} matrices. To make the coupling more easy to grasp, the accelerations for the coupled system are derived first:

$$\begin{aligned} (\mathbf{M}_{mech} + \mathbf{M}_{nc}) \ddot{q} = & - \left(\mathbf{D}_{mech} V_\infty \mathbf{D}_{nc} + V_\infty \mathbf{C}_{se} D_{aero}^{(1:2)} \mathbf{D}_{qs} \right) \dot{q} \\ & - \left(\mathbf{K}_{mech} + V_\infty^2 \mathbf{K}_{nc} + V_\infty^2 \mathbf{C}_{se} D_{aero}^{(1:2)} \mathbf{K}_{qs} \right) q \\ & + \mathbf{C} + \mathbf{I}_{T_c} \cdot T_{ctrl} - \mathbf{C}_{se} C_{aero} x - \mathbf{C}_{se} D_{aero}^{(3)} w_G, \end{aligned} \quad (2.27)$$

with the superscript $M^{(i_1:i_2)}$ representing the matrix M that consists only of the columns presented in the superscript.

The state-related system matrices

From (2.27), the aeroelastic plant is structured as a Linear-Parameter-Varying (LPV) system with V_∞ the varying parameter, hence the state space system in (2.26) is representation by:

$$\begin{aligned} A_{AEPL} &= A0 + A1 V_\infty + A2 V_\infty^2, \\ B_{AEPL} &= B0 + B1 V_\infty + B2 V_\infty^2. \end{aligned} \quad (2.28)$$

The above state matrices will be formulated on the next page, first however we define, the by the Degree Of Freedom (*i.e.* $DOF = \dim(q)$) sized matrices, as:

$$\begin{aligned} \mathbf{M} &= (\mathbf{M}_{mech} + \mathbf{M}_{nc}) \in \mathbb{R}^{DOF \cdot (2 \times 2)}, \\ \mathbf{D} &= \left(\mathbf{D}_{nc} + \mathbf{C}_{se} D_{aero}^{(1:2)} \mathbf{D}_{qs} \right) \in \mathbb{R}^{DOF \cdot (2 \times 2)}, \\ \mathbf{K} &= \left(\mathbf{K}_{nc} + \mathbf{C}_{se} D_{aero}^{(1:2)} \mathbf{K}_{qs} \right) \in \mathbb{R}^{DOF \cdot (2 \times 2)}. \end{aligned} \quad (2.29)$$

Using (2.27) and (2.29), the LPV state space A -matrices in (2.28) are constructed as follows:

$$A_0 = \begin{bmatrix} -\mathbf{M}^{-1}\mathbf{D}_{\text{mech}} & -\mathbf{M}^{-1}\mathbf{K}_{\text{mech}} & -\mathbf{M}^{-1}\mathbf{C}_{\text{se}}C_{a_0} \\ I & 0 & 0 \\ 0 & 0 & A_{a_0} \end{bmatrix}, \quad (2.30a)$$

$$A_1 = \begin{bmatrix} -\mathbf{M}^{-1}\mathbf{D} & 0 & -\mathbf{M}^{-1}\mathbf{C}_{\text{se}}C_{a_1} \\ 0 & 0 & 0 \\ B_{a_0}^{(1:2)}\mathbf{D}_{\text{qs}} & 0 & A_{a_1} \end{bmatrix}, \quad (2.30b)$$

$$A_2 = \begin{bmatrix} 0 & -\mathbf{M}^{-1}\mathbf{K} & -\mathbf{M}^{-1}\mathbf{C}_{\text{se}}C_{a_2} \\ 0 & 0 & 0 \\ 0 & B_{a_0}^{(1:2)}\mathbf{K}_{\text{qs}} & A_{a_2} \end{bmatrix}, \quad (2.30c)$$

The dimension of the matrices in (2.30) depend on the defined system's Degrees-Of-Freedom. The dimension of the LPV B -matrices dependent on the number of DOF in an equal same manner. Furthermore, since the system input is T_{ctrl} for $DOF = 3$ and δ for $DOF = 2$, the correct input terms are selected according to the specified DOF . Note that for a 2-DOF system the aerodynamic matrices change according to (2.20). Regarding the previous, the system's B -matrices can be derived as follows:

$$B_0 = \begin{bmatrix} \mathbf{M}^{-1}\mathbf{C} & \mathbf{M}^{-1}\mathbf{I}_{T_c} \cdot (DOF - 2) & -\mathbf{M}^{-1}\mathbf{C}_{\text{se}}D_{aero}^{(3)} \\ 0 & 0 & 0 \\ 0 & 0 & B_{a_0}^{(3)} \end{bmatrix}, \quad (2.31a)$$

$$B_1 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & B_{a_1}^{(3)} \end{bmatrix}, \quad (2.31b)$$

$$B_2 = \begin{bmatrix} -\mathbf{M}^{-1}\mathbf{C}_{\text{se}}D_{aero}^{(1:2)}\mathbf{K}_{\text{qs}}^{(2)} \cdot \theta_{twist} & -\mathbf{M}^{-1}\mathbf{K}^{(3)} \cdot (3 - DOF) & 0 \\ 0 & 0 & 0 \\ 0 & \left(B_{a_0}^{(1:2)}\mathbf{K}_{\text{qs}}^{(3)}\right) \cdot (3 - DOF) & B_{a_2}^{(3)} \end{bmatrix}, \quad (2.31c)$$

with θ_{twist} a constant initial lift term due to an incorporated structural twist or any other off-set from the $\theta = 0$ condition. The aeroelastic plant is, in case of (2.30) and (2.31), governed by unsteady aerodynamics. The aeroelastic plant can however also be represented for a quasi-steady aerodynamic model by defining $C(k) = S(k) = 1$ as indicated earlier. An LPV state-space formulation of this situation is however not part of the thesis.

The output-related system matrices

With the A_{AEPL} and B_{AEPL} state space matrices of the aeroelastic plant defined, the modeling is continued by giving a description of the output related state space matrices in (2.26). The simulation output variables of interest are defined as:

- Acceleration of the plunge and pitch motion, hence the C_{AEPL} and D_{AEPL} , matrices are identical to the first 2 or 3 rows, dependent on the number of DOF , of the earlier presented A_{AEPL} and B_{AEPL} matrices.

- The plunge and pitch motion of the airfoil, hence outputting q . Based on (2.26), this yields the output related matrices:

$$C_{AEPL} = \begin{bmatrix} 0 & I & 0 \end{bmatrix}, \quad D_{AEPL} = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}. \quad (2.32)$$

2.5 Verification and UpWind 5MW turbine implementation

In the previous sections the aeroelastic plant has been defined in a step-wise approach. An aeroelastic model that is specified for a wind turbine is not readily available in the literature, hence the implementation of the UpWind 5MW reference wind turbine's structural and dimensional parameters is of most importance to provide a simulation model for future reference. This is in contrast to aeroelastic models of aircraft wing, which are to some extent available in the literature. Therefore, the defined model in Section 2.4 is initially verified using a comparison with the available aerospace models in the literature, based on their included parameters. Thereafter, the 2-dimensional UpWind 5MW turbine model, with its parameters included in Table 2.1, is compared to provided data from a rotating 3-dimensional UpWind 5MW simulation model with a higher level of fidelity.

We start this section for this reason with parameterizing the aeroelastic model for the UpWind 5MW reference wind turbine according to Table 2.1). The poles/eigenvalues of the resulting 2-dimensional aeroelastic plant in (2.26) are given in Figure 2.6 as a function of free-stream velocity V_∞ . As can be seen from the figure, the pitch motion does not have much damping, and is therefore expected to have a high oscillatory behavior in the responses. Furthermore, the aeroelastic plant becomes unstable above $V_{flutter} = 113.83 \text{ ms}^{-1}$ due to the pitch related eigenvalues.

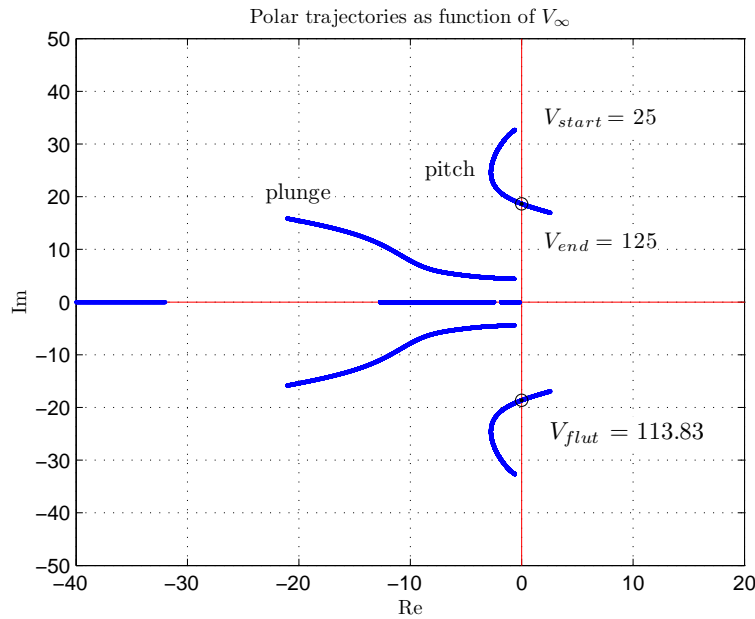


Figure 2.6: Polar plots with respect to V_∞ for the 5MW reference wind turbine, where the condition of marginal stability occurs at $V_{flutter} = 113.8297$.

2.5.1 Verification using aerospace models from literature

A perfect measure of model verification is an aeroelastic analysis with respect to the flutter speed, as performed for the UpWind 5MW reference wind turbine in Figure 2.6. This means that the particular V_∞ at which the aeroelastic model is marginally stable is determined. The pk-method is a more traditionally tool used in the field of aeroelasticity [26] for this purpose, but since the aeroelastic plant in Section 2.4 was constructed as an LPV system, it is also very suitable for this kind of aeroelastic analysis. Additionally, the flutter analysis can be performed on a state space model that is experimentally obtained with subspace model identification techniques (Chapter 3). In this manner, when the model identification procedure is performed on-line (*e.g.* recursively), a prediction can be made how close the system is to its flutter boundary. Alternatively, when a controller is active on the aeroelastic system, like an adaptive data-driven controller, estimated open-loop aeroelastic plant models can be obtained, which give the aeroelastic stability of the uncontrolled system.

The structural and dimensional parameters given in [32, 41, 48, 12] are implemented in the aeroelastic system presented in Section 2.4 and the computed flutter speed is compared to the one provided in the articles. The results from the literature comparison are summarized in Table 2.3. Although, differences are expected between the flutter speed obtained using the aeroelastic plant in this chapter and the flutter speed presented in the articles, due to for example a slight difference in $C(k)$ and $S(k)$ parameters or air density, the obtained flutter speeds fit the corresponding literature very well. Additionally, a 3-dimensional model of the UpWind 5MW reference turbine has been build in an earlier study using BEM theory [12]. Incorrect blade natural frequencies were used in this study. Nevertheless, implementation of the flap-wise and torsional natural frequencies from this study instead of the ones provided in Table 2.1 yields a nearly identical flutter speed as computed in this previous study.

Table 2.3: Comparison of the aeroelastic plant with model parameters and flutter speed provided in the literature. The flutter speed of the model in [48] was obtained from [32].

Reference:	[48] and [32]	[32]	[41]	[12]
From literature:	271.3 ms ⁻¹	139.3 ms ⁻¹	153.4 ms ⁻¹	120.0 ms ⁻¹
Model in Section 2.4:	273.6 ms ⁻¹	131.6 ms ⁻¹	154.9 ms ⁻¹	119.1 ms ⁻¹

2.5.2 Verification with gust responses from BLADED software package

In this section the 2-dimensional aeroelastic plant model of the UpWind 5MW reference wind turbine is simulated, and compared with plunge and pitch responses obtained from a rotating 3-dimensional simulation code that uses the BLADED software package. The simulated conditions include the varying disturbance gust, modeled in the 2-dimensional model as w_G , that is shown in Figure 2.7. Moreover, a constant wind speed below rated wind speed was used, so the pitch controller was not active in the simulations. This constant wind speed maps to a constant angular rotor speed, which is modeled with V_∞ in the 2-dimensional aeroelastic UpWind 5MW model.

The aeroelastic simulation model in Section 2.4 was run for the same conditions as the BLADED simulation package, namely, the gust w_G in figure 2.7, sectional properties at a rotor radius of $R = 48.65$ m, and a wind speed of 9 ms^{-1} . According to Figure 2.5(a), the

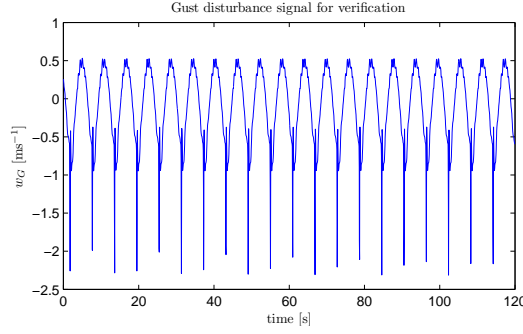


Figure 2.7: Wind Gust disturbance signal used in the BLADED wind turbine modeling software package that simulates the UpWind 5MW reference wind turbine.

latter results in a free-stream velocity of $V_\infty = 53 \text{ ms}^{-1}$ for a 48.65 m rotor radius. The mechanical and structural parameters were changed accordingly for this particular rotor annuli as well. The comparison simulations are shown in Figure 2.8. The 2-dimensional model of the UpWind 5MW reference wind turbine is incorporated without the constant terms, *e.g.* no structural twist and gravitational effects were incorporated. It should be noticed that defining the constant off-sets to be zero is consistent with a local linear model, where only perturbations around a certain operating point are considered. Indeed, as can be seen in Figure 2.8, the linear 2-dimensional aeroelastic plant model³ only shows the perturbations.

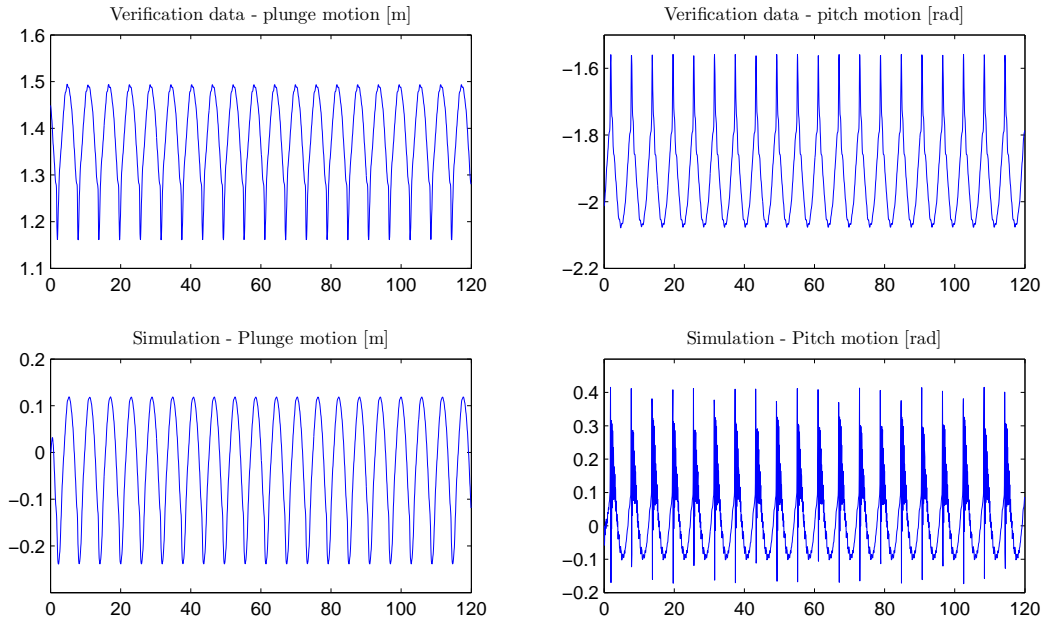


Figure 2.8: Verification data of the plunge and pitch Responses from the BLADED package (above) and the 2-dimensional UpWind 5MW model (below). Responses are for 9 ms^{-1} wind speed at a 48.65 m rotor radius (*i.e.* $V_\infty = 53 \text{ ms}^{-1}$).

³The condition includes only a constant free-stream velocity V_∞ , hence the result is indeed a linear model for the defined V_∞ .

The differences, in Figure 2.8, between the 2-dimensional aeroelastic plant model and the BLADED simulation data are only of small degree. It should be noticed that the 2-dimensional aeroelastic simulation model from Section 2.4 has a more oscillatory pitch motion. This is because the aeroelastic plant model that was constructed is only a 2-dimensional model representation, whereas the verification data is obtained from a 3-dimensional model which is much more damped with regard to torsional twist of the rotor blade. In conclusion, the 2-dimensional aeroelastic model is in compliance with the verification data that was provided from the BLADED software.

2.5.3 Frequency responses UpWind 5MW aeroelastic plant

In the previous sections we constructed a 2-dimensional aeroelastic ‘smart’ rotor blade model for the UpWind 5MW reference wind turbine. The parameters that make up the UpWind 5MW turbine are given in Table 2.1.

The 2-dimensional aeroelastic plant model is modeled in continuous time, however its intended use, *e.g.* digital controllers, adaptive data-driven control, and model identification, apply to discrete time systems. The system is therefore discretized using the zero-order-hold scheme with MATLAB’s `c2d`, function. It should be noted that the continuous time system needs to be discretized for every free-stream velocity V_∞ , even when quasi-steady velocity changes are made. A sample time of $h = 0.01s$ was chosen for the discretization, which was selected in accordance with the continuous time system dynamics for a range of V_∞ (see Fig. 2.6).

Figure 2.9 and Figure 2.10 show the frequency responses of the discretized aeroelastic plant model for free-stream velocities ranging from 30ms^{-1} to 120ms^{-1} . Directly noticeable is the observation that the magnitudes increase with an increasing free-stream velocity, which eventually results in an unstable system above the flutter speed. Figure 2.9 gives the Bode magnitude plot for the transfers $\delta \rightarrow [h, \theta]$, where we notice that the plunge motion has much higher gains compared to the pitch motion. The higher gains give a much better distinction with induced noise disturbance, like for example measurement noise. For these reason, the high-gain behavior of the plunge response compared to the pitch response, will come in favor when identifying the model using subspace model identification techniques. The frequency responses for the gust disturbance, *i.e.* the transfer $w_G \rightarrow [h, \theta]$, are shown in Figure 2.10. The effect of the wind gust on the system outputs is of smaller degree compared to the effect of the flap deflection δ on the outputs (Fig. 2.9). It is an important observation, since it means that the influence of the noise remains ‘low’ in the outputs. In Figure 2.10(b), a large downward peak can be seen, which indicates a frequency around which the wind gust has no real influence on the system output behavior anymore.

In Section 4, subspace model identification techniques are implemented on the 2-DOF UpWind 5MW wind turbine model shown in the figures. Based on the above observations, a couple of expectations can be made with regard to the identification. The identified model is expected to become more accurate when V_∞ increases, and is also expected to fit the plunge motion dynamics better than the pitch motion dynamics.

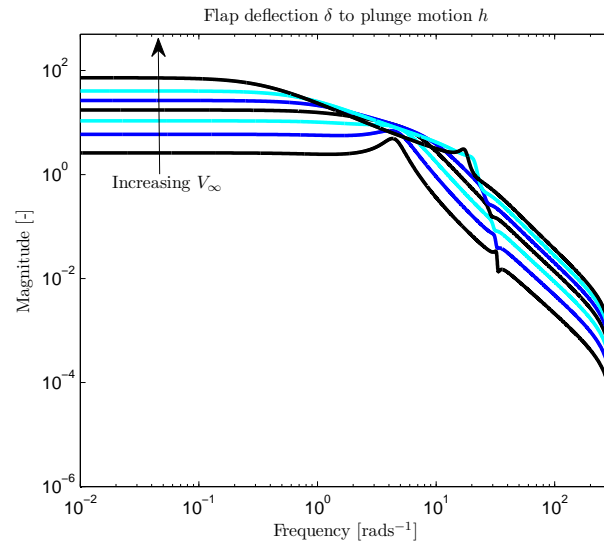
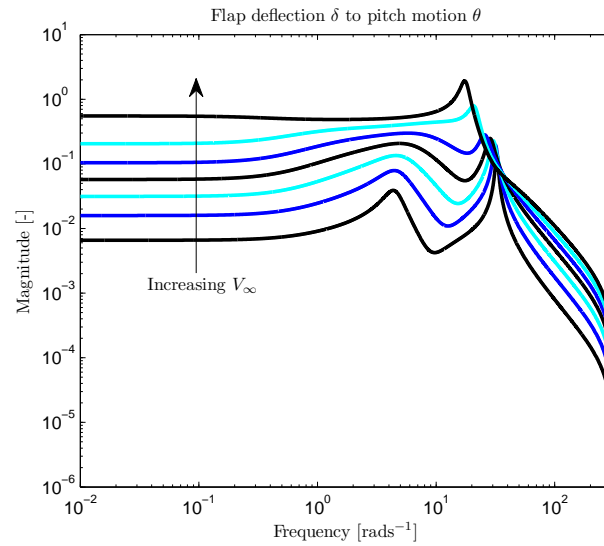
(a) $\delta \rightarrow h$ (b) $\delta \rightarrow \theta$

Figure 2.9: Bode diagram magnitude plot with respect to system input δ for discrete time system with $h = 0.01s$.

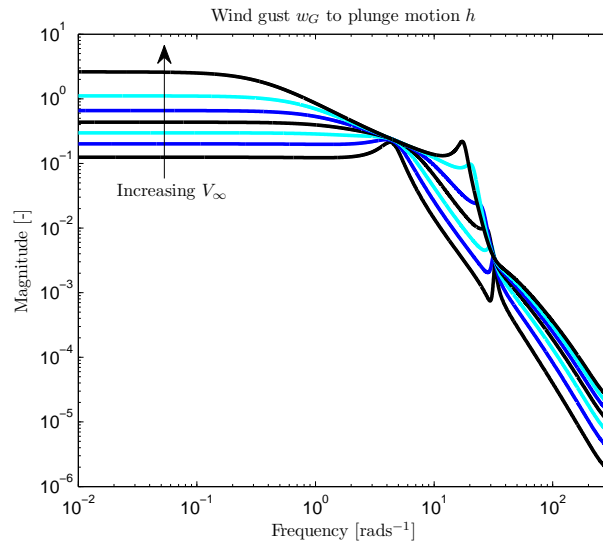
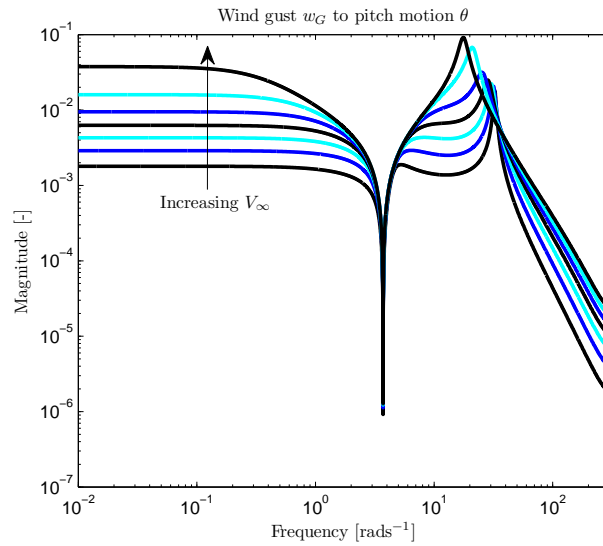
(a) $w_G \rightarrow h$ (b) $w_G \rightarrow \theta$

Figure 2.10: Bode diagram magnitude plot with respect to wind gust disturbance w_G for discrete time system with $h = 0.01s$.

2.6 Summary and final remarks

In this chapter, a model is developed that describes the 2-dimensional unsteady aeroelastic behavior of a ‘smart’ wind turbine rotor blade. Since aeroelasticity is characterized by the interaction between the flow and the structure, both a mechanical and an aerodynamic model are constructed based on the available literature. The aerodynamic model describes the aerodynamic loads, which depend on the orientation, shape and elastic deformation of the structure with respect to the flow. Moreover, the mechanical model describes the movements, orientation, and elastic deformations of the blade due to the aerodynamic loads. The aeroelastic plant model is thus formed by interconnecting the aerodynamic and mechanical sub-models. In the literature there are a number of 2-dimensional aeroelastic models available, but not like the novel aeroelastic plant model formed in this chapter. The aeroelastic system is constructed as a Linear Parameter Varying (LPV) state space system, with the free-stream velocity (*i.e.* V_∞) as the varying parameter. This enables the option to make quasi-steady changes in free-stream velocity with any LPV toolbox. In addition, the developed aeroelastic simulation model also incorporates a wind gust disturbance w_G . Since we can use both V_∞ and w_G as a varying input, it is possible to simulate a wind turbine ‘smart’ rotor blade for varying angular rotor speeds, as well as for varying gust disturbances (*e.g.* turbulence) that enter the rotor disc. However, only under the assumption that $V_\infty \gg w_G$ and V_∞ is slowly varying. We are thus able to simulate the wind turbine rotor blade also below rated wind speeds, where the slow changes in average wind speed are mapped to V_∞ using Figure 2.5, and small swift perturbations around this average wind speed are included as w_G .

The LPV structure of the constructed aeroelastic plant can also be quite useful for research into adaptive data-driven LPV controllers, for example in a feasibility study of adaptive data-driven control for a ‘smart’ wind turbine rotor blade in the future. Furthermore, the LPV format can also be used for the analysis of classical flutter. Traditionally the p-k method is often used in the field of aeroelasticity to find the classical flutter speed [26]. An alternative procedure based on the $A_{AEPL}(V_\infty)$ system matrix from (2.26) can however also be used. With the $A_{AEPL}(V_\infty)$ matrix we simply need to identify a value of V_∞ that results in some eigenvalues on the imaginary axis: the so-called flutter speed.

The constructed ‘smart’ rotor blade model can be used as a tool for aircraft flutter suppression, but also more corresponding to the subject of the thesis, for load alleviation or model identification on wind turbines. In this chapter we implemented the dimensional/structural parameters that correspond to the UpWind 5MW reference wind turbine. Because these parameters were available only for a 2-Degrees Of Freedom (DOF) system, the aeroelastic plant model of the wind turbine was restricted to 2-DOF. Nevertheless, in case the parameters get revised to a 3-DOF situation in the future, they can be easily incorporated in the aeroelastic model, since the aeroelastic simulation model is defined for both Degrees Of Freedom. It should be noted that, in contrast to the availability of aircraft wing models, there are not many models available in the literature that are specified for a wind turbine rotor blade with flap. The construction of the aeroelastic plant model, the implementation of the UpWind 5MW structural parameters, and the verification that was provided based on the more accurate rotating wind turbine code, can be considered as a welcome addition/contribution to the scarcely available 2-dimensional aeroelastic wind turbine models.

The used mechanical and aerodynamic sub-models are also known from literature, and are, as such, easily checked. The aeroelastic coupling is however performed differently compared to the aeroelastic simulation models in the literature. The challenge in this chapter was constructing the aeroelastic model as one coupled state space model, dependent on the free-stream velocity. It resulted in a highly structured derivation towards this combined mechanical-aerodynamic system. For this purpose, the mechanical and aerodynamic models needed to be redefined first, for example, it was shown in Section 2.3.4 that Theodorsen's and Sears' aerodynamic functions in (2.21) were transformed into state space (much like Leishman in [38]). Because of the novelty in structuring the coupled plant, a step of model verification is performed in Section 2.5, which implemented aerospace models from literature to make an evaluation of the developed aeroelastic model. The model presented in this chapter performs comparably well to the already existing aeroelastic models when these aircraft wing models are involved. After this, the UpWind 5MW reference wind turbine was implemented, and its gust responses compared to a simulation code with a higher fidelity, namely a rotating 3-dimensional aeroelastic model of the UpWind 5MW turbine. Also for this wind turbine model, the aeroelastic simulation model presented in this chapter performed very comparable to the higher fidelity model.

Chapter 3

Subspace Identification

This chapter presents a study of the current state-of-the-art in closed-loop Subspace Model Identification (SMI), presented in a novel ‘unifying’ closed-loop SMI framework that extends the existing identification methods with some new identification procedures. Furthermore, Chapter 3 provides the mathematical basis of the closed-loop SMI methods that are part of the comparative simulation study in the next chapter.

The SMI framework is presented in a stepwise approach, in which we sequentially handle a single-, double-, and multi-stage least squares procedure to estimate intermediate system parameters from which the open-loop system model can be determined. More stages indicate less computationally efficient methods. Furthermore, to solve for the state space matrices, most of the algorithms from literature require additional least squares procedures. This in contrast to the SMI algorithms formed in this thesis, which are able to determine the A matrix directly, making them, for example, very suitable for the earlier described (recursive) flutter prediction.

The description in this chapter is three-fold. Section 3.1 starts with a problem definition that provides a better understanding of closed-loop SMI, followed by a summary of the SMI methods incorporated under the ‘unifying’ description. Section 3.3 then presents the ‘unifying’ closed-loop SMI framework, including the novel identification procedures. Finally, the chapter ends by giving a short summary and conclusion.

3.1 The closed-loop identification problem

Control design involves the synthesis of a controller, based on a mathematical model of the system, hence obtaining an accurate representation of the system is of most importance. As indicated in Chapter 1, this model can be obtained using physical insight and first principles, or alternatively, it can be obtained experimentally from input-output data of the plant. Due to safety and operational restrictions, the experimental conditions might only include closed-loop conditions, which is the case for wind turbines. Furthermore, for an adaptive data-driven controller that uses a (partial) subspace identification step, the data is inherently also under feedback. The experimental identification of a system, using closed-loop Subspace

Identification Methods (SIM), is therefore the subject of this chapter. Figure 3.1 gives an illustrative example of a system under feedback. Considering this system, the closed-loop identification problem can be formulated as:

Problem formulation I

Find an estimated mathematical model of the plant, **given** joint input-output data under closed-loop conditions, like for example Figure 3.1.

Subspace Model Identification methods acquire a system parameterization in the form of a state space model, by application of linear algebra. This makes subspace identification computationally very attractive when compared to Prediction Error Methods (PEM) that use an iterative (optimization) procedure to find system parameters in a pre-defined model structure [49]. Since a state space model is estimated, SMI techniques are very suitable for estimation, filtering, prediction, and control of Multiple-Input Multiple-Output (MIMO) systems. The more traditional (open-loop) subspace identification methods, known as SSCVA [50], MOESP [51], and N4SID [52, 53], can however give biased results with closed-loop data [54, 55, 21]. This makes the use of special closed-loop SMI algorithms necessary. The open-loop subspace methods use so-called oblique projections to solve the identification problem, which is only valid under the condition that the input u_k is uncorrelated with the noise terms [51, 53]. Regarding Figure 3.1, this holds true when $C(q) = 0$, but for a closed-loop system, where u_k is formed by feedback of y_k (with its induced noise disturbances), this condition is no longer satisfied. In [54], Ljung and McKelvey clearly show that the traditional open-loop SMI methods result in a model mismatch for closed-loop data.

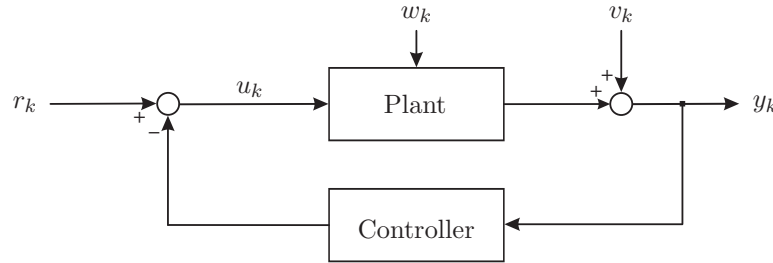


Figure 3.1: Illustrative block diagram of a closed-loop system with the controller $C(q)$, and the process, and measurement noise signals w_k and v_k , respectively.

The basic principle of the traditional open-loop subspace model identification procedures are illustrated in Figure 3.2. It shows that the methodology consists of two subsequent steps, namely:

1. the construction of a parallel bank of output predictors for the estimation of intermediate formatted matrices that define the predictors,
2. followed by a step where the state space matrices are determined.

The traditional open-loop SMI methods handle the second step in two different manners; MOESP [51] obtains estimates of the A and C state space matrices directly from the estimated bank of predictors, whereas N4SID [52] and SSCVA [50, 56] first find an estimate of the state x_k . The (remaining) state space matrices are then determined re-using the input-output data batches in a re-formulated problem that has to be solved for the unknown

system matrices. A Kalman filter state estimate can thus be obtained before the system is parameterized (Fig 3.2), one of the highly underestimated features of SMI, and especially useful for data-driven control, like SPC [57]. We start this chapter by defining some basic notations and formulations used in SMI, as well as giving the definition of the Kalman filter state estimate; the so-called Markov state.

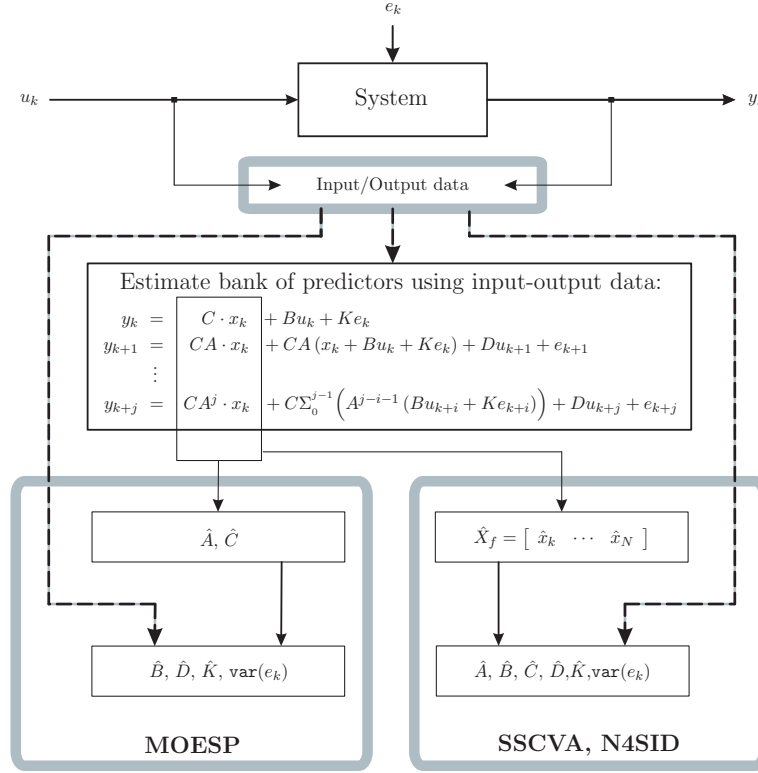


Figure 3.2: Basic principle behind subspace model identification techniques.

3.1.1 Formulations and the Markov state

The objective of identification methods is to find a mathematical representation of the open-loop plant model, given by the transfer from u_k to y_k as illustrated in Figure 3.1. The open-loop plant model used in the closed-loop subspace identification framework can be formulated by one of the following two forms.

- The **innovation form** representation:

$$x_{k+1} = Ax_k + Bu_k + Ke_k, \quad (3.1a)$$

$$y_k = Cx_k + Du_k + e_k. \quad (3.1b)$$

- The **predictor form** (*i.e.* the state observer) representation:

$$x_{k+1} = (A - KC)x_k + (B - KD)u_k + Ky_k, \quad (3.2a)$$

$$= \bar{A}x_k + \bar{B}u_k + Ky_k,$$

$$y_k = Cx_k + Du_k + e_k. \quad (3.2b)$$

In (3.1) and (3.2), the signals, $x_k \in \mathbb{R}^n$, $u_k \in \mathbb{R}^{n_u}$, $y_k \in \mathbb{R}^{n_y}$, and $e_k \in \mathbb{R}^{n_y}$ are the state, input, output, and innovation, respectively. Furthermore, K is the Kalman filter gain, and e_k is a zero-mean Gaussian distributed white noise signal. The relation between the two representations is explained by considering a Kalman filter (*i.e.* stochastic optimal observer), which estimates unknown/unmeasurable states using a system model and output measurements. In filter design, the innovation is defined as the error between the observer estimated outputs and the measured system outputs, *i.e.* $e_k = y_k - Cx_k - Du_k$. Substitution of this e_k into (3.1a) gives (3.2a).

Past and Future data separation

Traditionally subspace identification methods separate the joint input-output data in two subsequent batches, *i.e.* a ‘past’ related part to represent a Markov state sequence, and a ‘future’ related part to form a bank of output predictors. For this purpose we define a past ‘ p ’ and future ‘ f ’ horizon, and separate the joint input-output data according to the following block Hankel matrices:

$$Y_{k,\tau,\sigma} = \begin{bmatrix} y_k & \cdots & y_{k+\sigma-1} \\ \vdots & \ddots & \vdots \\ y_{k+\tau-1} & \cdots & y_{k+\tau+\sigma-2} \end{bmatrix}, \quad Y_p = Y_{0,p,N}, \quad Y_f = Y_{p,f,N}. \quad (3.3)$$

Note that the amount of measured data samples, denoted with \mathcal{N} , relates to $N = \mathcal{N} - p - f + 1$. For the remainder of the report, the ‘future’ related block Hankel matrices also need to be defined dependent on the index i :

$$Y_i = \begin{bmatrix} Y_{f1} \\ \vdots \\ Y_{fi} \end{bmatrix} \triangleq Y_{p,i,N} \quad \text{for} \quad 1 \leq i \leq f, \quad (3.4)$$

with each of its rows defined by Y_{fi} . Note that, in the subspace model identification problem, the block Hankel matrices in (3.3) and (3.4) are also defined for the signals u_k and e_k .

The Markov state sequence

Part of the subspace model identification problem is the inclusion of a reduced order estimate of the Markov states. Considering (3.2a), the state can be formulated as:

$$x_k = (qI - \bar{A})^{-1} \begin{bmatrix} \bar{B} & K \end{bmatrix} \begin{bmatrix} u_k \\ y_k \end{bmatrix}, \quad (3.5)$$

where q is the shift operator, *i.e.* $x_k \cdot q = x_{k+1}$. When the state is extended over multiple samples column-wise, and representing (3.5) with a numerical expansion over some past horizon p , we find the Markov state sequence:

$$\begin{aligned} X_{p,N} &= \begin{bmatrix} x_p & x_{p+1} & \cdots & x_{p+N-1} \end{bmatrix} = \bar{A}^p X_{0,N} + \bar{K} Z_p, \\ X_{p,N} &\approx \bar{K} Z_p, \end{aligned} \quad (3.6)$$

where:

$$\begin{aligned}\bar{\mathcal{K}} &\triangleq \begin{bmatrix} \bar{A}^{p-1}\bar{B} & \cdots & \bar{A}\bar{B} & \bar{B} & \bar{A}^{p-1}K & \cdots & \bar{A}K & K \end{bmatrix}, \\ Z_p &\triangleq \begin{bmatrix} U_p^T & Y_p^T \end{bmatrix}^T.\end{aligned}\tag{3.7}$$

In (3.6), the effect of the initial state condition $X_{0,N}$ is neglected by assuming that p goes to infinity, and under the condition that $\bar{A} = A - KC$ is stable [58]. Since only a finite past horizon p can be used, this induces a bias on the estimates. For this reason care must be taken to chose p high enough. It should be noted that the vector $C\bar{\mathcal{K}}$ contains the Markov parameters:

$$\Phi_\tau \triangleq \begin{bmatrix} \Phi_\tau^{\bar{B}} & \Phi_\tau^K \end{bmatrix} \triangleq C\bar{A}^{\tau-1} \begin{bmatrix} \bar{B} & K \end{bmatrix}.\tag{3.8}$$

3.1.2 Subspace model identification

According to the traditional open-loop SMI methods, the bank of predictors in Figure 3.2 are constructed in the innovation form system representation in (3.1). A likewise parallel bank of j -step ahead predictors can be defined for the predictor/observer form in (3.2), as will be provided in this section. Despite the fact that both representations are used for closed-loop subspace model identification, the objective is to find an estimate of the plant model in innovation form, thus computing A , B , C , D , and K from the measured plant input-output data. In Figure 3.2 we gave a basic overview of the methodology involved in estimating these system matrices. A more detailed explanation of the two-step SMI approach outlined in the figure is provided in this section.

The parameterization step that uses a Markov state estimate, *i.e.* the N4SID/SSCVA approach in Figure 3.2, is suitable for a bank of predictors constructed in both system forms. The use of the MOESP parameterization procedure is however limited to a problem formulation in innovation form, because only this form is directly dependent on the system's A matrix. Since the existing closed-loop SMI methods incorporated in the presented framework are mostly formulated using (3.2), and because of its applicability to both system representations, only the parameterization using the Markov state is included in this section. As previously mentioned, and indicated in Figure 3.2, the computation of a state space model involves the following two-stage procedure.

Stage 1a: Bank of predictors

The basis of SMI is a parallel bank of j -step ahead (output) predictors over some future horizon f , as indicated in Figure 3.2. These stacked $j = 0, \dots, f-1$ prediction models can either use a system in innovation (3.1) or predictor (3.2) form.

For a system representation in **innovation form** one can obtain a bank of predictors iterated over a future horizon f using the earlier presented Markov state estimate:

$$Y_f \approx \Gamma_f \bar{\mathcal{K}} Z_p + H_f U_f + G_f E_f + E_f.\tag{3.9}$$

In (3.9), the block Hankel matrices constructed according to (3.3), and the extended observability matrix Γ_f and Toeplitz matrices H_f and G_f defined as:

$$\Gamma_f \triangleq \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{f-1} \end{bmatrix}, \quad (3.10a)$$

$$H_f \triangleq \begin{bmatrix} D & 0 & \cdots & 0 \\ CB & D & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{f-2}B & CA^{f-3}B & \cdots & D \end{bmatrix}, \quad (3.10b)$$

$$G_f \triangleq \begin{bmatrix} 0 & 0 & \cdots & 0 \\ CK & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{f-2}K & CA^{f-3}K & \cdots & 0 \end{bmatrix}. \quad (3.10c)$$

Note that the matrix equation in (3.9) is identical to the set of equations contained in Figure 3.2. Equivalent to (3.9), a parallel bank of predictors can be build using the **predictor form** system representation in (3.2):

$$Y_f \approx \bar{\Gamma}_f \bar{K} Z_p + \bar{H}_f U_f + \bar{G}_f Y_f + E_f, \quad (3.11)$$

with the Toeplitz matrices defined in the same manner as (3.10), however now using the \bar{A} and \bar{B} matrices instead, see (3.29a) and (3.29b). Compared to (3.9), (3.11) depends solely on the Markov parameters $\bar{\Phi}_\tau$, and E_f is directly visible in the equation without a pre-multiplication matrix.

Stage 1b: Markov state estimation

The bank of predictors in (3.9) and (3.11) include an extended observability matrix, of which its column-wise subspace coincides with a certain characteristic subspace useful for identification [53]. This subspace, which has a dimension equal to the ‘low-order’ plant dynamics, can be used to estimate the Markov state sequence $X_{p,N}$ by a Singular Value Decomposition (SVD). Or alternatively, when considering the MOESP approach in Figure 3.2, the SVD can also be used to estimate the reduced order observability matrix $\Gamma_{f \times n}$. The latter is however only applicable for the bank of predictors defined in innovation form. This yields the following problem formulation specified for closed-loop SMI.

Problem formulation II

Find a for closed-loop input-output data unbiased estimate of either $\Gamma_f \bar{K}$ from (3.9), or $\bar{\Gamma}_f \bar{K}$ from (3.11), in order to obtain the estimated state sequence $X_{p,N}$ (or alternatively for (3.9) an estimate of $\Gamma_{f \times n}$).

Section 3.3 presents the closed-loop subspace model identification framework, which can be

used to solve the above closed-loop subspace identification problem. Thereafter an SVD can be performed on either of the two:

$$\widehat{\Gamma_f \bar{K}} Z_p = U \Sigma V^T \approx U_n \Sigma_n V_n^T, \quad (3.12a)$$

$$\widehat{\bar{\Gamma}_f \bar{K}} Z_p = U \Sigma V^T \approx U_n \Sigma_n V_n^T. \quad (3.12b)$$

In (3.12), Σ is a diagonal matrix containing the singular values. The purpose is to select an appropriate order n , based on the magnitude of the singular values, thus performing a step of model-order reduction. For more information on order determination using the singular values we refer to [51]. From the reduced order SVD in (3.12a) or (3.12b), the state can be computed with:

$$\hat{X}_{p,N} = V_n^T \in \mathbb{R}^{n \times N}. \quad (3.13)$$

Furthermore, the reduced extended observability matrix in innovation form can be obtained from the SVD in (3.12a):

$$\hat{\Gamma}_{f \times n} = U_n \in \mathbb{R}^{f \times n}. \quad (3.14)$$

Note that sometimes additional multiplicative weights based on the singular values are used, thus $\Sigma_n V_n^T$ or $\Sigma_n^{1/2} V_n^T$ and $U_n \Sigma_n$ or $U_n \Sigma_n^{1/2}$. This however produces nearly identical results as (3.13) and (3.14). This section continues with presenting how the system open-loop state-space matrices can be obtained using the estimated Markov state sequence in (3.13). The alternate procedure to find the state space matrices, using the reduced observability matrix in (3.14), is given in Section 3.3.1. This approach is however not of interest for the closed-loop identification problem in the predictor form, since it finds $\hat{\bar{\Gamma}}_{f \times n}$, which is *not* directly dependent on the open-loop A state space matrix.

Stage 2: System parameterization using the Markov state

The second step of SMI involves the computation of the quadruple of state space matrices by use of the estimated state sequence in (3.13). The state space output equation matrices C and D can be computed by performing linear least squares on:

$$Y_{p,N-1} = \mathcal{Y} = \underbrace{\begin{bmatrix} C & D \end{bmatrix}}_{\theta} \cdot \underbrace{\begin{bmatrix} \hat{X}_{p,N-1} \\ U_{p,N-1} \end{bmatrix}}_{\mathcal{W}} + \mathcal{E}_Y, \quad (3.15)$$

where $\mathcal{E}_Y = E_{p,N-1}$ is the unknown measurement noise data sequence, equivalent to the innovation e_k defined in (3.1). The solution of (3.15) is given by:

$$\begin{aligned} \hat{\theta} &= \mathcal{Y} \cdot \mathcal{W}^T (\mathcal{W} \mathcal{W}^T)^{-1} = \mathcal{Y} \cdot \mathcal{W}^\dagger, \\ \hat{\mathcal{Y}} &= \hat{\theta} \mathcal{W} = \mathcal{Y} \cdot \mathcal{W}^\dagger \mathcal{W} = \mathcal{Y} \cdot \Pi_{\mathcal{W}}, \\ \hat{\mathcal{E}}_Y &= \mathcal{Y} - \hat{\mathcal{Y}} = \mathcal{Y} \cdot (I - \Pi_{\mathcal{W}}) = \mathcal{Y} \cdot \Pi_{\mathcal{W}}^\perp, \end{aligned} \quad (3.16)$$

where $[\cdot]^\dagger$ denotes the Moore-Penrose pseudo-inverse. Furthermore, let us define the projection matrices: $Y_f \Pi_{\mathcal{W}}$ is the orthogonal projection of the row-space of Y_f onto the row-space of \mathcal{W} , and $\Pi_{\mathcal{W}}^\perp = I - \Pi_{\mathcal{W}}$. The projection matrices have the characteristics: $\mathcal{W} \Pi_{\mathcal{W}} = \mathcal{W}$ and $\mathcal{W} \Pi_{\mathcal{W}}^\perp = 0$. The remaining A , B , and K matrices can be determined using multiple approaches, of which two methods are handled in the following. We consider the methodologies provided in [51] and [58].

1. Qin solves, in [58], the following problem to obtain estimates for the A , B , and K system matrices by re-using the innovation $\hat{\mathcal{E}}_Y = \hat{E}_{p,N-1}$ found in (3.16):

$$\hat{X}_{p+1,N-1} = \begin{bmatrix} A & B & K \end{bmatrix} \cdot \begin{bmatrix} \hat{X}_{p,N-1} \\ U_{p,N-1} \\ \hat{E}_{p,N-1} \end{bmatrix}. \quad (3.17)$$

The Kalman filter gain K is obtained by solving (3.17), however there is no guarantee that this K is stabilizing. Therefore computing K by solving the Discrete Algebraic Riccati Equation (DARE), like in [51], seems more appropriate.

2. Verhaegen and Verdult estimate, in [51], the A and B matrices, as well as the stochastic signal \mathcal{E}_X , by solving a stochastic least squares problem analogous to (3.15):

$$\hat{X}_{p+1,N-1} = \begin{bmatrix} A & B \end{bmatrix} \cdot \begin{bmatrix} \hat{X}_{p,N-1} \\ U_{p,N-1} \end{bmatrix} + \underbrace{K \cdot E_{p,N-1}}_{\mathcal{E}_X}. \quad (3.18)$$

The Kalman filter gain K can then be computed by solving the DARE [51]:

$$\begin{aligned} \hat{P} &= \hat{A}\hat{P}\hat{A}^T + \hat{Q} - \left(\hat{S} + \hat{A}\hat{P}\hat{C}^T \right) \left(\hat{C}\hat{P}\hat{C}^T + \hat{R} \right)^{-1} \left(\hat{S} + \hat{A}\hat{P}\hat{C}^T \right)^T, \\ \hat{K} &= \left(\hat{S} + \hat{A}\hat{P}\hat{C}^T \right) \left(\hat{R} + \hat{C}\hat{P}\hat{C}^T \right)^{-1}, \end{aligned} \quad (3.19)$$

based on the noise (sample) covariances from the estimated noise sequences:

$$\begin{bmatrix} \hat{Q} & \hat{S} \\ \hat{S}^T & \hat{R} \end{bmatrix} = \lim_{N \rightarrow \infty} \frac{1}{N-1} \cdot \begin{bmatrix} \hat{\mathcal{E}}_X \\ \hat{\mathcal{E}}_Y \end{bmatrix} \cdot \begin{bmatrix} \hat{\mathcal{E}}_X^T & \hat{\mathcal{E}}_Y^T \end{bmatrix}$$

Section 3.3.1 includes a simulation based comparison of the above two parameterization methods, combined with a parametrization procedure that obtains the system's A and C matrices using the reduced observability matrix. As indicated in this comparison, the methodology in [51] produces biased results when the constraint $D = 0$ is imposed, whereas the methodology according to [58] gives unbiased estimates.

3.2 A combined framework: presenting the methods

This section gives an overview of the methods that are allocated under the ‘unifying’ closed-loop SMI formulation. It thus provides a basic introduction of the subspace model identification framework that is presented in Section 3.3. As such, it gives a good distinction between the closed-loop SMI methods from literature and the novel identification procedures formed in this thesis. Since the ‘unifying’ description in Section 3.3 only considers closed-loop methods, this section will start with providing a description of the traditional open-loop methods based on [53, 51].

3.2.1 The traditional open-loop methods described

For open-loop data, the input u_k is uncorrelated with the innovation e_k , so the following is applicable:

$$\lim_{N \rightarrow \infty} \frac{1}{N} E_{k,f,N} U_{k,f,N} = 0, \quad \text{hence: } E_{k,f,N} \Pi_{U_{k,f,N}}^\perp = E_{k,f,N}, \quad (3.20a)$$

$$\lim_{N \rightarrow \infty} \frac{1}{N} E_{k,f,N} Z_p^T = 0, \quad (3.20b)$$

Under these conditions, (3.9) yields the following unbiased stochastic linear least squares solution:

$$\widehat{\Gamma_f \mathcal{K}} = Y_f \Pi_{U_f}^\perp Z_p^T (Z_p \Pi_{U_f}^\perp Z_p^T)^{-1}. \quad (3.21)$$

The traditional open-loop SMI methods perform a Generalized Singular Value Decomposition (GSVD) on (3.21), as given by:

$$W_1 \widehat{\Gamma_f \mathcal{K}} W_2 \approx U_n \Sigma_n V_n^T, \quad (3.22)$$

where the weights that correspond to the traditional open-loop methods are presented in Table 3.1. The state space matrices can be estimated thereafter using the approaches illustrated in Figure 3.2, and:

$$\hat{X}_{p,N} = V_n^T W_2^{-1} Z_p, \quad (3.23a)$$

$$\hat{\Gamma}_{f \times n} = W_1^{-1} U_n. \quad (3.23b)$$

Table 3.1: The weights used for the traditional open-loop methods according to [51].

SSCVA	N4SID	PO-MOESP
$W_1 = (Y_f \Pi_{U_f}^\perp Y_f^T)^{-1/2}$	$W_1 = I$	$W_1 = I$
$W_2 = (Z_p \Pi_{U_f}^\perp Z_p^T)^{1/2}$	$W_2 = (Z_p Z_p^T)^{1/2}$	$W_2 = (Z_p \Pi_{U_f}^\perp Z_p^T)^{1/2}$

The GSVD with SSCVA weights is conceptually identical to Larimore's original SSCVA in [50, 56], which performs a Canonical Variate Analysis (CVA) between $Y_{f-} = Y_f \Pi_{U_f}^\perp$ and Z_p . The CVA between these two variables is performed according to the GSVD in (3.42), and as such yields a slightly different solution.

3.2.2 The closed-loop methods presented

The most recent closed-loop SMI algorithms solve the closed-loop identification problem by using the same methodology as the open-loop SMI algorithms, but since (3.20) is not valid for closed-loop systems, they either extend the approach with a High-Order Auto-Regressive eXogenous inputs (HOARX) pre-estimate, or by solving the open-loop identification problem in a multi-stage manner. The included closed-loop SMI methods can be arranged according to the number of stages in which a stochastic linear least squares procedure is used to solve **Problem formulation II**.

Single-stage procedures

The methods that only need to solve one stochastic linear least squares problem in the first step of SMI are:

- The optimally weighted Predictor Based Subspace IDentification method (**PBSID_{opt}**) using a High-Order Auto-Regressive with eXogenous inputs (HOARX) model estimate introduced by Chiuso and Picci [59]. It estimates the system's Markov parameters (defined only in predictor form), which are then used to construct a solution of the closed-loop identification problem in predictor form.
- A novel method that rewrites the **PBSID_{opt}** solution into an innovation form system representation, which yields a new solution to the identification problem. Two procedures that incorporate this methodology are developed in this thesis, namely a method that uses the MOESP approach to obtain the open-loop state space matrices, and a method that uses the N4SID procedure through the state. The former is denoted with **ARXISS**, whereas the latter has been given the name **IEM_{opt}**, to remain close to the existing terminology used in subspace model identification.

Ever-since notice was given of the novel subspace identification methodology explored with the ARXISS and IEM_{opt} algorithms, it has attained much attention within DCSC. Consequently this resulted in an extended insight to the approach. It should however be noted that the methodology was founded as part of this M.Sc. thesis, and as such its development is seen as one of the main contributions of the thesis.

Double-stage procedures

The double-stage procedures include closed-loop identification methods that require a pre-estimate from an estimated HOARX model. The following double-stage algorithms are considered:

- The **SSARX** subspace identification method of Jansson [60]. This SMI procedure uses pre-estimates in terms of Markov parameters that are used to formulate a bank of j -step ahead predictors in the predictor form, which is consistent with the Markov parameters. The bank of predictors is then solved using a Canonical Variance Analysis.
- A novel identification method that computes pre-estimates in innovation form, obtainable from the estimated Markov parameters. Hence, the use of a bank of predictors represented in innovation form is consistent with the pre-estimates. The bank of predictors is then solved using a Canonical Variance Analysis (CVA). Two methods to obtain the state space matrices explore this methodology, denoted with **CVA-MOESP** for a MOESP approach and **ISSCVA** that uses the state. The latter is, in methodology, an exact replica of the SSARX approach, however now defined for a system represented in innovation-form.
- A novel identification method that uses a pre-estimate of the innovation sequence in a MOESP like procedure. It borrows the idea from the IEM identification procedure of Qin and Ljung [61], but yields an entirely different approach to solve the identification problem. This method is presented with the name **IE-MOESP** for the remainder of the chapter.

Multi-stage procedures

The ‘unifying’ closed-loop SMI description includes two identification methods that follow a multi-stage procedure:

- The Innovation Estimate Method (IEM) of Qin and Ljung [61]¹. The **IEM1** technique, which uses a partial pre-estimated innovation sequence and solves the identification problem per prediction step (See Fig. 3.2), is included in the ‘unifying’ closed-loop identification framework.
- Predictor Based Subspace IDentification (**PBSID**) of Chiuso and Picci [62]². This method represents a predictor form based version of the IEM1 approach, but alternatively does not require a pre-estimate of the innovation sequence.

3.3 Unifying Closed-loop identification description

Section 3.1 clarified the basic subspace model identification approach. We start this section by referring to **Problem formulation II**. As indicated in the closed-loop identification problem, we desire to solve for $\Gamma_f \bar{K}$ or $\bar{\Gamma}_f \bar{K}$, from respectively (3.9) and (3.11), such that a low-order Markov state sequence, or low-order observability matrix (in innovation form), can be estimated using the SVD in (3.12). The unifying closed-loop SMI framework follows a step-wise approach describing the single-, double-, and multi-stage procedures that were outlined in Section 3.2. The presented framework is summarized in Figure 3.21.

3.3.1 HOARX estimate: single-stage procedures

The first block row of (3.9) and (3.11) are identical and is given by:

$$Y_{f1} = Y_{p,N} = C\bar{K}Z_p + DU_{f1} + E_{f1}. \quad (3.24)$$

Discarding the future horizon (*i.e.* $f = 1$), the first row can be represented over the longer, more accurate, sample length $\mathcal{N} - p = N + f - 1$:

$$Y_{p,\mathcal{N}-p} = C\bar{K}Z_{p+} + DU_{p,\mathcal{N}-p} + E_{p,\mathcal{N}-p}. \quad (3.25)$$

The past data block Hankel matrices are also changed accordingly,, hence we define:

$$\begin{aligned} Y_{p,\mathcal{N}-p} &= Y_{p,1,\mathcal{N}-p}, \\ U_{p,\mathcal{N}-p} &= U_{p,1,\mathcal{N}-p}, \\ Z_{P+} &= \begin{bmatrix} U_{0,p,\mathcal{N}-p}^T & Y_{0,p,\mathcal{N}-p}^T \end{bmatrix}^T, \end{aligned} \quad (3.26)$$

with the Hankel matrices defined according to (3.3). Assuming that either a delay in the inputs exist, or $D = 0$, this HOARX model can be estimated using linear least squares without a bias for closed-loop data, since u_k and e_k are uncorrelated under this condition.

¹Alternatively, the procedure is sometimes also referred to as Parsim-E(1).

²In the article presented as the Whitening Filter Algorithm (WFA).

With digital controllers, a delayed system input is a very reasonable assumption. Under this assumption the linear least squares solution yields:

$$\begin{bmatrix} \widehat{C\bar{K}} & \hat{D} \end{bmatrix} = Y_{p,\mathcal{N}} \cdot \begin{bmatrix} Z_{p^+} \\ U_{p,\mathcal{N}-p} \end{bmatrix}^\dagger. \quad (3.27)$$

Note that for the assumption that $D = 0$, the above linear least squares problem does not contain any ‘future’ inputs. From the estimated HOARX model in (3.27), one can obtain:

1. The first block row solution of (3.9) and (3.11), *i.e.* $\widehat{C\bar{K}}$ and \hat{D} .
2. An estimate of the future innovation block Hankel matrix \hat{E}_f from the entries of:

$$\hat{E}_{p,\mathcal{N}-p} = Y_{p,\mathcal{N}-p} - \begin{bmatrix} \widehat{C\bar{K}} & \hat{D} \end{bmatrix} \begin{bmatrix} Z_{p^+} \\ U_{p,\mathcal{N}-p} \end{bmatrix}. \quad (3.28)$$

Note that $\hat{E}_{p,\mathcal{N}-p}$ can also be estimated directly by means of (3.16), thus without the need to initially obtain the first block row solution.

3. The estimated Markov parameters $\hat{\Phi}_p, \dots, \hat{\Phi}_1$ from $\widehat{C\bar{K}}$ and \hat{D} , which can be used to construct estimates of the \bar{H}_f and \bar{G}_f Toeplitz matrices in (3.11):

$$\bar{H}_f \triangleq \begin{bmatrix} D & 0 & \dots & 0 \\ \Phi_1^{\bar{B}} & D & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{f-1}^{\bar{B}} & \Phi_{f-2}^{\bar{B}} & \dots & D \end{bmatrix}, \quad (3.29a)$$

$$\bar{G}_f \triangleq \begin{bmatrix} 0 & 0 & \dots & 0 \\ \Phi_1^K & 0 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ \Phi_{f-1}^K & \Phi_{f-2}^K & \dots & 0 \end{bmatrix}. \quad (3.29b)$$

Moreover, the solution of the Auto-Regressive implementation of the PBSID_{opt} method can also be obtained at this stage [59]:

$$\left\{ \widehat{\bar{\Gamma}_f \bar{K}} \right\}_{\text{PBSID}_{opt}} \triangleq \begin{bmatrix} \Phi_p^{\bar{B}} & \Phi_{p-1}^{\bar{B}} & \dots & \dots & \Phi_1^{\bar{B}} & \Phi_p^K & \Phi_{p-1}^K & \dots & \dots & \Phi_1^K \\ 0 & \ddots & \dots & \dots & \vdots & 0 & \ddots & \dots & \dots & \vdots \\ 0 & 0 & \Phi_p^{\bar{B}} & \dots & \Phi_f^{\bar{B}} & 0 & 0 & \Phi_p^K & \dots & \Phi_f^K \end{bmatrix}. \quad (3.30)$$

The PBSID_{opt} algorithm is the ‘optimized’ version of the Predictor Based Subspace IDentification method that performs multi-stage row-wise linear least squares on (3.11) presented Section 3.3.3. Note that the PBSID_{opt} procedure also imposes a Toeplitz structure on \bar{H}_f and \bar{G}_f .

4. The estimated Markov parameters and D -matrix can also be used to compute the innovation form Toeplitz matrices H_f and G_f in (3.10). Including the definition $\Phi_0^{\bar{B}} = D$, the entries in the H_f and G_f matrices are formed by the following series expansions:

$$\Lambda_i^K = CA^{i-1}K = \Phi_i^K + \sum_{j=1}^{i-1} \Phi_{i-j}^K \cdot \Lambda_j^K, \quad \text{for } i = 1, 2, 3, \dots \quad (3.31)$$

$$\begin{aligned}
\Lambda_i^B &= \Phi_i^{\bar{B}} + \sum_{j=0}^{i-1} \Phi_{i-j}^K \cdot \Lambda_j^B, \quad \text{giving:} \\
\Lambda_0^B &= D, \\
\Lambda_i^B &= CA^{i-1}B, \quad \text{for } i = 1, 2, 3, \dots
\end{aligned} \tag{3.32}$$

A relation between $\Gamma_f \bar{\mathcal{K}}$ and $\bar{\Gamma}_f \bar{\mathcal{K}}$ can also be formulated using an identical approach. When a solution of $\bar{\Gamma}_f \bar{\mathcal{K}}$ is known, *i.e.* at this stage the PBSID_{opt} solution in (3.30), we can use this solution to define each of its block rows as $\Xi_\tau = C\bar{A}^\tau \bar{\mathcal{K}}$, and compute:

$$\Theta_i = CA^i \bar{\mathcal{K}} = \Xi_i + \sum_{j=0}^{i-1} \Phi_{i-j}^K \cdot \Theta_j, \tag{3.33}$$

which can be stacked to form the novel solution to the identification problem:

$$\left\{ \widehat{\Gamma_f \bar{\mathcal{K}}} \right\}_{\text{ARXISS/IEM}_{opt}} = \left[\begin{array}{ccc} \Theta_0^T & \cdots & \Theta_{f-1}^T \end{array} \right]^T. \tag{3.34}$$

A derivation of (3.31), (3.32) and (3.33) is given in Appendix B. The solution in (3.34) can be used to determine the system's state space matrices using the two approaches shown in Figure 3.2. Like PBSID_{opt} , the novel IEM_{opt} procedure defines a regression problem based on the estimated Markov state sequence, as defined earlier. Since (3.34) is a solution in innovation form, the A , B , C , and D matrices can also be obtained in the same manner as the open-loop MOESP algorithm. The methodology to obtain the quadruple of state space matrices in this manner is presented in the next subsection, and is denoted with ARXISS: Auto-Regressive eXogenous inputs Innovation-based SubSpace. Section 3.3.2 continues with the next stage of the unifying description thereafter.

The single-stage PBSID_{opt} and IEM_{opt} methods are represented schematically in Figure 3.3 and are summarized in Algorithms 1 and 2. As can be seen from the figure, the estimated HOARX model service two purposes, namely the construction of pre-estimates that can be re-used in different stages of the presented framework, and three single-stage SMI procedures. The ARXISS algorithm is described in the next subsection.

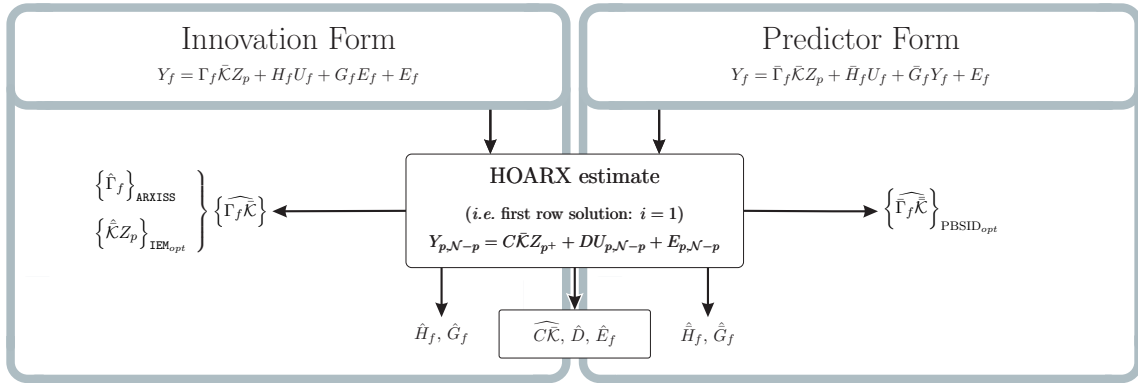


Figure 3.3: Overview of the single-stage procedures.

Algorithm 1 $PBSID_{opt}$

1. Construct the matrices $Y_{p,N-p}$, $U_{p,N-p}$, and Z_{P+} according to (3.26).
2. Solve the HOARX problem in (3.25), if desired imposing the constraint $D = 0$.
3. Construct $\widehat{\Gamma\mathcal{K}}$ for $PBSID_{opt}$ according to (3.30) using the Markov parameters from the HOARX problem solution in (3.27).
4. Determine the Markov state estimate, $\hat{X}_{p,N}$ using (3.12b) and (3.13).
5. Obtain an estimate of the A , B , C , D and K system matrices by solving the least squares problems in (3.15) and (3.17), if desired constraining $D = 0$.

Algorithm 2 IEM_{opt}

1. Construct the matrices $Y_{p,N-p}$, $U_{p,N-p}$, and Z_{P+} according to (3.26).
2. Solve the HOARX problem in (3.25), if desired imposing the constraint $D = 0$. The solution in (3.27) immediately gives an estimate of D .
3. Construct $\widehat{\Gamma\mathcal{K}}$ according to (3.30) using the Markov parameters from the HOARX problem solution in (3.27).
4. Transform (3.30) to the innovation form, which gives $\widehat{\Gamma\mathcal{K}}$, using (3.33) and (3.34).
5. Determine the Markov state estimate, $\hat{X}_{p,N}$ using (3.12a) and (3.13).
6. Obtain an estimate of the A , B , C , D and K system matrices by solving the least squares problems in (3.15) and (3.17), if desired constraining $D = 0$.

Example I. At this stage we have described the $PBSID_{opt}$ and IEM_{opt} algorithms in detail, and are able to start the identification procedure on a simulation example. We will use the instable system provided in example 2 in [62], which has the plant model $G(z)$ and the noise model $H(z)$:

$$G(z) = \frac{2.5}{z-3}, \quad H(z) = \frac{z+0.5}{z}, \quad (3.35)$$

which generates the system output: $y_k = Q(q)u_k + H(q)e_k$. Furthermore the system input is formed by the feedback law $u_k = r_k - y_k$ (see Fig.3.1), and r_k and e_k are zero mean white noise signals with the variances: $\text{var}(r_k) = \text{var}(e_k) = 1$. In this example we try to identify the poles of the combined system, hence $p_1 = 0$ and $p_2 = 3$. Note that only this combined system is in state space model (*i.e.* ARMAX) format. We will use the $PBSID_{opt}$ algorithm to estimate the pole positions of this system for different values of p (and f), as presented in Figure 3.4. Since, the system has no direct feedthrough, we have imposed the condition $D = 0$ in the identification problem. It should be noted that it is crucial that we impose this constraint, so unbiased estimates can be found in this example. This because the noise model has a delay (*i.e.* pole at $z = 0$), so a delay in the inputs is not a sufficient condition/assumption for the uncorrelated future input-noise. In this example, the parameterization is performed in two different manners, both based on the estimated state sequence. The C matrix is obtained with (3.15), whereas the A and B matrices are

determined with (3.18) according to [51] and (3.17) according to [58]. From the figure it is evident that, when we impose the $D = 0$ constraint, the parameterization method in (3.17) gives unbiased results, whereas the method in (3.18) gives biased results. The former obtains unbiased results because it reuses the estimated innovation from the output equation, which still holds the information that $D = 0$. Because of this characteristic, only (3.17) will be used for the state-based parameterization method in the remainder of the report.

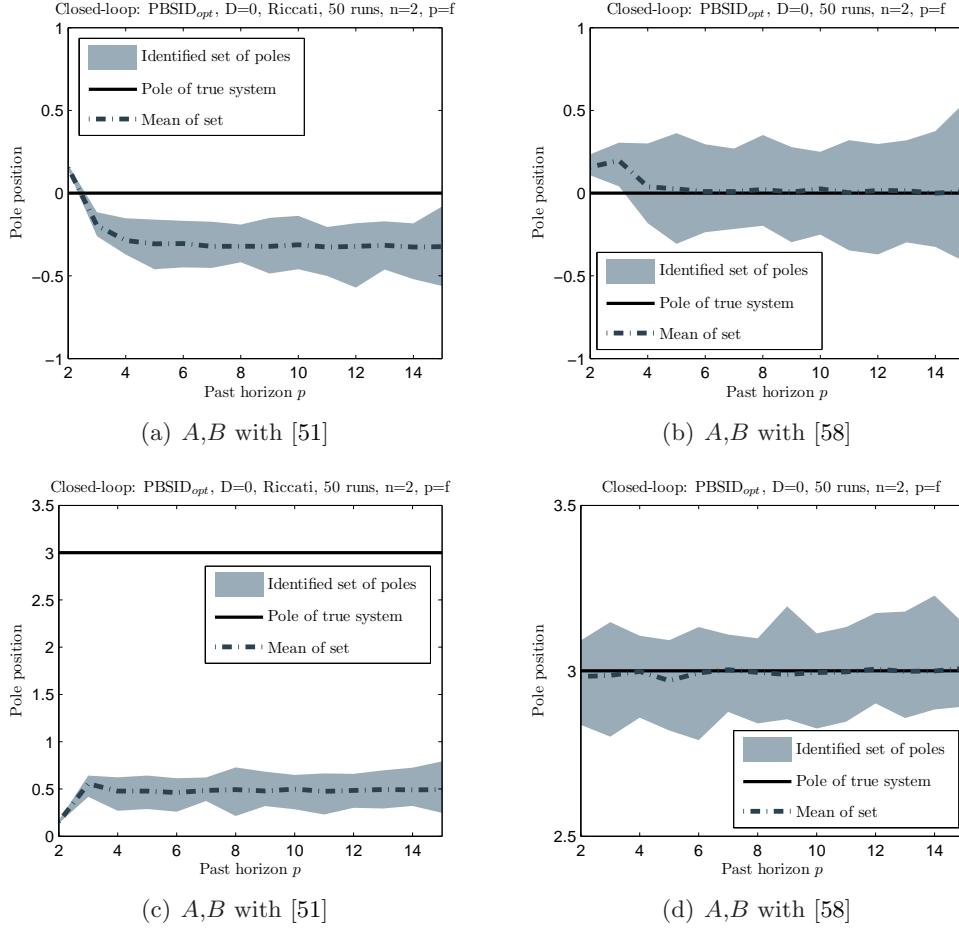


Figure 3.4: Pole estimates over 50 Monte Carlo simulations for PBSID_{opt} using the parameterization methods in [58] and [51] (denoted with 'Riccati'). Simulation example 2 from [62].

Example II. One of the main advantages of the SMI methods that use the predictor form is their applicability to identify unstable open-loop systems [62]. These algorithms define the identification problem in terms of $\bar{A} = A - KC$, which can still be stable if A is unstable. In [62], Chiuso and Picci show that this makes these methods very advantageous over, for example, the innovation form procedure IEM/Parsim-E. The IEM_{opt} algorithm, presented in this section, also obtains a solutions in the innovation-form. The least squares HOARX problem, from which IEM_{opt} essentially obtains it solution, consists however of Markov parameters, which contain the \bar{A} matrix. It is thus expected that IEM_{opt} also works well for unstable open-loop systems. In Figure 3.5(a)-3.5(b) we provide the same simulation as Example I for the IEM_{opt} algorithm. Indeed, from this result it is evident that the IEM_{opt} procedure can also handle unstable plants. Comparing the results in Figure 3.5 with

Figure 3.20(c) and 3.20(d), it is easily noticeable that PBSID_{opt} and IEM_{opt} find almost identical estimates for the pole at $z = 3$. However, for the pole at $z = 0$, IEM_{opt} performs somewhat worst than PBSID_{opt} . PBSID_{opt} only has a bias for small p , but becomes unbiased when p gets larger. This in contrast to IEM_{opt} , which has its best solution for $p = f = 4$, but gives biased estimates for higher values of p . The variance of the estimates for the pole at $z = 0$ is also larger for IEM_{opt} compared to PBSID_{opt} .

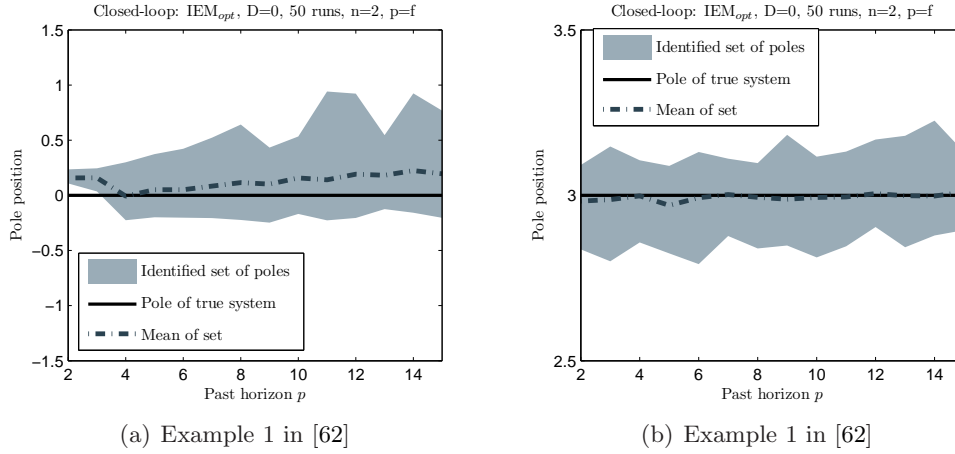


Figure 3.5: Pole estimates over 50 Monte Carlo simulations for IEM_{opt} (example 2 in [62]).

Example III. The article that provided the simulations in Example I and II, *i.e.* [62], also includes a simulation example of the instable plant, for the case of a filtered reference. This example, thus example 3 from [62], uses the model in (3.35), and a reference signal that is given by:

$$r_k = \frac{2}{10} \cdot \frac{z + 0.999}{z - 0.99} \cdot r'_k,$$

with r'_k a zero mean white noise signal with variance $\text{var}(r'_k)=1$. The power spectrum of this signal is given in Figure 3.6, which indicates that the signal has very poor power characteristics in the high frequency regions. With regard to model identification this makes it more likely to obtain biased estimates for the higher frequencies. Meaning, the reference signal does not contain enough power to excite the dynamics (in the form of system poles and zeros) at these high frequencies. In Figure 3.7(c)-3.7(d) the, with the IEM_{opt} algorithm, identified system poles, are shown for different past and future horizons ($p = f$). Furthermore, Figure 3.7 gives the PSBID_{opt} results run for the same simulations. When we compare both methods for this situation, we notice the opposite than for the previous simulation. For the filtered reference signal, the IEM_{opt} algorithm is able to estimate the poles very appropriately for high $p = f$ values, whereas PSBID_{opt} identifies decent pole positions for small $p = f$, but obtains biased results for large $p = f$ values. In Figure 3.8, we compared the identified model with the true system for both the PSBID_{opt} and IEM_{opt} algorithms. We have chosen a past and future horizon of $p = f = 3$. It can be seen from the figure that the PSBID_{opt} has a bias for high frequencies due to the filtered zero mean white noise reference. Remarkably the IEM_{opt} algorithm is able to provide a good estimate in this high frequency region, but at the cost of a worst estimate for the system poles (see Figure 3.7(c)). It should be noticed that, because of the filtered reference signal in this example, the variance at the

high frequencies is very large. This variance even gets larger as $p = f$ increases, since more parameters need to be estimated in the HOARX problem.

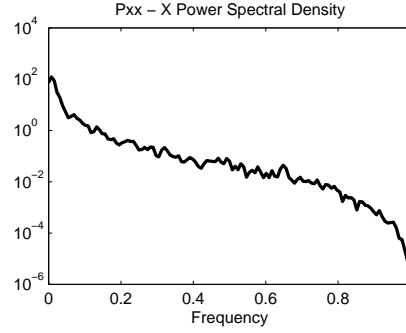


Figure 3.6: Power spectrum of the reference signal in Example III, that corresponds to example 3 in [62]. Since the sample time is $h = 1$, so the frequency is represented in Hz.

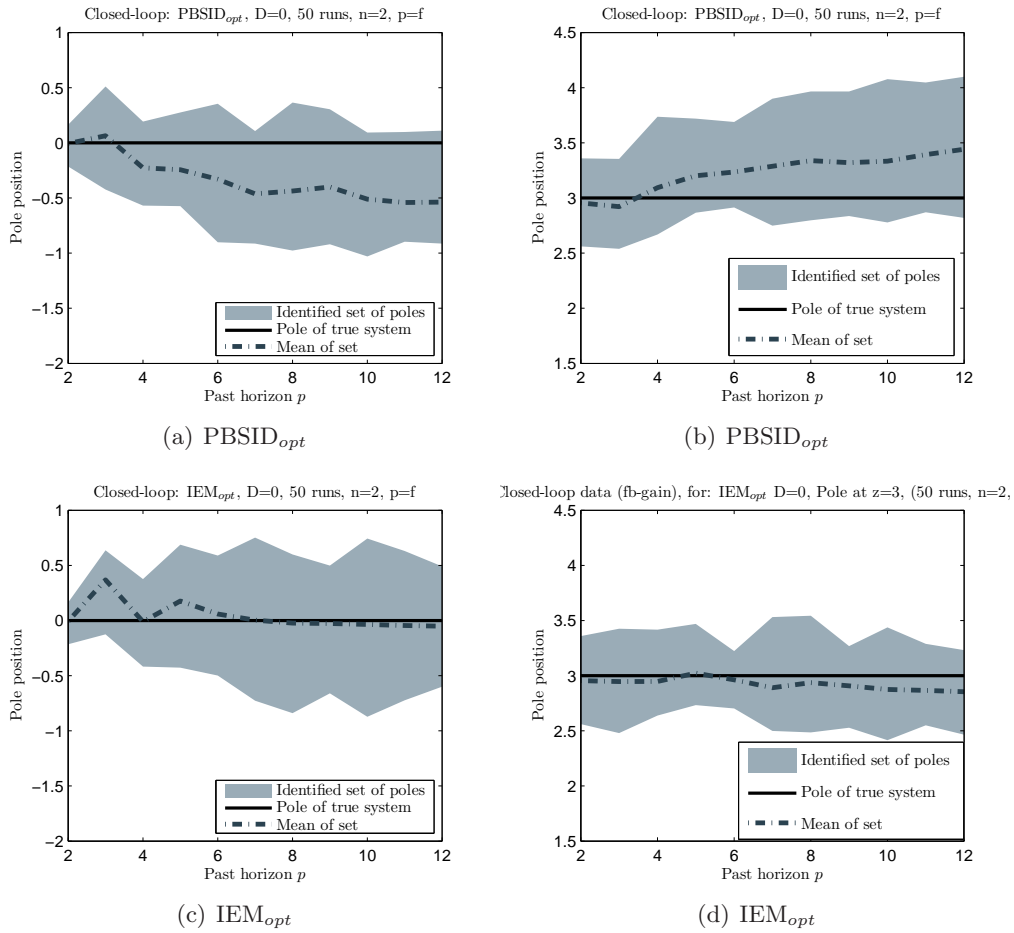


Figure 3.7: Pole estimates over 50 Monte Carlo simulations, for PBSID_{opt} and for IEM_{opt} , conditions from example 3 in [62].

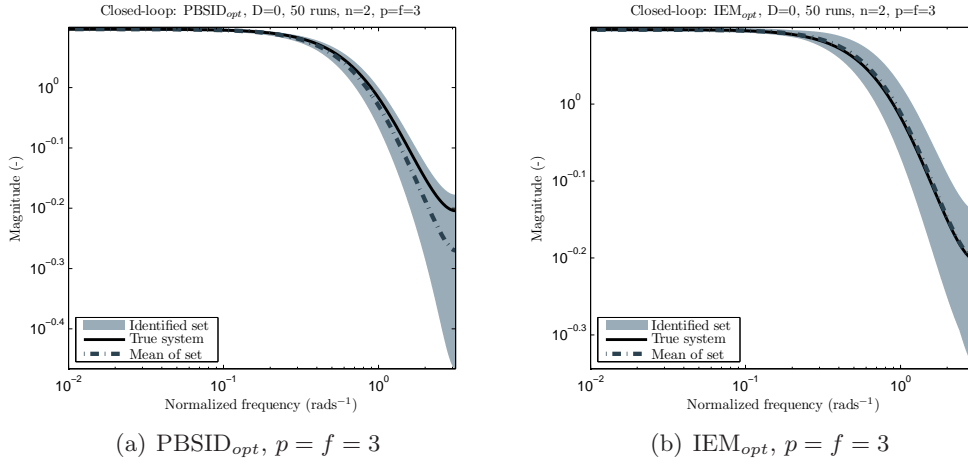


Figure 3.8: Frequency response of estimated model over 50 Monte Carlo simulations for PBSID_{opt} and for IEM_{opt}, example 3 provided in [62].

ARXISS: A novel SMI algorithm

At this stage we performed stochastic least squares to obtain a HOARX model estimate using input-output data, from which the state can be estimated using the PBSID_{opt} and new IEM_{opt} procedures. However, for these SMI methods, yet another set of stochastic least squares problems, based on the input-output data, need to be solved to obtain the A , B , C , and D system matrices using this state estimate. A MOESP like approach to obtain the system matrices, using (3.34), (3.31) and (3.32), gives the possibility to obtain the system matrices without solving any additional least squares problems based on input-output data. This can be advantages for a recursive implementation, since it only needs to recursively update one (*i.e.* HOARX) least squares problem.

This novel identification procedure is abbreviated with ARXISS, which stands for Auto-Regressive eXogenous inputs Innovation-based SubSpace. It seems to yield less computational complexity in obtaining the state-space matrices. Nevertheless, it must be noted that compared to a recursive implementation of PBSID_{opt}, a recursive implementation of the ARXISS procedure needs to perform the additional series expansions in (3.31), (3.32) and (3.33). The method that thus yields the least computational effort for a recursive implementation remains to be tested. Recursive SMI is however not part of the thesis, and as such the potential of the new ARXISS method for recursive implementation remains to be seen.

Corresponding to (3.12a), an SVD is performed on the solution in (3.34). From this Singular Value Decomposition the reduced-order observability matrix can be obtained by means of (3.10a), but omitting the hats:

$$\Gamma_{f \times n} = U_n = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{f-1} \end{bmatrix}. \quad (3.36)$$

From (3.36), the C matrix can be obtained very easily by taking the first n_y rows of $\Gamma_{f \times n}$, *i.e.* $\hat{C} = U_n(1 : n_y, 1 : n)$. The A matrix is obtained using (3.36) and solving the following

equation [51]:

$$\hat{A} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{f-2} \end{bmatrix}^\dagger \cdot \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{f-1} \end{bmatrix}. \quad (3.37)$$

Traditionally the MOESP procedure obtains the B and D matrices using an additional linear least squares regression problem. In this problem, the observability matrix is constructed using the obtained A and C matrices and the bank of predictors in (3.9) is solved for B and D and the initial state. For a more detailed description of the corresponding calculations, the reader is referred to [51], which in the simulation study is denoted with `dac2b(d)`. Since the series expansion in (3.32) and (3.31) provide $\Lambda_0^B = \hat{D}$, $\Lambda_i^B = CA^{i-1}B$, and $\Lambda_i^K = CA^{i-1}K$, based on the estimated Markov parameters, accurate estimates of the B D and K matrices can be found. Using this idea, one can use the $\Lambda_i^B = CA^{i-1}B$ and $\Lambda_i^K = CA^{i-1}K$ terms to obtain:

$$\hat{B} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{f-1} \end{bmatrix}^\dagger \cdot \begin{bmatrix} \Lambda_1^B \\ \Lambda_2^B \\ \vdots \\ \Lambda_f^B \end{bmatrix}, \quad \hat{K} = \begin{bmatrix} C \\ CA \\ \vdots \\ CA^{f-1} \end{bmatrix}^\dagger \cdot \begin{bmatrix} \Lambda_1^K \\ \Lambda_2^K \\ \vdots \\ \Lambda_f^K \end{bmatrix}. \quad (3.38)$$

The ARXISS subspace model identification method is shown schematically in Figure 3.3 and is summarized with Algorithms 3.

Algorithm 3 ARXISS

1. Construct the matrices $Y_{p,\mathcal{N}-p}$, $U_{p,\mathcal{N}-p}$, and Z_{P+} according to (3.26).
2. Solve the HOARX problem in (3.25), if desired imposing the constraint $D = 0$. The solution in (3.27) immediately gives an estimate of D .
3. Construct $\widehat{\Gamma\hat{K}}$ according to (3.30) using the Markov parameters from the HOARX problem solution in (3.27).
4. Transform (3.30) to the innovation form, which gives $\widehat{\Gamma\hat{K}}$, using (3.33) and (3.34).
5. Determine the reduced-order observability matrix, $\hat{\Gamma}_{f \times n}$, using (3.12a) and (3.36).
6. Determine the estimate C by taking the first n_y rows of $\hat{\Gamma}_{f \times n}$, and the an estimate of the A system matrix with the relation in (3.37).
7. Construct Λ_i^B and Λ_i^K , for $i = 1, \dots, f$, according to (3.32) and (3.31), and compute the B and K system matrices with (3.38).

Example IV. In Example I and II, simulation results of an unstable system were provided for the PBSID_{opt} and IEM_{opt} methods. The ARXISS procedure is also implemented on this model in (3.35), with the results shown in Figure 3.9. Compared to the IEM_{opt} procedure, the ARXISS method gives almost identical estimates. It should however be noted that, for $p = f = 2$, the ARXISS method found pole estimates on the imaginary axis, and as such not shown in the figure (since it only provides real valued system poles). This example, and the examples I-II, only provide estimates of the eigenvalues (*i.e.* poles) of the A matrix.

Since the ARXISS algorithm estimates A directly from the extended observability matrix, this methodology seems very sufficient for this situation. Nevertheless, the ARXISS (and IEM_{opt}) algorithm have some troubles to estimate the pole from the noise model, so the PBSID_{opt} might be a better choice if an accurate estimate of this pole is desired.

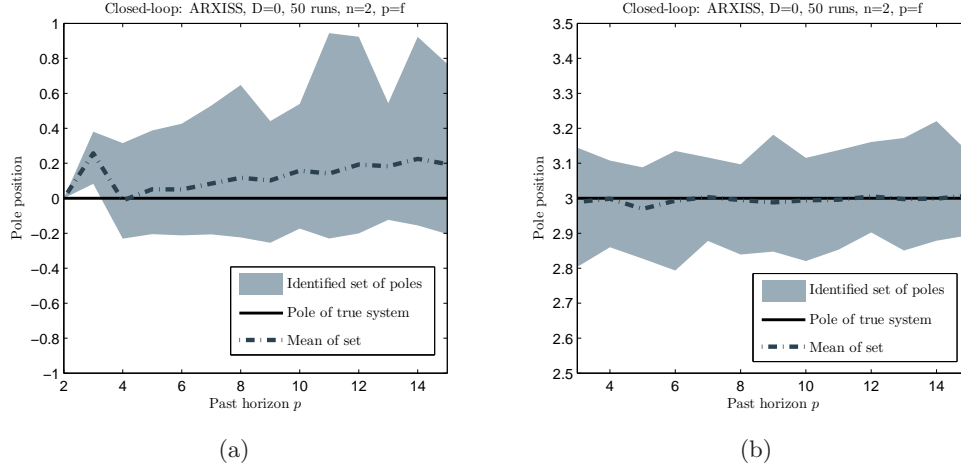


Figure 3.9: Pole estimates over 50 Monte Carlo simulations for ARXISS (example 2 in [62]).

Example V. To determine the A and C matrices, the ARXISS algorithm uses the observability matrix like MOESP. The B and D matrices are however not estimated in a manner that correspond to the MOESP procedure. ARXISS uses the series expansions in (3.32) and (3.31), whereas traditionally, the MOESP procedure uses the earlier described `dac2b(d)` MATLAB code that accompanies [51]. To make a comparison, we apply both methodologies to the ARXISS method and identify the first order model presented in [63] under closed-loop conditions. In [63] the following state space model is considered:

$$\begin{bmatrix} x_{k+1} \\ y_k \end{bmatrix} \begin{bmatrix} A & B & K \\ C & D & I \end{bmatrix} = \begin{bmatrix} 0.5 & 1 & 1 \\ 1 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_k \\ u_k \end{bmatrix}. \quad (3.39)$$

The example in [63] is officially used in an open-loop setting, however we apply it under closed-loop conditions with the feedback mechanism $u_k = r_k - y_k$, where the reference is given by:

$$r_k = \frac{z^2 + 0.8z + 0.55}{z^2 - 0.5z + 0.9} \cdot r'_k,$$

with r'_k and e_k defined as zero mean white noise signals with variance $\text{var}(e_k) = \text{var}(r'_k) = 1$. The power spectrum of this signal is given in Figure 3.10, which shown more weight on the import frequencies of the model. Since we again constraint $D = 0$, we use the existing `dac2b` code. The simulation results are shown in Figure 3.11. From these results it is noticeable that the novel ARXISS methodology to determine B and D outperforms the traditional approach for this particular simulation example. More simulations based on this simulation model will follow in other examples in this chapter.

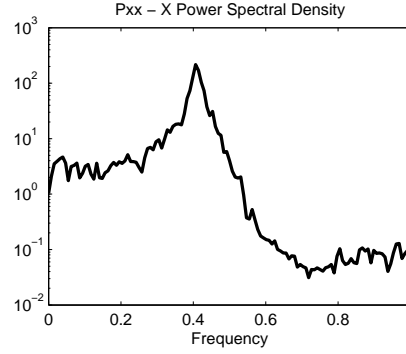


Figure 3.10: Power spectrum of the reference signal in Example V. Since the sample time is $h = 1$, so the frequency is represented in Hz.

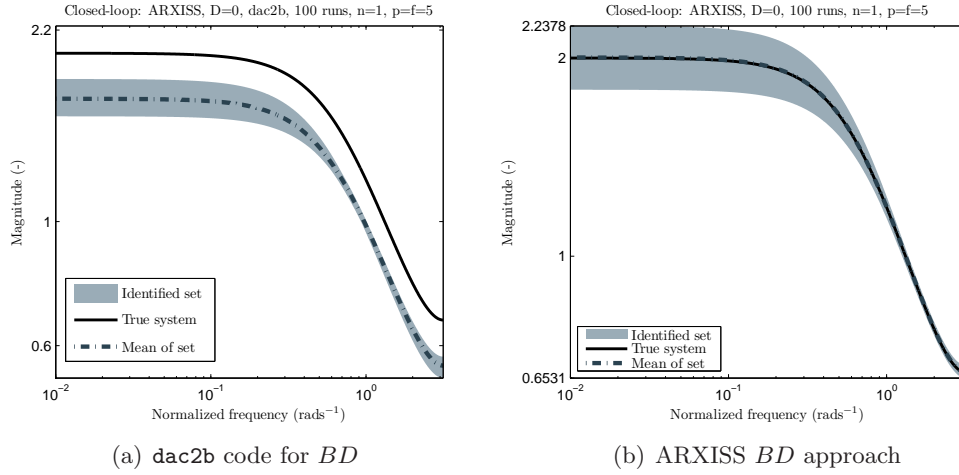


Figure 3.11: Frequency response estimated model over 100 Monte Carlo runs, using example in [63] under closed-loop conditions.

3.3.2 Pre-estimate based SMI: double-stage procedures

This section re-uses particular estimates from the previous single-stage step of the closed-loop identification framework, as shown in Figure 3.21. In Section 3.3.1, we estimated a HOARX model, a stage at which the pre-estimates $\hat{\hat{H}}_f$, $\hat{\hat{G}}_f$, \hat{H}_f , \hat{G}_f , and \hat{E}_f can be formed (see Fig. 3.3). The ‘unifying’ closed-loop SMI framework handles three approaches that use combinations of these pre-estimates. It starts with the methods that use the earlier obtained pre-estimates of the Toeplitz matrices for a system represented with the predictor form and innovation form, respectively. Then, it continues with a MOESP like algorithm that uses the pre-estimate of the innovation block Hankel matrix E_f .

CVA based methods for a system in predictor form

With the $\hat{\hat{H}}_f$ and $\hat{\hat{G}}_f$ pre-estimates from the HOARX estimation step, and defining the system in predictor form (3.11), the following can be deduced:

$$Y_{f-} = Y_f - \hat{\hat{H}}_f U_f - \hat{\hat{G}}_f Y_f = \bar{\Gamma} \bar{K} Z_p + E_f. \quad (3.40)$$

These pre-estimates make the future input-output terms and the innovation uncorrelated, so the linear least squares solution is unbiased for closed-loop data. Regarding this approach, a Canonical Variance Analysis (CVA) between Y_{f-} and Z_p can be considered to find an estimate of $\bar{\Gamma}\bar{\mathcal{K}}$. Since this subsection only provides the solution with respect to the closed-loop subspace model identification problem, we refer the reader to [50, 56, 64, 23] for more detailed information on the CVA procedure. Performing a CVA on (3.40) yields the following solution [60]:

$$\{\widehat{\bar{\Gamma}\bar{\mathcal{K}}}\}_{\text{SSARX}} = \left(Y_{f-} \cdot Y_{f-}^T\right)^{-1/2} \cdot Y_{f-} \cdot Z_p^T \cdot (Z_p \cdot Z_p^T)^{-1/2}, \quad (3.41)$$

on which an SVD is performed. It should be noted that a SVD on (3.41) corresponds to a Generalized Singular Value Decomposition (GSVD) on the stochastic linear least squares solution of (3.40), thus:

$$W_1 \cdot Y_{f-} \cdot Z_p^T \cdot (Z_p \cdot Z_p^T)^{-1} \cdot W_2 = U_n \Sigma_n V_n^T, \quad (3.42)$$

with the weights:

$$W_1 = \left(Y_{f-} \cdot Y_{f-}^T\right)^{-1/2},$$

$$W_2 = (Z_p \cdot Z_p^T)^{1/2}.$$

The Markov state sequence estimate, which can be determined based on the GVSD in (3.42) and the relation in (3.23a), is then used to obtain the system's state space matrices as described in Section 3.1.2. The SSARX subspace identification procedure is summarized in Algorithm 4, whereas Figure 3.12 provides a schematic representation.

Algorithm 4 SSARX

1. Construct the matrices $Y_{p,\mathcal{N}-p}$, $U_{p,\mathcal{N}-p}$, and Z_{P+} according to (3.26).
2. Solve the HOARX problem in (3.25), if desired imposing the constraint $D = 0$.
3. Construct the predictor-based Toeplitz matrices (i.e. $\hat{\hat{H}}_f$ and $\hat{\hat{G}}_F$), as stated in (3.29), using the Markov parameters from the HOARX problem solution in (3.27).
4. Construct the matrices Y_f , U_f , and Z_P according to (3.3), as well as Y_{f-} formulated in the problem in (3.40).
5. Perform a CVA between Y_{f-} and Z_P , hence solving the GSVD given in (3.42), and determine the Markov state estimate with (3.23a).
6. Obtain an estimate of the A , B , C , D and K system matrices by solving the least squares problems in (3.15) and (3.17), if desired constraining $D = 0$.

CVA based methods for a system in innovation form

An identical procedure as in the previous subsection can be performed for a system represented in the innovation form (3.9), as illustrated in Figure 3.12. In the literature there however exists only one method that uses pre-estimates for a system defined in innovation form in the same double-stage manner, namely Larimore's CVA subspace identification

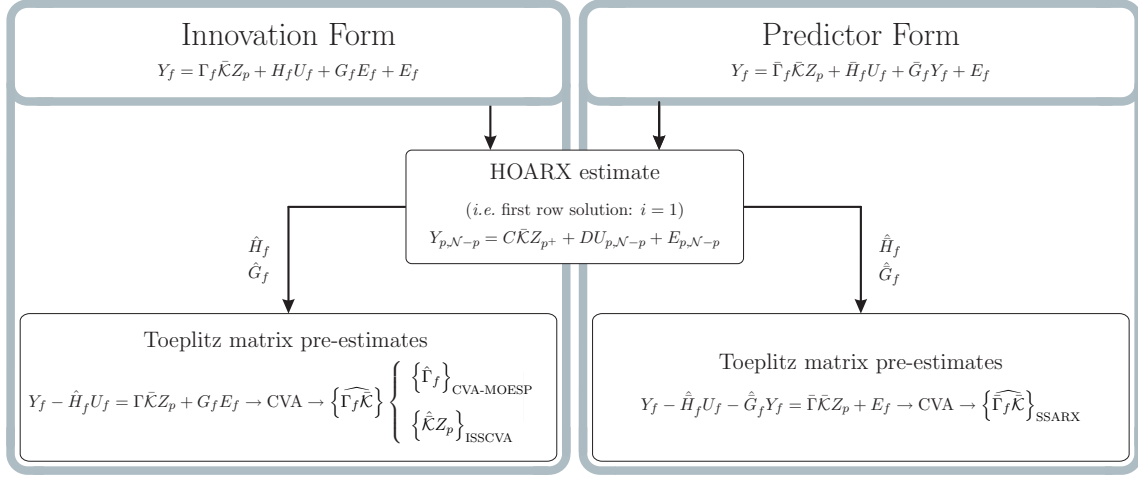


Figure 3.12: Overview of the Canonical Variate Analysis (CVA) based double-stage procedures.

method that uses a HOARX pre-estimate [60, 64, 58]. This method, referred to in this report as CLCVA, is suitable for closed-loop identification, but uses pre-estimates defined in the predictor form. With the pre-estimate provided in (3.29a), this procedure formulates the following:

$$Y_{f--} = Y_f - \hat{H}_f U_f = \Gamma_f \bar{K} Z_p + G_f E_f, \quad (3.43)$$

where a Canonical Variate Analysis (CVA) is performed between Y_{f--} and Z_p . The pre-estimate used in (3.43) is however inconsistent with the underlying system representation. It is therefore more correct to use the innovation form Toeplitz matrices constructed from the relations presented in (3.32), which consequently results in one of the new closed-loop SMI algorithms formed in this thesis and shown in Figure 3.12. As such, using the constructed \hat{H}_f pre-estimate, one can reformulate (3.9) into:

$$Y_{f-} = Y_f - \hat{H}_f U_f = \Gamma_f \bar{K} Z_p + G_f E_f, \quad (3.44)$$

where, for convenience, we have chosen the same terminology as (3.40). Performing a CVA between Y_{f-} and Z_p from (3.44) yields the following solution:

$$\{\hat{\Gamma}_f \bar{K}\}_{\text{ISSCVA/CVA-MOESP}} = (Y_{f-} \cdot Y_{f-}^T)^{-1/2} \cdot Y_{f-} \cdot Z_p^T \cdot (Z_p \cdot Z_p^T)^{-1/2}, \quad (3.45)$$

on which we apply an SVD to determine a reduced-order solution. Of course, this relates to the weighted SVD given in (3.42). The Markov state estimate, $X_{p,N}$, or the reduced observability matrix, $\hat{\Gamma}_{f \times n}$ can be distilled according to (3.23) thereafter:

- The approach that finds the state space matrices through the Markov state estimate, denoted with ISSCVA, is in methodology an exact mapping of the SSARX method to the innovation form based methods. The ISSCVA subspace model identification procedure is summarized in Algorithm 5.
- Additionally, using the reduced observability matrix the same parameterization procedure can be used as for the ARXISS method presented in Algorithm 3. Note that

from the constructed \hat{H}_f Toeplitz matrix the first column of \hat{H}_f can be reused to determine the B system matrix. The CVA-MOESP subspace model identification method is summarized in Algorithm 6.

Algorithm 5 ISSCVA

1. Construct the matrices $Y_{p,N-p}$, $U_{p,N-p}$, and Z_{P+} according to (3.26).
2. Solve the HOARX problem in (3.25), if desired imposing the constraint $D = 0$.
3. Compute the $\Lambda_i^B = CA^{i-1}B$ terms for $i = 1, \dots, f-1$, using the Markov parameters from the HOARX problem solution in (3.27), and construct the \hat{H}_f Toeplitz matrix according to (3.10b).
4. Construct the matrices Y_f , U_f , and Z_P according to (3.3), as well as Y_{f-} formulated in the problem in (3.44).
5. Perform a CVA between Y_{f-} and Z_P , hence solving the GSVD given in (3.42), and determine the Markov state estimate with (3.23a).
6. Obtain an estimate of the A , B , C , D and K system matrices by solving the least squares problems in (3.15) and (3.17), if desired constraining $D = 0$.

Algorithm 6 CVA-MOESP

1. Construct the matrices $Y_{p,N-p}$, $U_{p,N-p}$, and Z_{P+} according to (3.26).
2. Solve the HOARX problem in (3.25), if desired imposing the constraint $D = 0$. The solution in (3.27) immediately gives an estimate of D .
3. Compute the $\Lambda_i^B = CA^{i-1}B$ (and Λ_i^K) terms for $i = 1, \dots, f$, using the Markov parameters from the HOARX problem solution in (3.27), and construct the \hat{H}_f Toeplitz matrix according to (3.10b).
4. Construct the matrices Y_f , U_f , and Z_P according to (3.3), as well as Y_{f-} formulated in the problem in (3.44).
5. Perform a CVA between Y_{f-} and Z_P , hence solving the GSVD given in (3.42), and determine the reduced-order observability matrix, $\hat{\Gamma}_{f \times n}$, with (3.23b).
6. Determine the estimate of C by taking the first n_y rows of $\hat{\Gamma}_{f \times n}$, and the estimate of A , B and K system matrices with the relations in (3.37) and (3.38).

Example VI. In this example we will identify the pole of the first order plant model provided in [63]. Although the article uses this model in an open-loop setting, the model is simulated under closed-loop conditions, as provided in Example V. The example uses a filtered zero mean white noise reference signal, with its power spectrum shown in Figure 3.10. The filter gives a high magnitude around the system pole ($z = 0.5$), combined with a low-pass behavior. The results obtained with the CVA-based double-stage procedures are given in Figure 3.13. As shown in the figure, the SSARX and ISSCVA methods yield identical solutions to the closed-loop SMI problem. This is a remarkable result that shows that the underlying system representation is invariant for this particular situation where a predictor based algorithm is comparable in exact (CVA based) methodology to an innovation based

algorithm. CVA-MOESP gives totally different results compared to SSARX/ISSCVA. For Monte-Carlo simulations of this example, it is noticeable from the figure that the set of identified poles has more variance for the CVA-MOESP procedure.

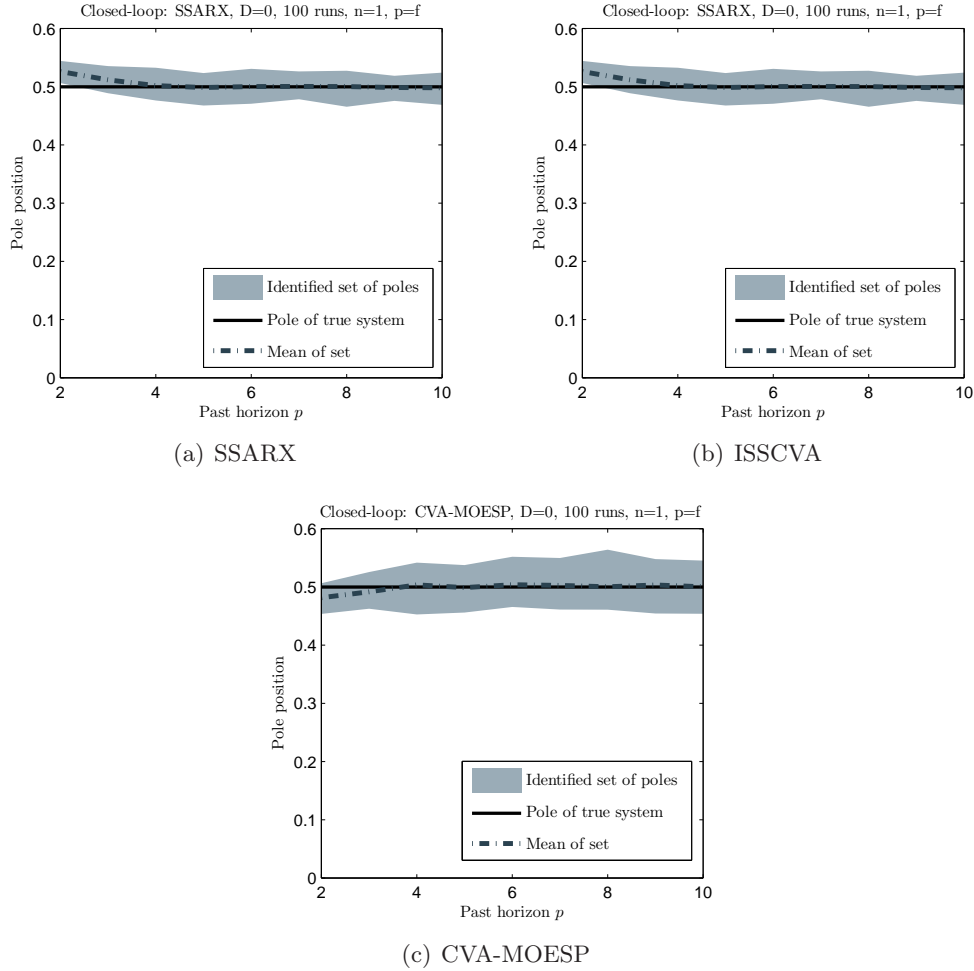


Figure 3.13: Pole estimates over 100 Monte Carlo simulations, using the first order model provided in [63] under closed-loop conditions.

Example VII. In Example VI, the SSARX/ISSCVA methods resulted in the exact same identified poles. Another example is given to provide some knowledge to the reader that the SSARX/ISSCVA consistently give the same model estimate³. Since the ISSCVA defined the identification problem in innovation form, one could ask if this method also works for unstable open-loop systems. For this reason we again consider the Monte Carlo simulations in Example III that used a (low-pass) filtered zero mean white noise reference signal. The effect this filter makes it very likely for the SMI algorithms to have estimates with a bias in the high frequencies (see Fig. 3.6). The simulation results are shown in Figure 3.14. As can be seen from the figures, the SSARX and ISSCVA procedures give again identical pole estimates. Moreover, for this simulation example, it is easily noticeable from the figures that the CVA-MOESP algorithm has a much smaller variance for the identified set of poles

³At least, for the many simulations run during the thesis

at $z = 3$, and a slightly higher variance for $z = 0$ at $p = 3, \dots, 5$, than SSARX/ISSCVA. Furthermore, it also has less biasedness with larger values of p .

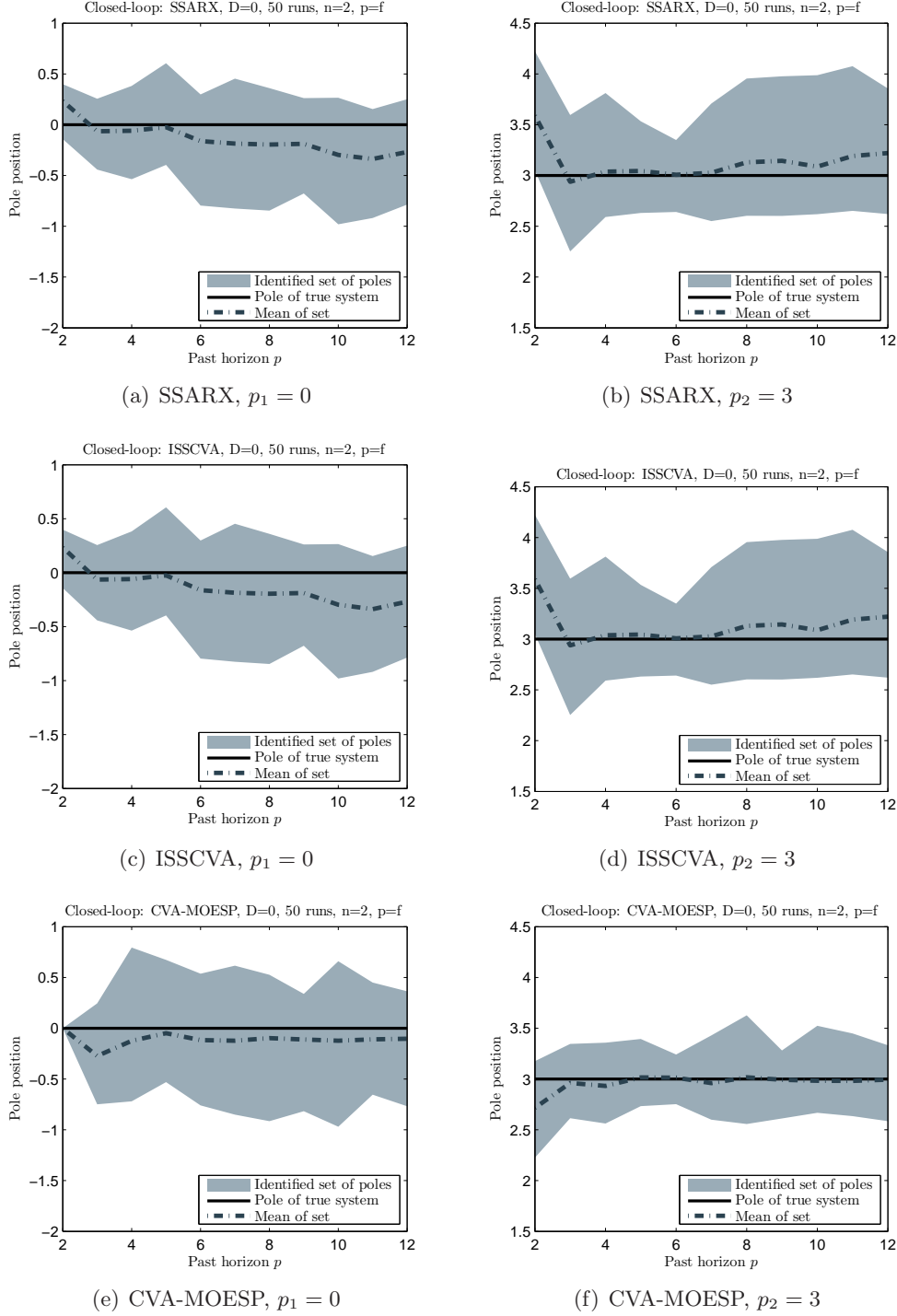


Figure 3.14: Pole estimates over 50 Monte Carlo simulations, using the simulation example 3, provided in [62].

Novel closed-loop MOESP method using the \hat{E}_f pre-estimate

This subsection presents a novel subspace model identification approach that re-uses the estimated innovation block Hankel matrix formed in Section 3.3.1. This method is inspired by the MOESP procedure in [51], which solves the identification problem using a numerical efficient LQ factorization, combined with the direct parameterization procedure illustrated in Figure 3.2. It should be noted that unlike the multi-stage IEM1 method, this approach uses the entire pre-estimated innovation sequence, and yields an entirely different solution to the closed-loop subspace model identification problem. This novel algorithm is denoted with the abbreviation IE-MOESP, which stands for Innovation Estimate Multivariable Output-Error State sPace.

A \hat{E}_f pre-estimate was constructed using $\hat{E}_{p,N-p}$ obtained from the HOARX problem in (3.25), which more faithful to the MOESP methodology can also be computed using the LQ factorization in (3.51). Using this pre-estimate, combined with the bank of predictors defined in innovation form (3.9), we can formulate the following:

$$Y_f = \Gamma_f X_{p,N} + \begin{bmatrix} H_f & (G_f + I) \end{bmatrix} \cdot \underbrace{\begin{bmatrix} U_f \\ \hat{E}_f \end{bmatrix}}_{\mathcal{W}}. \quad (3.46)$$

Due to the characteristic $\mathcal{W} \cdot \Pi_{\mathcal{W}}^\perp = 0$, described in (3.16), the deterministic bank of predictors in (3.46) reduce to:

$$Y_f \Pi_{\mathcal{W}}^\perp = \Gamma_f X_{p,N} \Pi_{\mathcal{W}}^\perp. \quad (3.47)$$

Assuming that $\text{range}(Y_f \Pi_{\mathcal{W}}^\perp) = \text{range}(\Gamma_f)$, then $Y_f \Pi_{\mathcal{W}}^\perp$ can be used to determine the reduced observability matrix by taking a Singular Value Decomposition (SVD). Like the MOESP procedure we can use the LQ decomposition:

$$\begin{bmatrix} \begin{bmatrix} U_f \\ E_f \\ Y_f \end{bmatrix} \end{bmatrix} = \begin{bmatrix} R_{11} & 0 & 0 \\ R_{21} & R_{22} & 0 \end{bmatrix} \cdot \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \end{bmatrix}, \quad (3.48)$$

to compute $Y_f \Pi_{\mathcal{W}}^\perp$ in a computationally efficient/robust manner. This approach does not require a matrix inverse to be computed, since it uses the orthogonal property of the R_{ij} and Q matrices in the LQ decomposition. From the orthogonality of these matrices it follows that $R_{ij} R_{ij}^T = R_{ij}^T R_{ij} = I$, and:

$$\begin{aligned} Q^T Q &= I \rightarrow Q_1^T Q_1 + Q_2^T Q_2 + Q_3^T Q_3 = I, \\ Q Q^T &= I \rightarrow \begin{cases} Q_i Q_j^T = 0, & \text{for } i \neq j \\ Q_i Q_j^T = I, & \text{for } i = j \end{cases}. \end{aligned}$$

Using the LQ decomposition in (3.48) one can formulate:

$$\begin{aligned} \Pi_{\mathcal{W}}^\perp &= I - \mathcal{W}^T (\mathcal{W} \mathcal{W}^T)^{-1} \mathcal{W}, \\ &= I - Q_1^T R_{11}^T (R_{11} Q_1 Q_1^T R_{11}^T)^{-1} R_{11} Q_1, \\ &= I - Q_1^T Q_1 = Q_2^T Q_2 + Q_3^T Q_3. \end{aligned}$$

Therefore, with $Y_f = R_{21}Q_1 + R_{22}Q_2$, we find the left-side term in (3.47) to be equal to:

$$\begin{aligned} Y_f \Pi_{\mathcal{W}}^\perp &= (R_{21}Q_1 + R_{22}Q_2) \cdot (Q_2^T Q_2 + Q_3^T Q_3), \\ Y_f \Pi_{\mathcal{W}}^\perp &= R_{22}Q_2. \end{aligned} \quad (3.49)$$

With the characteristic (Q_2 has full row rank):

$$\text{range}(Y_f \Pi_{\mathcal{W}}^\perp) = \text{range}(\Gamma_f) = \text{range}(R_{22}Q_2) = \text{range}(R_{22}),$$

the reduced observability matrix, $\Gamma_{f \times n}$, can be found performing an SVD on:

$$R_{22} = U \Sigma V \approx U_n \Sigma_n V_n, \quad \Gamma_{f \times n} = U_n. \quad (3.50)$$

From (3.50), the C matrix can be obtained very easily by taking the first n_y rows of $\Gamma_{f \times n}$, i.e. $\hat{C} = U_n(1 : n_y, 1 : n)$, whereas the A matrix is obtained according (3.37). Traditionally the MOESP procedure obtains the B and D matrices using an additional linear least squares regression problem. In this problem, the observability matrix is constructed using the obtained A and C matrices and the bank of predictors in (3.9) is solved for B and D and the initial state, see [51]. The original approach, which has earlier been denoted with the name of the corresponding MATLAB function `dac2b(d)`, can also be used with the estimated innovation sequence. Providing this estimated innovation sequence, $\hat{E}_{p, \mathcal{N}-p}$, as a second system input to the existing `dac2b(d)` code, we are able to find accurate estimates of the B , D and K system matrices. As indicated in, this novel approach results in better estimates than the original intended use of the `dac2b(d)` code.

The IE-MOESP procedure, summarized in Algorithm 7, is presented schematically in Figure 3.15. It shows this exact methodology only to be applicable for a system representation in innovation form. As a small side note, the pre-estimate $\hat{E}_{p, \mathcal{N}-p}$ can also be determined more elegantly following the MOESP methodology with:

$$\begin{bmatrix} \begin{bmatrix} Z_{p+} \\ U_{p, \mathcal{N}-p} \\ Y_{p, \mathcal{N}-p} \end{bmatrix} \end{bmatrix} = \begin{bmatrix} R_{11} & 0 & 0 \\ R_{21} & R_{22} & 0 \end{bmatrix} \cdot \begin{bmatrix} Q_1 \\ Q_2 \\ Q_3 \end{bmatrix} \Rightarrow \hat{E}_{p, \mathcal{N}-p} = R_{22}Q_2, \quad (3.51)$$

Algorithm 7 IE-MOESP

1. Construct the matrices $Y_{p, \mathcal{N}-p}$, $U_{p, \mathcal{N}-p}$, and Z_{p+} according to (3.26).
2. Solve the HOARX problem in (3.25) for $\hat{E}_{p, \mathcal{N}-p}$, using the LQ factorization in (3.51), and construct \hat{E}_f according to (3.3).
3. Construct the matrices Y_f , U_f , and Z_P according to (3.3), and apply an LQ factorization as shown in (3.48).
4. Perform a SVD on R_{22} , and determine the reduced-order observability matrix, $\hat{\Gamma}_{f \times n}$, like in (3.50).
5. Determine the estimate of C by taking the first n_y rows of $\hat{\Gamma}_{f \times n}$, and the estimate of the A system matrix using (3.37).
6. Find the B and D and K matrices using the methodology provided in [51], using the existing `dac2b(d)` MATLAB code accompanying [51]. However, contrary to its ordinary use, provide the system input as $\begin{bmatrix} U_{p, \mathcal{N}-p}^T & \hat{E}_{p, \mathcal{N}-p}^T \end{bmatrix}^T$.

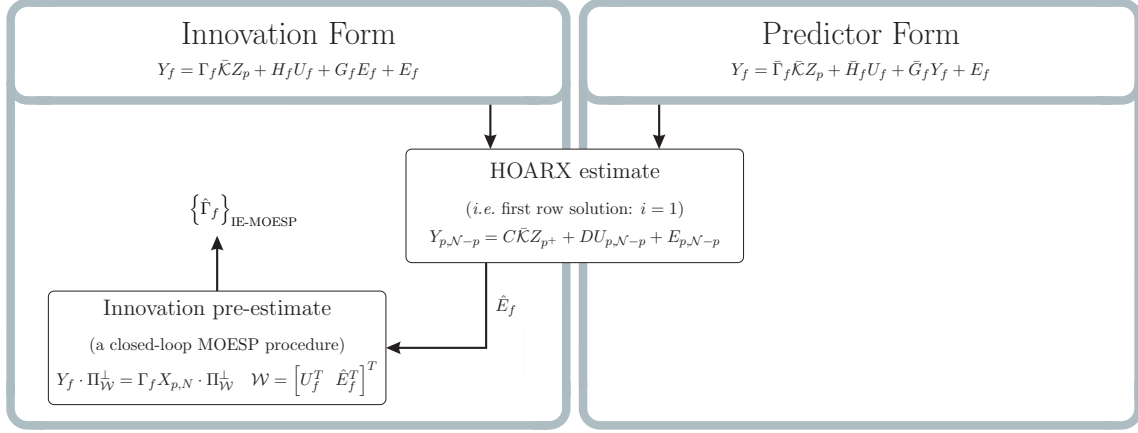


Figure 3.15: Innovation pre-estimate double-stage procedure that follows the MOESP approach.

Example VIII. Since, IE-MOESP uses the existing `dac2bd` MATLAB function to find the estimates of B and D , or `dac2b` for the situation when we impose $D = 0$, the IE-MOESP procedure can not be used for unstable open-loop systems. This is because `dac2bd` solves a bank of predictors that is defined in innovation form, which will consists of an unstable open-loop A matrix. Another methodology to obtain the B and D matrices is thus required to make the system suitable for open-loop unstable systems. In this example we use the same Monte Carlo simulations as Example V. First, the IE-MOESP identification procedure was performed for the original `dac2b` usage, and then following the manner in which IE-MOESP uses it. Both approaches are performed in the IE-MOESP algorithms for a comparison. As shown in Figure 3.16, the manner in which IE-MOESP uses the `dac2b` code gives much better estimates than its original use.

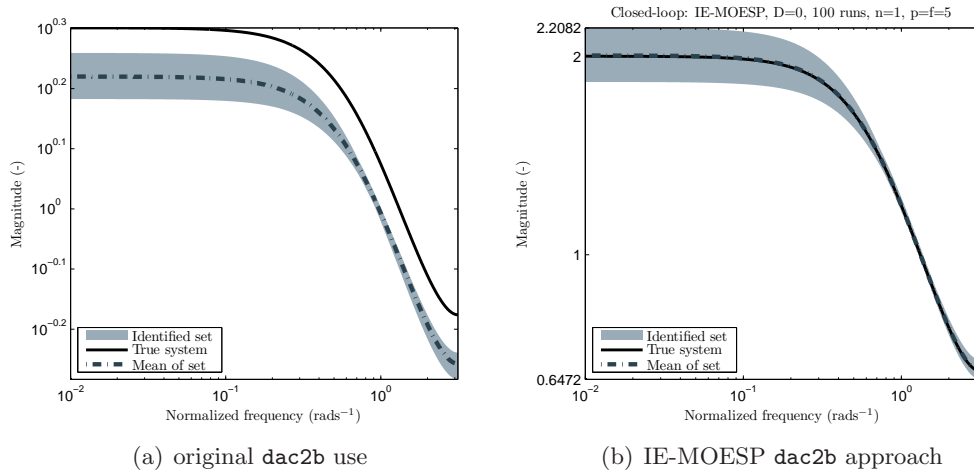


Figure 3.16: Frequency response estimated model over 100 Monte Carlo runs for IE-MOESP, using example in [63] under closed-loop conditions.

Example IX. To make a comparison, the $\text{PBSID}_{\text{opt}}$, IEM_{opt} , SSARX, ISSCVA, and CVA-MOESP algorithms are also applied to the same simulation as in Example VI and VIII. The

results are shown in Figure 3.17, whereas the ARXISS method for this simulation example was already given in Figure 3.11(b). All methods yield very nice estimates of the open-loop model. A comparison can however be made on the variance of the estimates, which corresponds to the identified model set. It should be noted that the difference between the ARXISS and CVA-MOESP estimates is only very minimal. The same is observed for the PBSID_{opt} and SSARX algorithms. When we compare IEM_{opt} with PBSID_{opt} , it is evident that the IEM_{opt} algorithm has slightly less variance in the low frequency regions, but that this comes at the cost of more uncertain high frequency estimates. A slightly larger variance is thus visible there for the high frequencies. In fact, this observation corresponds exactly to the simulations in example III and VII (note the variances in Fig. 3.8). Meaning that the transformation of the PBSID_{opt} method into the innovation form, using (3.33), has the same effect as applying a low-pass filter over the signals. Compared to the other methods described so far, the IE-MOESP estimate in Figure 3.16 corresponds the most to the other methods that follow the MOESP tradition in using the extended observability matrix. Here, the IE-MOESP algorithm yields some more certainty in the low frequencies, but also at the cost of more variance of the solutions in the high frequencies. This seems to favor the IE-MOESP approach a little, when compared to ARXISS and CVA-MOESP.

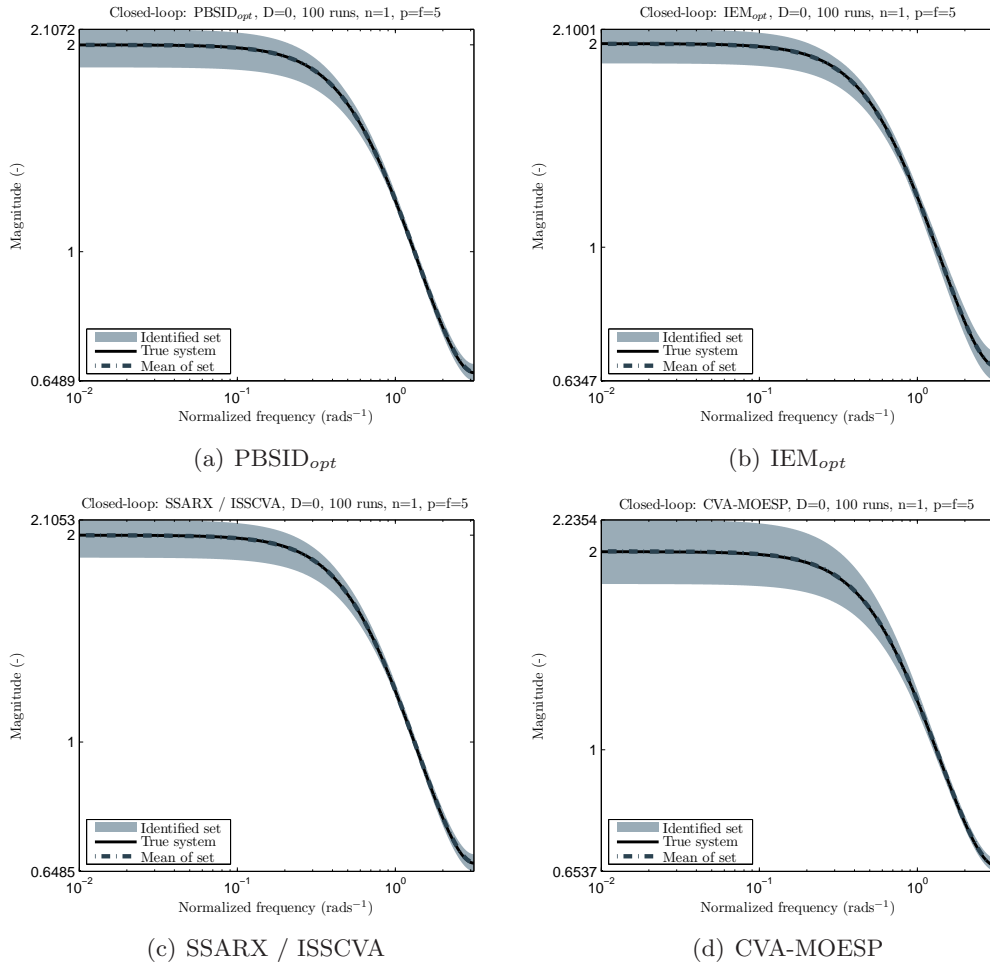


Figure 3.17: Frequency response estimated model over 100 Monte Carlo runs, using example in [63] under closed-loop conditions.

3.3.3 Multi-stage row-wise least squares estimates

In (3.9) and (3.11), H_f and \bar{H}_f are lower block triangular matrices making the model causal, a constraint normally not imposed in open-loop SMI [53]. To avoid inconsistencies when there is no proper excitation of the input, these terms should not contain these non-causal terms. Parsimonious identification algorithms, like the open-loop Parsim-P method [61], remove these non-causal effects by performing f linear least squares in parallel. This methodology is also especially useful in the closed-loop SMI framework. The Innovation Estimate Method (IEM1) of Qin and Ljung [61, 65, 58]⁴, and Predictor Based Subspace IDentification (PB-SID) of Chiuso and Picci [62]⁵ perform row-wise linear least squares on (3.9) and (3.11), respectively, to obtain closed-loop unbiased results under the assumption that a delay in u_k exists or $D = 0$. Since we obtained a solution for the first block row, *i.e.* $i = 1$, of (3.9) and (3.11) with the HOARX model estimate (Sec. 3.3.1), only the solutions for $i = 2, \dots, f$ need to be acquired here.

System in innovation form

The IEM1 algorithm presented in [61] by Qin and Ljung solves (3.9) per row i :

$$Y_{fi} = \Gamma_{fi} \bar{K} Z_p + H_{fi} U_i + G_{fi} \hat{E}_{i-1} + E_{fi}, \quad (3.52)$$

with the Hankel matrices defined according to (3.4). In (3.52), \hat{E}_{i-1} , and the solution of the first block row, $i = 1$, are obtained from the HOARX model in Section 3.3.1. Furthermore:

$$\Gamma_{fi} \triangleq C A^{i-1}, \quad (3.53a)$$

$$H_{fi} \triangleq \begin{bmatrix} C A^{i-2} B & \dots & C B & D \end{bmatrix} \in \mathbb{R}^{n_y \times n_u \cdot i}, \quad (3.53b)$$

$$G_{fi} \triangleq \begin{bmatrix} C A^{i-2} K & \dots & C K \end{bmatrix} \in \mathbb{R}^{n_y \times n_y \cdot (i-1)}. \quad (3.53c)$$

After performing linear least squares for every $i = 2, \dots, f$, the solutions are combined to give estimates of $\Gamma_f \bar{K}$, H_f , and G_f . Mathematically the solution for every block-row i reduces to:

$$\left[\left\{ \widehat{\Gamma_{fi} \bar{K}} \right\}_{\text{IEM1}} \quad \hat{H}_{fi} \quad \hat{G}_{fi} \right] = Y_{fi} \begin{bmatrix} Z_p \\ U_i \\ \hat{E}_{i-1} \end{bmatrix}^\dagger. \quad (3.54)$$

Notice that the obtained \hat{H}_f and \hat{G}_f are lower block triangular matrices, but that the Toeplitz structure is not necessarily imposed. The IEM closed-loop SMI methods is summarized in Algorithms 8.

System in predictor form

For a system representation in predictor (*i.e.* observer) form, (3.11) can be solved in the same multi-stage manner as the innovation form presented earlier:

$$Y_{fi} = \bar{\Gamma}_{fi} \bar{K} Z_p + \bar{H}_{fi} U_i + \bar{G}_{fi} Y_{i-1} + E_{fi}, \quad (3.55)$$

⁴IEM1 is sometimes also denoted with Parsim-E1. Parsim-E (IEM) estimates \hat{E}_{i-1} per row, which is computationally less efficient, and hence not included in this unifying description.

⁵Presented under the name Whitening Filter Algorithm (WFA).

with all terms defined in the same manner as in (3.53), except that it uses \bar{A} and \bar{B} to build the extended observability and Toeplitz matrices. Again, the first block row solution, $i = 1$, is solved as the HOARX model. Since the predictor form is used, solving (3.55) for the remaining $i = 2, \dots, f$ does not require the estimate \hat{E}_f :

$$\left[\left\{ \widehat{\Gamma_{fi} \mathcal{K}} \right\}_{\text{PBSID}} \quad \hat{H}_{fi} \quad \hat{G}_{fi} \right] = Y_{fi} \begin{bmatrix} Z_p \\ U_i \\ Y_{i-1} \end{bmatrix}^\dagger. \quad (3.56)$$

Adding all i terms results in the lower block triangular estimates \hat{H}_f and \hat{G}_f , as well as the PBSID solution of the closed loop subspace identification problem. It is said that the PBSID_{opt} algorithm can be performed using the same parallel row-wise methodology, however alternatively it performs an ‘optimally’ weighted least squares that yields the least asymptotic variance of the estimators and also imposes a Toeplitz structure on \hat{H}_f and \hat{G}_f [66]. The PBSID subspace model identification procedure is summarized in Algorithms 9.

Example X. The IEM1 and PBSID subspace model identification algorithms are applied to the same simulation model as Example VI, VIII and IX. In summary, the first order model with a pole at $z = 0.5$ from [63] is implemented in closed-loop, with a filtered reference signal. This filter is also supplied in [63]. Figure (3.18) shows the Bode magnitude plots of the identified models. It is easily noticeable that the innovation-form IEM1 method has more variance in the high frequencies, but also a little lower variance in low frequencies because of this. It is easily noticeable when one compares Figure 3.18(a) with Figure 3.16(b) that the IEM1 and IE-MOESP algorithms yield totally different solutions to the closed-loop identification problem, and that IEM1 has less variance in the low frequencies.

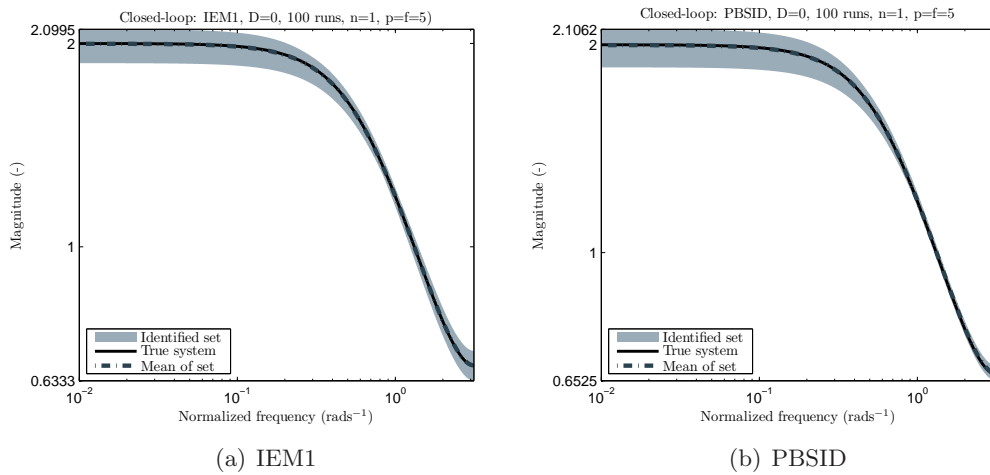


Figure 3.18: Frequency response estimated model for IEM1 and PNSID over 100 Monte Carlo runs, using example in [63] under closed-loop conditions.

Summary of the multi-stage procedures

This section gives a small summary of the closed-loop SMI methods that were described in this section, which are illustrated schematically in Figure 3.19. From the figure it is

immediately evident that IEM1 uses a HOARX pre-estimate of the innovation sequence, whereas PBSID does not require this step. The two multiple-stage closed-loop SMI methods contained in this section are summarized in Algorithms 8-9.

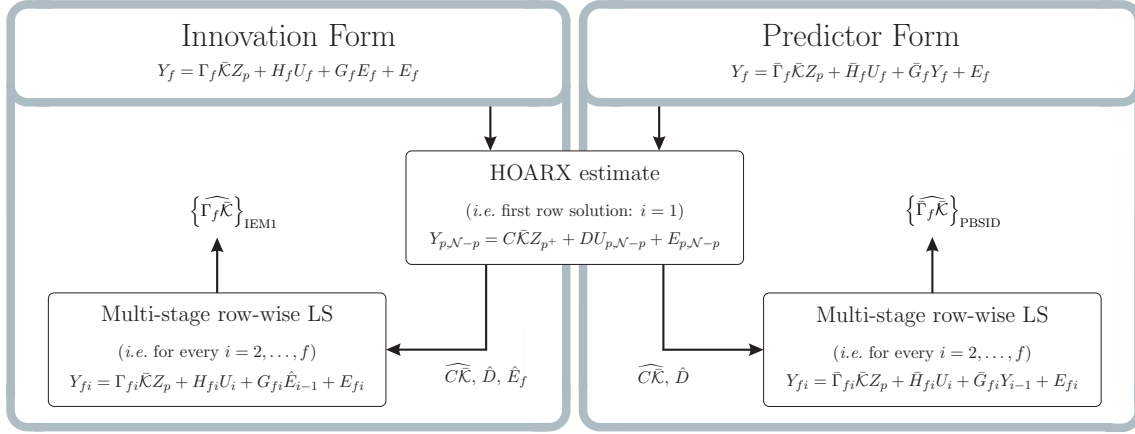


Figure 3.19: Overview of the multi-stage procedures.

Algorithm 8 IEM1 (Parsim-E1)

1. Construct the matrices $Y_{p, \mathcal{N}-p}$, $U_{p, \mathcal{N}-p}$, and Z_{P+} according to (3.26).
2. Solve the HOARX problem in (3.25) (first block row: $i = 1$), if desired imposing the constraint $D = 0$, and construct \hat{E}_i from its solution according to (3.4).
3. Construct the block Hankel matrices Z_P provided in (3.3), as well as Y_i and U_i according to (3.4).
4. Solve the multi-stage problem defined in (3.52) for the remaining block rows $i = 2, \dots, f$, if desired imposing the constraint $D = 0$, and stack the obtained $i = 1, \dots, f$ solutions in (3.54) to form $\widehat{\Gamma_f \bar{K}}$.
5. Determine the Markov state estimate, $\hat{X}_{p, \mathcal{N}}$ using (3.12a) and (3.13).
6. Obtain an estimate of the A, B, C, D and K system matrices by solving the least squares problems in (3.15) and (3.17), if desired constraining $D = 0$.

Algorithm 9 PBSID (WFA)

1. Construct the block Hankel matrices Z_P provided in (3.3), as well as Y_i and U_i according to (3.4).
2. Solve the multi-stage problem defined in (3.55), for all block rows $i = 1, \dots, f$, if desired imposing the constraint $D = 0$. The first block-row solution is, in essence, identical to the HOARX problem in (3.25).
3. Stack the $i = 1, \dots, f$ solutions in (3.56) to form $\widehat{\Gamma_f \bar{K}}$.
4. Determine the Markov state estimate, $\hat{X}_{p, \mathcal{N}}$ using (3.12b) and (3.13).
5. Obtain an estimate of the A, B, C, D and K system matrices by solving the least squares problems in (3.15) and (3.17), if desired constraining $D = 0$.

Example XI. Simulation results of the unstable plant example, used in Example I and II, are provided in Figure 3.20. The simulation show that the IEM1 method produces biased results, whereas PBSID obtains an accurate solution. The PBSID_{opt} and IEM_{opt} algorithms are also shown in the figure. For this model, with a past/future horizon of $p = f = 6$, the IEM_{opt} method performs slightly better than the other algorithms in the low frequency range. The same applies for PBSID_{opt} compared to the PBSID algorithm, since PBSID_{opt} performs slightly better for low frequencies.

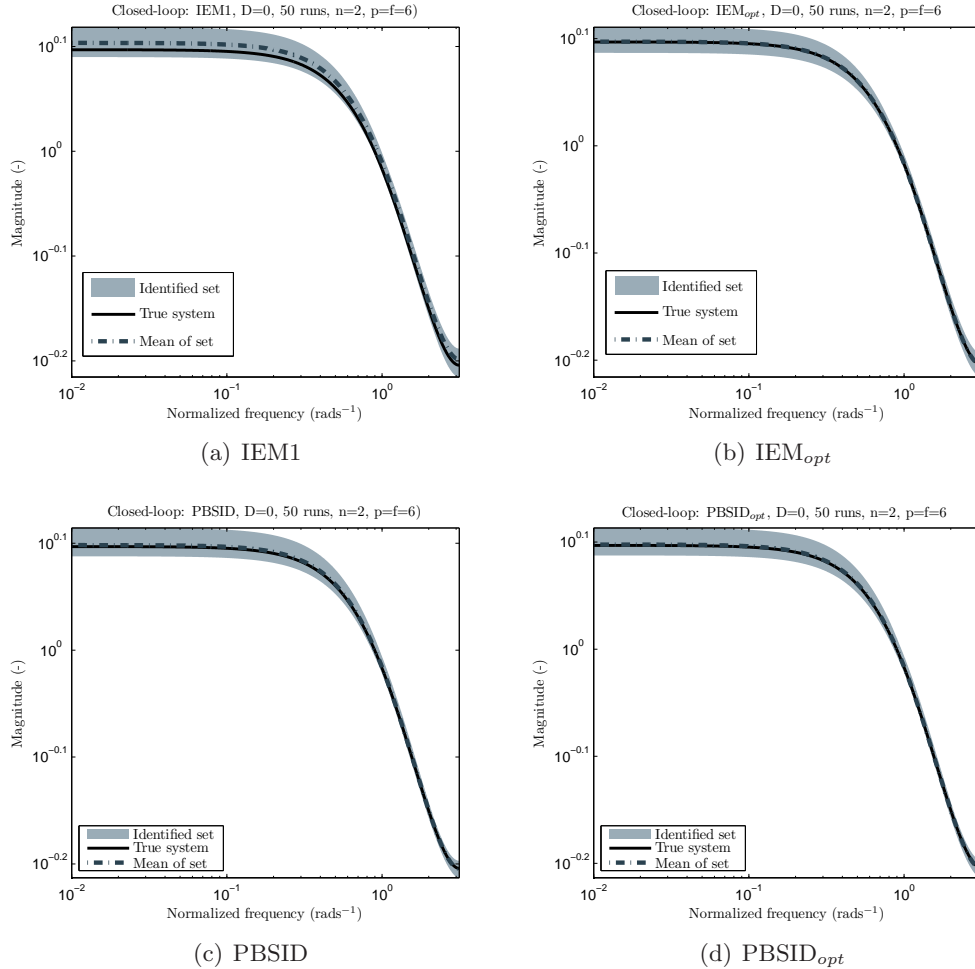


Figure 3.20: Frequency response estimated model over 50 Monte Carlo simulations for IEM_{opt} (example 2 in [62]).

3.4 Summary and concluding remarks

This chapter provided a detailed description, and explanation, of subspace model identification techniques, especially for use with systems under closed-loop (controlled) conditions. It gave an overview of the most recent closed-loop SMI methods from literature, namely SSARX developed by Jansson [60], Parsim-E1/IEM1 developed by Qin and Ljung [61, 58], and PBSID and PBSID_{opt} developed by Chiuso and Picci [62, 59]. The Parsim-E/IEM algorithm is

not included in the presented closed-loop SMI framework, since it is computationally intensive, and its successor, IEM1, conformed better with the ‘unifying’ description. This section gives a summary of the presented framework, as well as some concluding remarks.

Summary

The closed-loop methods that are contained in the ‘unifying’ closed-loop SMI framework all initially estimate a HOARX model in some way. This was, for the subspace identification procedures that already exist in the literature, remarked by Chiuso and Picci in [63], but the methods that were developed in this thesis are no exception. It is this observation that forms the basis of the ‘unifying’ description. The manner in which the closed-loop SMI methods use the HOARX estimate is however very different, as indicated in Figure 3.21. A summary of the closed-loop SMI procedures from literature is given, for PBSID_{opt} in Alg. 1, for SSARX in Alg. 4, for IEM1 in Alg. 8, and for PBSID in Alg. 9. From these summaries, and from Figure 3.21, one can clearly see that PBSID_{opt} obtains its solution directly from the HOARX model estimate, whereas the other existing methods need to reformulate the problem in a next step. This makes the PBSID_{opt} advantageously simple and numerically very efficient to implement, *e.g.* in a recursive manner [25]. Furthermore, the SSARX method uses the identified Markov parameters from the HOARX model to construct the Toeplitz matrices in the identification problem (3.11). These pre-estimated Toeplitz matrices are then used to append the influence of the future inputs onto the future outputs, after which a Canonical Variance Analysis (CVA) is performed between the future and past related terms. This in contrast with the PBSID and IEM1 methods that interpret the identification problem, thus (3.9) and (3.11) respectively, per block-row. These methods solve the problem per block-row, and stack the block-row solutions to form the total solution to the problem. It should be noticed that in this methodology, the first block-row forms the HOARX problem. Moreover, compared to PBSID, the IEM1 method requires an extra pre-estimate of the innovation sequence from this HOARX problem.

In the former we have shortly described the SMI procedures from literature. With this knowledge, and referring to Figure 3.21, it is easily indicated that there would have been some empty positions on the innovation-form side of the figure for the situation without the novel methods. In this thesis we use a mapping to, in a sense, mirror the predictor-from based methods to an innovation-form system representation. The mapping from predictor-from to innovation-form is provided as the series expansions in (3.32), (3.33), and (3.33), and forms the basis of the novel IEM_{opt}, ARXISS, ISSCVA, and CVA-MOESP methods. The IEM_{opt} and ARXISS methods are obtainable from the PBSID_{opt} solution by using the series expansions, as summarized in Alg. 2 and Alg. 3. Furthermore, the ISSCVA procedure is summarized in Alg. 5, and the CVA-MOESP method in Alg. 6. These methods use an identical approach as the existing SSARX method, however with pre-estimates that are re-defined in innovation form, again using the series-expansions. Simulation results, of which some provided in this chapter, have shown that the SSARX and ISSCVA methods give the identical results. Both methods find the same state space model, even though their formulations are entirely different. Besides these methodologies, also another novel closed-loop SMI algorithm has been developed in this thesis, namely the IE-MOESP procedure provided in Algorithm 7. The basis of this approach is conceptually almost identical to the IEM1 algorithm that re-uses the estimated innovation block Hankel matrix formed with the HOARX estimate. Nevertheless, unlike the multi-stage IEM1 method, this approach uses the entire

pre-estimated innovation sequence, and applies an LQ factorization in a MOESP like procedure. As such, this yields an entirely different (and unknown) solution to the closed-loop subspace model identification problem.

After we have obtained these 'versions' of the solution to the identification problem, the system can be parameterized using the approaches illustrated in Figure 3.2. These approaches use either an estimate of the Markov state, or a low-order estimate of the observability matrix, formed from a Singular value Decomposition (SVD) on the solutions of the identification

Concluding remarks

With respect to the subjects in this chapter, the following final remarks can be made:

- This chapter presented a (mostly) self-sustaining description/explanation of (closed-loop) subspace model identification, which includes many algorithms in a highly structured manner. It is considered, together with the developed closed-loop SMI methods, as one of the main contributions of the thesis. Moreover, the presented framework also resulted in the development of a MATLAB toolbox that follows the interlinked nature of the 'unifying' description (see Fig. 3.2 and Fig. 3.21). In fact, it is this toolbox that was used for the comparative simulation study in Chapter 4, and the practical verification of the developed SMI methods. The SMI description in this chapter, with the accompanying MATLAB toolbox can also be used as an interactive introductory tool into closed-loop subspace model identification for students or beginners.
- This chapter provided five novel closed-loop SMI algorithms, of which a simulation based comparison with SMI methods from literature will be given in the next chapter. In this chapter, only a few simulations were provided. These show that novel innovation-form based ISSCVA algorithm and the already existing predictor-form based SSARX method give identical results. It is a remarkable result that shows the system representation to be invariant when algorithms are comparable in exact (CVA) methodology. Thus bringing the, more or less, isolated innovation-form based and predictor-form based methods closer together, something that is worth a further investigation.
- A comparison between two parameterization methods that use the state sequence was also given in this chapter. With simulations we have shown that the method provided in [51] gives biased results when we constrain $D = 0$. If we constrain the set of identified systems to a system with no direct feed through, it is highly recommended to use the approach in [58], since it gives unbiased results. Simulations in the next chapter will show the need for this constraint in the identification procedure.

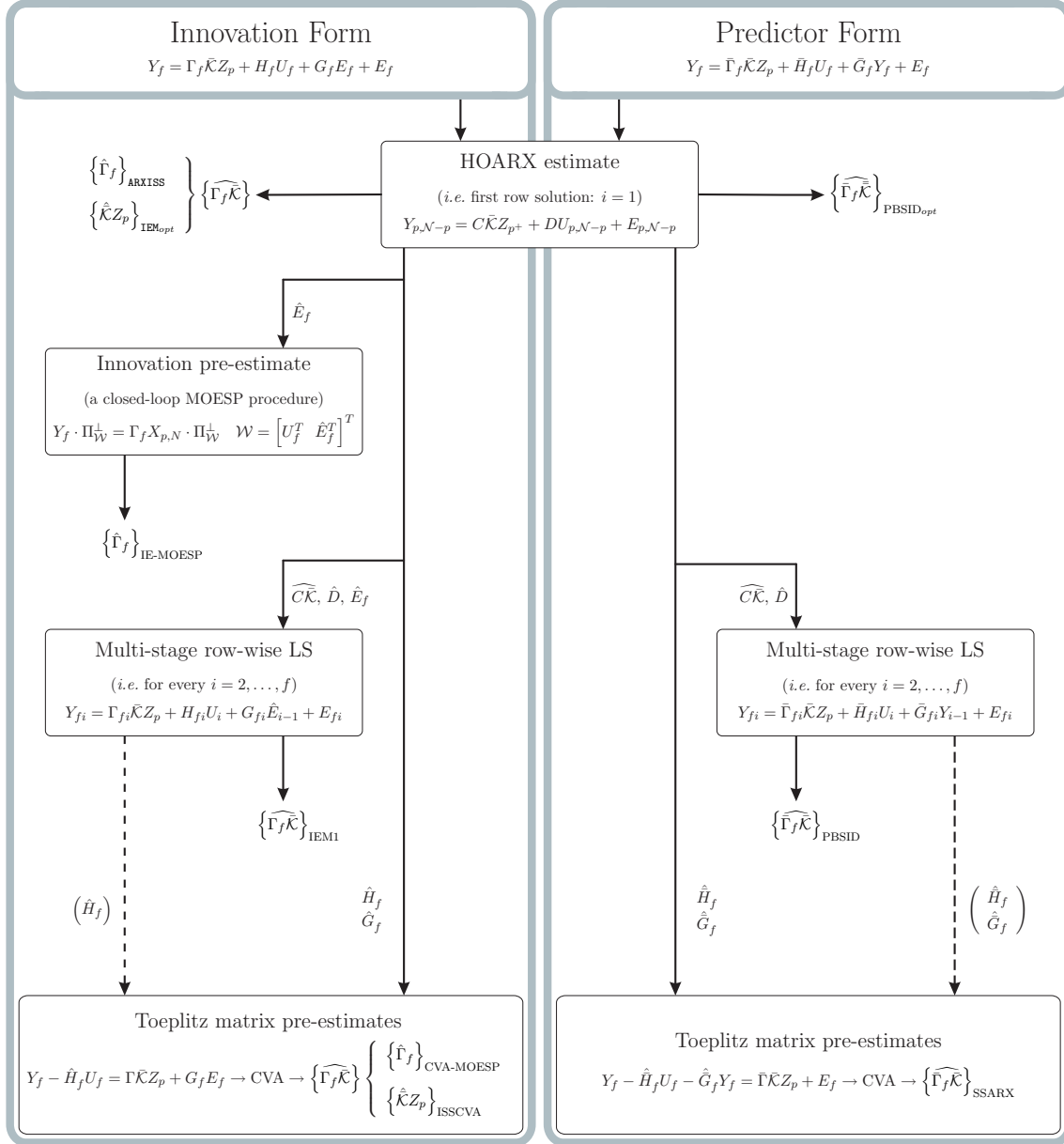


Figure 3.21: Schematic of the closed-loop subspace identification framework according to the 'unified' description.

Closed-loop SMI on UpWind 5MW

In the previous sections we developed an accurate model of a ‘smart’ rotor blade for the UpWind 5MW reference wind turbine, and presented a novel closed-loop Subspace Model Identification (SMI) framework that incorporated some new methods. It is the purpose of this section to combine the former subjects in a wind turbine simulation example that provides a comparison between the presented SMI methods, and show how the developed SMI methods compare to the existing algorithms.

The SMI methods were already implemented on some simulation examples from literature in Section 3. This section will provide some additional simulation results on the aeroelastic UpWind 5MW reference wind turbine model in Section 2. The ‘smart’ rotor blade will be controlled with a LQG controller that provides stable responses, with an enhanced performance. It should be noted that we will only perform SMI statically, so the algorithms are not implemented on-line (e.g. recursively).

The contents of this chapter can be divided in two separate parts. It starts, in Section 4.1, with presenting the conditions of the simulation, where for example the controller will be presented. Section 4.2 continues with the SMI results from the UpWind 5M simulation model thereafter. Finally, the chapter ends by giving a short summary and conclusion.

4.1 The controlled aeroelastic plant

This chapter performs Subspace Model Identification (SMI) on the 2-dimensional UpWind 5MW reference wind turbine model from Chapter 2. The previous section provided a detailed description of the closed-loop subspace model identification procedures in a step-wise approach. This description structured existing closed-loop SMI methods, together with some new identification methods, in a ‘unifying’ closed-loop subspace identification framework. The closed-loop SMI methods that are allocated under this ‘unifying’ description are summarized in Algorithm 1 to Algorithm 9. In this section we will apply each of these algorithms on simulated data from a closed-loop ‘smart’ rotor blade set-up. Hence, a feedback controller will be applied on the UpWind 5MW reference wind turbine model in this section.

We will refer to this open-loop ‘smart’ rotor blade model simply as the aeroelastic plant in the remainder of the report. The simulations serve the purpose to generate input-output data batches that will be used to identify a state space model in compliance with the SMI algorithms presented earlier. For this reason only constant free-stream velocities are applied to the system. It should be noted that this static/constant free-stream velocity relates to the aerodynamic operating point around which the aerodynamic model was linearized in Section 2.3.

The simulations that are considered in this section relate to each of the following constant free-stream velocities:

- $V_\infty = 60 \text{ mss}^{-1}$, which is the free-stream velocity that corresponds to a 75% rotor annuli, for wind speeds above the rated wind speed, as shown in Figure 2.5(b).
- $V_\infty = V_{flutter} = 113.83 \text{ mss}^{-1}$, which is the model’s flutter speed. For this free-stream velocity the system has two marginally stable poles, as indicated in Figure 2.6
- $V_\infty = 120 \text{ mss}^{-1}$, which is the free-stream velocity for which the aeroelastic system of the UpWind 5MW turbine is unstable.

Since only one of the above three test cases reflects, in fact, a stable system, controllers need to be designed that stabilize the marginally stable and unstable open-loop models. Furthermore, the objective of the feedback controllers is to enhance the performance as much as possible. Three Linear Quadratic Gaussian (LQG) controllers are constructed in the following.

In Chapter 2 the aeroelastic plant model was defined in a very general manner that is applicable to both a 2-DOF and 3-DOF system, and also incorporated some constant terms, like gravitational effect. In this section we will redefine this general notation into a more specific model that is suitable to be used for closed-loop simulations. Only the plunge response $h(t)$ and the pitch response $\theta(t)$ will define the system outputs in this section. Moreover, the system inputs are formed by the quasi-steady flap deflection δ , the wind gust disturbance $w_G(t)$, and some additional terms to incorporate measurement and process noise. The closed-loop system is shown in Figure 4.1, which corresponds to the following redefined aeroelastic plant description:

$$\begin{bmatrix} \ddot{q}(t) \\ \dot{q}(t) \\ \dot{x}(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A_{AEPL}(V_\infty) & B_{AEPL}(V_\infty) & K \\ C_{AEPL} & D_{AEPL} & I \end{bmatrix} \cdot \begin{bmatrix} \dot{q}(t) \\ q(t) \\ x(t) \\ \delta(k) \\ w_G(t) \\ e(t) \end{bmatrix}. \quad (4.1)$$

It should be noted that the formulation in (4.1) is in accordance with the predictor/observer form in (3.2). The system matrices are given in (2.28) and (2.32). The open-loop model is however a continuous time system. As described in Chapter 2, the continuous time aeroelastic plant is implemented as follows (for constant V_∞):

1. Define the operating point V_∞ , and compute $A_{AEPL}(V_\infty)$, $B_{AEPL}(V_\infty)$ accordingly.

2. Discretize the system in (2.26) using the zero-order-hold discretization method with MATLAB's `c2d` function.

After the discretization, the Kalman filter gain K can be obtained by solving the Discrete Algebraic Riccati Equation (DARE), given in (3.19). We define the noise covariance matrices as follows: $S = 0 \cdot I$, $Q = 0.001 \cdot I$, $R = 0.01 \cdot I$, and I the identity matrix. The innovation e_k is, for the simulations, defined as a zero mean white noise signal with the variance of 0.01. There are however some restrictions. The wind gust disturbance represents an unknown external influence to the system, and as such, one can not use it as a known term in the computation of K . We are thus only allowed to use the columns of B_{AEPL} that correspond to the known input δ . Consequently, the two aerodynamic lag states that correspond to the gust disturbance need to be removed, since only in that case, the system with $B_{AEPL}(1:n,1)$ yields a minimal realization and the DARE can be solved. Note that the aeroelastic plant in Figure 4.1 still contains the gust related terms. They were only temporarily removed in the computation of the gain K . The same approach needs to be taken in the LQG controller synthesis.

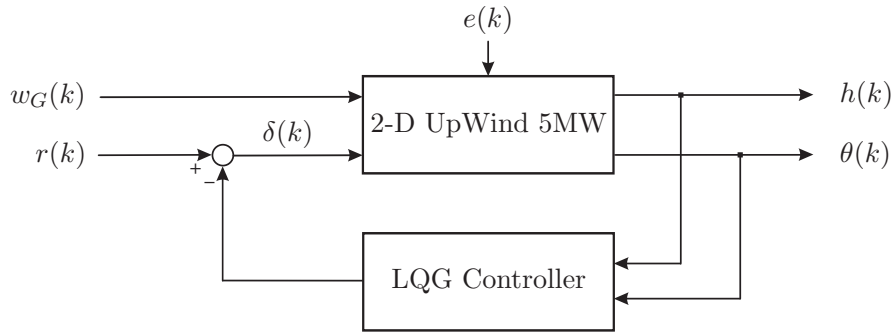


Figure 4.1: Closed-loop block diagram of the aeroelastic plant model with LQG controller, and the innovation noise signal $e(k)$.

LQG controller synthesis and the closed-loop system

In Figure 4.1, a block diagram of the controlled ‘smart’ rotor blade is given. In this section we consider an LQG controller to control the closed-loop behavior of the system. An LQG controller combines a Kalman filter with a linear quadratic cost function that needs to be minimized. The cost function has the form:

$$J = \mathbf{x}^T \cdot Q_x \cdot \mathbf{x} + \mathbf{u}^T \cdot R_u \cdot \mathbf{u},$$

where the state x_k is estimated using the Kalman filter, of which we already described how to compute the filter gain K .

The LQG controllers that are constructed, all define the weighting matrices as:

$$Q_x = C_{AEPL}^T C_{AEPL}, \quad R_u = 0.001.$$

The same tuning parameters are used in the LQG control law synthesis for every free-stream velocity defined earlier. Since the Kalman filter is based on a different open-loop model for each V_∞ , different controllers are found for each test case. The open-loop and closed-loop

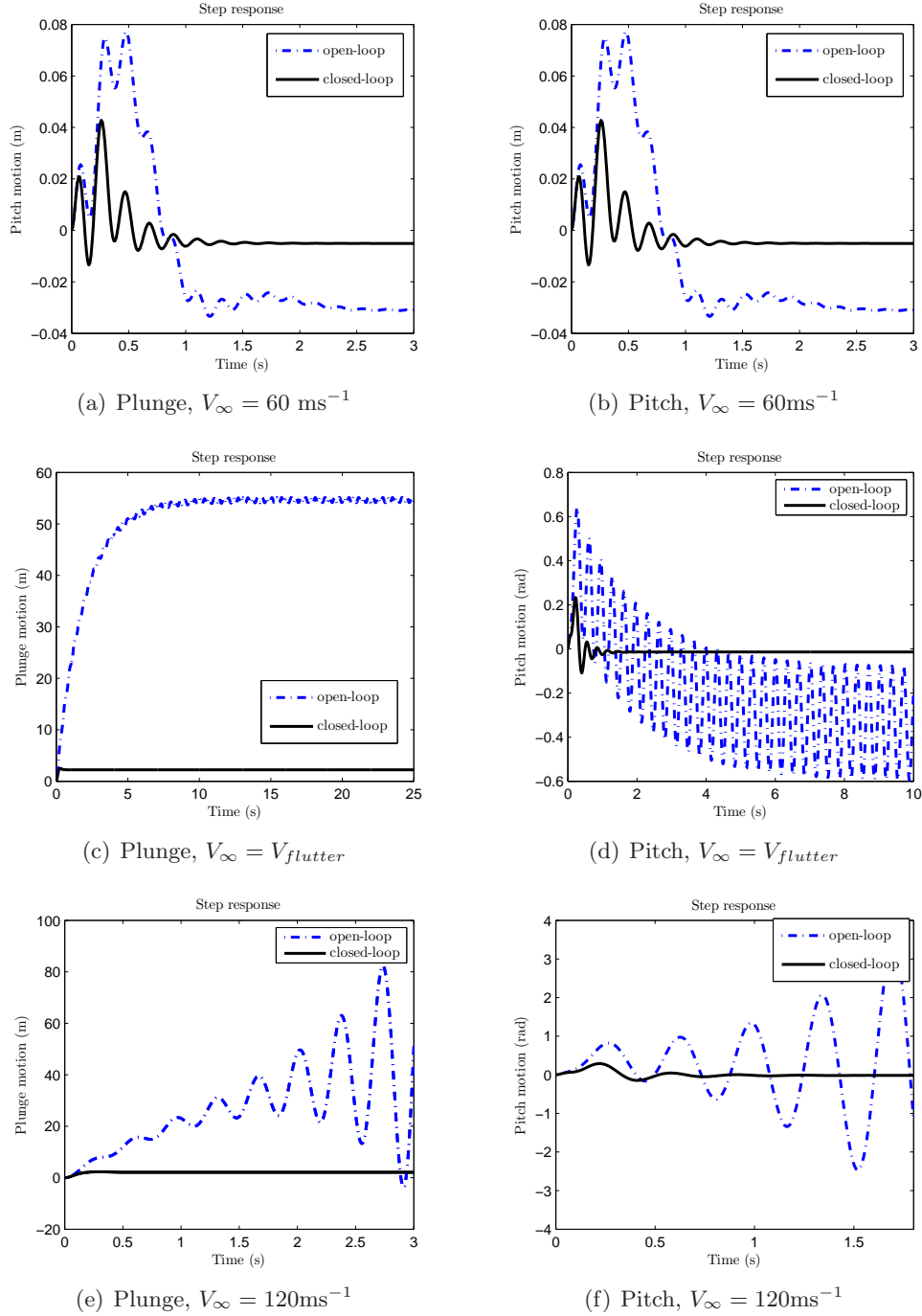


Figure 4.2: Step responses of the open-loop and closed-loop UpWind 5MW aeroelastic plant for different free-stream velocities.

step response are shown in Figure 4.2. The step responses in the figures correspond to the situation without wind gust disturbance.

The figures clearly shows that the feedback controller results in better performance, compared to the open-loop situation, for each implemented free-stream velocity. A good controller however often corresponds to high feedback gains, which is not ideal for closed-loop subspace

model identification. After all, the outputs with their induced noises are fed back to the inputs, so a high feedback gain might provide a more distinct relation between the inputs and noises, which can give a model mismatch in the identification procedure. Figure 4.3 provides the Bode magnitude plots for the LQG controller that corresponds to the test case $V_\infty = 60 \text{ms}^{-1}$. The plots show high controller gains in the high frequencies, which is the area where normally noise instead of the system dynamics reside. With this in mind, the controlled aeroelastic models clearly provide a challenge to the identification algorithms.

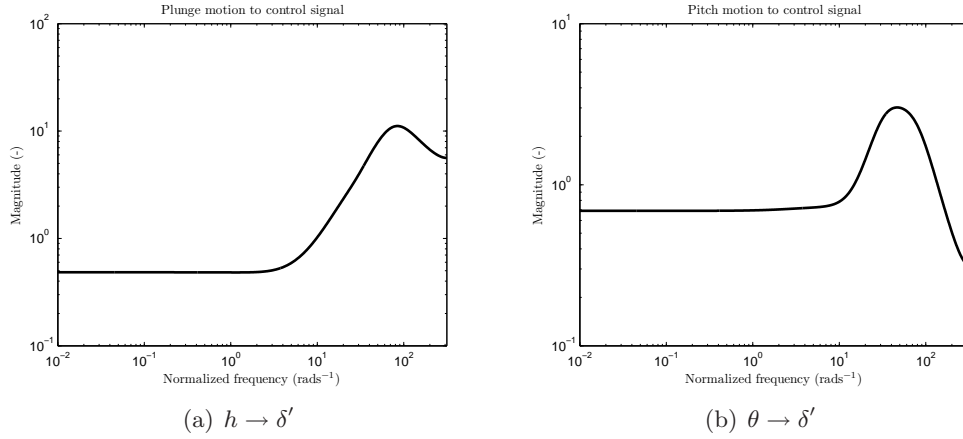


Figure 4.3: Bode magnitude plot of the discrete time LQG controller used in Chapter 4.

Defining the wind gust disturbance signal

The wind gust disturbance signal is defined by the following:

$$w_G = \frac{0.00999}{z - 0.998} \cdot w'_G,$$

where w'_G is a zero mean white noise signal with the variance: $\text{var}(w'_G)=2$. The power spectrum of the wind gust disturbance signal is shown in Figure 4.4. The filter was designed in such a way that its continuous time equivalent is a low-pass filter with a cut-off frequency that equals the frequency at which the rotor blade is rotating, hence $\omega_c = 12.1 \text{rpm} = 0.2017 \text{rads}^{-1}$.

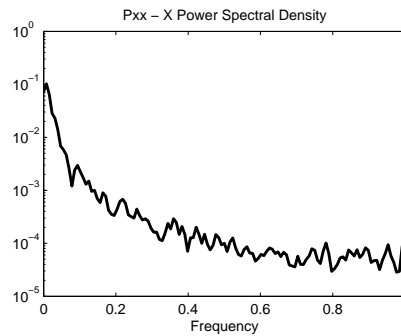


Figure 4.4: Power spectrum of the wind gust disturbance signal w_G . Note that the sample time is $h = 0.01$, so the 1 in the plot corresponds to 100Hz

4.2 closed-loop UpWind 5MW simulations

In this section the subspace model identification methods, that are summarized in Algorithm 1-9, will be applied on data-batches obtained with the closed-loop systems defined in the previous section. Monte Carlo simulations over 50 runs are performed, meaning that the stochastic signals use a new random realization per run. Remember that the LQG controllers have very high feedback gains at large frequencies, so a bias is expected to occur in these frequency regions. The aeroelastic plant model has an order of $n = 8$, but we identify a sixth order model, with success. The poles related to wind gust disturbance do not need to be identified to obtain a proper model description. Furthermore, we use past and future horizons of $p = f = 24$ for every identification procedure.

First, the SMI procedures are performed on the test case with $V_\infty = 60 \text{ ms}^{-1}$. For this situation we will compare the frequency response of the each estimated model with the frequency response of the true ‘smart’ rotor blade model. For the remaining two test cases another comparison is provided, based on the flutter prediction problem. In the flutter prediction problem one tries to identify the complex pole pair that is responsible for the flutter instability. This pole pair is estimated for the situation at the flutter instability point, as well as for the unstable system.

4.2.1 UpWind 5MW identification at 75% annuli

In this section we include all SMI results for the test case with $V_\infty = 60 \text{ ms}^{-1}$. It should be noted that most methods produce comparable estimates, but there are some distinct difference noticeable. We will start this section with a reference to Figure 2.9(b), in which Bode magnitude plots were provided for different free-stream velocities, where it can be noticed that the plunge motion has a much higher magnitude than the pitch motion. This high magnitude results in a high signal to noise ratio (given that the same measurement noise characteristics are applied on both outputs) and hence the possibility for a very good identified model fit and low variances. For this reason, the algorithms have more trouble to estimate the dynamics with regard to the pitch motion. The pitch motion has a much smaller magnitude compared to the plunge motion, and thus a lower signal to noise ratio (again given the identical measurement noise characteristics). A low signal to noise ratio implies less certainty of the estimates, which results in more variance of the estimates.

Figure 4.5 and Figure 4.6, show the results that are obtained with the PBSID_{opt} and IEM_{opt} algorithms, respectively. The first thing that is noticeable when we compare both identified frequency responses is that the PSBID_{opt} approach has a much smaller variance for the low frequencies. However, based on the observations in Example IX, we know that a lower variance for the low frequencies most likely yields in a somewhat higher variance in the high frequencies, and vice versa. This is also this case here, and we can see that PBSID_{opt} has a somewhat larger variance in the high frequency regions. The same observations are also applicable to a bias, although the difference are only of minimal degree then. PBSID_{opt} has a slightly smaller bias for low frequencies and a slightly larger bias for the high frequencies, when compared to IEM_{opt} .

In Figure 4.7 the results of the ARXISS methods are given, which are very comparable to the Bode magnitude plots of IEM_{opt} . Nevertheless, the ARXISS algorithms results in

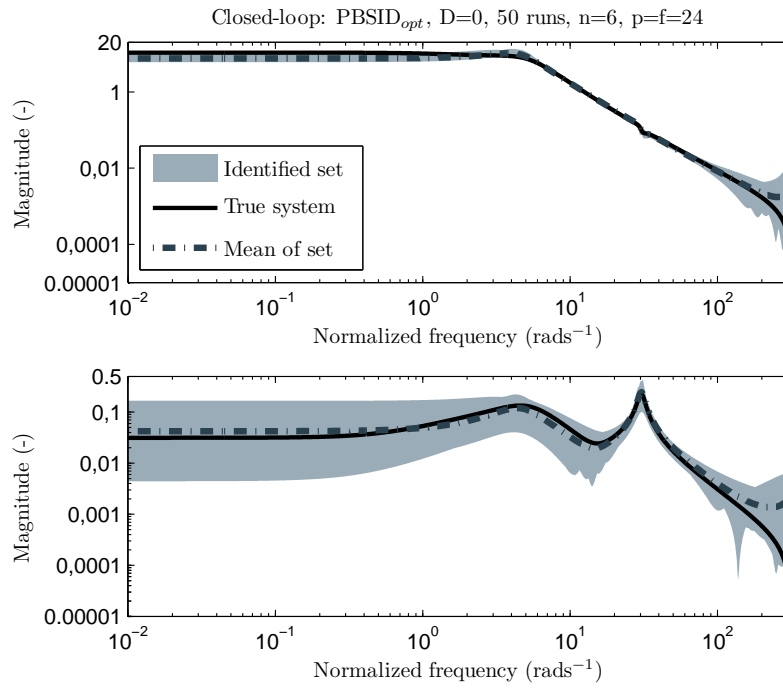


Figure 4.5: 50 Monte Carlo simulations on UpWind 5MW for $V = 60$. Frequency response of estimated model for the PBSID_{opt} algorithm with $n = 6$, $p = f = 24$, and imposing $D = 0$.

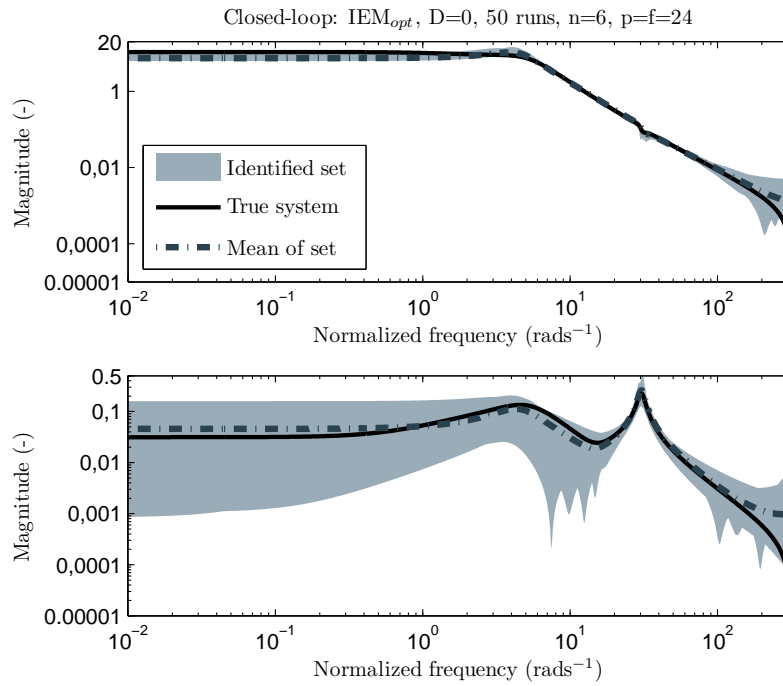


Figure 4.6: 50 Monte Carlo simulations on UpWind 5MW for $V = 60$. Frequency response of estimated model for the IEM_{opt} algorithm with $n = 6$, $p = f = 24$, and imposing $D = 0$.

slightly more variance/bias in the very low frequencies compared to the IEM_{opt} method. Nevertheless, around $10\text{-}20 \text{ rads}^{-1}$ the ARXISS procedure is in favor, since it has somewhat less variance/bias there than IEM_{opt} . For the ‘smart’ rotor blade model it seems more favorable to have slightly better results around $10\text{-}20 \text{ rads}^{-1}$, since this is where most of the dynamics are. It should be noticed that the ARXISS and IEM_{opt} methods only differ in the manner in which they perform the parameterization step: ARXISS uses the low-order observability matrix and IEM_{opt} uses the Markov state.

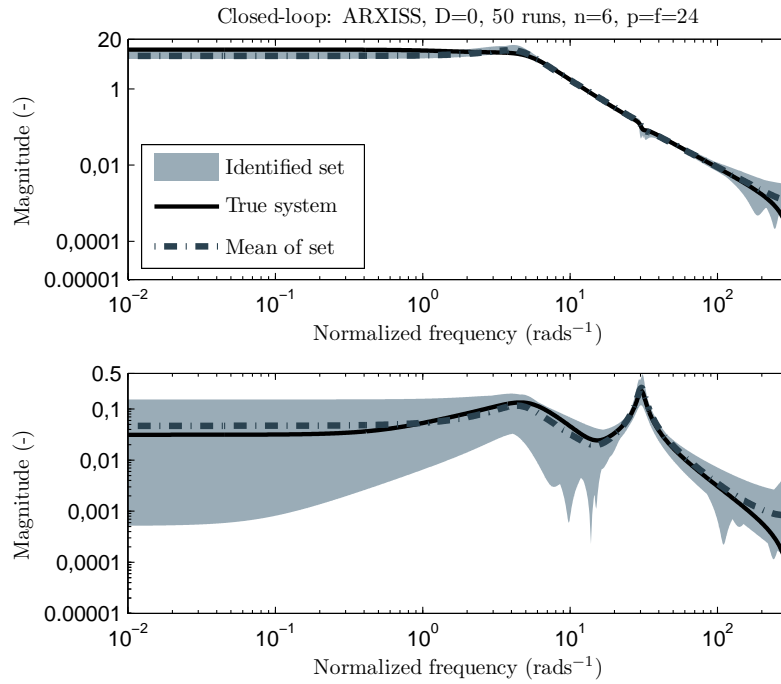


Figure 4.7: 50 Monte Carlo simulations on UpWind 5MW for $V = 60$. Frequency response of estimated model for the ARXISS algorithm with $n = 6$, $p = f = 24$, and imposing $D = 0$.

The results of the SSARX/ISSCVA methods are provided in Figure 4.8. Also for this simulation example both the SSARX and the ISSCVA method produce identical solutions. The first thing that can be observed, is that SSARX/ISSCVA has more variance at the very low frequencies, when it is compared to the three previously described algorithms. However, in the same low frequency region it produces estimates with much smaller bias. Moreover, it is also noticeable that around $10\text{-}20 \text{ rads}^{-1}$ the SSARX/ISSCVA method has less variance than IEM_{opt} and ARXISS. $PBSID_{opt}$ on the other hand seems to yield the least variance of all methods described so far in these low and middle frequencies. SSARX/ISSCVA seems to have less bias than $PBSID_{opt}$, IEM_{opt} , and ARXISS, over practically the whole frequency range.

Figure 4.9 presents the results that were obtained with the CVA-MOESP procedure. Compared to all other algorithms, the CVA-MOESP method provides the best fit of the dynamics at $5\text{-}50 \text{ rads}^{-1}$. Note that the other CVA based methods, SSARX and ISSCVA, already had some favorable variance characteristics around the $5\text{-}50 \text{ rads}^{-1}$ frequency range. A comparison between the IEM_{opt} and the ARXISS approach clearly showed ARXISS in favor in approximately the same $5\text{-}50 \text{ rads}^{-1}$ range. Clearly the parameterization steps used in the ARXISS and CVA-MOESP algorithms put more weight on the frequency response estimates

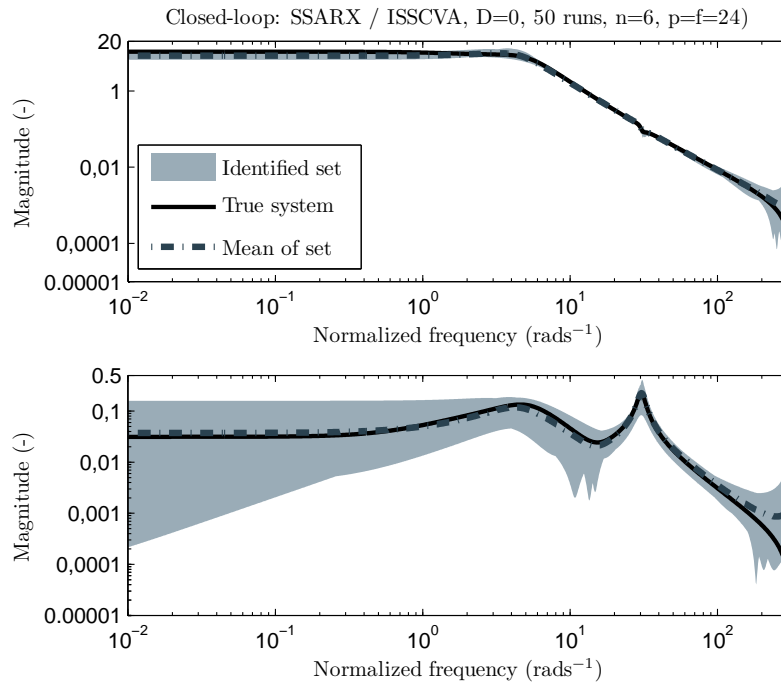


Figure 4.8: 50 Monte Carlo simulations on UpWind 5MW for $V = 60$. Frequency response of estimated model for the SSARX/ISSCVA algorithm with $n = 6$, $p = f = 24$, and imposing $D = 0$.

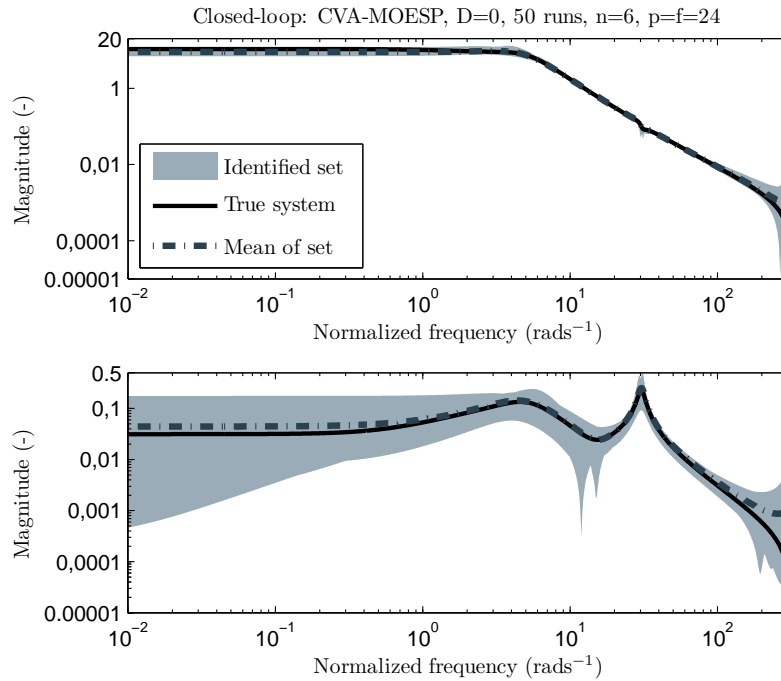


Figure 4.9: 50 Monte Carlo simulations on UpWind 5MW for $V = 60$. Frequency response of estimated model for the CVA-MOESP algorithm with $n = 6$, $p = f = 24$, and imposing $D = 0$.

in this frequency region. It must also be observed that CVA-MOESP has however somewhat more bias at the low frequencies compared to the SSARX/ISSCVA algorithms. [2.5ex] At this stage it seems that the SSARX/ISSCVA algorithm and the CVA-MOESP method found very decent estimates compared to the other methods. It might be because of the weightings used in the CVA solutions, which are sometimes said to be optimal [59]. Although PBSID_{opt} is implemented without weights in the ‘unifying’ description in Section 3, Chiuso and Picci apply in [59] an additional CVA based weight for their PBSID_{opt} methods that is said to yield the least variance of all the estimators. The results for the ‘optimally’ weighted PBSID_{opt} method are presented in Figure 4.10. When we compare Figure 4.5 with Figure 4.10 there is not much difference noticeable between PBSID_{opt} and its weighted version.

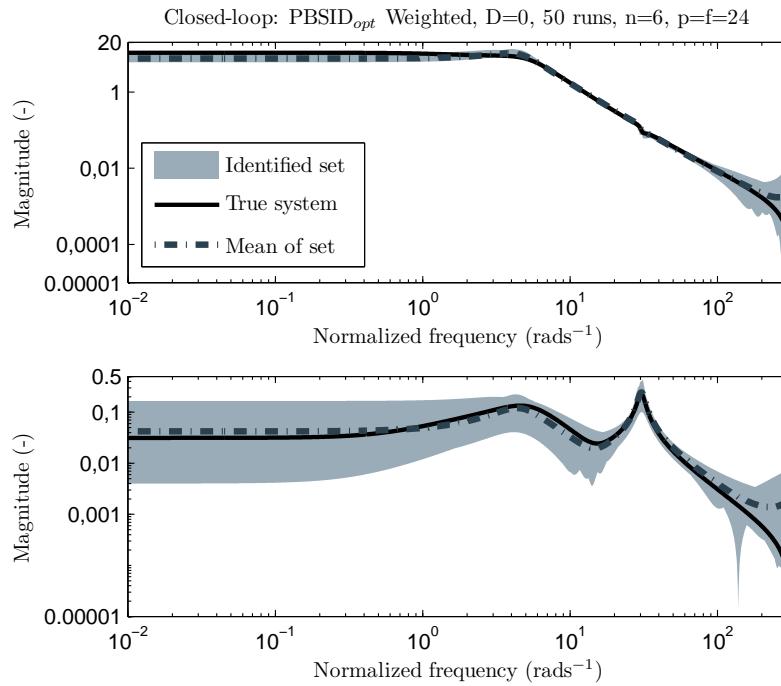


Figure 4.10: 50 Monte Carlo simulations on UpWind 5MW for $V = 60$. Frequency response of estimated model for PBSID_{opt} that uses an optimal CVA-based weighting (see [59] with $n = 6$, $p = f = 24$, and imposing $D = 0$).

Figure 4.11 gives the frequency responses of the estimated model for the IE-MOESP procedure. Its results are from all methods most alike the results obtained with the IEM_{opt} procedure in Figure 4.6. Compared to this method the IE-MOESP procedure has slightly less variance in the lower frequency range and slightly more variance in the other regions. Furthermore, compared to IEM_{opt} the IE-MOESP algorithm has less biased results in the range of approximately $1\text{--}10 \text{ rads}^{-1}$. The results that were obtained with the IEM1 algorithm are given in Figure 4.12, whereas the results for the PSBID algorithm is provided in Figure 4.13. First, it should be noted that the results for the IEM1 method is not comparable to the results of the IE-MOESP approach, even though they both use an innovation pre-estimate from the HOARX problem. When we compare the result of PBSID with IEM1, it is noticeable that PBSID has less bias in the low frequencies. Furthermore it should be noted that PBSID and the SSARX/ISSCVA method produce very similar results, but that they both use methodologies that are very different.

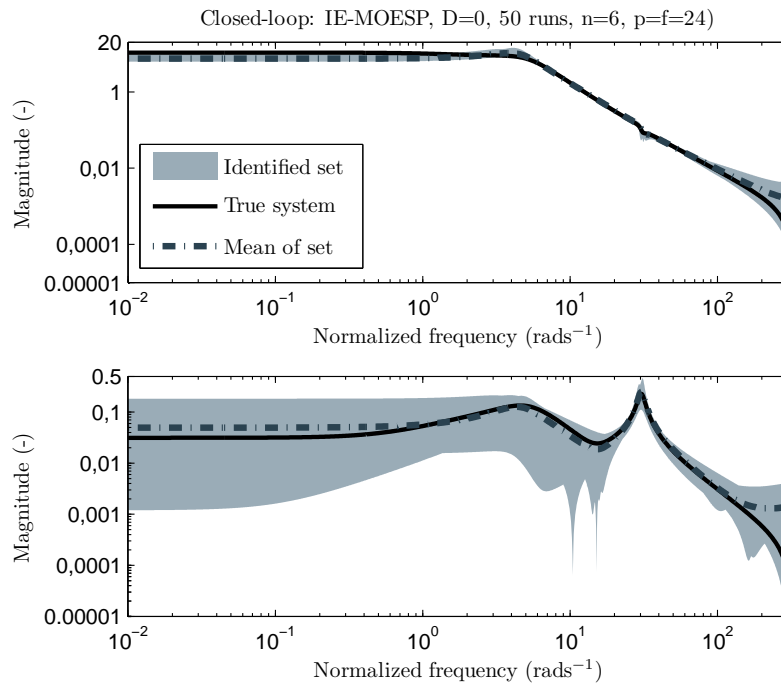


Figure 4.11: 50 Monte Carlo simulations on UpWind 5MW for $V = 60$. Frequency response of estimated model for the IE-MOESP algorithm with $n = 6$, $p = f = 24$, and imposing $D = 0$.

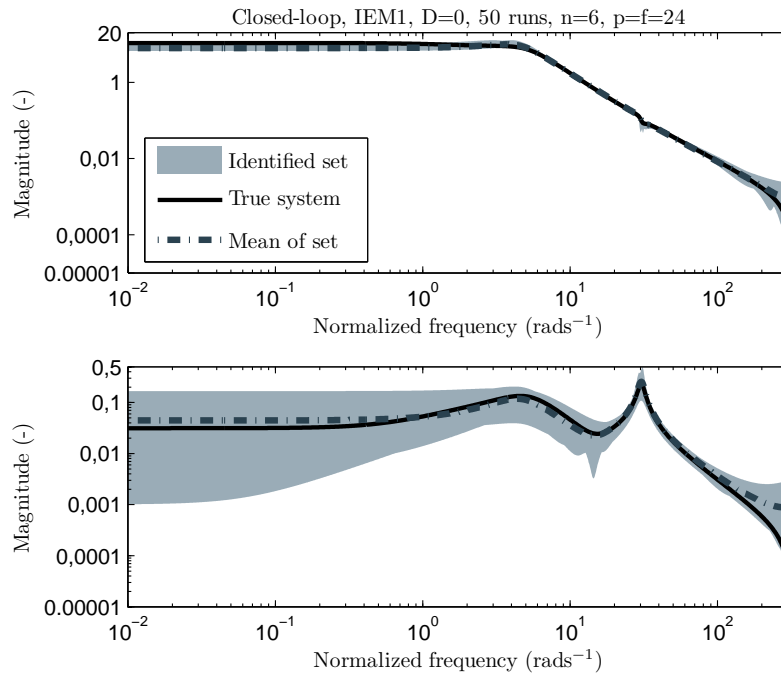


Figure 4.12: 50 Monte Carlo simulations on UpWind 5MW for $V = 60$. Frequency response of estimated model for the IEM1 algorithm with $n = 6$, $p = f = 24$, and imposing $D = 0$.

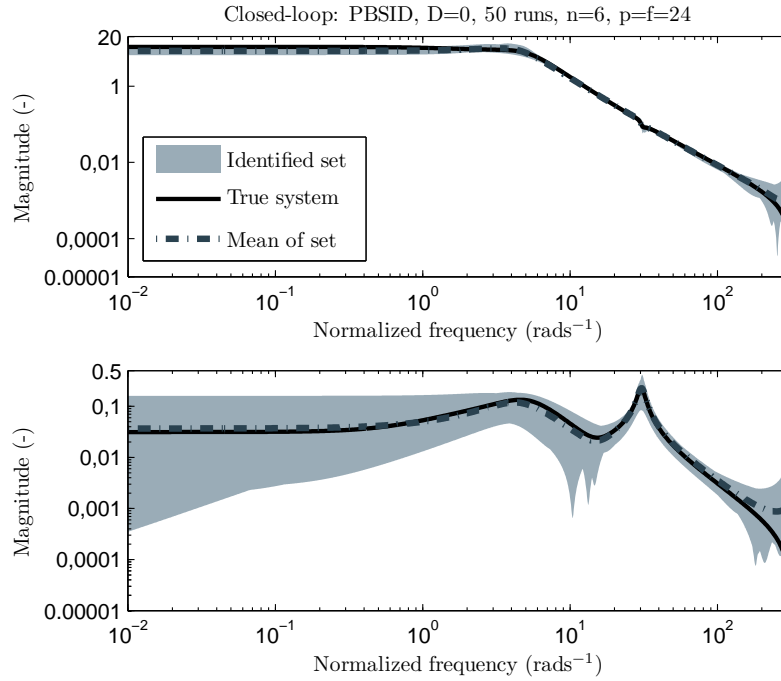


Figure 4.13: 50 Monte Carlo simulations on UpWind 5MW for $V = 60$. Frequency response of estimated model for the PBSID algorithm with $n = 6$, $p = f = 24$, and imposing $D = 0$.

4.2.2 Pole estimates for marginally stable model

In this section we look at the aeroelastic simulations of the model that has the free-stream velocity identical to the flutter speed, meaning that this system yields a marginally stable plant. It has the complex pole pair, that relates to the pitch motion, on the unit circle. In this section we use the nine algorithms that are included in the ‘unifying’ closed-loop SMI framework, to estimate this marginally stable system pole. Hence in this manner, we give a small view on the concept of flutter prediction using subspace model identification techniques. The results are shown in figure 4.14. The first thing noticeable is the comparable results the algorithms produce. Probably because at this high free-stream velocity the system gains of the aeroelastic model get quite high. This simulation provides us yet another example that SSARX and ISSCVA produce identical results, and that PBSID is very much related to these algorithms. Furthermore it seems, from the figure, that each algorithm produces a very slight bias/off-set in its pole estimates. The CVA-MOESP algorithm has the middle of its identified set of poles the closest near the true system pole.

4.2.3 Pole estimates for unstable model

The simulation results that are presented in this section are applicable to the aeroelastic simulation model that has a free-stream velocity of 120 ms^{-1} , and as such has an instable pole pair. Again this simulation example relates to the idea of flutter prediction using subspace model identification. All algorithms find appropriate estimates of this instable pole pair, thus also the methods that are defined in innovation from. First, let's note that SSARX and ISSCVA have the exact same solution. Furthermore, it is noticeable that PBSID also

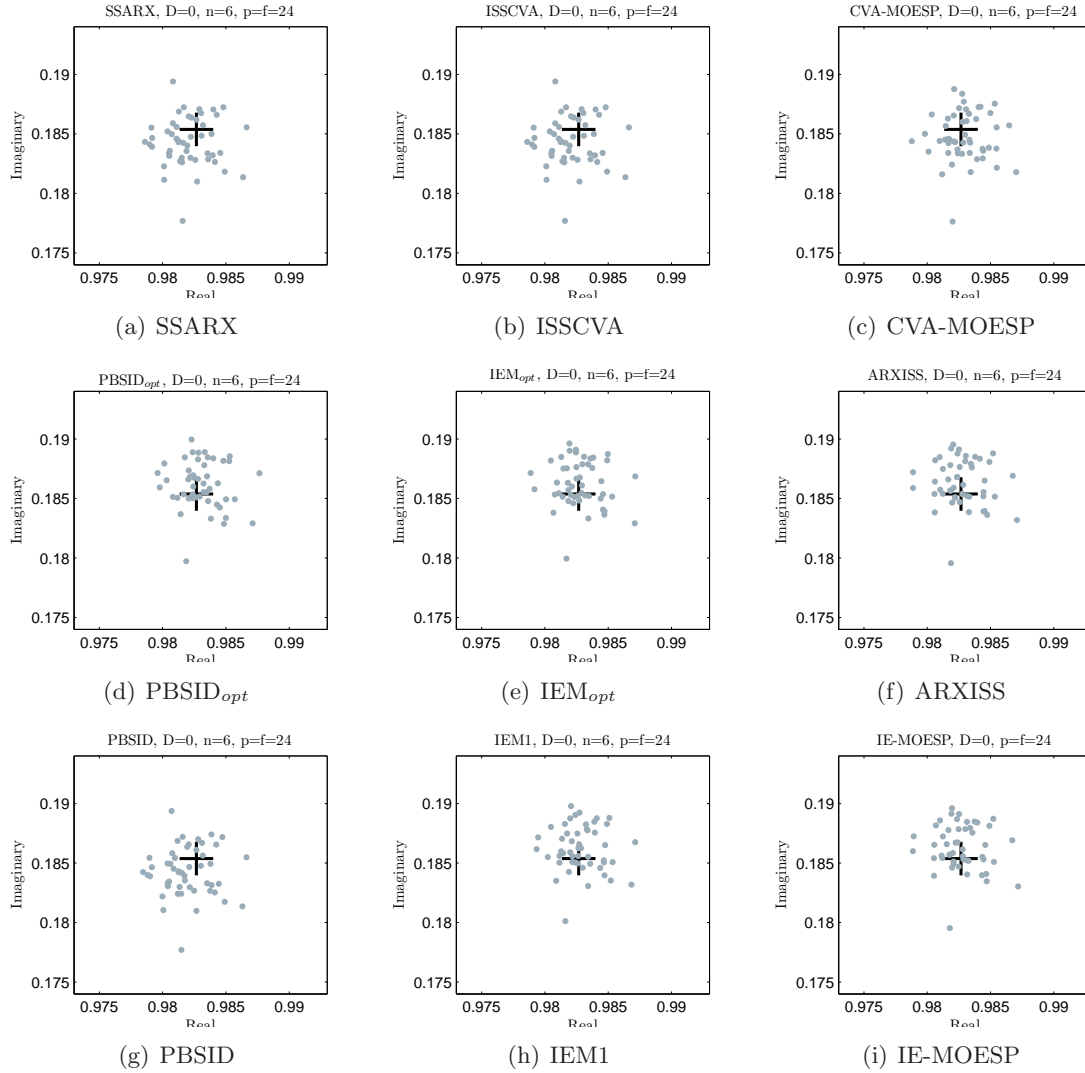


Figure 4.14: 50 Monte Carlo simulations of UpWind 5MW for $V = V_{flutter}$: Pole estimates of all closed-loop SMI algorithms described in Chapter 3, with the parameters: $n = 6$, $p = f = 24$, as well as the constraint $D = 0$.

find estimates that are closely related to SSARX and ISSCVA. Furthermore, CVA-MOESP finds an estimate with a somewhat smaller bias than SSARX/ISSVCA. It should be noticed that the $PBSID_{opt}$, IEM_{opt} , $IEM1$, IE -MOESP, and ARXISS methods find pole estimates that are a little bit too high compared to the true system pole.

4.3 Summary and concluding remarks

In this section the aeroelastic model of the ‘smart’ wind turbine rotor blade from Chapter 3 was redefined to a form more suitable to be used in a closed-loop simulation example. The closed-loop system involves a ‘smart’ wind turbine rotor blade that is controlled using an LQG controller. A controller was implemented for three different situations, namely for a free-stream velocity at 75% rotor annuli, for the flutterspeed (marginal stability), and

for $V_\infty = 120 \text{ ms}^{-1}$, which yields an unstable system. For each situation a controller was designed. The simulations of these closed-loop systems provide data-batches that are used by the SMI algorithms allocated under the 'unifying' closed-loop SMI framework. The simulation results show that all methods perform decently well. Furthermore, the SSARX and the ISSCVA methods produce identical results, and the PBSID method and SSARX/ISSCVA produce very comparable results. It seems that each closed-loop SMI procedure has its own frequency range (low,middle,high) where it works best. A further investigation into this observation is most desired. For the examples of the aeroelastic plant, the CVA-MOESP algorithm was found to work remarkably well around the system dynamics, when compared to the other algorithms.

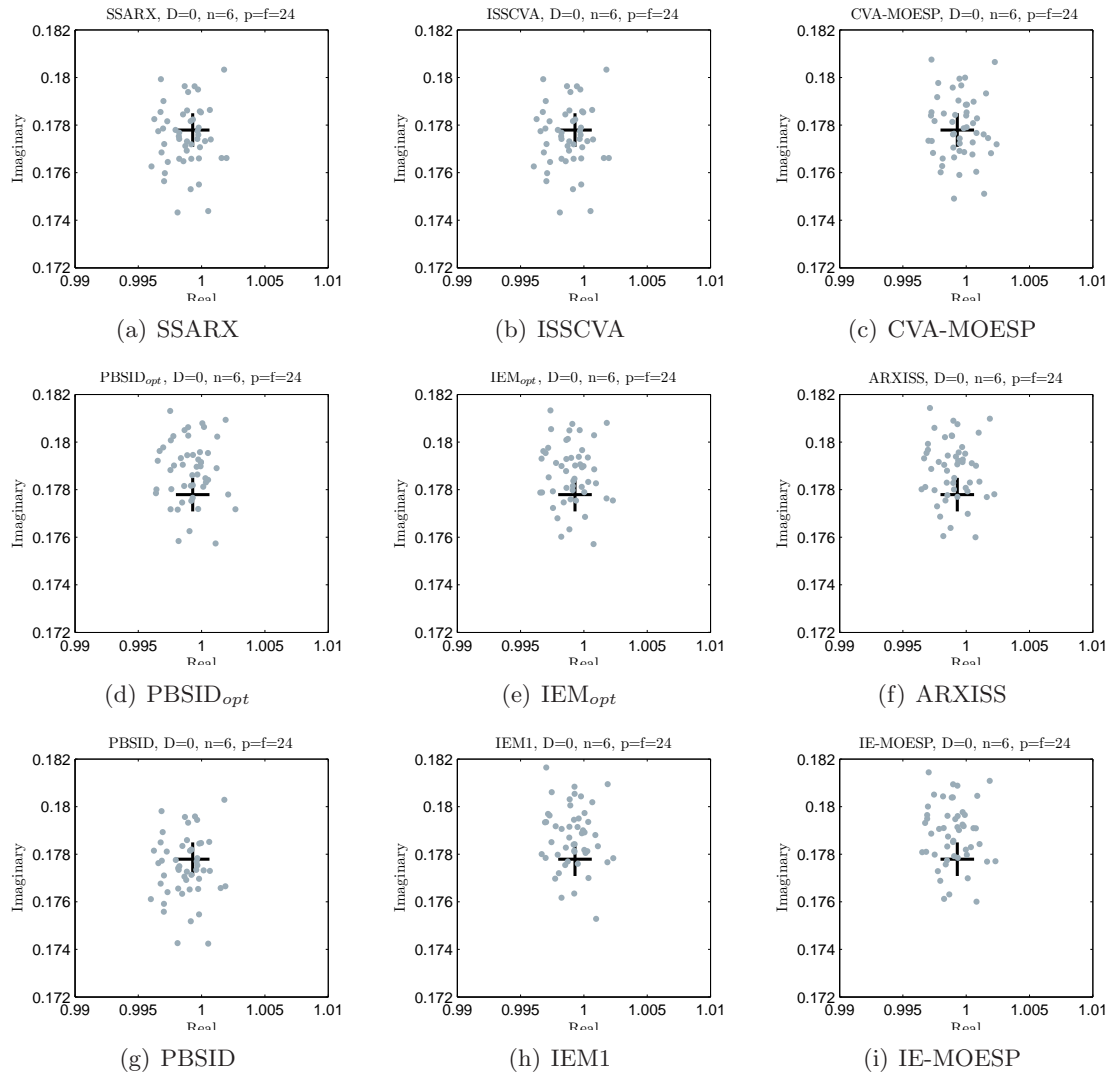


Figure 4.15: 50 Monte Carlo simulations of UpWind 5MW for $V = 120 \text{ ms}^{-1}$: Pole estimates of all closed-loop SMI algorithms described in Chapter 3, with the parameters: $n = 6$, $p = f = 24$, as well as the constraint $D = 0$.

Conclusion and contributions

The report contained two major subjects, namely the construction of an aeroelastic simulation model and a study into novel closed-loop Subspace Model Identification (SMI) techniques. In the thesis, the latter started as small ‘unifying’ description of some closed-loop SMI methods from literature, but ended as a full ‘unifying’ closed-loop subspace model identification framework, which included some novel identification techniques that were developed as part of the thesis.

An aeroelastic model was developed that describes the 2-dimensional unsteady aeroelastic behavior of a ‘smart’ wind turbine rotor blade. The aeroelastic plant model is formed by interconnecting an aerodynamic and a mechanical sub-model. In the literature there are a number of 2-dimensional aeroelastic models available, but not like the novel aeroelastic plant model formed in this thesis. The aeroelastic system is constructed as a Linear Parameter Varying (LPV) state space system, with the free-stream velocity (*i.e.* V_∞) as the varying parameter. This enables the option to make quasi-steady changes in free-stream velocity with any LPV simulation toolbox. In addition, the developed aeroelastic simulation model also incorporates a modeled wind gust disturbance w_G . Since we can use both V_∞ and w_G as a varying input, it is possible to simulate a wind turbine ‘smart’ rotor blade for varying angular rotor speeds, as well as for varying gust disturbances (*e.g.* turbulence) that enter the rotor disc. However, only under the assumption that $V_\infty \gg w_G$ and V_∞ is slowly varying. The construction of the aeroelastic plant model, as well as the implementation of the Up-Wind 5MW structural parameters, and a verification based on a more accurate rotating wind turbine 3D simulation package, can be considered a welcome addition/contribution to the scarcely available 2-dimensional aeroelastic wind turbine models. It should be noted that, in contrast to the availability of aircraft wing models, there are not many models available in the literature that are specified for a wind turbine rotor blade with flap.

The closed-loop Subspace Model Identification (SMI) methods that are contained in the ‘unifying’ closed-loop SMI framework all initially estimate a HOARX model in some way. It should be noted that this was already remarked by Chiuso and Picci in [63] for the subspace identification procedures that already exist in the literature. The SMI methods that were developed in this thesis are however no exception on this observation. It has been the initial HOARX step in the algorithms that forms the basis of the ‘unifying’ description presented in the thesis. The manner in which the closed-loop SMI methods use the HOARX estimate is however entirely different.

The existing PBSID_{opt} algorithms can be seen as one of the easiest SMI algorithm that exists in literature nowadays. PBSID_{opt} obtains its solution directly from the HOARX model estimate, whereas the other existing methods need to reformulate the problem in some additional steps. This makes the PBSID_{opt} advantageously simple and numerically very efficient to implement, *e.g.* in a recursive manner [25]. Furthermore, the SSARX method uses the identified Markov parameters from the HOARX model to construct the Toeplitz matrices in the identification problem. These pre-estimated Toeplitz matrices are then used to append the influence of the future inputs onto the future outputs, after which a Canonical Variance Analysis is performed between the future and past related terms. In this thesis we use a mapping to, in a sense, mirror the predictor-from based methods to an innovation-form system representation. The mapping from predictor-from to innovation-form is provided as a couple of series expansions, and forms the basis of the novel IEM_{opt} , ARXISS, ISSCVA, and CVA-MOESP methods. The IEM_{opt} and ARXISS methods are obtainable from the PBSID_{opt} solution by using the presented transformations. The CVA-MOESP and ISSCVA methods use an identical approach as the existing SSARX method, however with pre-estimates that are redefined in innovation form, again using the series-expansions. Simulation results, of which some provided in this chapter, have shown that the SSARX and ISSCVA methods give identical results. Both methods find the same state space model, even though their formulations are entirely different. Besides these methodologies, also another novel closed-loop SMI algorithm has been developed in this thesis, namely the IE-MOESP procedure. This approach uses the entire pre-estimated innovation sequence, and applies an LQ factorization in a MOESP like procedure. As such, this also yields an entirely new solution to the closed-loop subspace model identification problem.

The thesis presented a (mostly) self-sustaining description/explanation of (closed-loop) subspace model identification, which includes many algorithms in a highly structured manner. It is considered, together with the developed closed-loop SMI methods, as one of the main contributions of the thesis. Moreover, the presented framework also resulted in the development of a MATLAB toolbox that follows the interlinked nature of the ‘unifying’ description. In fact, it is this toolbox that was used for a comparative simulation study, and the practical verification of the developed SMI methods. The SMI description in this chapter, with the accompanying MATLAB toolbox can also be used as an interactive introductory tool into closed-loop subspace model identification for students or beginners.

The novel IEM_{opt} and ARXISS algorithms can be very useful for recursive implementations, since they can be considered numerically as efficient as the existing PBSID_{opt} . Especially ARXISS can be especially useful for recursive implementation since it estimates the system matrices directly from the HOARX problem. The recursive implementation (as well as theoretical performance) of these novel SMI methods can be considered nice areas for future research, for example in a small recursive simulation study that can precede the topic of adaptive subspace-based data-driven control for a ‘smart’ wind turbine rotor blade.

Five novel closed-loop SMI algorithms were developed, of which a simulation based comparison with SMI methods from literature was given. The new algorithms work remarkably well, with the CVA-MOESP algorithm considered a very satisfying SMI algorithm for use with the constructed 2-dimensional ‘smart wind turbine rotor blade model.

Bibliography

- [1] Yang, M., “Climate Change Drives Wind Turbines,” *Energy Policy*, Vol. 35, 2007, pp. 6546–6548.
- [2] Marsh, G., “Wind Turbines: How Big Can They Get,” *Rofocus*, Vol. 6, No. 2, 2005, pp. 22–28.
- [3] Snel, H., “Review of the Present Status of Rotor Aerodynamics,” *Wind Energy*, Vol. 1, 1998, pp. 46–69.
- [4] Jonkman, J., “NRELOffshrBslne5MW.doc,” Tech. rep., NREL/NWTC, 2007.
- [5] Joncas, S., Bergsma, O., and Beukers, A., “Power Regulation and Optimization of Offshore Wind Turbines Through Trailing Edge Flap Control,” *43rd AIAA Aerospace Sciences Meeting and Exhibit (AIAA 2005-394)*, Reno, 2005.
- [6] Andersen, P. B., Gaunaa, M., Bak, C., and Buhl, T., “Load Alleviation on Wind Turbine Blades using Variable Airfoil Geometry,” *European Wind Energy Conference, Athens*, 2006.
- [7] Bak, C., Gaunaa, M., Andersen, P. B., Buhl, T., Hansen, P., Clemmensen, K., and Moeller, R., “Wind Tunnel Test on Wind Turbine Airfoil with Adaptive Trailing Edge Geometry,” *45th AIAA Aerospace Sciences Meeting and Exhibit, Reno (AIAA-2007-1016)*, 2007.
- [8] Van Wingerden, J. W., Hulskamp, A. W., Barlas, T., Marrant, B., Van Kuik, G. A. M., Molenaar, D. P., and Verhaegen, M., “On the Proof of Concept of a ‘Smart Wind Turbine Rotor Blade for Load Alleviation,” *Wind Energy*, 2008.
- [9] Barlas, T., Van Wingerden, J. W., Hulskamp, A. W., and Van Kuik, G. A. M., “Closed-Loop Control Wind Tunnel Tests on an Adaptive Wind Turbine Blade for Load Reduction,” *46th AIAA Aerospace Science Meeting and Exhibit, Reno (AIAA-2008-1318)*, 2008.
- [10] Larsen, T. J., Madsen, H. A., and Thomsen, K., “Active load reduction using individual pitch, based on local blade flow measurements,” *Wind Energy*, Vol. 8, No. 1, 2005, pp. 67–80.

- [11] Buhl, T., Gaunaa, M., and Bak, C., "Load Reduction Potential Using Airfoils With Variable Trailing Edge Geometry," *AIAA-2005-1183*, 2005.
- [12] Politakis¹, G., W., H., and van Bussel, G., "Suppression of Classical Flutter using a Smart Blade," *46th AIAA Aerospace Sciences Meeting and Exhibit, Nevada*, 2008.
- [13] *Proceedings of the 56th IEA Topical Expert Meeting: The Application of Smart Structures for Large Wind Turbine Rotor Blades*, Albuquerque, USA, May 2008.
- [14] Buhl, T., Bak, C., Gaunaa, M., and Andersen, P. B., "Load Alleviation through Adaptive Trailing Edge Control Surfaces: ADAPWING Overview," *EWEC 2007, Milan*, 2007.
- [15] Basualdo, S., "Load Alleviation on Wind Turbine Blades Using Variable Airfoil Geometry," *Wind Engineering*, Vol. 29, 2005, pp. 169–182.
- [16] Buhl, T., Gaunaa, M., and Bak, C., "Potential Load Reduction Using Airfoils with Variable Trailing Edge Geometry," *Journal of Solar Energy Engineering*, Vol. 127, 2005, pp. 503–516.
- [17] Simms, D., Schreck, S., Hand, M., and Fingersh, L. J., "NREL Unsteady Aerodynamics Experiment in the NASA-Ames Wind Tunnel: A Comparison of Predictions to Measurements," Tech. rep., National Renewable Energy Laboratory, 2001.
- [18] de Korte, R., *Adaptive Data-Driven Control for a Smart Wind Turbine Rotor Blade: A Literature Study*, Master's thesis, Delft University of technology, 2008.
- [19] Pires, L., *Closed-Loop Identification for Wind-Turbines: A Literature Survey*, Master's thesis, Delft University of technology, 2008.
- [20] Dong, J. and Verhaegen, M., "Closed-loop Subspace Predictive Control for Fault Tolerant MPC design," Accepted for publication in 17th IFAC World Congress, 2008.
- [21] Hallouzi, R. and Verhaegen, M., "Reconfigurable Fault Tolerant Control of a Boeing 747 using Subspace Predictive Control," *AIAA Guidance, Navigation and Control Conference and Exhibit (AIAA-2007-6665)*, South Carolina, 2007.
- [22] Dong, J. and Verhaegen, M., "Cautious H₂ optimal control using uncertain Markov parameters identified in Closed Loop," Submitted for publication, Jan. 2008.
- [23] Peloubet, R., Haller, R., and Bolding, R., "On-line Adaptive Control of Unstable Aircraft Wing Flutter," *Proceedings of the 29th Conference on Decision and Control*, 1990.
- [24] Lau, E. and Krener, A. J., "LPV Control of Two Dimensional Wing Flutter," *Proceedings of the 38th Conference on Decision & Control, Phoenix*, 1999, pp. 3005–3010.
- [25] Van Wingerden, J. W., *Control of Wind Turbines with 'Smart' Rotors: Proof of Concept and LPV Subspace Identification*, Ph.D. thesis, Delft University of Technology, 2008.
- [26] Dowell, E. H., *A Modern Course in Aeroelasticity*, Kluwer Academy Publishers, 2004.
- [27] Favoreel, W. and De Moor, B., "SPC: Subspace Predictive Control," *Proceedings of the IFAC World Congress, Beijing, China*, 1999.

- [28] Hulshoff, S. J., *Course Notes AE4-930 Aeroelasticity*, Delft University of Technology, Course Period 2006-2007.
- [29] Waszak, M. R., "Modeling The Benchmark Active Control Technology Windtunnel Model for Application to Flutter Suppression," *AIAA Atmospheric Flight Mechanics Conference (AIAA Paper No. 96-3437)*, 1996.
- [30] Hariharan, N. and Leishman, J. G., "Unsteady Aerodynamics of a Flapped Airfoil in Subsonic Flow by Indicical Concepts," *Journal of Aircraft*, Vol. 33, 1996, pp. 855–868.
- [31] Ko, Jeonghwan & Kurdila, A. J. . S. T. W., "Nonlinear Control of a Prototypical Wing Section with Torsional Non-linearity," *Journal of Guidance, Control and Dynamics*, Vol. 20, No. 6, 1997.
- [32] Librescu, L., Na, S., Marzocca, P., Chung, C., and Kwak, M. K., "Active aeroelastic control of 2-D wing-flap systems operating in an incompressible flowfield and impacted by a blast pulse," *Journal of Sound and Vibration*, Vol. 283, 2005, pp. 685–706.
- [33] Lee, K. W. and Singh, S. N., "Global Robust Control of an Aeroelastic System Using Output Feedback," *Journal of Guidance, Control, and Dynamics*, Vol. 30, No. 1, 2007.
- [34] Andersen, P. B., *Load Alleviation on Wind Turbine Blades using Variable Airfoil Geometry (2D and 3D study)*, Master's thesis, Technical University of Denmark, Section of Fluid Mechanics, 2005.
- [35] Anderson Jr., J. D., *Fundamentals of Aerodynamics*, McGraw-Hill Higher Education, 3rd ed., 2001.
- [36] Snel, H., "Review of Aerodyanmics for Wind Turbines," *Wind Energy*, Vol. 6, 2003, pp. 203–211.
- [37] Hansen, M. O. L., Sorensen, J. N., Voutsinas, S., Srensen, N., and Madsen, H. A., "State of the art in wind turbine aerodynamics and aeroelasticity," *Progress in Aerospace Sciences*, Vol. 42, 2006, pp. 285–330.
- [38] Leishman, J. G., "Unsteady Lift of a Flapped Airfoil by Indicical Concepts," *Journal of Aircraft*, Vol. 31, 1994, pp. 288–297.
- [39] Gaunaa, M., "Unsteady 2D potential-flow forces on a thin variable geometry airfoil undergoing arbitrary motion," Tech. rep., Ris-R-1478(EN), 2006.
- [40] Theodorsen, T., "General Theory of Aerodynamic Instabiltiy and the Mechanism of Flutter," Tech. rep., Report No. 496, NACA, 1935.
- [41] Marretta, R. M. A. and Marino, F., "Wing flutter suppression enhancement using a well-suited active control model," *Proceedings of the I MECH E Part G Journal of Aerospace Engineering*, Vol. 221, 2007, pp. 441–452.
- [42] Kalogiros, J. and Wang, Q., "Aerodynamic Effects on Wind Turbulence Measurements with research Aircraft," *Journal of Atmospheric and Oceanic Technology*, Vol. 19, No. 10, 2002, pp. 1567–1576.

- [43] Leishman, J. G., "Challenges in Modeling the Unsteady Aerodynamics of Wind Turbines," *21st ASME Wind Energy Symposium and the 40th AIAA Aerospace Sciences Meeting (AIAA 2002-0037)*, Reno, 2002.
- [44] Hansen, M. H., Gaunaa, M., and Madsen, H. A., "A Beddoes-Leishman type dynamic stall model in state-space and indicial formulations," Tech. rep., Ris-R-1354(EN), 2004.
- [45] Van der Wall, B. G. and Leishman, J. G., "On the influence of Time-Varying Flow Velocity on Unsteady Aerodynamics," *Journal of the American Helicopter Society*, Vol. 39, No. 4, 1994, pp. 25–36.
- [46] Leishman, J., *Principles of Helicopter Aerodynamics*, Cambridge University Press, 2nd ed., 2006.
- [47] Ogata, K., *Modern Control Engineering*, Prentice-Hall, Int., 3rd ed., 1997.
- [48] Olds, S., *Modeling and LQR Control of a Two-Dimensional Airfoil*, Master's thesis, Virginia Polytechnic Institute and State University, Department of Mathematics, 1997.
- [49] Ljung, L., *System Identification: Theory for the user*, Prentice-Hall, Inc., 1999.
- [50] Larimore, W. E., "Canonical Variate Analysis in Identification, Filtering, and Adaptive Control," *Proceedings of the 29th Conference on Decision and Control, Hawaii*, 1990, pp. 596–604.
- [51] Verhaegen, M. and Verdult, V., *Filtering and System Identification: A Least Squares Approach*, Cambridge Univ. Pr., 2007.
- [52] Van Overschee, P. and De Moor, B., "N4SID : Subspace Algorithms for the Identification of Combined Deterministic-Stochastic Systems," *Automatica (Special Issue on Statistical Signal Processing and Control)*, Vol. 30, No. 1, 1994, pp. 75–93.
- [53] Van Overschee, P. and De Moor, B., "A Unifying Theorem for three Subspace System Identification Algorithms," *In Proc. 10th IFAC Symp. on Syst. Id., Copenhagen, Denmark*, 1994.
- [54] Ljung, L. and McKelvey, T., "Subspace identification from closed loop data," *Signal Processing*, Vol. 52, No. 2, 1996, pp. 209–215.
- [55] Favoreel, W., De Moor, B., Gevers, M., and Van Overschee, P., "Closed-loop model-free subspace-based LQG-design," *Proceedings of the 7th Mediterranean Conference on Control and Automation, Haifa, Israel*, 1999.
- [56] Larimore, W. E., "Automated Multivariable System Identification and Industrial Applications," *Proceedings of the American Control Conference*, 1999, pp. 1148–1162.
- [57] Favoreel, W., De Moor, B., and Van Overschee, P., "Model-free subspace-based LQG-design," *Proceedings of the American Control Conference, San Diego*, 1999.
- [58] Qin, J. S., "An overview of subspace identification," *Computers & Chemical Engineering*, Vol. 30, No. 10-12, 2006, pp. 1502–1513.
- [59] Chiuso, A., "The role of Vector Auto Regressive Modeling in Predictor-Based Subspace Identification," *Automatica*, Vol. 43, No. 6, 2007, pp. 1034–1048.

- [60] Jansson, M., “Subspace Identification and ARX Modeling,” *IFAC SYmp. on System Identification*, 2003.
- [61] Qin, J. S. and Ljung, L., “Closed-loop Subspace Identification with Innovation Estimation,” *Proceedings of SYSID 2003, Rotterdam*, 2003.
- [62] Chiuso, A. and Picci, G., “Consistency analysis of some closed-loop subspace Identification methods,” *Automatica*, Vol. 41, No. 3, 2005, pp. 377–391.
- [63] Chiuso, A., “The Role of Vector AutoRegressive Modeling in Subspace Identification,” *Proceedings of the 45th IEEE Conference on Decision & Control, San Diego*, 2006.
- [64] Larimore, W. E., “Maximum Likelihood Subspace Identification for Linear, Nonlinear, and Closed-loop Systems,” *American Control Conference, Portland*, 2005.
- [65] Lin, W., Qin, J. S., and Ljung, L., “Comparisons of Subspace Identification Methods for Systems Operating on Closed-Loop,” *Proceedings of IFAC World Congress, Prague*, 2005.
- [66] Chiuso, A., “On the relation between CCA and predictor-based subspace identification.” *IEEE Transactions on Automatic Control*, Vol. 52, No. 10, 2007, pp. 1795–1812.

Appendix A

Theodorsen's geometrical functions

This appendix gives the geometrical functions defined by Theodorsen in his solution for the harmonically oscillating thin airfoil in a incompressible irrotational flow [40]. The geometrical functions T_1, \dots, T_{14} are given in Table A.1.

Table A.1: Geometrical constants defined by Theodorsen [40].

T_1	$-\frac{1}{3} \cdot (2 + x_f^2) \cdot \sqrt{1 - x_f^2} + x_f \cdot \text{acos}(x_f)$
T_2	$x_f \cdot (1 - x_f^2) - \sqrt{1 - x_f^2} \cdot (1 + x_f^2) \cdot \text{acos}(x_f) + x_f \cdot \text{acos}^2(x_f)$
T_3	$-(\frac{1}{8} + x_f^2) \cdot \text{acos}^2(x_f) + \frac{1}{4} \cdot x_f \cdot \sqrt{1 - x_f^2} \cdot \text{acos}(x_f) \cdot (7 + 2 \cdot x_f^2) +$ $-\frac{1}{8} \cdot (1 - x_f^2) \cdot (5 \cdot x_f^2 + 4)$
T_4	$x_f \cdot \sqrt{1 - x_f^2} - \text{acos}(x_f)$
T_5	$-(1 - x_f^2) - \text{acos}^2(x_f) + 2 \cdot x_f \cdot \sqrt{1 - x_f^2} \cdot \text{acos}(x_f)$
T_6	T_2
T_7	$-(\frac{1}{8} + x_f^2) \cdot \text{acos}(x_f) + \frac{1}{8} \cdot x_f \cdot \sqrt{1 - x_f^2} \cdot (7 + 2 \cdot x_f^2)$
T_8	$-\frac{1}{3} \cdot \sqrt{1 - x_f^2} \cdot (2 \cdot x_f^2 + 1) + x_f \cdot \text{acos}(x_f)$
T_9	$\frac{1}{2} \cdot (\frac{1}{3} \cdot \sqrt{1 - x_f^2}^3 + x_a \cdot T_4)$
T_{10}	$\sqrt{1 - x_f^2} + \text{acos}(x_f)$
T_{11}	$(1 - 2 \cdot x_f) \cdot \text{acos}(x_f) + (2 - x_f) \cdot \sqrt{1 - x_f^2}$
T_{12}	$\sqrt{1 - x_f^2} \cdot (2 + x_f) - \text{acos}(x_f) \cdot (2 \cdot x_f + 1)$
T_{13}	$\frac{1}{2} \cdot (-T_7 - (x_f - x_a) \cdot T_1)$
T_{14}	$\frac{1}{16} + \frac{1}{2} \cdot x_a \cdot x_f$

Appendix B

Derivation predictor terms

This appendix shows the derivation of the terms in the H_f and G_f Toeplitz matrices in (3.10), which we define with:

$$\Lambda_\tau = \begin{bmatrix} \Lambda_\tau^B & \Lambda_\tau^K \end{bmatrix} = CA^{\tau-1} \begin{bmatrix} B & K \end{bmatrix}.$$

We show that these parameters are dependent on solely the Markov parameters Φ_τ in (3.8) and the D -matrix. Furthermore we also relate $\Gamma_f \tilde{K}$ with the predictor based solutions.

Derivation of Λ_τ^K

It is easy to see that $\Lambda_1^K = CK = \Phi_1^K$. For the derivation of Λ_2^K we first write out Φ_2^K :

$$\Phi_2^K = C(A - KC)K = \underbrace{CAK}_{\Lambda_2^K} - \underbrace{CKCK}_{\Phi_1^K \Lambda_1^K},$$

where we directly observe that Λ_2^K is given by:

$$\Lambda_2^K = CAK = \Phi_2^K + \Phi_1^K \Lambda_1^K.$$

We do the same for $\tau = 3$, given by the following:

$$\Phi_3^K = \underbrace{CA^2K}_{\Lambda_3^K} + \underbrace{-CAKCK + CKCKCK}_{-C(A-KC)KCK = -\Phi_2^K \Lambda_1^K} - \underbrace{CKCAK}_{\Phi_1^K \Lambda_2^K},$$

from which one can observe that:

$$\Lambda_3^K = CA^2K = \Phi_3^K + \Phi_2^K \Lambda_1^K + \Phi_1^K \Lambda_2^K.$$

And the derivation for $\tau = 4$ is shown next:

$$\begin{aligned} \Phi_4^K = & \underbrace{CA^3K}_{\Lambda_4^K} + \underbrace{-CA^2KCK + CAKCKCK + CKCAKCK - CKCKCKCK}_{-C(A-KC)(A-KC)KCK = -\Phi_3^K \Lambda_1^K} + \\ & + \underbrace{-CAKCAK + CKCKCAK}_{-C(A-KC)KCAK = -\Phi_2^K \Lambda_2^K} - \underbrace{CKCA^2K}_{\Phi_1^K \Lambda_3^K}, \end{aligned}$$

where we find:

$$\Lambda_4^K = CA^2K = \Phi_4^K + \Phi_3^K \Lambda_1^K + \Phi_2^K \Lambda_2^K + \Phi_1^K \Lambda_3^K.$$

Looking at the previous derivations of Λ_τ^K , we observe that the solutions for any τ can be written as the series expansion:

$$\Lambda_i^K = CA^{i-1}K = \Phi_i^K + \sum_{j=1}^{i-1} \Phi_{i-j}^K \cdot \Lambda_j^K, \quad \text{for } i = 1, 2, 3, \dots \quad (\text{B.1})$$

Derivation of Λ_τ^B

We show the derivation of $\Lambda_\tau^B = CA^{\tau-1}B$ in the same manner as done earlier for Λ_τ^K . However, because the D -matrix enters into the equation we also define the situation $\Phi_0^{\bar{B}} = \Lambda_0^B = D$. The situations for $\tau = 1, 2, 3$ are shown by the following. Thereafter a sort-like series expansion as for Λ_τ^K is presented.

$$\Lambda_1^B = CB = \underbrace{C(B - KD)}_{\Phi_1^{\bar{B}}} + \underbrace{CKD}_{\Phi_1^K \Lambda_0^B}.$$

$$\Phi_2^{\bar{B}} = C(A - KC)(B - KD) = \underbrace{CAB}_{\Lambda_2^B} + \underbrace{-CAKD + CKCKD}_{-C(A-KC)KD = -\Phi_2^K \Lambda_0^B} - \underbrace{CKCB}_{\Phi_1^K \Lambda_1^B},$$

$$\Lambda_2^B = CAB = \Phi_2^{\bar{B}} + \Phi_2^K \Lambda_0^B + \Phi_1^K \Lambda_1^B.$$

$$\begin{aligned} \Phi_3^{\bar{B}} &= C(A - KC)^2(B - KD) = \underbrace{CA^2B}_{\Lambda_3^B} + \underbrace{-CAKCB + CKCKCB}_{-C(A-KC)KCB = -\Phi_2^K \Lambda_1^B} - \underbrace{CKCAB}_{\Phi_1^K \Lambda_2^B} + \\ &\quad + \underbrace{-CA^2KD + CAKCKD + CKCAKD - CKCKCKD}_{-C(A^2 - AKC - KCA + KCKC)KD = C\bar{A}^2KD = \Phi_3^K \Lambda_0^B}, \end{aligned}$$

$$\Lambda_3^B = CA^2B = \Phi_3^{\bar{B}} + \Phi_3^K \Lambda_0^B + \Phi_2^K \Lambda_1^B + \Phi_1^K \Lambda_2^B.$$

Based on the above equations, it is pretty straight forward to derive each Λ_i^B by additionally defining $\Phi_0^K = 0$, as the series expansion:

$$\begin{aligned} \Lambda_i^B &= \Phi_i^{\bar{B}} + \sum_{j=0}^{i-1} \Phi_{i-j}^K \cdot \Lambda_j^B, \quad \text{giving:} \\ \Lambda_0^B &= D, \\ \Lambda_i^B &= CA^{i-1}B, \quad \text{for } i = 1, 2, 3, \dots \end{aligned} \quad (\text{B.2})$$

Derivation of $\Gamma_\tau \bar{\mathcal{K}}$

With an analogous procedure it is possible to relate the solution $\bar{\Gamma}_f \bar{\mathcal{K}}$ with $\Gamma_f \bar{\mathcal{K}}$, which has the extended observability matrix in innovation form. Beforehand let us define the matrices:

$$\Xi_\tau = C\bar{A}^\tau \bar{\mathcal{K}} \quad \Theta_\tau = CA^\tau \bar{\mathcal{K}}$$

Assuming that a solution of $\bar{\Gamma}_f \bar{\mathcal{K}}$ (and thus Ξ_τ) is known, for example from the PBSID_{opt} algorithm obtainable from the HOARX estimate, than a solution of $\Gamma_f \bar{\mathcal{K}}$ can be obtained from the series expansion:

$$\Theta_i = CA^i \bar{\mathcal{K}} = \Xi_i + \sum_{j=0}^{i-1} \Phi_{i-j}^K \cdot \Theta_j. \quad (\text{B.3})$$

A derivation of this is given by the following.

$$\Theta_0 = C\bar{\mathcal{K}} = \Xi_0$$

$$\Xi_1 = C(A - KC)\bar{\mathcal{K}} = \underbrace{CA\bar{\mathcal{K}}}_{\Theta_1} - \underbrace{CKC\bar{\mathcal{K}}}_{\Phi_1^K \Theta_0},$$

$$\Theta_1 = CA\bar{\mathcal{K}} = \Xi_1 + \Phi_1^K \Theta_0.$$

$$\Xi_2 = C(A - KC)(A - KC)\bar{\mathcal{K}} = \underbrace{CA^2\bar{\mathcal{K}}}_{\Theta_2} + \underbrace{-CAKCK\bar{\mathcal{K}} + CKCKC\bar{\mathcal{K}}}_{-C(A-KC)KC\bar{\mathcal{K}} = -\Phi_2^K \Theta_0} - \underbrace{CKCA\bar{\mathcal{K}}}_{\Phi_1^K \Theta_1},$$

$$\Theta_2 = CA^2\bar{\mathcal{K}} = \Xi_2 + \Phi_2^K \Theta_0 + \Phi_1^K \Theta_1$$

Note that the solutions in (3.32) and (B.3) are identical (for $= 0$) to the closed-loop Subspace Predictive Control (SPC) problem in [20, 21]. Nevertheless, in these papers the purpose is not to perform Subspace Model Identification and the series expansions are not directly linked with the solutions for a system representation in innovation form. Moreover, SPC is said to use the PBSID_{opt} method, and although the equivalence is probably satisfied for the SPC problem, when looking at SMI the fact rises that it represents an alternate identification method for a system in innovation form. Furthermore, it is nice to see the relation between innovation and predictor form the derivation in this section presents for the identification problem.