Estimation of temporal and spatio-temporal nonlinear descriptor systems

by

Julian Mercieca

Submitted to the Department of Automatic Control and Systems Engineering
in partial fulfilment of the requirements for the degree of

Doctor of Philosophy

at the

University of Sheffield

January 2018

The University Of Sheffield.

Acknowledgments

First and foremost, I am obliged to my supervisor, Prof. Visakan Kadirkamanathan, who provides an atmosphere that truly allows graduate students to transcend their traditional role and assume a position as an equal contributor to our discipline.

A special thanks is due to Dr. Parham Aram who has always generously made himself available for discussion through the years. I am grateful for his unwavering commitment in providing technical help and support whenever I needed it.

I have also been fortunate to have had the opportunity to collaborate with Dr. Bryn Jones, whose knowledge on fluid flow led to informative and enjoyable collaborations on wind field estimation. I would also like to acknowledge Dr. Paul Towers for the insightful discussions on fluid dynamics.

In my time in Sheffield, I have had the pleasure of sharing experiences with the rest of the research group, namely Angesh, Anastasia, Yang, Andrew, Krish, Geoff, Yashar, Lingli, Tara, William, Jinny, Buket, Adam, Peter and others whom I thank for having contributed to my entire educational experience. I would also like to thank my friends in Malta, particularly Emmanuel and Kris, for providing a smile when much needed.

Finally, I am forever indebted to my parents and sister for their support throughout my entire education, not only my graduate studies. Their love and encouragement was worth far more than any degree.

Abstract

As advances in the remote sensing of fluid flows forge ahead at an impressive rate, we face an increasingly compelling question of how best to exploit this progress. Light detection and ranging (LIDAR) measurement equipment still presents the problems of having only radial (line-of-sight) wind speed measurements (Cyclops' dilemma). Substantial expanses of unmeasured flow still remain and range weighting errors have a considerable influence on LIDAR measurements. Clearly, more information needs to be extracted from LIDAR data and an estimation problem naturally arises. A key challenge is that most established estimation techniques, such as Kalman filters, cater for systems that are finite-dimensional and described by ordinary differential equations (ODEs). By contrast, many fluid flows are governed by the Navier-Stokes equations, which are nonlinear partial differential-algebraic equations (PDAEs).

With this motivation in mind, this thesis proposes a novel statistical signal processing framework for the model-based estimation of a class of spatio-temporal nonlinear partial differential-algebraic equations (PDAEs). The method employs finite-dimensional reduction that converts this formulation to a nonlinear DAE form for which new unscented transform-based filtering and smoothing algorithms are proposed. Gaussian approximations are derived for differential state variables and more importantly, extended to algebraic state variables. A mean-square error lower bound for the nonlinear descriptor filtering problem is obtained based on the posterior Cramér-Rao inequality.

The potential of adopting a descriptor systems approach to spatio-temporal estimation is shown for a wind field estimation problem. A basis function decomposition method is used in conjunction with a pressure Poisson equation (PPE) formulation to yield a spatially-continuous, strangeness-free, reduced-order descriptor flow model which is used to estimate unmeasured wind velocities and pressure over the entire spatial region of interest using sparse measurements from wind turbine-mounted LIDAR instruments. The methodology is validated for both synthetic data generated from large eddy simulations of the atmospheric boundary layer and real-world LIDAR

measurement data. Results show that a reconstruction of the flow field is achievable, thus presenting a validated estimation framework for potential applications including wind gust prediction systems, the preview control of wind turbines and other spatio-temporal descriptor systems spanning several disciplines.

Contents

List of Figures

List of Tables

List of Acronyms

CML	coupled map lattice
DAE	differential-algebraic equation
EKF	extended Kalman filter
EM	expectation-maximisation
EnKF	ensemble Kalman filter
FHKF	Fourier-Hermite Kalman filter
GP	Gaussian process
IDE	integro-difference equation
LES	large eddy simulation
LIDAR	light detection and ranging
MC	Monte Carlo
MCMC	Markov chain Monte Carlo
MHE	moving horizon estimator
MRMSE	mean root mean square error
ODE	ordinary differential equation
PCRB	posterior Cramér-Rao bound
PDE	partial differential equation
PDAE	partial differential-algebraic equation
PF	particle filter
PPE	pressure Poisson equation
RMSE	root mean square error
SLF	statistically linearised filter
SOWFA	simulator for offshore wind farm applications

STARMA	space-time auto-regressive moving-average
UKF	unscented Kalman filter
UKS	unscented Kalman smoother
URTSS	unscented Rauch-Tung-Striebel smoother
UT	unscented transformation

Chapter 1

Introduction

The burgeoning offshore wind energy industry has been fuelling widespread research efforts to improve the economic efficiency of energy production. The overall cost of energy is impacted by various factors that are negatively affected by wind unsteadiness. Violent gusts of wind often disrupt construction and maintenance operations. This forces the maintenance crew to follow a more conservative approach than would otherwise be required if prior knowledge of oncoming gusts is available. Although meteorological forecasts do provide averaged wind speeds over large areas, this is often of insufficient fidelity to obtain finer spatial and faster temporal dynamics of wind gusts. Wind gusts are a primary cause of extreme and fatigue load and result in a decreased operational life expectancy and increased maintenance. Construction costs are increasingly high as heavier and stronger towers are needed [1]. It is therefore unsurprising that the work in [2] reports on the crucial need to develop technologies that improve turbine reliability, in clear view of the significant maintenance costs endured particularly at remote offshore locations.

This has sparked off a keen interest in developing the ability to sample an oncoming wind field using light detection and ranging (LIDAR) instrumentation in order to maximise energy production and mitigate structural loads. Such wind information is vital for the preview control of wind turbines and constitutes one of the present control research challenges as documented in *The Impact of Control Technology* report published by the IEEE Control Systems Society [3]. LIDAR equipment now provides

useful bandwidth and ranging specifications that enable accurate and fast radial velocity measurements at specific locations away from the turbine [4, 5]. However, LIDAR equipment only provides radial (line-of-sight) wind speed measurements (Cyclops' dilemma). Substantial expanses of unmeasured flow still remain and range weighting errors have a considerable influence on LIDAR measurements. Clearly, more information needs to be extracted from LIDAR data. How to exploit such measurements for oncoming gust detection, or how to employ this information within preview control schemes, remains unclear [6, 7, 8]. Such controllers would depend on the accuracy of wind field prediction which therefore require wind flow estimation tools that predict wind gusts using sparse spatio-temporal wind velocity measurements [9, 4, 10]. This action prevents the possible blade damage due to strong wind gusts if the blade pitch is varied accordingly in a timely fashion [6, 7, 11, 12, 13]. The estimation process should link observations to regions of flow which are not directly measured.

Many fluid flows are described by the Navier-Stokes equations [14]. However, owing to their intractable nature in the original partial differential-algebraic equation (PDAE) form, the estimation of spatio-temporal systems described by these equations remains challenging since most established estimation methods cater for systems that are finite-dimensional and described by ordinary differential equations (ODEs). Keeping the full descriptor formulation is a particularly attractive consideration since the resulting pressure field description would become important for other flow applications, such as air flow for transport vehicles. The difference in pressure across a vehicle is known to result in pressure drag [15] that in turn makes up 80% of ground transportation drag [16]. It is estimated that 16% of the total energy consumption in the US is a consequence of aerodynamic drag [16].

Several strategies can tackle the aforementioned problems, however the literature treating appropriate methodologies is somewhat fragmented and lacks a principled approach that caters for different situations. This thesis intends to address this gap by proposing a new approach to the study of estimation for temporal and spatio-temporal generalised state-space systems. The methodology presented is applicable to a variety of situations and delivers a framework that adopts a model-based approach

to estimation.

The research scope is given in detail together with a clear thesis outline in Section 1.2. The motivation, however, is first put in context by first elaborating on the concepts of descriptor and spatio-temporal models and the associated estimation process in Section 1.1.

1.1 An introduction to models and estimation

1.1.1 Descriptor system models

The *model* is a key building block for the analysis of data. The behaviour of an observable phenomenon may be described in terms of independent variables, such as space and/or time using a model. The underlying system is usually dynamic, which means that the current output value depends not only on the present external stimuli, but also on earlier values [17]. Many complex phenomena, however, are naturally described by formulations that extend beyond the traditional purely dynamic or purely static models [18]. Whenever systems are described by their interconnection of subsystems comprising differential equations and algebraic constraints, such situations naturally arise [19]. For these cases, systems are modelled as a descriptor system, also known as a differential-algebraic equation (DAE) system, generalised state-space system, singular system or implicit system [20]. Descriptor system applications are abundant and include chemical engineering [21, 22], fluid dynamics [23], robotics [24], electronic network modelling [25], image modelling [26] and economy [27, 18] to name a few. This general class of models enables modelling several time-evolutionary phenomena such as ordinary state-space equations, combinations of static and dynamic equations and noncausal systems [18], and has therefore been a topic of research over recent decades.

Written in semi-explicit formulation, DAEs are systems of equations having the

form [28]:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{z}, t), \tag{1.1a}$$

$$\mathbf{0} = \mathbf{g}(\mathbf{x}, \mathbf{z}, t), \tag{1.1b}$$

where $\mathbf{x} \in \mathbb{R}^{n_x}$ is a vector of differential state variables, $\mathbf{z} \in \mathbb{R}^{n_z}$ is a vector of algebraic state variables, $\mathbf{f}(\cdot)$ is the dynamic model function of differential states, $\mathbf{g}(\cdot)$ is a mapping of algebraic equations and t denotes time. In general, descriptor systems differ from systems of ordinary differential equations (ODEs) [29]. In constrained ODE systems, all process states evolve as differential equations, subject to algebraic constraints that restrict state evolution. In descriptor systems, however, the evolution of some states is not described by differential equations. Such states, known as algebraic, follow an evolution that is entirely governed by the evolution of differential states, such that all algebraic constraints are obeyed [30]. As a result of this important difference, a descriptor system cannot be simply handled as a constrained ODE system [30]. Descriptor systems are generally characterised by their differentiation index, typically defined as the minimum number of differentiations required in order to obtain an explicit ODE formulation [31]. A more formal definition of the differentiation index is provided in Appendix A.

1.1.2 Spatio-temporal models

Spatio-temporal systems are systems that evolve over both space and time [32]. As a result, they are ubiquitous in an array of scientific disciplines including bacterial and viral infection spread [33, 34], biology [35, 36, 37], neuroscience [38, 39, 40], environmental science [41, 42, 43, 44], conflict dynamics [45], mobile sensor networks [46] and image processing [47]. This widespread occurrence has motivated numerous research works over recent decades in a bid to provide mathematical models that capture the behaviour of spatio-temporal phenomena, where the data spanning both space and time should not be treated as statistically independent variables [48]. These models make up the basis for the simulation, design and analysis of systems.

The numerous spatio-temporal models that appear in the literature may be generally grouped into two model classes: geostatistical models [49] and dynamic models [48, Chapters 7-9]. The former modelling paradigm employs a statistical description, usually as mean and covariance functions, such as underlying Gaussian random functions [50, 49]. Dynamic models generally consist of differential or difference equations that explicitly describe the systems' evolution in space and/or time. Dynamic models are therefore typically associated with a mechanistic approach to systems, where estimated parameters usually have a physical meaning or a direct connection to the system behaviour such as partial differential equations [51]. Both modelling paradigms may sometimes be interchangeable descriptions of the same process [52]. A number of advantages of dynamic models over geostatistical models are highlighted in [53], including the more computationally efficient parameter estimation framework when making use of signal processing tools for dynamic models and the associated covariance functions that could be representative of models describing unnatural features. Additionally, estimation mechanisms related to dynamic models can easily handle missing or incomplete data. Their amenability to engineering scenarios make dynamic models the main focus of this thesis.

In numerous circumstances, spatio-temporal processes can only be partially observed and an estimation problem naturally arises. The estimation of internal states is critical when controlling, monitoring and diagnosing several engineering processes. A cost-efficient approach to regularly monitor state variables employs model-based state estimation methods to infer unmeasured and/or rarely measured variables.

State estimators may be designed using stochastic (Bayesian) or deterministic approaches [54, 55], with the former approach being our main focus throughout this thesis. In the majority of cases, a system is deterministic and free of any unexplained events. In such a case, using statistical tools to estimate the states and/or parameters does not imply that the system is, of itself, stochastic. Rather, exact prediction is very difficult, measurements are noisy and any non-systematic errors introduced must be removed. For these situations, a separate model class, known as stochastic models, naturally describe elements that appear to be random. The concept of uncertainty

is therefore introduced in the model in the form of a stochasticity without obscuring the dominant dynamics of the underlying process. Taking for instance groundwater flow in aquifers [56] and pollution spread [57], these would require inordinate amounts of deterministic models for their exact prediction. In such circumstances, the apparent randomness is a result of several factors including climate factors (e.g. precipitation and wind) and social factors (e.g. urbanisation and market volatility) that are impossible to model deterministically. For our ultimate purpose of estimation, stochastic models make it possible to approximate the effects for which exact deterministic models are unavailable.

In this context, whilst models may further our understanding of the underlying physical phenomena, their strength lies in the predictive ability of what will happen over the next seconds, hours, days or years, depending on the application. Knowledge of future events would allow one to anticipate certain situations and act accordingly, such as adjusting the blade pitch in response to a predicted wind gust in a wind turbine preview control scheme. Stochastic models, therefore, allow the future to be predicted with a degree of probability. Taking meteorology as an example, the wind velocity at a specific spatial location is hard to be exactly predicted in advance, but if an appropriate stochastic model is available, wind velocity may be predicted with confidence intervals. In summary, stochasticity therefore provides a measure of uncertainty. The more the unexplained and unmodelled dynamics in the model, the higher the uncertainty in the estimates. Consequently, stochastic models allow for an adequate degree of flexibility in systems characterised by high uncertainties and is therefore the class of models that will be extensively studied in this work.

A lengthy exposure to nonlinear Bayesian state estimation is given in several books [58, 59, 60, 61, 62], while more recent progress is described in [63, 64, 30, 65]. Recent efforts also present the emergence of deterministic and random sampling-based estimation methods as part of research works in the area of unconstrained sequential estimation. Significant achievements in sigma point and particle filters are discussed in [66, 67, 68, 69, 70].

1.1.3 Bayesian state estimation

In mathematical terms, estimation methods are aimed at performing optimal filtering and smoothing which are essentially statistical inversion problems in which the unknown quantities, for descriptor systems in discrete time k, are time series values $\{\mathbf{X}_0, \mathbf{X}_1, \mathbf{X}_2, \ldots\}$, which are observed using a set of noisy measurements $\{\mathbf{y}_1, \mathbf{y}_2, \ldots\}$, where $\mathbf{X}_k = (\mathbf{x}_k^\top, \mathbf{z}_k^\top)^\top$ and the superscript $^\top$ denotes the transpose operator.

The statistical inversion estimates the hidden states $\mathbf{X}_{0:K} = \{\mathbf{X}_0, \mathbf{X}_1, \mathbf{X}_2, \ldots \mathbf{X}_K\}$ using the observed measurements $\mathbf{y}_{1:K} = \{\mathbf{y}_1, \mathbf{y}_2, \mathbf{y}_3, \ldots, \mathbf{y}_K\}$, where the discrete time steps, denoted by k, run from 0 to K. In a Bayesian context, we therefore need to obtain the joint posterior distribution of all states given all observations. This can in theory be achieved by applying Bayes' rule, as follows [65]:

$$p(\mathbf{X}_{0:K}|\mathbf{y}_{1:K}) = \frac{p(\mathbf{y}_{1:K}|\mathbf{X}_{0:K})p(\mathbf{X}_{0:K})}{p(\mathbf{y}_{1:K})}, \tag{1.2}$$

where $p(\mathbf{y}_{1:K}|\mathbf{X}_{0:K})$ is the likelihood model for the measurements, $p(\mathbf{X}_{0:K})$ is the prior distribution given by the descriptor model and $p(\mathbf{y}_{1:K})$ is a constant of normalisation given by

$$p(\mathbf{y}_{1:K}) = \int p(\mathbf{y}_{1:K}|\mathbf{X}_{0:K})p(\mathbf{X}_{0:K})\ d\mathbf{X}_{0:K}. \tag{1.3}$$

A critical disadvantage of this full posterior formulation is that for every new measurement, the full posterior distribution has to be recomputed. For the problem of dynamic estimation being solved in this thesis, measurements are usually obtained one at a time. After each measurement, the best possible estimate must be computed. The dimensionality of the full posterior distributions rapidly increases as the number of time steps increases, leading to a surge in the computational complexity of every single time step. Irrespective of the available computational power, the computations would eventually become intractable and without additional information or approximations, the problem of full posterior computation will still stand.

The aforementioned problem is only an issue if at each time interval, the full

posterior distribution is computed. If a portion of the marginal state distributions would suffice, the computations will carry an order of magnitude less [65]. This may be achieved by restricting the class of dynamic models to probabilistic Markov sequences where (i) an initial distribution $p(\mathbf{x}_0)$ is specified for $\mathbf{x}_o$ at the initial time interval $k = 0$, (ii) a descriptor model describes the system dynamics together with the associated uncertainties in terms of a Markov sequence given as transition probability distributions $p(\mathbf{x}_{k+1}|\mathbf{X}_k)$ and $p(\mathbf{z}_k|\mathbf{x}_k)$ and (iii) a measurement model describes the dependence of measurements $\mathbf{y}_k$ on the present sate $\mathbf{X}_k$ through the conditional probability distribution of the observation given the states, i.e. $p(\mathbf{y}_k|\mathbf{X}_k)$.

The descriptor system formulation given by equations (1) may therefore be interpreted in terms of the following general probabilistic generalised state-space model given in discrete time:

$$\mathbf{x}_0 \sim p(\mathbf{x}_0), \tag{1.4a}$$

$$\mathbf{x}_{k+1} \sim p(\mathbf{x}_{k+1}|\mathbf{X}_k), \tag{1.4b}$$

$$\mathbf{z}_k \sim p(\mathbf{z}_k|\mathbf{x}_k), \tag{1.4c}$$

$$\mathbf{y}_k \sim p(\mathbf{y}_k|\mathbf{X}_k). \tag{1.4d}$$

Since the full joint distribution of states is computationally tedious and unnecessary for use in real-time scenarios, this thesis considers Bayesian filtering where filtering distributions computed by the Bayesian filter are the marginal distributions of the present state $\mathbf{X}_k$ given the present and past observations $\mathbf{y}_{1:k} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_k\}$, that is, $p(\mathbf{X}_k|\mathbf{y}_{1:k})$ for $k = 1, \ldots, K$.

All state estimation algorithms make the assumption that the corresponding model parameters are accurately known. However, if a number of unknown parameters must be estimated in addition to the states, a joint state-parameter estimation scheme is required [71]. Estimating and building system models from experimental data is referred to as system identification and is essential for system emulation, system response prediction for various inputs and investigation of various design circumstances [72]. As a result, the accuracy of the system representation would influence the va-

lidity of all the simulation, design and analysis and has thus motivated significant efforts to derive joint state-parameter estimation methods, namely the Markov Chain Monte Carlo (MCMC) methods [73], expectation-maximisation (EM) algorithms [74] and their variants.

The EM algorithm requires smoothed state estimates and this thesis therefore also considers Bayesian smoothing, which computes the marginal state distributions given an interval $\mathbf{y}_{1:K} = \{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_K\}$ of measurements with $K > k$, that is, $p(\mathbf{X}_k|\mathbf{y}_{1:K})$ for $k = 1, \ldots, K$. The computation of the filtering and smoothing distributions only need a constant number of computations for each time interval, thereby avoiding processing a long time series.

1.2 Motivation and thesis organisation

Although the state estimation for linear descriptor systems has been well developed over the last few decades, the nonlinear descriptor system problem remains an open-ended research area and the design of observers and filters for such systems is relatively recent [75]. This is probably due to the complexity of this class of systems [76]. This section describes the limitations with the present state-of-the-art and proceeds to highlight a summary of contributions in the thesis organisation.

1.2.1 Research scope and objectives

With high-dimensional estimation problems in mind, this work considers the Kalman filter [77], which yields closed form solutions for linear Gaussian filtering problems where by assuming linear Gaussian models, the posterior distribution is exactly Gaussian with no numerical approximation required. Since the Bayesian optimal filtering equations are typically computationally cumbersome and all problems considered in this thesis are nonlinear, the unscented Kalman filter (UKF) [78, 79] will be the main filtering algorithm studied here. The UKF approximates the propagation of densities as these undergo the nonlinear transformations of observation functions using the unscented transform, resulting in Gaussian approximations.

The smoothing problem is tackled using a Rauch-Tung-Striebel-based smoothing algorithm [80], which is a closed form smoothing technique for Gaussian linear state-space models. For our nonlinear problems, the unscented Rauch-Tung-Striebel smoother (URTSS) [81, 82] is considered, which is an approximate nonlinear smoother that corresponds to the UKF method.

Despite the considerable research efforts in the field of state estimation for nonlinear descriptor systems, several voids still remain and a principled methodology adapted to different classes of systems is lacking in the literature. The main issues and their implications will now be discussed.

- A careful analysis of the recent advances in extending Kalman filtering to nonlinear descriptor systems (e.g. [21, 83, 84, 85] and references therein) show that the covariance of the algebraic states remains unknown. The statistical properties of the algebraic states are inherent to the system dynamics and any estimator should make use of this information to complete the state estimation process and in turn be utilised through successive computations of the algorithm.

- In the Kalman filtering application to nonlinear descriptor systems, the algebraic equations are always assumed to be free of any uncertainty. Although this is typically the case, whenever model reduction schemes are introduced and model uncertainties exist, treating the algebraic equations deterministically becomes infeasible and virtually impossible if the effects of model approximations and uncertainties are to be accounted for. The availability of a stochastic estimation framework is of paramount importance if such effects are to be approximated. This motivates further research into the estimation of nonlinear descriptor systems.

- From the literature, it is evident that most of the work has considered the observer and filtering problems for nonlinear DAE systems. For EM algorithms, however, smoothed estimates are a requirement but to the best of our knowledge, smoothers for nonlinear descriptor systems are not available in the literature and remain largely unexplored. This would be essential for any state and

parameter estimation problem that arise for applications requiring a nonlinear DAE formulation.

- The nonlinear PDAE estimation problem has hitherto not been extensively addressed in the signal processing literature. This general class of systems, known to accurately describe several physical processes, is a natural description of the underlying phenomena and easily describes spatial heterogeneity. The challenge here is to convert the nonlinear PDAE model to one that is amenable to signal processing techniques in finite-dimensional form whilst retaining the original physical meaning which would make it a good fit for control purposes.

- An important application of nonlinear spatio-temporal descriptor models is fluid flow. A generalised state-space approach to the estimation of fluid flow still represents an open problem, both as a consequence of the complexity of the original Navier-Stokes equations and its pressure field description, as well as the lack of nonlinear PDAE estimation tools available.

- For the nonlinear DAE estimation problem at hand, a filtering error analysis in terms of the posterior Cramér-Rao bound (PCRB) is lacking in the literature. Knowing the achievable filtering performance is crucial for estimation algorithms. The computation of lower bounds is an important error analysis method that is indicative of performance limitations and would validate any imposed performance requirements.

1.2.2 Original contributions and thesis outline

In the light of the foregoing, this work addresses the problem of how to create a useful estimation framework for nonlinear temporal and spatio-temporal descriptor systems using a model-based approach for estimation. A summary of original contributions is given throughout the following thesis outline:

- *Chapter 2* reviews methods that dominate the spatio-temporal model and estimation literature with the aim of outlining the context that surrounds this

current avenue of research. The concepts introduced here describe dynamic spatio-temporal models and how these are reduced to a tractable form that may be readily handled by state estimation tools discussed in the same chapter. This highlights the gaps that form the basis for the novel estimation frameworks put forward in this thesis.

- *Chapter 3* proposes filters and smoothers for nonlinear temporal descriptor systems having deterministic or stochastic algebraic equations. The estimation methods retain the original DAE formulation for use in the filtering and smoothing algorithms. This paves the way for the spatio-temporal estimation schemes implemented in later chapters.

  The first part of the chapter develops the discrete-time UKF and URTSS algorithms for nonlinear DAE systems having stochastic differential equations and deterministic algebraic equations. The filtered and smoothed solutions yield the mean and covariance of both algebraic and differential state estimates. In addition to differential state sigma points in standard UKF and URTSS formulations, the algebraic state sigma points are also incorporated to ensure consistency in the DAE solution. An electrochemical case study demonstrates the performance of the proposed filter and smoother.

  The second part of the chapter derives a new descriptor form of the discrete-time UKF and URTSS algorithms having stochastic differential and algebraic equations. Unlike previous approaches, we propose to compute and exploit both the mean and the covariance of the algebraic state estimates and ensure that the sigma points encode the mean and covariance of all differential and algebraic states. The differential and algebraic state filtering and smoothing distributions are derived as unscented transform-based Gaussian approximations. The performance of the proposed estimation methods is demonstrated by a numerical example.

- *Chapter 4* presents an estimation framework for spatio-temporal nonlinear DAE systems, where a basis function decomposition method converts this formula-

tion to a reduced finite-dimensional nonlinear descriptor form. Transforming the nonlinear PDAE to a DAE requires the consideration of model approximation effects. By representing the latter effects stochastically, the new unscented transform-based filtering algorithm derived in Chapter 3 is employed. For important spatio-temporal applications, an estimation error performance analysis is a requirement and this is therefore tackled by deriving the mean square error lower bounds for the nonlinear descriptor filtering case, based on the posterior Cramér-Rao inequality. The novel methods are successfully implemented and validated through the accurate estimation of wind flow velocity and pressure given sparse noisy velocity measurements from realistic atmospheric boundary layer wind flow data.

- *Chapter 5* derives a generalised state-space flow model in nonlinear, spatially-continuous form. The key to obtaining a strangeness-free DAE of differentiation index 1 is the reformulation of the Navier-Stokes equations using the pressure Poisson equation (PPE) in conjunction with basis function decomposition. The latter enables the user to represent the flow field by choosing an appropriate number and placement of basis functions, or state variables, that is independent of the number and placement of observations and which allows a computationally efficient estimation procedure. The performance of the proposed reduced-order model and estimator are validated for both synthetic data obtained using large eddy simulations of the atmospheric boundary layer and real-world LIDAR measurement data obtained from a nacelle-mounted LIDAR unit.

- *Chapter 6* provides a summary of the thesis. The contributions of this work are placed in the bigger picture and a discussion of future work is presented. The potential application of the thesis contributions to different real-world problems is discussed. It is envisaged that the proposed framework contributes towards further research into estimation and control of spatio-temporal descriptor systems.

This thesis has resulted in the following publications:

- J. Mercieca and V. Kadirkamanathan, "Estimation and identification of spatio-temporal models with applications in engineering, healthcare and social science," Annual Reviews in Control, vol. 42, pp. 285-298, 2016.
- J. Mercieca, P. Aram, and V. Kadirkamanathan, "Unscented Rauch-Tung-Striebel smoothing for nonlinear descriptor systems," in Proceedings of the 2015 European Control Conference (ECC), pp. 491-496, 2015.

The following three papers are currently under review:

- J. Mercieca, P. Aram, B. Jones, V. Kadirkamanathan, "State Estimation of Spatio-Temporal Nonlinear Descriptor Systems," Under review.
- J. Mercieca, P. Aram, B. Jones, V. Kadirkamanathan, "A Spatio-temporal Estimation Framework for Real-World LIDAR Wind Speed Measurements," Under review.
- J. Mercieca, P. Aram, V. Kadirkamanathan, "An Unscented Rauch-Tung-Striebel Smoother for Nonlinear Differential-Algebraic Equations," Under review.

Chapter 2

Estimation of spatio-temporal models

This chapter gives the background context necessary for the development of estimation schemes for temporal and spatio-temporal descriptor formulations provided in later chapters. The first portion of this chapter, Section 2.1, reviews various dynamic spatio-temporal models and accompanying theoretical properties. Whilst several models have been proposed, only the most relevant are discussed, namely the space-time auto-regressive moving-average (STARMA) model, the coupled map lattice (CML), the integro-difference equation (IDE), the partial differential equation (PDE) and the partial differential-algebraic equation (PDAE). The strengths and limitations of each model scheme are described and contrasted. Of particular interest for our work is the PDAE, which is a general model class which describes fields over a continuous spatial domain that is appropriate for spatially heterogenous dynamics and provides an appropriate physical description of the underlying phenomenon.

Section 2.2 is devoted to describing model reduction methods to ultimately obtain a finite-dimensional representation which facilitates prediction and control schemes. In this thesis, temporal and spatio-temporal behaviour is represented in a generalised state-space form, so Section 2.3 treats common methods associated with state estimation for state-space models. Section 2.4 provides a literature review of state estimation methods for nonlinear descriptor systems ahead of the estimation schemes developed in Chapter 3. Algorithms which are implemented and investigated throughout this work are particularly emphasised.

2.1 Dynamic spatio-temporal models

This section gives a brief overview of the most popular dynamic spatio-temporal models appearing in the literature, namely, the STARMA, the CML, the IDE, the PDE and the PDAE models.

2.1.1 Space-time auto-regressive moving-average models

After the successful development of the auto-regressive moving-average (ARMA) class of models for stochastic temporal processes [86], the ARMA models were extended throughout the 1970s to include spatial dynamics in the time series evolution, resulting in the introduction of space-time ARMA (STARMA) models [87, 88].

STARMA models are essentially linear relationships that are lagged in space and time. STARMA formulations are obtained by having observations $y_{k,i}$ of the random variable $Y_{k,i}$ which need to reside at each of the N fixed locations (or sites) situated in the spatial field, over K time steps of the discrete time k. The spatio-temporal auto-regressive format expresses $y_{k,i}$ in terms of a linear combination of previous measurements at site i [88] and neighbouring sites. If an identical relationship is true for every site, spatial stationarity or homogeneity is said to exist.

In the classical STARMA formulation by [88], a spatial lag operator $L^{(l)}$ of order l is initially defined as follows:

$$L^{(0)} y_{k,i} = y_{k,i}, \tag{2.1}$$

$$L^{(l)} y_{k,i} = \sum_{j=1}^{N} w_{ij}^{(l)} y_{k,j}, \tag{2.2}$$

where $w_{ij}^{(l)}$ denotes a set of weights such that

$$\sum_{j=1}^{N} w_{ij}^{(l)} = 1 \tag{2.3}$$

for all i and $w_{ij}^{(l)} \neq 0$ if sites i and j are l^{th} order neighbours. STARMA models are

typically given in vector form, where the measurements are represented by the vector $\mathbf{y}_k = [y_{k,1}\ y_{k,2}\ \cdots\ y_{k,N}]^\top$. The weights $w_{ij}^{(l)}$ may be represented in matrix form as $\mathbf{W}^{(l)} \in \mathbb{R}^{N\times N}$. The spatial lag operator for stacked measurements is then given by

$$L^{(0)}\mathbf{y}_k = \mathbf{W}^{(0)}\mathbf{y}_k = \mathbf{I}_N\mathbf{y}_k, \tag{2.4}$$

$$L^{(l)}\mathbf{y}_k = \mathbf{W}^{(l)}\mathbf{y}_k, \text{ for } l > 0, \tag{2.5}$$

where $\mathbf{I}_N$ denotes the $N \times N$ identity matrix. The STARMA model may therefore be written in vector form as

$$\mathbf{y}_k = \sum_{\tau=1}^{p}\sum_{l=0}^{\lambda_\tau}\phi_{\tau l}\mathbf{W}^{(l)}\mathbf{y}_{k-\tau} - \sum_{\tau=1}^{q}\sum_{l=0}^{m_\tau}\phi_{\tau l}\mathbf{W}^{(l)}\epsilon_{k-\tau} + \epsilon_k, \tag{2.6}$$

where ϵ_k denotes a random normal error vector. This form is known as a STARMA (p,q) model.

In [89], popular neighbourhood definitions are discussed. A local interaction hypothesis that yields less parameters for estimation is typically assumed. Unrestricted models were put forward [90], however such spatio-temporal models needed a considerably bigger parameter space which requires lower spatial dimensions and less observation locations to be included [91]. Despite STARMA models being shown to outperform univariate ARMA models in forecasting applications [92], researchers started losing interest in this class of models, mainly as a consequence of their inadequate consideration of spatial dependence and heterogeneity of measurements [93, 94].

Over the last decade, however, the increased computational power revived the potential of STARMA models. The issue of instantaneous spatial correlation was addressed in [91] by commencing the first summation appearing in both terms of equation (2.6) from $\tau = 0$ so that innovations may be representative of a spatial spread over one sampling interval. Model modifications which allowed the use of larger data sets enabled heterogeneous model definitions. One example is the toroidal space definition proposed in [95]. The model dimension still depends on the number of observations and these aforementioned limitations had sparked the researchers'

interest to propose alternative modelling schemes. In the 1980s, one such alternative became known as the coupled map lattice, which was especially popular for its ability to represent spatio-temporal dynamics of very complex or poorly understood systems.

2.1.2 Coupled map lattices

Various natural phenomena exhibit chaotic and complex spatio-temporal trends [96, 97] that motivated the development of Coupled Map Lattices (CMLs) [98, 99, 100]. The extensive applicability of CMLs is confirmed by the wide spectrum of uses described in the literature and has been the model class of choice for studying physical and chemical processes, including studying Bénard convection [101, 102], modelling reaction-diffusion dynamics [103], modelling the physics of boiling [104], describing cloud dynamics [105], modelling open fluid flow [106, 107, 108] and modelling crystal growth [109, 103]. CMLs were also used for the study of complex spatio-temporal relationships observed in ecology [110, 111, 112]. Other applications include image processing [113], computer theory [114] and electroencephalography (EEG) signal processing [115, 116].

CMLs are part of a more general class of systems known as lattice dynamic systems [117] and share a number of similarities with cellular automata (CA), with the difference that the system states are not required to be discrete [118]. CMLs are defined in discrete space and discrete time. Letting a set of lattice points be $i = 1, \ldots, N$, where every element defines a discrete location in space, and denoting the field by $\psi_{k,i}$ at discrete-time instant k, the time evolution at site i is described by the nonlinear mapping $M_i : \mathbb{R}^N \to \mathbb{R}$, so that $\psi_{k+1,i} = M_i \boldsymbol{\psi}_k$, where $\boldsymbol{\psi}_k = [\psi_{k,1}\ \psi_{k,2}\ \ldots\ \psi_{k,N}]^\top$. A minority of works treat spatial heterogeneity [119], however a spatially homogeneous behaviour is typically assumed so the standard nonlinear evolution equation becomes $\psi_{k+1,i} = M\boldsymbol{\psi}_k$, with the mapping dependence on i being omitted.

The mapping M clearly governs the behaviour of the CML. A popular mapping is the nearest neighbour coupling map [96, 120, 100, 121] that consists of a spatial

coupling function f_c and a local interaction term f_l. This may be written as

$$\begin{aligned} \psi_{k+1,i} &= f_c(\psi_{k,i-1}, \psi_{k,i+1}) + f_l(\psi_{k,i}) \\ &= \frac{\epsilon}{2}\left(f(\psi_{k,i-1}) + f(\psi_{k,i+1})\right) + (1-\epsilon) f(\psi_{k,i}), \end{aligned} \tag{2.7}$$

where $\epsilon \in [0, 1]$ and $f(\cdot)$ represents a pre-defined nonlinear function, for instance the logistic map $f : \psi_{k,i} \to 1 - a\psi_{k,i}^2$. This logistic map [122] is the most commonly used local map [100], however chaotic behaviour may be represented by numerous other mappings [123]. Alternative mappings which consider larger neighbourhoods produce significantly different output patterns and result in 'global coupling' [123].

A CML is typically derived using the natural laws obeyed by the system under study. However, the mapping M may not always be determined or derived, in which case model structure detection and parameter estimation may need to be performed [124, 118, 125, 126, 127]. Most CMLs described in the literature are deterministic, but stochastic CMLs, such as that reported to use randomly perturbed lattice points [126], also exist.

CMLs are therefore dynamic, can represent systems characterised by large uncertainties and carry a meaningful representation of the system's underlying processes. Such properties make them very useful for several modelling scenarios, but since they are built bottom-up on a discrete grid, observations must be taken on a regular lattice. This may be impossible in certain situations, such as control schemes involving mobile agents. Although heterogeneous CMLs may provide a spatially varying mapping, parametrising the heterogeneity and choosing the appropriate inference mechanism to cater for the heterogeneity in parameter estimation is unclear.

2.1.3 Integro-difference equation models

The main weakness of the CML is the discrete spatial lattice construction and the integro-difference equation (IDE) [128, 129] remedies the situation by employing a continuous-space representation. The deterministic IDE was first proposed in [129, 130] to model the spread of invading organisms. The IDE was developed by modelling

a population in two separate stages. The first stage is known as the sedentary stage and is represented by a nonlinear map $f(\cdot)$ that determines local growth. The second stage is referred to as the dispersion stage and is described by an integral operator which represents physical diffusion or migration dynamics in a population. On the basis of such applications, the IDE was shown to model these systems better than the reaction-diffusion equation proposed in [131]. Since then, IDEs have been employed to model several phenomena such as cloud dynamics [132, 128] and precipitation nowcasting [133].

The IDE is continuous in space and discrete in time. It describes the evolution of the spatio-temporal field ψ given by

$$\psi_k(\mathbf{s}) = \int_{\mathcal{S}} \kappa_k(\mathbf{s}, \mathbf{r}) f(\psi_{k-1}(\mathbf{r})) d\mathbf{r}, \tag{2.8}$$

where $\mathbf{s}, \mathbf{r} \in \mathcal{S} \subset \mathbb{R}^d$ represent the spatial locations in a d-dimensional space and $\kappa_k(\mathbf{s}, \mathbf{r}) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is a time-varying, heterogeneous spatial convolution kernel that controls the spatio-temporal interactions of the system.

Equation (2.8) represents a heterogeneous IDE model with nonlinear growth. In environmental literature, however, this is usually simplified and linear growth and homogeneity is assumed, giving rise to the following model:

$$\psi_k(\mathbf{s}) = f'(0) \int_{\mathcal{S}} \kappa_k(\mathbf{s} - \mathbf{r}) \psi_{k-1}(\mathbf{r}) d\mathbf{r}, \tag{2.9}$$

where f' represents the first derivative of f. This simplification catalysed the development of spatio-temporal methods such as spatio-temporal Kalman filtering [134, 135] and new classes of non-separable covariance functions for geostatistical models [136, 137, 53]. In time, IDEs became a popular choice in representing complex spatio-temporal behaviour spanning fields as diverse as ecology [129], signal processing [138, 139] and environmental applications [134, 133, 128, 135, 140, 132].

Most of the research, particularly in ecological literature, has concentrated on analysing the effect of the shape and growth term of the convolution kernel on the spatio-temporal process stability [130] and the shape and speed of the invading waves

generated [129, 141, 142, 143, 144, 145]. More recent works in modelling population dynamics using IDE models have been addressing improvements in the basic representations given by equations (2.8) and (2.9), namely by employing the Allee effect [129, 144, 145] and by analysing the effect of environmental variables and population structure on propagation [146, 143]. Other efforts report the estimation of the travelling wave shape [147], the numerical estimation of the invading wave speed [145] and the prediction of the future invasion speed [143].

The IDE was put into a stochastic framework in [128], where additive spatial noise is included using a spatial Gaussian processes (GP) [148]. For every time instant of this stochastic IDE, the propagated field is superimposed by draws from a zero-mean spatial GP, $\epsilon_k(\mathbf{s}) \sim \mathcal{GP}(0, \Sigma(\mathbf{s}, \mathbf{r}))$. This stochasticity can model uncertainties and represent any random forcing functions or model mismatch. The set generated, $\epsilon_k(\mathbf{s})$, is usually taken to be independent and identically distributed (i.i.d.) over time, so the behaviour of the model is principally dictated by the mixing kernel and the form of $f(\cdot)$. For example, in EEG studies, $f(\cdot)$ is set to be a sigmoid function [149]. On the other hand, the Ricker growth models or the standard logistic models are often used in ecology [130]. The function $f(\cdot)$, however, may take different forms, such as Gompertz or Malthusian formats [150].

A novel basis function decomposition for the IDE was derived in [138], where a state-space representation that decouples the number of states from the number of observation locations or parameters is presented. By using a state-space representation for the IDE, the work in [139] employs ideas from multidimensional sampling theory to propose an approach that provides the minimum model and parameter vector dimensions required for an adequate system representation. The method is based on the spatial bandwidth of the system and the frequency support of the redistribution kernel of the IDE.

When modelling systems using the IDE, the kernel provides a meaningful insight into the system dynamics. Various works have estimated basis functions which shape $\kappa_k(\mathbf{s}, \mathbf{r})$ [151, 149, 152, 138]. However, as argued in [153], a key limitation of the IDE is its lack of description of the evolution process at a physical level. The IDE

may obscure the underlying physical dynamics and may be consequently challenging to represent heterogeneity. A more mechanistic approach to modelling is therefore necessary for a principled way of representing spatially varying systems. This is the key strength of another class of models, the partial differential equations.

2.1.4 Partial differential equation models

PDE models, which are continuous in both space and time, enjoy a widespread interest due to their impressive array of natural phenomena that they can describe. These include elasticity, quantum physics, thermodynamics, fluid dynamics, mechanics and electromagnetic theory [154, 155, 156, 157]. PDEs have been central to several applications in oceanography [158], ecology [159], flexible structures [160] and wildfire control [161].

A PDE is defined as any equation that describes a function of two or more independent variables and at least one of their partial derivatives [162, Chapter 1]. For the case of spatio-temporal systems, the independent variables are restricted to be space and time. Let time $t \in \mathcal{T} \subset \mathbb{R}^+$ and consider a spatio-temporal field $\psi(\mathbf{s}, t) : \mathcal{S} \times \mathcal{T} \to \mathbb{R}$. The general form of the PDE may be written as

$$F\left(\mathbf{s}, t, \psi, \frac{\partial \psi}{\partial \mathbf{s}}, \frac{\partial \psi}{\partial t}, \frac{\partial^2 \psi}{\partial \mathbf{s}^2}, \frac{\partial^2 \psi}{\partial t^2}, \frac{\partial^2 \psi}{\partial \mathbf{s} \partial t}, \ldots\right) = 0. \tag{2.10}$$

Whenever the function $F(\cdot)$ is linear, the PDE is referred to as linear, otherwise the PDE is said to be nonlinear or quasilinear. The system is known as space and time invariant if $F(\cdot)$ is independent of both space and time.

The study of linear PDEs has been substantial given the broad applicability to so many areas of mathematical physics including vibrations, heat flow and so on [163, 155, 157]. However, several other phenomena are modelled using nonlinear PDEs, including fluid pressure effects solved using Navier-Stokes equations, superconductivity based on the Ginzburg-Landau equation and general relativity described by Einstein's field equations and the Dym equation [164, 165].

Although PDEs represent spatio-temporal dynamics of physical phenomena, for

which practical experiments prove the existence of a stable unique solution, their mathematical description might not determine a solution. While for the ODE case the general solution to an nth-order equation is given by a family of functions with n independent arbitrary constants, PDE models are different. In fact, even the solution space for simple linear homogeneous PDEs is infinite dimensional. In the control literature, systems yielding an infinite dimensional solution space are termed distributed parameter systems [166].

PDEs are typically defined on some bounded domain and the PDE formulation must have some conditions for ψ that should be obeyed on the domain boundary $\partial\mathcal{S}$. The conditions are generally first-type (Dirichlet), where ψ takes some fixed value on $\partial\mathcal{S}$, or second-type (Neumann), where ψ must have fixed derivatives on $\partial\mathcal{S}$. If both boundary conditions and initial conditions are specified, the problem of solving for the field ψ is referred to as the initial/boundary-value problem.

Although various methods exist for finding an analytical solution to PDEs [156, 154], most practical physical systems cannot be solved analytically and therefore a numerical method is the viable alternative. Two main techniques are available in the literature, namely the finite element [167] and the finite difference methods [168].

The solution of PDEs becomes more complex when the model parameters, such as the thermal conductivity of a material in a heat flow equation, are unknown. The system identification community has dedicated significant research efforts towards obtaining models of spatio-temporal systems using measured data, often assuming little or no knowledge of the underlying physical dynamics [169, 170, 171, 172]. This problem was first treated in [173] by proposing tests for the identifiability of PDE model parameters. An estimation method that assumes identifiability and is based on alternating conditional algorithms was later developed in [174] and was shown to successfully estimate the Swift-Hohenberg equation.

More recently, works progressively allowed more assumptions to be relaxed. The assumption of knowing the structural form of the PDE taken in [171] is relaxed in [169, 170], where PDE estimation is carried out using the orthogonal least squares algorithm and the Adams integration.

Stochastic PDEs (SPDEs) become the necessary form of representation whenever the initial or boundary conditions are stochastic [175, Section 1.1], where a random forcing term exists [176] or when the physical system is not fully known. This intricate model can describe all kinds of processes characterised by a stochastic influence in nature or man-made complex systems [177]. This is clearly evidenced by the use of SPDEs for modelling systems in hydrology [56], neurophysiology [178], geophysics [179] and signal denoising [180]. Despite their wide applicability, choosing SPDEs for analysis is particularly challenging in the context of parameter estimation. Here, most of the literature considers deterministic PDEs observed in noise [181, 169, 160], whilst for the stochastic case, fewer works have been published [182]. New estimation and identification tools for SPDEs have been recently explored in [183], where the variational approximation and the consideration of both continuous and point-process observations were investigated and implemented for SPDE models.

2.1.5 Partial differential-algebraic equation models

A more general class of models to the PDEs are partial differential-algebraic equation (PDAE) models, which have a descriptor formulation. Several natural processes, such as fluid flow [14], electrochemical reactions in a molten carbonate fuel cell [184], packed-bed chromatographic adsorption [185], electronic integrated circuit processes [186] and slender inextensible elastica dynamics [187] are modelled by a nonlinear PDAE model of the form

$$\frac{\partial \mathbf{b}(\mathbf{s},t)}{\partial t} = \mathbf{F}\left(D^{n_o}\boldsymbol{\psi}(\mathbf{s},t), D^{n_o-1}\boldsymbol{\psi}(\mathbf{s},t), \ldots, D\boldsymbol{\psi}(\mathbf{s},t), \boldsymbol{\psi}(\mathbf{s},t)\right), \tag{2.11a}$$

$$\mathbf{0} = \mathbf{G}\left(D^{n_o}\boldsymbol{\psi}(\mathbf{s},t), D^{n_o-1}\boldsymbol{\psi}(\mathbf{s},t), \ldots, D\boldsymbol{\psi}(\mathbf{s},t), \boldsymbol{\psi}(\mathbf{s},t)\right), \tag{2.11b}$$

where $\boldsymbol{\psi}(\mathbf{s},t) : \mathbb{R}^{d+1} \to \mathbb{R}^{n_\psi}$ represents n_ψ spatio-temporal fields and $\boldsymbol{\psi}(\mathbf{s},t) = [\mathbf{b}^\top(\mathbf{s},t), \mathbf{c}^\top(\mathbf{s},t)]^\top$, where $\mathbf{b} : \mathbb{R}^{d+1} \to \mathbb{R}^{n_b}$ and $\mathbf{c} : \mathbb{R}^{d+1} \to \mathbb{R}^{n_c}$ denote the differential and algebraic state variables, respectively. For a non-negative integer n_o, $D^{n_o}\boldsymbol{\psi}(\mathbf{s},t)$ is the set of all partial spatial derivatives of order n_o.

It is noteworthy to point out a key difference between a PDAE and a constrained

PDE system. For constrained PDE systems, the evolution of all process states $\boldsymbol{\psi}(\mathbf{s},t) = \mathbf{b}(\mathbf{s},t)$ is described by PDEs, subject to algebraic constraints that restrict their evolution. For a PDAE system, there exist some states $\mathbf{c}(\mathbf{s},t)$, termed algebraic, whose evolution is not governed by PDEs, but is completely controlled by the evolution of differential states $\mathbf{b}(\mathbf{s},t)$, such that all algebraic constraints are satisfied [30]. This key difference means that the spatio-temporal descriptor system cannot be estimated as a constrained PDE system, requiring an alternative estimation scheme [30].

2.2 Model reduction, state-space models and descriptor systems

Most standard signal processing techniques are usually tailored for finite-dimensional systems, so model reduction methods were proposed to reduce infinite-dimensional spatio-temporal models to a finite-dimensional form. An important spatial and temporal discretisation scheme is the method of finite differences, usually employed for PDE-based models [188]. This method uses difference quotients to approximate spatial and temporal derivatives of the PDE.

Another model reduction technique is the method of moments which is typically used for spatial dimensionality reduction [189], where a finite set of linearly independent basis functions $\{\phi_i(\mathbf{s})\}_{i=1}^{n_\phi}$ decomposes a spatio-temporal field $\psi(s,t)$ such that

$$\psi(\mathbf{s},t) \approx \sum_{i=1}^{n_\phi} \phi_i(\mathbf{s}) x_i(t) = \boldsymbol{\phi}^\top(\mathbf{s})\mathbf{x}(t), \tag{2.12}$$

where $\mathbf{x}(t) \in \mathbb{R}^{n_\phi}$ is a vector of state variables that weights the field basis functions $\phi_i(\mathbf{s})$. The spatio-temporal field is then projected under an inner-product transformation with respect to a set of test functions $\{\chi_i(\mathbf{s})\}_{i=1}^{n_\phi}$ [189]. A popular choice for the test functions is to set $\{\phi_i(\mathbf{s})\}_{i=1}^{n_\phi} = \{\chi_i(\mathbf{s})\}_{i=1}^{n_\phi}$, which is a special case of the method of moments known as the Galerkin method. The method of moments enjoys a num-

ber of advantages over standard finite-difference schemes, largely due to their more convenient use in complex geometry spaces and their ability to systematically handle Dirichlet boundary conditions by an appropriate choice of basis functions [183]. Since the observation process is usually temporally discrete, an Euler step provides a discrete-time representation for the finite-dimensional system [149].

By following finite-dimensional reduction, the popular stochastic state-space model framework is obtained for which several signal processing techniques are readily available and algorithm development is greatly facilitated:

$$\mathbf{x}_{k+1} = \mathbf{f}_k(\mathbf{x}_k) + \mathbf{q}_k, \tag{2.13a}$$

$$\mathbf{y}_k = \mathbf{h}_k(\mathbf{x}_k) + \mathbf{r}_k, \tag{2.13b}$$

where $\mathbf{x}_k := \mathbf{x}(k\Delta_t)$ and $\mathbf{y}_k := \mathbf{y}(k\Delta_t) \in \mathbb{R}^{n_y}$ are vectors of the system state variables and observations, respectively, $\mathbf{q}_k$ and $\mathbf{r}_k$ are noise sequences, $\mathbf{f}_k(\cdot)$ is a dynamic model function, $\mathbf{h}_k(\cdot)$ is a measurement model function and Δ_t is the time step.

In Chapter 1, it was seen how descriptor models are characterised by a collection of variables which is adequate to describe the system. Such variables, conveniently described as descriptor variables, generally have an inherent meaning, or natural interpretation, within the context of the particular system. These might represent, for instance, position, velocity or acceleration in the case of Newtonian systems, price or quantity in economic systems and so on. Once variables are defined, system laws dictate the relationship between descriptor variables, some of which will be dynamic, in that they involve variables at different time instants [18]. Some of these relationships will be purely static, which represent identity relations that hold between variables. This results in a set of equations expressed in terms of variables, aptly called descriptor variables for their role of being natural descriptors of the system, rendering descriptor models widely applicable to several physical systems. The equations (2.13)

may be written as a more general descriptor formulation given by

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{z}_k) + \mathbf{q}_k, \tag{2.14a}$$

$$\mathbf{0} = \mathbf{g}(\mathbf{x}_k, \mathbf{z}_k) + \mathbf{e}_k, \tag{2.14b}$$

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k, \mathbf{z}_k) + \mathbf{r}_k, \tag{2.14c}$$

where $\mathbf{z}_k \in \mathbb{R}^{n_z}$ is a vector of algebraic state variables, $\mathbf{e}_k$ is the algebraic state process noise and $\mathbf{g}(\cdot)$ is a mapping of algebraic equations. The functions $\mathbf{f}$, $\mathbf{g}$ and $\mathbf{h}$ may depend on the time instant k, however this dependence will no longer be explicitly noted throughout this thesis for notational convenience.

As will be discussed in the next section, the purpose of a filter is then to approximate the filtering distributions $p(\mathbf{X}_k|\mathbf{y}_{1:k})$ for time steps k running from 0 to K. A smoothing algorithm approximates the smoothing distributions $p(\mathbf{X}_k|\mathbf{y}_{1:K})$ for time steps k running from 0 to K. In this thesis, the approximations and noise processes are chosen to be Gaussian where $\mathbf{q}_k \sim \mathcal{N}_q(0, \mathbf{Q}_k)$, $\mathbf{e}_k \sim \mathcal{N}_e(0, \mathbf{E}_k)$ and $\mathbf{r}_k \sim \mathcal{N}_r(0, \mathbf{R}_k)$. The notation $\mathcal{N}_q(0, \mathbf{Q}_k)$ denotes the normal distribution of $\mathbf{x}_k$ with zero mean and covariance matrix $\mathbf{Q}_k = \sigma_q^2\mathbf{I}$, where $\mathbf{I}$ denotes the identity matrix of appropriate dimensions. A similar notation holds for $\mathcal{N}_e(0, \mathbf{E}_k)$ and $\mathcal{N}_r(0, \mathbf{R}_k)$, where $\mathbf{E}_k = \sigma_e^2\mathbf{I}$ and $\mathbf{R}_k = \sigma_r^2\mathbf{I}$. The filtering and smoothing distributions may be given respectively as

$$p(\mathbf{X}_k|\mathbf{y}_{1:k}) \approx \mathcal{N}(\mathbf{X}_k|\boldsymbol{\mu}_k, \mathbf{P}_k), \tag{2.15}$$

$$p(\mathbf{X}_k|\mathbf{y}_{1:K}) \approx \mathcal{N}(\mathbf{X}_k|\boldsymbol{\mu}_k^{(s)}, \mathbf{P}_k^{(s)}), \tag{2.16}$$

where $\boldsymbol{\mu}_k$ and $\boldsymbol{\mu}_k^{(s)}$ denote the associated mean and $\mathbf{P}_k$ and $\mathbf{P}_k^{(s)}$ represent the associated covariance matrices.

2.3 State estimation for nonlinear state-space models

An important problem in dynamic spatio-temporal systems is field reconstruction in some spatial domain at a given time instant from some observation process. If

the data is sufficiently informative, such as data obtained from an infrared camera [190], the spatio-temporal field is practically known and no further signal processing is required. If, however, the field is only observed at a few, isolated points, such as in neural field or ocean sampling [149, 191], state estimation for $\mathbf{x}_{1:K} = \{\mathbf{x}_1, \ldots, \mathbf{x}_K\}$ (for K regularly spaced time steps) is carried out using observed data $\mathbf{y}_{1:K} = \{\mathbf{y}_1, \ldots, \mathbf{y}_K\}$ needed for field reconstruction. The optimal estimation of the states from some data set is referred to as the smoothing problem. This problem is typically solved using the forward-backward algorithm, where the forward pass represents filtering and the backward pass represents smoothing. Another approach is the two-filter smoother that uses forward messages (equivalent to the filtering result) with backward messages calculated in reverse time so that smoothed estimates are obtained.

With much improved computational power being widely available, state estimation is now being increasingly performed for on-line monitoring and control in various application domains such as robotics, digital communications, computer vision and process control [67, 71]. Historical developments in state estimation are excellently introduced in [192] and the important publications [77] and [193] spurred great research efforts in the area of dynamic model online state estimation. While such initial developments used only linear dynamic models, their nonlinear counterpart was the main topic of research in later years. It is noteworthy that despite significant advances employing dynamic models considering continuous observations [194, 138, 149], very few efforts have considered the problem of having observations available as isolated events, i.e. point-process observations [45].

A general state-space model has the form

$$\mathbf{x}_{k+1} \sim p(\mathbf{x}_{k+1}|\mathbf{x}_k), \tag{2.17a}$$

$$\mathbf{y}_k \sim p(\mathbf{y}_k|\mathbf{x}_k), \tag{2.17b}$$

of which the model (2.13) is a special case. The filtering distributions are formally

described by the optimal filtering equations [195, 196] given by

$$p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k}) = \int p(\mathbf{x}_{k+1}|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}_{1:k})\ \mathrm{d}\mathbf{x}_k, \tag{2.18a}$$

$$p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1}) = \frac{p(\mathbf{y}_{k+1}|\mathbf{x}_{k+1})p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})}{\int p(\mathbf{y}_{k+1}|\mathbf{x}_{1:k+1})p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})\mathrm{d}\mathbf{x}_{k+1}}, \tag{2.18b}$$

where equations (2.18a) and (2.18b) are the prediction and update steps, respectively.

The optimal smoothing equations [196] may be expressed in the two-filter smoother form, as follows:

$$p(\mathbf{x}_k|\mathbf{y}_{1:K}) \propto p(\mathbf{x}_k|\mathbf{y}_{1:k})p(\mathbf{y}_{k+1:K}|\mathbf{x}_k), \tag{2.19}$$

where $p(\mathbf{x}_k|\mathbf{y}_{1:k})$ is obtained using the optimal filter and $p(\mathbf{y}_{k+1:K}|\mathbf{x}_k)$ is calculated by running a filter that goes backwards in time. The unscented Kalman smoother (UKS) described in [197] may be considered to approximate this form of smoother. An alternative form is the forward-backward smoother which is described by the equation

$$p(\mathbf{x}_k|\mathbf{y}_{1:K}) = p(\mathbf{x}_k|\mathbf{y}_{1:k}) \int \frac{p(\mathbf{x}_{k+1}|\mathbf{x}_k)p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})}{p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})}\mathrm{d}\mathbf{x}_{k+1}, \tag{2.20}$$

where $p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})$ represents the predicted distribution at time instant $k+1$, that may be obtained from equation (2.18a). The filtering distribution at time k is $p(\mathbf{x}_k|\mathbf{y}_{1:k})$. The recursion that performs the smoothing action is started from the last time step $k = K$ and proceeds back in time.

The filtering and smoothing equations just described are only formal in that they can rarely be implemented directly for practical computation purposes because of the computational intractability. This explains the significant research efforts towards numerical approximation methods.

Consider discrete-time state-space models of the form given by equations (2.13). In the context of conditional-density-approximation-based state estimators for linear models, the analytical solution is the Kalman filter that describes the optimal re-

cursive solution to the problem of sequential state estimation [77]. Combining the predicted state estimate $\hat{\mathbf{x}}_{k+1|k}$ with the measurement $\mathbf{y}_{k+1}$, the optimal state estimate is constructed recursively as follows:

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{L}_{k+1}\mathbf{e}_{k+1}, \tag{2.21}$$

where $\mathbf{e}_{k+1} = \mathbf{y}_{k+1} - \mathbf{C}_{k+1}\hat{\mathbf{x}}_{k+1|k}$ is the innovation and $\mathbf{L}_{k+1}$ is the Kalman gain matrix given by

$$\mathbf{L}_{k+1} = \mathbf{P}_{k+1}^{(\epsilon,e)}[\mathbf{P}_{k+1}^{(e,e)}]^{-1}, \tag{2.22}$$

where

$$\mathbf{P}_{k+1}^{(\epsilon,e)} = \mathrm{E}[(\boldsymbol{\epsilon}_{k+1|k})(\mathbf{e}_{k+1})^T] \tag{2.23}$$

and

$$\mathbf{P}_{k+1}^{(e,e)} = \mathrm{E}[(\mathbf{e}_{k+1})(\mathbf{e}_{k+1})^T]. \tag{2.24}$$

The a priori estimation error is denoted by $\boldsymbol{\epsilon}_{k+1|k} = \mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k}$ and $\mathrm{E}(\cdot)$ is the expectation operator. A key feature of the Kalman filter is that whenever $\mathbf{q}_k$ and $\mathbf{r}_k$ are additive Gaussian noise processes and $\mathbf{x}_0$ is Gaussian distributed, the conditional densities $p(\mathbf{x}_{k+1}|\mathbf{y}_k)$ and $p(\mathbf{x}_{k+1}|\mathbf{y}_{k+1})$, and the innovation sequence $\mathbf{e}_{k+1}$, are also Gaussian. This is a consequence of the preservation of the Gaussian distributions under linear transformations.

In practical applications, the dynamic and measurement models are often nonlinear and the Kalman filter becomes inappropriate. When dealing with nonlinear systems, the sequential Bayesian estimation problem needs approximate and computationally tractable sub-optimal solutions. For such cases, even when $\mathbf{q}_k$, $\mathbf{r}_k$ and $\mathbf{x}_0$ are Gaussian, the conditional densities $p(\mathbf{x}_{k+1}|\mathbf{y}_k)$ and $p(\mathbf{x}_{k+1}|\mathbf{y}_{k+1})$ are non-Gaussian. The key to obtaining approximate solutions is making appropriate simplifying assumptions to approximate the non-Gaussian conditional densities and estimating their first and second moments. In his seminal publication, Kalman [77] noted how when the Kalman filter is used, all statistical computations are based on the first and second order averages and no other statistical data is needed. For Gaussian distributions, this

is indeed the case and the first and second moments are sufficient to characterize the whole distribution. However, this is not the case for non-Gaussian distributions and most of the approximate methods that extend the Kalman filter to handle nonlinear problems focus on estimating the first and second moments of the conditional densities. These techniques continue the use of the Kalman filter update step of equation (2.21) and the observer gain computation of equation (2.22). As will be discussed, the main differences lie in the methods used to estimate the moments.

A popular choice for the simplification of the Bayesian estimation problem is approximating the conditional densities by a Gaussian distribution. This approximation, owing to its analytical tractability, is a simple approach to approximate numerical integration computations in the prediction and update steps. The simplifying assumption also makes it sufficient to update the mean and covariances of the conditional distributions while performing sequential estimation. In this section, particular emphasis is given to two types of methods for forming the Gaussian approximations, both of which will be implemented in this work: the extended Kalman filter (EKF), based on the Taylor series approximation and the unscented Kalman filter (UKF), based on the unscented transform approximation. The statistically linearised filter (SLF), based on statistical linearisation, is briefly described and compared.

The UKF differs from the other filters since it is not a series expansion-based technique, despite being initially justified by considering a nonlinear function given as a series expansion. The UKF is, however, considered to approximate the SLF and to converge to the EKF in an appropriate parameter limit.

2.3.1 Extended Kalman filter

The EKF [58, 198] is an extension of the Kalman filter to nonlinear filtering problems and approximates the filtering densities using the Gaussian approximation

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \mathcal{N}(\mathbf{x}_k|\boldsymbol{\mu}_k, \mathbf{P}_k). \tag{2.25}$$

Such approximation is computed using Taylor series approximations to the nonlinearities. Assuming additive process and measurement noise as in equations (2.13), the joint distribution of $\mathbf{x}_{k+1}$ and $\mathbf{x}_k$ is non-Gaussian, however, a Gaussian approximation may be formed as follows:

$$p(\mathbf{x}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k}) \approx \mathcal{N}\left(\begin{pmatrix}\mathbf{x}_k \\ \mathbf{x}_{k+1}\end{pmatrix} |\boldsymbol{\mu}'_k, \mathbf{P}'_k\right), \tag{2.26}$$

where

$$\boldsymbol{\mu}'_k = \begin{pmatrix}\boldsymbol{\mu}_k \\ \mathbf{f}(\boldsymbol{\mu}_k)\end{pmatrix}, \tag{2.27}$$

$$\mathbf{P}'_k = \begin{pmatrix}\mathbf{P}_k & \mathbf{P}_k\mathbf{F}^{(x)\ \top} \\ \mathbf{F}^{(x)}\mathbf{P}_k & \mathbf{F}^{(x)}\mathbf{P}_k\mathbf{F}^{(x)\top} + \mathbf{Q}_k\end{pmatrix}, \tag{2.28}$$

and the Jacobian matrix $\mathbf{F}^{(x)}$ of $\mathbf{f}(\mathbf{x})$ is computed at $\mathbf{x} = \boldsymbol{\mu}_k$. The marginal mean and covariance of $\mathbf{x}_k$ are approximated by

$$\boldsymbol{\mu}^-_{k+1} = \mathbf{f}(\boldsymbol{\mu}_k), \tag{2.29}$$

$$\mathbf{P}^-_{k+1} = \mathbf{F}^{(x)}\mathbf{P}_k\mathbf{F}^{(x)\top} + \mathbf{Q}_k. \tag{2.30}$$

Considering now the non-Gaussian joint distribution of $\mathbf{y}_{k+1}$ and $\mathbf{x}_{k+1}$, this may be approximated by the following distribution:

$$p(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}|\mathbf{y}_{1:k}) \approx \mathcal{N}\left(\begin{pmatrix}\mathbf{x}_{k+1} \\ \mathbf{y}_{k+1}\end{pmatrix} |\boldsymbol{\mu}''_{k+1}, \mathbf{P}''_{k+1}\right), \tag{2.31}$$

where

$$\boldsymbol{\mu}''_{k+1} = \begin{pmatrix} \boldsymbol{\mu}^-_{k+1} \\ \mathbf{h}(\boldsymbol{\mu}^-_{k+1}) \end{pmatrix}, \tag{2.32}$$

$$\mathbf{P}''_{k+1} = \begin{pmatrix} \mathbf{P}^-_{k+1} & \mathbf{P}^-_{k+1}\mathbf{H}^{(x)\ \top} \\ \mathbf{H}^{(x)}\mathbf{P}^-_{k+1} & \mathbf{H}^{(x)}\mathbf{P}_k\mathbf{H}^{(x)\top} + \mathbf{R}_{k+1} \end{pmatrix}, \tag{2.33}$$

and the Jacobian matrix $\mathbf{H}^{(x)}$ of $\mathbf{h}(\mathbf{x})$ is calculated at $\mathbf{x} = \boldsymbol{\mu}^-_{k+1}$. Finally, the conditional distribution of $\mathbf{x}_{k+1}$ may be obtained as

$$p(\mathbf{x}_{k+1}|\mathbf{y}_{k+1}, \mathbf{y}_{1:k}) \approx \mathcal{N}(\mathbf{x}_{k+1}|\boldsymbol{\mu}_{k+1}, \mathbf{P}_{k+1}), \tag{2.34}$$

where

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}^-_{k+1} + \mathbf{P}^-_{k+1}\mathbf{H}^{(x)\top}(\mathbf{H}^{(x)}\mathbf{P}^-_{k+1}\mathbf{H}^{(x)\top} + \mathbf{R}_{k+1})^{-1}[\mathbf{y}_{k+1} - \mathbf{h}(\boldsymbol{\mu}^-_{k+1})], \tag{2.35}$$

$$\mathbf{P}_{k+1} = \mathbf{P}^-_{k+1} - \mathbf{P}^-_{k+1}\mathbf{H}^{(x)\top}(\mathbf{H}^{(x)}\mathbf{P}^-_{k+1}\mathbf{H}^{(x)\top} + \mathbf{R}_{k+1})^{-1}\mathbf{H}^{(x)}\mathbf{P}^-_{k+1}. \tag{2.36}$$

Algorithm 2.1 summarises the EKF algorithm.

Algorithm 2.1 (*The EKF algorithm*) The prediction-correction technique of the first order EKF algorithm for additive noise is as follows:

- Initialisation: Recursions start from the prior mean $\boldsymbol{\mu}_0$ and covariance $\mathbf{P}_0$.
- Prediction:

$$\boldsymbol{\mu}^-_{k+1} = \mathbf{f}(\boldsymbol{\mu}_k), \tag{2.37}$$

$$\mathbf{P}^-_{k+1} = \mathbf{F}^{(x)}(\boldsymbol{\mu}_k)\mathbf{P}_k\mathbf{F}^{(x)\top}(\boldsymbol{\mu}_k) + \mathbf{Q}_k. \tag{2.38}$$

- Correction:

$$\mathbf{e}_{e,k+1} = \mathbf{y}_{k+1} - \mathbf{h}(\boldsymbol{\mu}^{-}_{k+1}), \tag{2.39}$$

$$\mathbf{S}_{e,k+1} = \mathbf{H}^{(x)}(\boldsymbol{\mu}^{-}_{k+1})\mathbf{P}^{-}_{k+1}\mathbf{H}^{(x)\top}(\boldsymbol{\mu}^{-}_{k+1}) + \mathbf{R}_{k+1}, \tag{2.40}$$

$$\mathbf{K}_{k+1} = \mathbf{P}^{-}_{k+1}\mathbf{H}^{(x)\top}(\boldsymbol{\mu}^{-}_{k+1})\mathbf{S}^{-1}_{e,k+1}, \tag{2.41}$$

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}^{-}_{k+1} + \mathbf{K}_{k+1}\mathbf{e}_{e,k+1}, \tag{2.42}$$

$$\mathbf{P}_{k+1} = \mathbf{P}^{-}_{k+1} - \mathbf{K}_{k+1}\mathbf{S}_{e,k+1}\mathbf{K}^{\top}_{k+1}. \tag{2.43}$$

The EKF has been particularly successful in solving several industrial problems [199]. The EKF algorithm has the edge over its nonlinear filtering counterparts when noting its relative simplicity in comparison to the filtering performance. Linearisation is a popular strategy for approximating nonlinear systems and is therefore easy to understand and implement.

Many nonlinear systems remain challenging, however, and the price to pay for the simplicity of the EKF is that owing to its local linear approximation method, it will not perform well for significant nonlinearities. Also, since the noise processes can only be Gaussian, the model cannot have, for instance, random variables having discrete values. The Gaussian limitation further prevents the use of hierarchical models or models for which considerably non-Gaussian distributions are required. The Taylor series approximation needs smooth and at least twice differentiable nonlinear function vectors. In certain situations, the Jacobian matrices are difficult to compute and program, or simply not possible to obtain.

Although the second-order EKF attempts to contain the significant error involved by correcting estimates of the mean $\hat{\mathbf{x}}_{k+1|k}$ and $\hat{\mathbf{x}}_{k+1|k+1}$ using the second order terms in the Taylor series expansion of $\mathbf{f}(\mathbf{x}_k)$, severe nonlinearities remain hard to handle.

This requirement for smooth or differentiable nonlinear functions is relaxed when the first order Taylor series approximation employed in the first order EKF algorithm is replaced by approximations based on statistical linearisation [59], which represents a better alternative for approximating the nonlinear function of a random variable.

The statistically linearised filter [59] represents an improvement over EKF in the sense that the filter is a more global approximation when compared to the EKF, since the linearisation covers the whole range of function values instead of only the local region around the mean. The nonlinear functions are not required to be smooth or differentiable [59]. A notable disadvantage of the SLF in comparison to the EKF is the computation of the expected values of nonlinear functions in closed form, which is clearly impossible for some functions. It is noteworthy that when the first order truncation of the series is utilized, the SLF algorithm is a special case of the Fourier-Hermite Kalman filter (FHKF) [200].

2.3.2 Unscented Kalman filter

A common approach in the class of conditional-density-approximation-based estimators are sigma point filters. The popular unscented Kalman filter [78, 79] has been an attractive filtering strategy for several researchers and numerous application studies have appeared [201, 202, 203, 204].

The relatively recent unscented transform (UT) numerical method is another strategy for obtaining approximations to the joint distribution of random variables $\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \mathbf{P})$ (dimension n_x) and $\mathbf{y} = \mathbf{h}(\mathbf{x})$. The UT is different from linearisation and statistical linearisation in that it attempts to yield direct approximations to the mean and covariance of the target distribution rather than approximating the nonlinear function [205].

By first deterministically choosing a number of sigma points which exactly encode the mean and covariance of the original distribution of $\mathbf{x}$, the UT algorithm propagates the points through the nonlinearity. The mean and covariance of the transformed variable are then approximated using the sigma points. This might seem similar to Monte Carlo estimation, however, the fundamental difference of having sigma points being selected deterministically distinguishes the two methods [79].

To illustrate the unscented transform action, we consider the example of having the random variable $\mathbf{x}$ undergoing the nonlinear transformation $\mathbf{h}(\mathbf{x})$ to form the Gaussian approximation to $\mathbf{y} = \mathbf{h}(\mathbf{x})$. A set of $2n_x + 1$ sigma points is first formed

using the following equations:

$$\begin{aligned}
\boldsymbol{\mathcal{X}}_0 &= \tilde{\boldsymbol{\mu}}_k, \\
\boldsymbol{\mathcal{X}}_i &= \tilde{\boldsymbol{\mu}}_k + (\sqrt{(n_x+\lambda)\mathbf{P}})_i, \quad i = 1, \ldots, n_x, \\
\boldsymbol{\mathcal{X}}_i &= \tilde{\boldsymbol{\mu}}_k - (\sqrt{(n_x+\lambda)\mathbf{P}})_i, \quad i = n_x+1, \ldots, 2n_x,
\end{aligned} \tag{2.44}$$

where the vector $(\sqrt{(n_x+\lambda)\mathbf{P}})_i$ represents the ith column of the matrix square root and λ denotes a scaling parameter defined by the equation

$$\lambda = \alpha^2(n_x + \kappa) - n_x, \tag{2.45}$$

where the constants α and κ (here set to $3-n_x$) govern the sigma point spread around the mean [197]. The matrix square root represents a matrix for which $\sqrt{\mathbf{P}}\sqrt{\mathbf{P}}^\top = \mathbf{P}$. The sigma point propagation through the nonlinear function $\mathbf{h}(\cdot)$ yields

$$\boldsymbol{\mathcal{Y}}_i = \mathbf{h}(\boldsymbol{\mathcal{X}}_i), \quad i = 0, \ldots, 2n_x, \tag{2.46}$$

which gives the transformed sigma points $\boldsymbol{\mathcal{Y}}_i$. The estimates for the mean and covariance of the transformed variable may be determined using the sigma points:

$$\mathrm{E}(\mathbf{h}(\mathbf{x})) \approx \boldsymbol{\mu}_s = \sum_{i=0}^{2n_x} W_i^{(m)} \boldsymbol{\mathcal{Y}}_i, \tag{2.47}$$

$$Cov(\mathbf{h}(\mathbf{x})) \approx \mathbf{S}_s = \sum_{i=0}^{2n_x} W_i^{(c)} (\boldsymbol{\mathcal{Y}}_i - \boldsymbol{\mu}_s)(\boldsymbol{\mathcal{Y}}_i - \boldsymbol{\mu}_s)^\top, \tag{2.48}$$

where $W_i^{(m)}$ and $W_i^{(c)}$ are constant weights given by [197]

$$\begin{aligned}
W_0^{(m)} &= \frac{\lambda}{n_x + \lambda}, \\
W_0^{(c)} &= \frac{\lambda}{n_x + \lambda} + 1 - \alpha^2 + \beta, \\
W_i^{(m)} &= W_i^{(c)} = \frac{1}{2(n_x + \lambda)}, \ i = 1, \ldots, 2n_x.
\end{aligned} \tag{2.49}$$

The parameter β may be employed to incorporate prior knowledge of the non-Gaussian distribution of $\mathbf{x}$ [197].

Since the estimate for the mean of $\mathbf{h}(\cdot)$ is exact for polynomials up till the third order, the unscented transform is said to be a third order method. Therefore, if $\mathbf{h}(\cdot)$ happens to be a multi-variate third order polynomial, the mean is exact. However, the approximation for the covariance is only exact for first order polynomials since squaring a second order polynomial results in a fourth order polynomial for which the UT cannot determine exact results. In this context, the UT is known as a first order method. With an appropriate parameter selection (namely, setting $\kappa = 3 - n_x$ as in [205]), there exists a possibility to have some correct fourth order terms appearing in the covariance computation for quadratic functions. Note that the UT may also be implemented without the Gaussian assumption, but Guassianity facilitates the Bayesian interpretation of the UT.

The UT is used in the unscented Kalman filter [205, 79, 197], which is an approximate filtering algorithm that approximates the filtering distributions of models having a similar form as those used with the EKF and SLF algorithms, as in equations (2.13). A non-additive noise filtering model may also be used, but is not given here in the interest of brevity. Similar to the EKF and the SLF, the UKF approximates the filtering distribution given by equation (2.25) as a Guassian approximation. Algorithm 2.2 summarises the UKF algorithm.

Algorithm 2.2 (*The UKF algorithm*) In its additive form, the UKF algorithm may be used for additive models of the form given by equations (2.13). For this additive form, the following steps are performed for each measurement instant $k = 0, 1, 2, 3, \ldots$:

- Initialisation: Recursions start from the prior mean $\boldsymbol{\mu}_0$ and covariance $\mathbf{P}_0$.
- Prediction:

1. Sigma points are formed:

$$\begin{aligned}
\boldsymbol{\mathcal{X}}_{k,0} &= \boldsymbol{\mu}_k, \\
\boldsymbol{\mathcal{X}}_{k,i} &= \boldsymbol{\mu}_k + (\sqrt{(n_x+\lambda)\mathbf{P}_k})_i, \quad i = 1, \ldots, n_x, \\
\boldsymbol{\mathcal{X}}_{k,i} &= \boldsymbol{\mu}_k - (\sqrt{(n_x+\lambda)\mathbf{P}_k})_i, \quad i = n_x+1, \ldots, 2n_x,
\end{aligned} \tag{2.50}$$

   where λ is given by equation (2.45).

2. The sigma points are propagated through the dynamic model:

$$\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} = \mathbf{f}(\boldsymbol{\mathcal{X}}_{k,i}), \quad i = 0, \ldots, 2n_x. \tag{2.51}$$

3. The predicted mean $\boldsymbol{\mu}^-_{k+1}$ and the predicted covariance $\mathbf{P}^-_{k+1}$ are obtained:

$$\boldsymbol{\mu}^-_{k+1} = \sum_{i=0}^{2n_x} W_i^{(m)} \hat{\boldsymbol{\mathcal{X}}}_{k+1,i}, \tag{2.52}$$

$$\mathbf{P}^-_{k+1} = \sum_{i=0}^{2n_x} W_i^{(c)} (\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \boldsymbol{\mu}^-_{k+1})(\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \boldsymbol{\mu}^-_{k+1})^\top + \mathbf{Q}_k, \tag{2.53}$$

   where the weights $W_i^{(m)}$ and $W_i^{(c)}$ are defined in equation (2.49).

- Correction:

1. Sigma points are formed:

$$\begin{aligned}
\boldsymbol{\mathcal{X}}^-_{k+1,0} &= \boldsymbol{\mu}^-_{k+1}, \\
\boldsymbol{\mathcal{X}}^-_{k+1,i} &= \boldsymbol{\mu}^-_{k+1} + (\sqrt{(n_x+\lambda)\mathbf{P}^-_{k+1}})_i, \quad i = 1, \ldots, n_x, \\
\boldsymbol{\mathcal{X}}^-_{k+1,i} &= \boldsymbol{\mu}^-_{k+1} - (\sqrt{(n_x+\lambda)\mathbf{P}^-_{k+1}})_i, \quad i = n_x+1, \ldots, 2n_x.
\end{aligned} \tag{2.54}$$

2. The sigma points are propagated through the measurement model:

$$\hat{\boldsymbol{\mathcal{Y}}}_{k+1,i} = \mathbf{h}(\boldsymbol{\mathcal{X}}^-_{k+1,i}), \quad i = 0, \ldots, 2n_x. \tag{2.55}$$

3. The predicted observation mean $\mathbf{m}_{k+1}$, the predicted measurement covari-

ance $\mathbf{S}_{k+1}$ and the state and measurement cross-covariance matrix $\mathbf{C}_{k+1}$ are computed:

$$\mathbf{m}_{k+1} = \sum_{i=0}^{2n_x} W_i^{(m)} \hat{\boldsymbol{\mathcal{Y}}}_{k+1,i}, \tag{2.56}$$

$$\mathbf{S}_{k+1} = \sum_{i=0}^{2n_x} W_i^{(c)} (\hat{\boldsymbol{\mathcal{Y}}}_{k+1,i} - \mathbf{m}_{k+1})(\hat{\boldsymbol{\mathcal{Y}}}_{k+1,i} - \mathbf{m}_{k+1})^\top + \mathbf{R}_{k+1}, \tag{2.57}$$

$$\mathbf{C}_{k+1} = \sum_{i=0}^{2n_x} W_i^{(c)} (\boldsymbol{\mathcal{X}}^-_{k+1,i} - \boldsymbol{\mu}^-_{k+1})(\hat{\boldsymbol{\mathcal{Y}}}_{k+1,i} - \mathbf{m}_{k+1})^\top. \tag{2.58}$$

4. The filter gain $\mathbf{D}_{k+1}$, filtered state mean $\boldsymbol{\mu}_{k+1}$ and covariance $\mathbf{P}_{k+1}$, conditional on the measurement $\mathbf{y}_{k+1}$ are computed:

$$\mathbf{D}_{k+1} = \mathbf{C}_{k+1}\mathbf{S}^{-1}_{k+1}, \tag{2.59}$$

$$\boldsymbol{\mu}_{k+1} = \boldsymbol{\mu}^-_{k+1} + \mathbf{D}_{k+1}(\mathbf{y}_{k+1} - \mathbf{m}_{k+1}), \tag{2.60}$$

$$\mathbf{P}_{k+1} = \mathbf{P}^-_{k+1} - \mathbf{D}_{k+1}\mathbf{S}_{k+1}\mathbf{D}^\top_{k+1}. \tag{2.61}$$

The filtering equations above may be similarly derived as the EKF equations, replacing the linear approximations with the UT-based approximations. As can be easily seen, the advantage that the UKF algorithm enjoys over EKF is that the nonlinearity is approximated using several points instead of having a linear approximation at one point. This makes the UT more capable of capturing the higher order moments caused by the nonlinear transformation in comparison with Taylor series-based approximations [79]. Furthermore, the UKF does not require the dynamic and measurement models to be differentiable and the computation of their Jacobian matrices is not necessary. As opposed to the SLF algorithm, the UKF does not need any closed form evaluation of expected values since only computations of dynamic and measurement models must be performed. The UKF accuracy, however, trails that of the SLF, since the latter makes use of a more extensive area in its approximation, whilst the UKF considers only a specific number of points within the region. One

disadvantage of the UKF over the EKF is the higher computational cost.

2.3.3 Particle filtering and optimisation-based estimators

Although Gaussian approximations work well for many filtering problems, the filtering distributions may occasionally be, for instance, multi-modal and non-Gaussian, or a number of the state components may be discrete, for which Gaussian approximations become inappropriate. In such situations, particle filtering can be a better alternative. A particle filter (PF) [66, 70] employs Monte Carlo sampling to approximate the multi-dimensional integrations involved in the prediction and update steps of the Bayesian filtering equations (2.18). The problem with such integrals is that closed form evaluation is only possible in a few specific cases and in general, numerical techniques are required. Monte Carlo methods are therefore exploited so that obtaining samples from the distribution and using sample averages to estimate the quantities replaces the closed form calculation of statistical values.

Belonging to the class of particle filters is the ensemble Kalman Filter (EnKF) [206, 207] that is based on the idea of determining estimates for $\mathbf{P}_{k+1}^{(\epsilon,e)}$ and $\mathbf{P}_{k+1}^{(e,e)}$ using random samples instead of deterministic ones. Detailed expositions on algorithmic and theoretical aspects of particle filtering are given in [67, 66, 70, 208].

In addition to conditional-density-approximation-based estimators, several nonlinear state estimation methods that exploit an optimisation strategy to solve nonlinear state estimation problems have been developed. These techniques were proposed with the specific goal of handling constraints on states and parameters in the estimation process [30]. An estimator that makes use of an explicit optimisation-based approach for state and parameter estimation of nonlinear dynamic processes (described by ODEs) is the moving horizon estimator (MHE) [209]. At every time step, the MHE solves an optimisation problem, thus readily handling constraints and bounds on state variables. The standard MHE formulation includes an arrival cost term that accounts for the accumulated state estimate uncertainties till the current window of interest. For general nonlinear constrained state estimation, the arrival cost cannot be determined analytically, however, for constrained linear ODE systems, the arrival

cost can be approximated using the cost evaluated for the unconstrained problem [210]. For nonlinear systems, the arrival cost is typically calculated by approximating a constrained nonlinear ODE system by an unconstrained linear time-varying system [211].

With this work's high-dimensional and real-time applications in mind, particle filters are known to become computationally expensive for large dimensions. For high-dimensional systems and long window sizes, optimisation-based formulations require a computationally intensive optimisation problem to be solved at every time step [30]. Kalman filtering is therefore the main focus in this thesis.

2.3.4 Unscented Rauch-Tung-Striebel smoothing

So far, only filtering algorithms have been discussed. These make use of measurements obtained before and at the present instant for determining the optimal estimate of the present state (and potentially future states). However, on occasions, it is important to obtain state estimates for every time instant conditional on all observations, including future measurements. Bayesian smoothing solves such a problem, which is described here for the unscented Rauch-Tung-Striebel smoother (URTSS) [81, 82], which is an unscented transform-based RTS smoother (or a forward-backward UKS smoother) used to approximate Bayesian smoothing solutions for nonlinear state-space models using Gaussian approximations, where the nonlinearity is approximated using the unscented transform.

To obtain such an approximation, the approximate mean and covariance of the filtering distribution, given in equation (2.25), are assumed to be available. The joint distribution of $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ given $\mathbf{y}_{1:k}$ may be given by

$$p(\mathbf{x}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k}) = p(\mathbf{x}_{k+1}|\mathbf{x}_k)p(\mathbf{x}_k|\mathbf{y}_{1:k}). \tag{2.62}$$

Equation (2.62) is approximated by a Gaussian approximation using the unscented

transform, as follows:

$$\begin{pmatrix} \mathbf{x}_k \\ \mathbf{x}_{k+1} \end{pmatrix} |\mathbf{y}_{1:k} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{\mu}_k \\ \boldsymbol{\mu}_{k+1}^- \end{pmatrix}, \begin{pmatrix} \mathbf{P}_k & \mathbf{C}_{k+1} \\ \mathbf{C}_{k+1}^\top & \mathbf{P}_{k+1}^- \end{pmatrix} \right). \tag{2.63}$$

This can be computed using the unscented transform for the nonlinear transformation given by equation (2.13a). By re-writing equation (2.62) as a conditional distribution of $\mathbf{x}_k$ given $\mathbf{x}_{k+1}$ and $\mathbf{y}_{1:k}$, we have that

$$p(\mathbf{x}_k|\mathbf{x}_{k+1}, \mathbf{y}_{1:k}) = \frac{p(\mathbf{x}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k})}{p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})}, \tag{2.64}$$

but using the Markov property yields

$$p(\mathbf{x}_k|\mathbf{x}_{k+1}, \mathbf{y}_{1:K}) = \frac{p(\mathbf{x}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k})}{p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})}. \tag{2.65}$$

Distribution (2.63) is Gaussian and we may therefore exploit the Gaussian distribution computation rules to obtain the Gaussian conditional distribution of equation (2.65), resulting in the following approximation:

$$p(\mathbf{x}_k|\mathbf{x}_{k+1}, \mathbf{y}_{1:K}) \approx \mathcal{N}(\mathbf{x}_k|\boldsymbol{\mu}_{k+1}^{'}, \mathbf{P}_{k+1}^{'}), \tag{2.66}$$

where

$$\mathbf{D}_k = \mathbf{C}_{k+1}(\mathbf{P}_{k+1}^-)^{-1}, \tag{2.67}$$

$$\boldsymbol{\mu}_{k+1}^{'} = \boldsymbol{\mu}_k + \mathbf{D}_k(\mathbf{x}_{k+1} - \boldsymbol{\mu}_{k+1}^-), \tag{2.68}$$

$$\mathbf{P}_{k+1}^{'} = \mathbf{P}_k - \mathbf{D}_k\mathbf{P}_{k+1}^-\mathbf{D}_k^\top. \tag{2.69}$$

The joint distribution of $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ given $\mathbf{y}_{1:K}$ is

$$p(\mathbf{x}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:K}) = p(\mathbf{x}_k|\mathbf{x}_{k+1}, \mathbf{y}_{1:K})p(\mathbf{x}_{k+1}|\mathbf{y}_{1:K}), \tag{2.70}$$

so that assuming knowledge of the smoothing distribution of the following time instant

and taking this to be Gaussian and given by

$$p(\mathbf{x}_{k+1}|\mathbf{y}_{1:K}) \approx \mathcal{N}(\mathbf{x}_{k+1}|\boldsymbol{\mu}^s_{k+1}, \mathbf{P}^s_{k+1}), \tag{2.71}$$

the distribution (2.70) then becomes

$$\begin{pmatrix} \mathbf{x}_k \\ \mathbf{x}_{k+1} \end{pmatrix} |\mathbf{y}_{1:K} \sim \mathcal{N}(\boldsymbol{\mu}^{''}_{k+1}, \mathbf{P}^{''}_{k+1}), \tag{2.72}$$

where

$$\boldsymbol{\mu}^{''}_{k+1} = \begin{pmatrix} \boldsymbol{\mu}_k + \mathbf{D}_k(\mathbf{x}_{k+1} - \boldsymbol{\mu}^-_{k+1}) \\ \boldsymbol{\mu}^s_{k+1} \end{pmatrix}, \tag{2.73}$$

$$\mathbf{P}^{''}_{k+1} = \begin{pmatrix} \mathbf{D}_k\mathbf{P}^s_{k+1}\mathbf{D}^\top_k + \mathbf{P}^{'}_{k+1} & \mathbf{D}_k\mathbf{P}^s_{k+1} \\ \mathbf{P}^s_{k+1}\mathbf{D}^\top_k & \mathbf{P}^s_{k+1} \end{pmatrix}. \tag{2.74}$$

At time instant k, the smoothing distribution is obtained as a Gaussian approximation by marginalizing over $\mathbf{x}_{k+1}$:

$$p(\mathbf{x}_k|\mathbf{y}_{1:K}) \approx \mathcal{N}(\mathbf{x}_k|\boldsymbol{\mu}^s_k, \mathbf{P}^s_k), \tag{2.75}$$

where

$$\boldsymbol{\mu}^s_k = \boldsymbol{\mu}_k + \mathbf{D}_k(\boldsymbol{\mu}^s_{k+1} - \boldsymbol{\mu}^-_{k+1}), \tag{2.76}$$

$$\mathbf{P}^s_k = \mathbf{P}_k + \mathbf{D}_k(\mathbf{P}^s_{k+1} - \mathbf{P}^-_{k+1})\mathbf{D}^\top_k. \tag{2.77}$$

Algorithm 2.3 summarises the unscented RTS smoother algorithm.

Algorithm 2.3 (*The URTSS algorithm*) The unscented RTS smoother algorithm for the additive model of equations (2.13) is as follows:

- Initialisation: The algorithm starts from the filtered estimates of the last time instant, with mean $\boldsymbol{\mu}^s_K = \boldsymbol{\mu}_K$ and covariance $\mathbf{P}^s_K = \mathbf{P}_K$. The following recur-

sions run backward for $k = K-1, \ldots, 0$.

- Sigma points are formed:

$$\begin{aligned}
\boldsymbol{\mathcal{X}}_{k,0} &= \boldsymbol{\mu}_k, \\
\boldsymbol{\mathcal{X}}_{k,i} &= \boldsymbol{\mu}_k + (\sqrt{(n_x+\lambda)\mathbf{P}_k})_i, \quad i = 1, \ldots, n_x, \\
\boldsymbol{\mathcal{X}}_{k,i} &= \boldsymbol{\mu}_k - (\sqrt{(n_x+\lambda)\mathbf{P}_k})_i, \quad i = n_x+1, \ldots, 2n_x.
\end{aligned} \tag{2.78}$$

- Sigma point propagation through the dynamic model yields

$$\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} = \mathbf{f}(\boldsymbol{\mathcal{X}}_{k,i}), i = 0, \ldots, 2n_x. \tag{2.79}$$

- The predicted mean $\boldsymbol{\mu}_{k+1}^-$, its covariance $\mathbf{P}_{k+1}^-$ and cross-covariance $\mathbf{C}_{k+1}$ are computed as follows:

$$\boldsymbol{\mu}_{k+1}^- = \sum_{i=0}^{2n_x} W_i^{(m)} \hat{\boldsymbol{\mathcal{X}}}_{k+1,i}, \tag{2.80}$$

$$\mathbf{P}_{k+1}^- = \sum_{i=0}^{2n_x} W_i^{(c)} (\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \boldsymbol{\mu}_{k+1}^-)(\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \boldsymbol{\mu}_{k+1}^-)^\top + \mathbf{Q}_k, \tag{2.81}$$

$$\mathbf{C}_{k+1} = \sum_{i=0}^{2n_x} W_i^{(c)} (\boldsymbol{\mathcal{X}}_{k,i} - \boldsymbol{\mu}_k)(\hat{X}_{k+1,i} - \boldsymbol{\mu}_{k+1}^-)^\top, \tag{2.82}$$

where the weights $W_i^{(m)}$ and $W_i^{(c)}$ are given by equation (2.49).

- The smoother gain $\mathbf{D}_k$, smoothed mean $\boldsymbol{\mu}_k^s$ and smoothed covariance $\mathbf{P}_k^s$ are calculated using the following equations:

$$\mathbf{D}_k = \mathbf{C}_{k+1}(\mathbf{P}_{k+1}^-)^{-1}, \tag{2.83}$$

$$\boldsymbol{\mu}_k^s = \boldsymbol{\mu}_k + \mathbf{D}_k(\boldsymbol{\mu}_{k+1}^s - \boldsymbol{\mu}_{k+1}^-), \tag{2.84}$$

$$\mathbf{P}_k^s = \mathbf{P}_k + \mathbf{D}_k(\mathbf{P}_{k+1}^s - \mathbf{P}_{k+1}^-)\mathbf{D}_k^\top. \tag{2.85}$$

The two-filter smoother [212, 213, 214] is another approach which differs from RTS-style forward-backward smoothing. Despite the many advantages of two-filter smoothing, this is not particularly well suited for the nonlinear case due to the required construction of artificial probability densities which ensure a normalisable backward filter. In view of this difficulty, this thesis bases the smoothing algorithms on the forward-backward smoothing methodology.

2.4 State estimation for nonlinear descriptor systems

Several researchers tend to convert the descriptor problem to a standard state-space formulation (e.g. [1, 215]), however this may introduce significant numerical errors where ODE solvers are used [21]. The conversion may also lead to the violation of algebraic constraints and makes measurements that are functions of algebraic states redundant for state estimation [21]. This was an important motivation for researchers to retain a descriptor formulation for estimation purposes [216, 21, 85].

The problem of state estimation for linear descriptor systems has been well developed by various authors [217, 218, 219, 220]. By contrast, most research efforts in designing filters and observers for the nonlinear case are more recent [75, 76].

A full-order observer is reported in [221] for a nonlinear descriptor system class subject to unknown inputs and faults. The observer design is based on the idea of dividing the system into a dynamic system and a static one. Other researchers have designed observers for nonlinear DAE systems described in terms of a linear part and a nonlinearity that is assumed to be Lipschitz [222, 223, 224]. This assumption is relaxed in [225, 226], however the nonlinearity is taken to obey a quadratic inequality so that the observer error system is represented as a Lur'e descriptor system. Another estimation scheme is proposed in [227], where the authors reformulate the original descriptor model into an ODE form on a restricted manifold [228]. Other efforts exploit linearisation methods, such as those presented in [229, 230]. In [230], index reduction techniques are further proposed to handle high-index models. The observer design put forward in [231] proposes a Lyapunov-based approach. Åslund and Frisk

[232] consider index 1 nonlinear DAE systems and extend the work of Nikoukhah [233] so that an observer formulated as an index 1 DAE is developed. The authors linearise the error dynamics where under the right conditions, the estimator error dynamics are guaranteed to be locally stable. Another approach to the state estimation of nonlinear DAEs is the MHE scheme [234], however this becomes computationally intensive for large horizons or large-scale models [85].

The application of Kalman filtering for nonlinear DAE systems is described in [85], where a modified extended Kalman filter is proposed. However, this technique only treats measurements as functions of differential state variables. The technique was therefore extended in [235, 21] to accommodate measurements including algebraic state variables. More recently, this method was further modified in [236] to handle measurements taken at multiple sampling rates. An unscented Kalman filter [79] approach to nonlinear descriptor systems was presented in [84], who linearise the DAE model to obtain a recursive least squares estimation approach in which the unscented transformation is applied. The work in [235, 21] proposes a heuristic extension of the standard UKF approach to nonlinear DAEs, which was later heuristically revised in [83], where the scaled unscented transformation [237] is employed whilst excluding augmented state calculations.

2.5 Conclusion

This chapter has discussed different classes of spatio-temporal models which are commonly used and reveals how the study of nonlinear PDAEs is warranted for the applications we consider in this thesis. Nonlinear PDAEs retain the physical meaning of the underlying process governing the system whilst being able to represent the spatially heterogeneous dynamics. Model reduction methods were also described to show how a spatio-temporal field may be approximated in finite-dimensional form which is readily applicable to the available state estimation tools which were also described in this same chapter. The prevalent state estimation schemes were discussed with particular emphasis to those pertaining to our work, namely, the UKF

and URTSS methods. This leaves the development of appropriate state estimation tools for nonlinear descriptor systems to be addressed in the next chapter.

Chapter 3

Estimation of temporal descriptor systems

As it was described in Chapter 2, descriptor systems consist of a general form of sets of equations that are often a product of modelling physical systems or large-scale systems and involve both dynamic and static relationships between variables. Since such systems are typically only partially observed, an estimation problem arises. In their original form, the optimal filtering equations for such systems are rarely used directly in practical situations due to high computational cost. Numerical approximations are therefore required so the optimal filtering equations are divided into steps. Since we are primarily interested in the first and second moments of the filtering distributions, we employ unscented transform-based Gaussian approximations [197].

Section 3.1 extends the UKF and URTSS algorithms to temporal nonlinear descriptor systems with stochastic differential equations and deterministic algebraic equations, for which an electrochemical case study is used to validate the proposed methodology. Section 3.2 derives modified UKF and URTSS techniques for temporal nonlinear DAE systems having stochastic differential and algebraic equations, as would be the case when model uncertainties exist in all equations. A numerical example illustrates the derived solutions. All proposed methods allow for measurements that are functions of both differential and algebraic states.

3.1 Nonlinear descriptor systems with deterministic algebraic equations

Deterministic algebraic equations typically arise from physical principles that constrain the evolution of differential states such that all states obey the descriptor formulation. Taking chemical kinetics as an example, reaction rates are governed by differential equations, while algebraic equations typically represent equilibrium properties, pseudo-stationary assumptions and charge or mass balances [22]. The foregoing means that the algebraic state covariance would encode useful information which in previously developed filtering algorithms, remains unknown. In our method, we ensure that statistical information from all states is used throughout since this would best capture all system properties.

3.1.1 Problem statement

We consider nonlinear DAE models of the form given by equations (2.14) and remove the algebraic state process noise $\mathbf{e}_k$ to have deterministic algebraic equations. Our aim is to obtain Gaussian approximations to the filtering distributions $p(\mathbf{X}_k|\mathbf{y}_{1:k})$ and smoothing distributions $p(\mathbf{X}_k|\mathbf{y}_{1:K})$ for time steps k running from 0 to K. This thesis shall only consider semi-explicit DAEs of differentiation index 1 unless otherwise stated.

3.1.2 An unscented Kalman filter for nonlinear DAEs with deterministic algebraic equations

This section presents a filtering algorithm that is employed in the forward step of the proposed RTS smoother for nonlinear descriptor systems. In contrast with previously proposed UKF algorithms, our technique provides the covariance of both the differential and algebraic state estimates. For this purpose, we make use of the discrete-time UKF [205, 78, 197, 79] - an optimal filtering algorithm that employs the unscented transformation. Since the evolution of algebraic states is completely described by the

evolution of differential states such that all algebraic constraints are satisfied [30], such typical natural constraints enable algebraic states to be considered as noise-free [85, 21]. In the algebraic equation, the differential states are nonlinearly transformed to yield the algebraic states. A way of transforming statistical information through the nonlinear function $\mathbf{g}$ must therefore be sought. Since approximating a probability distribution is an easier approach than approximating a nonlinear function [238, 79], the unscented transformation is employed to propagate the mean and covariance of the differential state estimates through the nonlinear function.

The theoretical results obtained in [239, 197] may be used to quantify the accuracy of the resulting algebraic state estimates mean and covariance. The true posterior mean and the UT predicted mean of the algebraic state estimates are identical till the third order and errors are only present in the fourth and higher-order terms. Similarly, it can be shown that the true posterior covariance and the UT posterior covariance of the algebraic state estimates are identical for the first two terms and errors are introduced only for the fourth and higher order moments. If, however, a priori knowledge of the shape of the prior distribution is available, a non-zero value for the UT parameter β may be set, thus minimising the error in some of the higher order (≥ 4) moments. For a Gaussian random variable, Julier [237] shows how the error in the kurtosis of the posterior distribution is minimised when $\beta = 2$. A detailed treatment of the UT is given in [239, 79, 197, 237].

The standard formulation of the UKF algorithm is used to obtain the filtered differential state estimates. A notable difference, however, is that the computation of the predicted differential state and predicted observation mean and covariance requires algebraic state sigma points obtained using algebraic equations. This ensures consistency with respect to the descriptor model. Additionally, we note that measurements of algebraic states are functions of differential states and are therefore treated as such.

The filtered algebraic state estimates at $k+1$ are computed by using a set of $2n_x+1$ sigma points that are selected such that the mean and covariance of the sigma points are $\mathbf{x}_{k+1}$ and $\mathbf{P}^{(x)}_{k+1}$, respectively. By propagating each sigma point through the non-

linear function $\mathbf{g}$, the statistics of the transformed sigma points may be calculated to obtain an estimate for the nonlinearly transformed mean and covariance. Algorithm 3.1 summarises the UKF algorithm for nonlinear difference-algebraic equations with deterministic algebraic equations.

Algorithm 3.1(*The UKF algorithm for nonlinear DAEs with deterministic algebraic equations*). The UKF computations are initialised with a prior distribution that has a mean $\mathbf{x}_0$ and covariance $\mathbf{P}_0^{(x)}$. The following steps are executed at every measurement time instant $k = 0, 1, 2, 3, \ldots$:

1. **Prediction step:**

   (a) The differential state sigma points $\boldsymbol{\mathcal{X}}_{k,i}$ are first computed, as follows:

$$\begin{aligned}
\boldsymbol{\mathcal{X}}_{k,0} =& \mathbf{x}_k, \\
\boldsymbol{\mathcal{X}}_{k,i} =& \mathbf{x}_k + (\sqrt{(n_x + \lambda)\mathbf{P}_k^{(x)}})_i, i = 1, \ldots, n_x, \\
\boldsymbol{\mathcal{X}}_{k,i} =& \mathbf{x}_k - (\sqrt{(n_x + \lambda)\mathbf{P}_k^{(x)}})_i, i = n_x + 1, \ldots, 2n_x,
\end{aligned} \tag{3.1}$$

   where λ is defined in equation (2.45). The vector $(\sqrt{(n_x + \lambda)\mathbf{P}_k^{(x)}})_i$ denotes the ith column of the matrix square root, $\mathbf{x}_k$ is the filtered estimate of differential states and $\mathbf{P}_k^{(x)}$ is the associated covariance matrix.

   (b) The algebraic state sigma points $\boldsymbol{\mathcal{Z}}_{k,i}$ are obtained using the algebraic equations $\mathbf{g}(\boldsymbol{\mathcal{X}}_{k,i}, \boldsymbol{\mathcal{Z}}_{k,i}) = \mathbf{0}$ to ensure consistency.

   (c) The sigma points $\boldsymbol{\mathcal{X}}_{k,i}$ and $\boldsymbol{\mathcal{Z}}_{k,i}$ are propagated through the nonlinear DAE model to get $\hat{\boldsymbol{\mathcal{X}}}_{k+1,i}$.

   (d) The predicted differential state estimate $\mathbf{x}_{k+1}^-$ and the predicted covariance

$\mathbf{P}_{k+1}^{(x)-}$ are computed as follows:

$$\mathbf{x}_{k+1}^{-} = \sum_{i=0}^{2n_x} W_i^{(m)} \hat{\boldsymbol{\mathcal{X}}}_{k+1,i}, \tag{3.2}$$

$$\mathbf{P}_{k+1}^{(x)-} = \sum_{i=0}^{2n_x} W_i^{(c)} (\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \mathbf{x}_{k+1}^{-})(\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \mathbf{x}_{k+1}^{-})^{\top} + \mathbf{Q}_k, \tag{3.3}$$

where the constant weights $W_i^{(m)}$ and $W_i^{(c)}$ are given by equations (2.49).

2. **Update step:**

   (a) The differential state sigma points $\boldsymbol{\mathcal{X}}_{k+1,i}^{-}$ are computed by performing unscented sampling with mean $\mathbf{x}_{k+1}^{-}$ and covariance $\mathbf{P}_{k+1}^{(x)-}$.

   (b) The algebraic state sigma points $\boldsymbol{\mathcal{Z}}_{k+1,i}^{-}$ are determined from the algebraic equations $\mathbf{g}(\boldsymbol{\mathcal{X}}_{k+1,i}^{-}, \boldsymbol{\mathcal{Z}}_{k+1,i}^{-}) = \mathbf{0}$.

   (c) The sigma points are propagated through the measurement model:

$$\hat{\boldsymbol{\mathcal{Y}}}_{k+1,i} = \mathbf{h}(\boldsymbol{\mathcal{X}}_{k+1,i}^{-}, \boldsymbol{\mathcal{Z}}_{k+1,i}^{-}). \tag{3.4}$$

   (d) The predicted observation mean $\mathbf{y}_{k+1}^{-}$, the predicted measurement covariance $\mathbf{S}_{k+1}$ and the state and measurement cross-covariance matrix $\mathbf{B}_{k+1}$ are computed as follows:

$$\mathbf{y}_{k+1}^{-} = \sum_{i=0}^{2n_x} W_i^{(m)} \hat{\boldsymbol{\mathcal{Y}}}_{k+1,i}, \tag{3.5}$$

$$\mathbf{S}_{k+1} = \sum_{i=0}^{2n_x} W_i^{(c)} (\hat{\boldsymbol{\mathcal{Y}}}_{k+1,i} - \mathbf{y}_{k+1}^{-})(\hat{\boldsymbol{\mathcal{Y}}}_{k+1,i} - \mathbf{y}_{k+1}^{-})^{\top} + \mathbf{R}_{k+1}, \tag{3.6}$$

$$\mathbf{B}_{k+1} = \sum_{i=0}^{2n_x} W_i^{(c)} (\boldsymbol{\mathcal{X}}_{k+1,i}^{-} - \mathbf{x}_{k+1}^{-})(\hat{\boldsymbol{\mathcal{Y}}}_{k+1,i} - \mathbf{y}_{k+1}^{-})^{\top}. \tag{3.7}$$

   (e) The filter gain $\mathbf{K}_{k+1}$, filtered differential state mean $\mathbf{x}_{k+1}$ and covariance $\mathbf{P}_{k+1}^{(x)}$

are calculated as follows:

$$\mathbf{K}_{k+1} = \mathbf{B}_{k+1}\mathbf{S}_{k+1}^{-1}, \tag{3.8}$$

$$\mathbf{x}_{k+1} = \mathbf{x}_{k+1}^{-} + \mathbf{K}_{k+1}(\mathbf{y}_{k+1} - \mathbf{y}_{k+1}^{-}), \tag{3.9}$$

$$\mathbf{P}_{k+1}^{(x)} = \mathbf{P}_{k+1}^{(x)-} - \mathbf{K}_{k+1}\mathbf{S}_{k+1}\mathbf{K}_{k+1}^{\top}. \tag{3.10}$$

(f) To obtain the filtered algebraic states $\mathbf{z}_{k+1}$, the differential state sigma points $\boldsymbol{\mathcal{X}}_{k+1,i}$ are first computed by performing unscented sampling with mean $\mathbf{x}_{k+1}$ and covariance $\mathbf{P}_{k+1}^{(x)}$.

(g) The algebraic state sigma points $\boldsymbol{\mathcal{Z}}_{k+1,i}$ are generated from the algebraic equations $\mathbf{g}(\boldsymbol{\mathcal{X}}_{k+1,i}, \boldsymbol{\mathcal{Z}}_{k+1,i}) = \mathbf{0}$.

(h) The filtered algebraic state mean $\mathbf{z}_{k+1}$ and its covariance $\mathbf{P}_{k+1}^{(z)}$ are computed:

$$\mathbf{z}_{k+1} = \sum_{i=0}^{2n_x} W_i^{(m)} \boldsymbol{\mathcal{Z}}_{k+1,i}, \tag{3.11}$$

$$\mathbf{P}_{k+1}^{(z)} = \sum_{i=0}^{2n_x} W_i^{(c)} (\boldsymbol{\mathcal{Z}}_{k+1,i} - \mathbf{z}_{k+1})(\boldsymbol{\mathcal{Z}}_{k+1,i} - \mathbf{z}_{k+1})^{\top}. \tag{3.12}$$

3.1.3 An unscented Rauch-Tung-Striebel smoother for nonlinear DAEs with deterministic algebraic equations

An unscented RTS smoothing (URTSS) algorithm that extends the work presented by Särkkä in [81, 82] is now proposed for application to nonlinear descriptor system models of the form given by equations (2.14). The URTSS method is used to obtain the differential state estimates. Once again, the algebraic state sigma points are computed using the algebraic equations. Then proceeding similarly as in Section 3.1.2, the unscented transform is used to obtain the algebraic state estimates from the differential state estimates. Algorithm 3.2 summarises the URTSS method for nonlinear DAEs with deterministic algebraic equations.

Algorithm 3.2 (*The URTSS algorithm for nonlinear DAEs with deterministic algebraic equations*). After computing the filtered state estimates, the following URTSS algorithm can be executed to get the smoothed state estimates for each time instant conditional to all measurements spanning time K:

1. The differential state sigma points $\boldsymbol{\mathcal{X}}_{k,i}$ are first generated by performing unscented sampling with mean $\mathbf{x}_k$ and covariance $\mathbf{P}_k^{(x)}$.

2. Consistency is ensured by obtaining the algebraic state sigma points $\boldsymbol{\mathcal{Z}}_{k,i}$ using the equation $\mathbf{g}(\boldsymbol{\mathcal{X}}_{k,i}, \boldsymbol{\mathcal{Z}}_{k,i}) = \mathbf{0}$.

3. The sigma points $\boldsymbol{\mathcal{X}}_{k,i}$ and $\boldsymbol{\mathcal{Z}}_{k,i}$ are propagated through the nonlinear DAE model to obtain $\hat{\boldsymbol{\mathcal{X}}}_{k+1,i}$.

4. The predicted state estimates $\mathbf{x}_{k+1}^-$ and the corresponding cross-covariance $\mathbf{C}_{k+1}$ and predicted covariance $\mathbf{P}_{k+1}^{(x)-}$ are determined using the following equations:

$$\mathbf{x}_{k+1}^- = \sum_{i=0}^{2n_x} W_i^{(m)} \hat{\boldsymbol{\mathcal{X}}}_{k+1,i}, \tag{3.13}$$

$$\mathbf{C}_{k+1} = \sum_{i=0}^{2n_x} W_i^{(c)} (\boldsymbol{\mathcal{X}}_{k,i} - \mathbf{x}_k)(\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \mathbf{x}_{k+1}^-)^\top, \tag{3.14}$$

$$\mathbf{P}_{k+1}^{(x)-} = \sum_{i=0}^{2n_x} W_i^{(c)} (\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \mathbf{x}_{k+1}^-)(\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \mathbf{x}_{k+1}^-)^\top + \mathbf{Q}_k. \tag{3.15}$$

5. The smoother gain $\mathbf{D}_k$ and associated smoothed mean $\mathbf{x}_k^s$ and covariance $\mathbf{P}_k^{(x)s}$ are calculated as follows:

$$\mathbf{D}_k = \mathbf{C}_{k+1} (\mathbf{P}_{k+1}^{(x)-})^{-1}, \tag{3.16}$$

$$\mathbf{x}_k^s = \mathbf{x}_k + \mathbf{D}_k (\mathbf{x}_{k+1}^s - \mathbf{x}_{k+1}^-), \tag{3.17}$$

$$\mathbf{P}_k^{(x)s} = \mathbf{P}_k^{(x)} + \mathbf{D}_k (\mathbf{P}_{k+1}^{(x)s} - \mathbf{P}_{k+1}^{(x)-}) \mathbf{D}_k^\top. \tag{3.18}$$

6. To determine the smoothed algebraic states $\mathbf{z}_k^s$, the differential state sigma points

$\boldsymbol{\mathcal{X}}^s_{k,i}$ are generated by carrying out unscented sampling with mean $\mathbf{x}^s_k$ and covariance $\mathbf{P}^{(x)s}_k$.

7. The algebraic state sigma points $\boldsymbol{\mathcal{Z}}^s_{k,i}$ are calculated using the algebraic equations $\mathbf{g}(\boldsymbol{\mathcal{X}}^s_{k,i}, \boldsymbol{\mathcal{Z}}^s_{k,i}) = \mathbf{0}$.

8. The smoothed algebraic state mean $\mathbf{z}^s_k$ and its covariance $\mathbf{P}^{(z)s}_k$ are computed:

$$\mathbf{z}^s_k = \sum_{i=0}^{2n_x} W^{(m)}_i \boldsymbol{\mathcal{Z}}^s_{k,i}, \tag{3.19}$$

$$\mathbf{P}^{(z)s}_k = \sum_{i=0}^{2n_x} W^{(c)}_i (\boldsymbol{\mathcal{Z}}^s_{k,i} - \mathbf{z}^s_k)(\boldsymbol{\mathcal{Z}}^s_{k,i} - \mathbf{z}^s_k)^\top. \tag{3.20}$$

At the last time step K, the smoothing and filtering distributions are identical and hence $\mathbf{x}^s_K = \mathbf{x}_K$, $\mathbf{z}^s_K = \mathbf{z}_K$, $\mathbf{P}^{(x)s}_K = \mathbf{P}^{(x)}_K$ and $\mathbf{P}^{(z)s}_K = \mathbf{P}^{(z)}_K$. Therefore, by running the above computations starting from the filtered estimates of the last time instant and proceeding backwards for $k = K-1, \ldots, 0$, the smoothed estimates are obtained.

3.1.4 Case study: a galvanostatic charge process of a thin-film nickel hydroxide electrode

The estimation performance of the UKF and URTSS algorithms developed here for nonlinear descriptor systems with deterministic algebraic equations is demonstrated by an electrochemical reaction considering a galvanostatic charge process of a thin-film nickel hydroxide electrode [240]. The problem was studied in [235, 21] to investigate the performance of EKF and UKF algorithms.

The operation can be charging, open-circuit or discharging if the value of the applied current density on the nickel electrode, i_{app}, is positive, zero or negative,

respectively. The nonlinear descriptor system is described by the following equations:

$$\frac{dx}{dt} = \frac{l_1 W}{\rho F V}, \tag{3.21}$$

$$0 = l_1 + l_2 - i_{app}, \tag{3.22}$$

where

$$l_1 = i_1[2(1-x)e^{\frac{0.5F}{RT}(z-\phi_1)} - 2xe^{\frac{-0.5F}{RT}(z-\phi_1)}], \tag{3.23}$$

$$l_2 = i_2[e^{\frac{F}{RT}(z-\phi_2)} - e^{\frac{-F}{RT}(z-\phi_2)}]. \tag{3.24}$$

The differential state x represents the mole fraction of nickel hydroxide and the algebraic state z denotes the potential difference across the solid-liquid interface. The differential equation (3.21) is known as the species balance equation and the algebraic equation (3.22) is referred to as the charge balance equation. The terms l_1 and l_2 are described by the Butler-Volmer relationship. The rest of the terms, that remain constant throughout the simulation, are process parameters that are summarised in Table 3.1. Their values and the initial state values are chosen as given in [235, 21]. In this simulation, the measurement is the algebraic state z and the DAE solver used follows the forward Euler scheme. It is assumed that x is affected by process noise q, z is noise-free and y is corrupted with observation noise r.

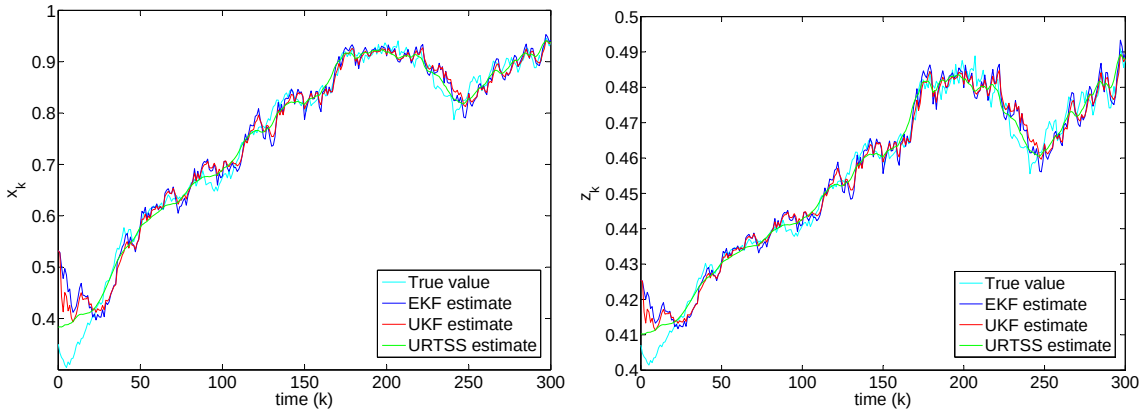
The estimation methods tested here are the EKF reported in [235, 21] and the UKF and URTSS proposed here. To our knowledge, the extended Kalman smoother (EKS) and UKS (two-filter smoother) methods have not yet been extended to nonlinear DAE systems and are therefore not investigated here. The parameters used for the EKF and the proposed UKF and URTSS techniques are given in Table 3.1.

A typical simulation result for the system under study is shown in Figures 3-1 and 3-2. Table 3.2 gives the root mean square error (RMSE) value statistics of EKF, UKF and URTSS estimates following 100 Monte Carlo simulations of the electrochemical reaction. Figure 3-3 further illustrates the results using RMSE plots. As expected, the smoother's performance is superior with a more significant improvement observed

Table 3.1: Parameters for the nonlinear DAE model and the estimation procedure.

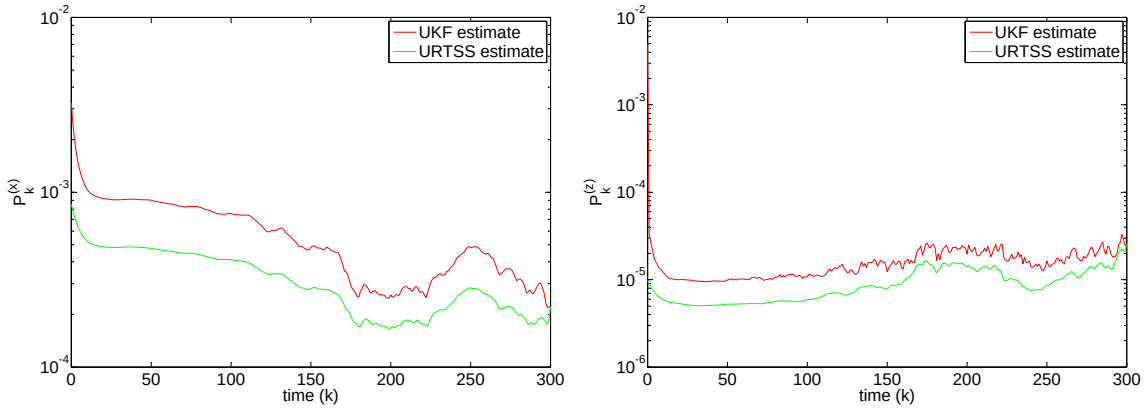
Symbol	Description	Value	Units
Time Domain			
Δt	Time step	15	s
K	Number of time steps used in estimation	300	n.a.
DAE Model			
F	Faraday's constant	96487	C/mol
R	Ideal gas constant	8.314	J/mol K
T	Temperature	298.15	K
ϕ_1	Equilibrium potential of nickel reaction	0.420	V
ϕ_2	Equilibrium potential of oxygen reaction	0.303	V
ρ	Density of nickel active material	3.4	g/cm^3
W	Molecular weight	92.7	g/mol
V	Effective length	10^{-5}	cm
i_{app}	Applied current density on nickel electrode	10^{-5}	A/cm^2
i_1	Exchange current density of nickel reaction	10^{-4}	A/cm^2
i_2	Exchange current density of oxygen reaction	10^{-8}	A/cm^2
q_k	Differential state process noise variance	10^{-4}	n.a.
r_k	Observation noise variance	10^{-4}	V^2
EKF state estimation [235, 21]			
$\mathbf{P}_0$	State estimation error covariance	$0.005 \times \mathbf{I}_2$	n.a.
Proposed UKF and URTSS state estimation			
$P_0^{(x)}$	Differential state estimation error variance	0.005	n.a.
α	Controls spread of sigma points	1	n.a.
β	Prior knowledge of sigma points distribution	2	n.a.
κ	Scaling parameter $(3 - n_x)$	2	n.a.
λ	Scaling parameter	2	n.a.

for the initial time steps since smoothed estimates are based on the entire interval of measurements rather than just prior means. The remaining initial smoother error is then largely a result of the Gaussian approximations employed. Furthermore, it may be noted that the unscented transformation is a better approximation to the model nonlinearities, as evidenced by the lower UKF and URTSS errors.



(a) (b)

Figure 3-1: True value, EKF estimate, UKF estimate and URTSS estimate of (a) differential state x_k (dimensionless) and (b) algebraic state z_k (in Volts) for a single instantiation of the electrochemical reaction.

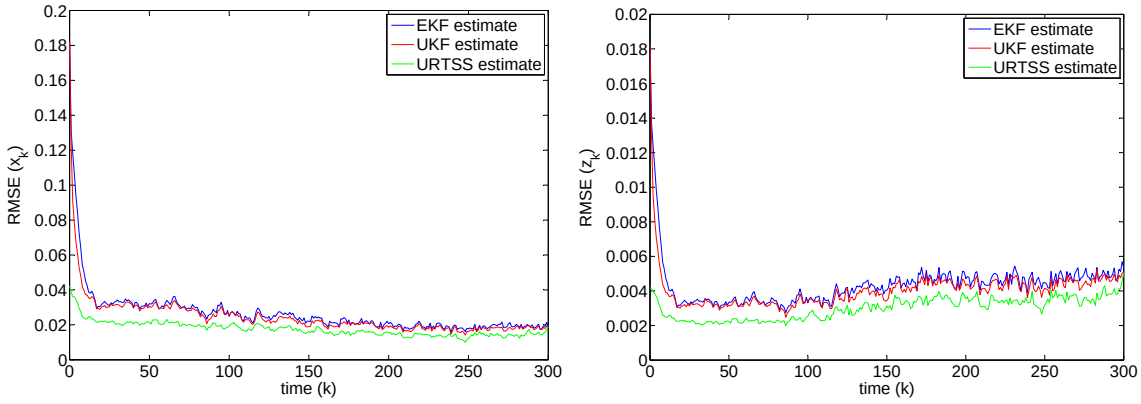


(a) (b)

Figure 3-2: Covariance plots of the UKF and URTSS state estimates of (a) differential state x_k and (b) algebraic state z_k, plotted on a log scale for the single instantiation of the electrochemical reaction shown in Figure 3-1.

Table 3.2: Mean and standard deviation of RMSE values for 100 Monte Carlo runs of the electrochemical process.

Method	EKF ([235, 21])	UKF	URTSS
E (RMSE (x_k))	0.0268	0.0247	0.0177
STD (RMSE (x_k))	0.0156	0.0137	0.0043
E (RMSE (z_k))	0.0044	0.0041	0.0030
STD (RMSE (z_k))	0.0014	0.0012	0.0006



(a) (b)

Figure 3-3: RMSE plots for (a) differential state x_k and (b) algebraic state z_k following 100 Monte Carlo simulations of the electrochemical process.

3.2 Nonlinear descriptor systems with stochastic algebraic equations

In several situations, significant model uncertainties exist in both differential and algebraic equations, often as a result of modelling errors, unmodelled dynamics, model reduction and associated approximations. The effect of such uncertainties may be approximated stochastically and algebraic equations would therefore also be stochastic in nature.

This section extends UKF and URTSS algorithms to nonlinear DAE systems where all states are affected by stochastic uncertainties. The mean and covariance of both differential and algebraic state estimates are computed and utilized for sigma point generation through the backward smoothing pass that recursively computes corrections to the forward filtering result. The unscented transform is employed to form Gaussian approximations to smoothing distributions. The performance of the proposed smoothing algorithm is demonstrated by a simulation.

The aforementioned advances in modifying Kalman filtering to nonlinear DAE systems assume that the algebraic equations are free of any uncertainty. Although this is typically the case, in the transformation from a nonlinear PDAE to a DAE, effects of model approximations have to be included. By representing the latter effects

stochastically, we propose to compute and exploit the mean and covariance of the algebraic state estimates that are required to complete the estimate information obtained and, unlike previously developed methods, ensure that the sigma points encode the mean and covariance of both differential and algebraic states. In this context, we derive a new descriptor form of the discrete-time UKF algorithm, where differential and algebraic state filtering distributions are derived as unscented transform-based Gaussian approximations. Motivated by the need for accurate estimation techniques that enable successful preview control of wind turbines, the performance of the proposed descriptor filtering framework is demonstrated through the accurate estimation of wind flow velocity and pressure given sparse noisy velocity measurements from realistic atmospheric boundary layer wind flow data.

3.2.1 Problem statement

This section considers general descriptor models of the form given by equations (2.14). For such systems where all equations are now stochastic in nature, the aim is to approximate the filtering distributions $p(\mathbf{X}_k|\mathbf{y}_{1:k})$ and smoothing distributions $p(\mathbf{X}_k|\mathbf{y}_{1:K})$ using Gaussian approximations, for time steps k running from 0 to K.

3.2.2 Unscented Kalman filtering for nonlinear DAEs with stochastic algebraic equations

Most standard state estimation tools do not handle nonlinear descriptor systems and we cannot treat a DAE system as a constrained ODE system [30]. An appropriate estimator must therefore be studied and developed for system models where uncertainties and unmodelled dynamics in both difference and algebraic equations are represented stochastically.

For this purpose, a new descriptor form of the UKF algorithm is derived, where all equations may be stochastic. The resulting method differs from the standard UKF algorithm for ODE systems in that the differential and algebraic states are neither estimated by running two separate standard filters, nor augmented into one

state vector throughout the whole algorithm. Instead, one forward filtering pass yields the differential and algebraic state filtering distributions derived as unscented transform-based Gaussian approximations. In particular, state and measurement predictions make use of statistical information from all states to ensure that the set of sigma points computing the projected prediction correctly encode the mean and covariance of both differential and algebraic states (see equations (3.28) and (3.33)). Furthermore, unlike [85, 21], the unscented transform is employed to determine the predicted and filtered algebraic state estimates (see equations (3.33) and (3.40)). This conceptually simple step provides the required statistical information (mean and covariance) for algebraic state estimates with sufficient accuracy, after differential state estimates are nonlinearly transformed in equation (2.14b). An approximation to the optimal filtering solution is now derived for nonlinear descriptor systems, based on the unscented transform methodology.

Consider the joint distribution of $\mathbf{x}_k$, $\mathbf{z}_k$ and $\mathbf{x}_{k+1}$ given $\mathbf{y}_{1:k}$:

$$p(\mathbf{x}_k, \mathbf{z}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k}) = p(\mathbf{z}_k|\mathbf{x}_k, \mathbf{x}_{k+1}, \mathbf{y}_{1:k})\; p(\mathbf{x}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k}). \tag{3.25}$$

Re-arranging equation (3.25) as

$$p(\mathbf{x}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k}) = \frac{p(\mathbf{x}_k, \mathbf{z}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k})}{p(\mathbf{z}_k|\mathbf{x}_k, \mathbf{x}_{k+1}, \mathbf{y}_{1:k})}, \tag{3.26}$$

a Gaussian approximation

$$\begin{pmatrix} \mathbf{x}_k \\ \mathbf{x}_{k+1} \end{pmatrix} |\mathbf{y}_{1:k} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{\mu}_k^{(x)} \\ \boldsymbol{\mu}_{k+1}^{(x)-} \end{pmatrix}, \begin{pmatrix} \mathbf{P}_k^{(x)} & \mathbf{C}_{k+1}^{(x,x)-} \\ \mathbf{C}_{k+1}^{(x,x)-\top} & \mathbf{P}_{k+1}^{(x)-} \end{pmatrix} \right) \tag{3.27}$$

may be generated using the unscented transform, where $\boldsymbol{\mu}^{(\cdot)}$ represents the mean and $\mathbf{P}^{(\cdot)}$ and $\mathbf{C}^{(\cdot,\cdot)}$ denote the corresponding state covariance and cross-covariance matrices, respectively. The '$-$' superscript denotes the predicted quantities. Distribution

(3.27) is computed by evaluating the distribution

$$\begin{pmatrix} \mathbf{x}_k \\ \mathbf{z}_k \end{pmatrix} |\mathbf{y}_{1:k} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{\mu}_k^{(x)} \\ \boldsymbol{\mu}_k^{(z)} \end{pmatrix}, \begin{pmatrix} \mathbf{P}_k^{(x)} & \mathbf{C}_k^{(x,z)} \\ \mathbf{C}_k^{(x,z)\top} & \mathbf{P}_k^{(z)} \end{pmatrix} \right), \tag{3.28}$$

using the unscented transform, so the joint distribution of $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ may now be computed as a Gaussian approximation, again making use of the unscented transform. Marginalising distribution (3.27) over $\mathbf{x}_k$ yields

$$\mathbf{x}_{k+1}|\mathbf{y}_{1:k} \sim \mathcal{N}(\boldsymbol{\mu}_{k+1}^{(x)-}, \mathbf{P}_{k+1}^{(x)-}). \tag{3.29}$$

The joint distribution of $\mathbf{x}_{k+1}$, $\mathbf{z}_{k+1}$ and $\mathbf{y}_{k+1}$ given $\mathbf{y}_{1:k}$ is

$$p(\mathbf{x}_{k+1}, \mathbf{z}_{k+1}, \mathbf{y}_{k+1}|\mathbf{y}_{1:k}) = p(\mathbf{z}_{k+1}|\mathbf{x}_{k+1}, \mathbf{y}_{1:k+1})\ p(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}|\mathbf{y}_{1:k}), \tag{3.30}$$

which may be written as

$$p(\mathbf{x}_{k+1}, \mathbf{y}_{k+1}|\mathbf{y}_{1:k}) = \frac{p(\mathbf{x}_{k+1}, \mathbf{z}_{k+1}, \mathbf{y}_{k+1}|\mathbf{y}_{1:k})}{p(\mathbf{z}_{k+1}|\mathbf{x}_{k+1}, \mathbf{y}_{1:k+1})}. \tag{3.31}$$

An unscented transform-based Gaussian approximation can be calculated for this distribution:

$$\begin{pmatrix} \mathbf{x}_{k+1} \\ \mathbf{y}_{k+1} \end{pmatrix} |\mathbf{y}_{1:k} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{\mu}_{k+1}^{(x)-} \\ \mathbf{m}_{k+1} \end{pmatrix}, \begin{pmatrix} \mathbf{P}_{k+1}^{(x)-} & \mathbf{C}_{k+1}^{(x,y)} \\ \mathbf{C}_{k+1}^{(x,y)\top} & \mathbf{S}_{k+1} \end{pmatrix} \right), \tag{3.32}$$

where $\mathbf{m}_{k+1}$ is the predicted observation mean and $\mathbf{S}_{k+1}$ is the predicted measurement covariance. Once again, this requires the concatenation of the differential and algebraic states, as follows:

$$\begin{pmatrix} \mathbf{x}_{k+1} \\ \mathbf{z}_{k+1} \end{pmatrix} |\mathbf{y}_{1:k} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{\mu}_{k+1}^{(x)-} \\ \boldsymbol{\mu}_{k+1}^{(z)-} \end{pmatrix}, \begin{pmatrix} \mathbf{P}_{k+1}^{(x)-} & \mathbf{C}_{k+1}^{(x,z)-} \\ \mathbf{C}_{k+1}^{(x,z)-\top} & \mathbf{P}_{k+1}^{(z)-} \end{pmatrix} \right). \tag{3.33}$$

Since the distribution (3.32) is Gaussian, the conditional distribution $p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1})$

may be determined to approximate the differential state filtering distribution of the instant $k+1$ using a Gaussian approximation:

$$\mathbf{x}_{k+1}|\mathbf{y}_{1:k+1} \sim \mathcal{N}(\boldsymbol{\mu}_{k+1}^{(x)}, \mathbf{P}_{k+1}^{(x)}), \tag{3.34}$$

where

$$\boldsymbol{\mu}_{k+1}^{(x)} = \boldsymbol{\mu}_{k+1}^{(x)-} + \mathbf{D}_{k+1}(\mathbf{y}_{k+1} - \mathbf{m}_{k+1}), \tag{3.35}$$

$$\mathbf{P}_{k+1}^{(x)} = \mathbf{P}_{k+1}^{(x)-} - \mathbf{D}_{k+1}\mathbf{S}_{k+1}\mathbf{D}_{k+1}^{\top}, \tag{3.36}$$

$$\mathbf{D}_{k+1} = \mathbf{C}_{k+1}^{(x,y)}\mathbf{S}_{k+1}^{-1}, \tag{3.37}$$

where $\mathbf{D}_{k+1}$ denotes the filter gain. The joint distribution of $\mathbf{x}_{k+1}$ and $\mathbf{z}_{k+1}$ given $\mathbf{y}_{1:k+1}$ is as follows:

$$p(\mathbf{x}_{k+1}, \mathbf{z}_{k+1}|\mathbf{y}_{1:k+1}) = p(\mathbf{x}_{k+1}|\mathbf{z}_{k+1}, \mathbf{y}_{1:k+1})\ p(\mathbf{z}_{k+1}|\mathbf{y}_{1:k+1}). \tag{3.38}$$

The algebraic state filtering distribution of the instant $k+1$ is hence given by:

$$p(\mathbf{z}_{k+1}|\mathbf{y}_{1:k+1}) = \frac{p(\mathbf{x}_{k+1}, \mathbf{z}_{k+1}|\mathbf{y}_{1:k+1})}{p(\mathbf{x}_{k+1}|\mathbf{z}_{k+1}, \mathbf{y}_{1:k+1})}. \tag{3.39}$$

The generated Gaussian approximation to equation (3.38) may be written as an unscented transform-based approximation as

$$\begin{pmatrix} \mathbf{x}_{k+1} \\ \mathbf{z}_{k+1} \end{pmatrix} |\mathbf{y}_{1:k+1} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{\mu}_{k+1}^{(x)} \\ \boldsymbol{\mu}_{k+1}^{(z)} \end{pmatrix}, \begin{pmatrix} \mathbf{P}_{k+1}^{(x)} & \mathbf{C}_{k+1}^{(x,z)} \\ \mathbf{C}_{k+1}^{(x,z)\top} & \mathbf{P}_{k+1}^{(z)} \end{pmatrix} \right), \tag{3.40}$$

so that marginalising over $\mathbf{x}_{k+1}$ gives a Gaussian approximation for the algebraic state filtering distribution of the instant $k+1$:

$$\mathbf{z}_{k+1}|\mathbf{y}_{1:k+1} \sim \mathcal{N}(\boldsymbol{\mu}_{k+1}^{(z)}, \mathbf{P}_{k+1}^{(z)}). \tag{3.41}$$

In summary, the developed descriptor filtering algorithm is characterised by novel

aspects that include the inclusion of both differential and algebraic state distributions in the computation of the state and measurement predictions (see Algorithm 3.3, steps 1(a) and 2(a), respectively). Additionally, instead of obtaining the predicted and filtered algebraic state estimates by directly solving the algebraic equation (2.14b), the unscented transform is used for the computation of the mean and covariance of the algebraic state estimates after the differential states undergo the nonlinear transformation $\mathbf{g}$ (see Algorithm 3.3, steps 1(d)-(f) and 2(e)-(g)).

A prior distribution with mean $\boldsymbol{\mu}_0^{(x)}$ and covariance $\mathbf{P}_0^{(x)}$ is used to initialise the UKF computations. Equations (3.56)-(3.58) are used to get $\boldsymbol{\mu}_0^{(z)}$, $\mathbf{P}_0^{(z)}$ and $\mathbf{C}_0^{(x,z)}$. A single step of the UKF is summarised in Algorithm 3.3 (the procedure is executed for each time step $k = 0, 1, 2, \ldots$).

Algorithm 3.3 (*UKF algorithm for nonlinear descriptor systems with stochastic algebraic equations*)**.**

1. **Prediction step:**

(a) The sigma points of $\mathbf{X}_k$ are first formed:

$$\begin{aligned}
\tilde{\boldsymbol{\mathcal{X}}}_{k,0} &= \tilde{\boldsymbol{\mu}}_k, \\
\tilde{\boldsymbol{\mathcal{X}}}_{k,i} &= \tilde{\boldsymbol{\mu}}_k + (\sqrt{(n+\lambda')\tilde{\mathbf{P}}_k})_i, \quad i = 1, \ldots, n, \\
\tilde{\boldsymbol{\mathcal{X}}}_{k,i} &= \tilde{\boldsymbol{\mu}}_k - (\sqrt{(n+\lambda')\tilde{\mathbf{P}}_k})_i, \quad i = n+1, \ldots, 2n,
\end{aligned} \tag{3.42}$$

where

$$\tilde{\boldsymbol{\mu}}_k = \begin{pmatrix} \boldsymbol{\mu}_k^{(x)} \\ \boldsymbol{\mu}_k^{(z)} \end{pmatrix}, \tilde{\mathbf{P}}_k = \begin{pmatrix} \mathbf{P}_k^{(x)} & \mathbf{C}_k^{(x,z)} \\ \mathbf{C}_k^{(x,z)\top} & \mathbf{P}_k^{(z)} \end{pmatrix}. \tag{3.43}$$

The vector $(\sqrt{(n+\lambda')\tilde{\mathbf{P}}_k})_i$ denotes the ith column of the matrix square root and $n = n_x + n_z$ is the total number of states, where n_x and n_z are the number of differential and algebraic states, respectively. The scaling parameter λ' is defined

by equation (2.45) but replacing n_x by n and setting $\kappa = 3 - n$.

(b) The sigma points are propagated through the dynamic model to yield

$$\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} = \mathbf{f}(\tilde{\boldsymbol{\mathcal{X}}}^{(x)}_{k,i}, \tilde{\boldsymbol{\mathcal{X}}}^{(z)}_{k,i}), \quad i = 0, \ldots, 2n, \tag{3.44}$$

where $\tilde{\boldsymbol{\mathcal{X}}}^{(x)}_{k,i}$ represents the first n_x components of $\tilde{\boldsymbol{\mathcal{X}}}_{k,i}$ and $\tilde{\boldsymbol{\mathcal{X}}}^{(z)}_{k,i}$ represents the last n_z components. It is assumed that if a numerical solver is used, the latter can accept the sigma points $\tilde{\boldsymbol{\mathcal{X}}}_{k,i}$ as its arguments.

(c) The predicted differential state mean $\boldsymbol{\mu}^{(x)-}_{k+1}$ and its predicted covariance $\mathbf{P}^{(x)-}_{k+1}$ are computed as follows:

$$\boldsymbol{\mu}^{(x)-}_{k+1} = \sum_{i=0}^{2n} W_i^{(m)'} \hat{\boldsymbol{\mathcal{X}}}_{k+1,i}, \tag{3.45}$$

$$\mathbf{P}^{(x)-}_{k+1} = \sum_{i=0}^{2n} W_i^{(c)'} (\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \boldsymbol{\mu}^{(x)-}_{k+1})(\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \boldsymbol{\mu}^{(x)-}_{k+1})^\top + \mathbf{Q}_k, \tag{3.46}$$

where the weights $W_i^{(m)'}$ and $W_i^{(c)'}$ are as defined in equation (2.49), but replacing λ and n_x by λ' and n, respectively.

(d) The sigma points $\bar{\boldsymbol{\mathcal{X}}}^{(x)-}_{k+1,i}$ of the predicted differential states are generated using equation (3.42), replacing $\tilde{\boldsymbol{\mu}}_k$, $\tilde{\mathbf{P}}_k$, λ' and n by $\boldsymbol{\mu}^{(x)-}_{k+1}$, $\mathbf{P}^{(x)-}_{k+1}$, λ and n_x, respectively.

(e) The algebraic state sigma points $\bar{\boldsymbol{\mathcal{X}}}^{(z)-}_{k+1,i}$ are computed using the algebraic equation

$$\mathbf{g}(\bar{\boldsymbol{\mathcal{X}}}^{(x)-}_{k+1,i}, \bar{\boldsymbol{\mathcal{X}}}^{(z)-}_{k+1,i}) = \mathbf{0}, \ i = 0, \ldots, 2n_x. \tag{3.47}$$

(f) The algebraic state predicted mean $\boldsymbol{\mu}^{(z)-}_{k+1}$, associated covariance $\mathbf{P}^{(z)-}_{k+1}$ and cross-

covariance $\mathbf{C}_{k+1}^{(x,z)-}$ may be calculated next:

$$\boldsymbol{\mu}_{k+1}^{(z)-} = \sum_{i=0}^{2n_x} W_i^{(m)} \bar{\boldsymbol{\mathcal{X}}}_{k+1,i}^{(z)-}, \tag{3.48}$$

$$\mathbf{P}_{k+1}^{(z)-} = \sum_{i=0}^{2n_x} W_i^{(c)} (\bar{\boldsymbol{\mathcal{X}}}_{k+1,i}^{(z)-} - \boldsymbol{\mu}_{k+1}^{(z)-})(\bar{\boldsymbol{\mathcal{X}}}_{k+1,i}^{(z)-} - \boldsymbol{\mu}_{k+1}^{(z)-})^\top + \mathbf{E}_{k+1}, \tag{3.49}$$

$$\mathbf{C}_{k+1}^{(x,z)-} = \sum_{i=0}^{2n_x} W_i^{(c)} (\bar{\boldsymbol{\mathcal{X}}}_{k+1,i}^{(x)-} - \boldsymbol{\mu}_{k+1}^{(x)-})(\bar{\boldsymbol{\mathcal{X}}}_{k+1,i}^{(z)-} - \boldsymbol{\mu}_{k+1}^{(z)-})^\top, \tag{3.50}$$

The noise $\mathbf{e}_k$ is assumed to enter the algebraic equation (2.14b) additively with respect to $\mathbf{z}_k$.

2. **Update step**:

(a) The sigma points $\tilde{\boldsymbol{\mathcal{X}}}_{k+1,i}^{-}$ of $\mathbf{X}_{k+1}$ are computed using equation (3.42) and replacing $\tilde{\boldsymbol{\mu}}_k$ and $\tilde{\mathbf{P}}_k$ by $\tilde{\boldsymbol{\mu}}_{k+1}^{-}$ and $\tilde{\mathbf{P}}_{k+1}^{-}$, respectively. The mean $\tilde{\boldsymbol{\mu}}_{k+1}^{-}$ and covariance $\tilde{\mathbf{P}}_{k+1}^{-}$ are given by equation (3.33), which has the form $\mathbf{X}_{k+1}|\mathbf{y}_{1:k} \sim \mathcal{N}(\tilde{\boldsymbol{\mu}}_{k+1}^{-}, \tilde{\mathbf{P}}_{k+1}^{-})$.

(b) The sigma points are propagated through the observation model:

$$\hat{\boldsymbol{\mathcal{Y}}}_{k+1,i} = \mathbf{h}(\tilde{\boldsymbol{\mathcal{X}}}_{k+1,i}^{(x)-}, \tilde{\boldsymbol{\mathcal{X}}}_{k+1,i}^{(z)-}),\ i = 0, \ldots, 2n. \tag{3.51}$$

(c) The predicted observation mean $\mathbf{m}_{k+1}$, the predicted measurement covariance $\mathbf{S}_{k+1}$ and the state and measurement cross-covariance matrix $\mathbf{C}_{k+1}^{(x,y)}$ are obtained as follows:

$$\mathbf{m}_{k+1} = \sum_{i=0}^{2n} W_i^{(m)'} \hat{\boldsymbol{\mathcal{Y}}}_{k+1,i}, \tag{3.52}$$

$$\mathbf{S}_{k+1} = \sum_{i=0}^{2n} W_i^{(c)'} (\hat{\boldsymbol{\mathcal{Y}}}_{k+1,i} - \mathbf{m}_{k+1})(\hat{\boldsymbol{\mathcal{Y}}}_{k+1,i} - \mathbf{m}_{k+1})^\top + \mathbf{R}_{k+1}, \tag{3.53}$$

$$\mathbf{C}_{k+1}^{(x,y)} = \sum_{i=0}^{2n} W_i^{(c)'} (\tilde{\boldsymbol{\mathcal{X}}}_{k+1,i}^{(x)-} - \boldsymbol{\mu}_{k+1}^{(x)-})(\hat{\boldsymbol{\mathcal{Y}}}_{k+1,i} - \mathbf{m}_{k+1})^\top. \tag{3.54}$$

(d) The filter gain $\mathbf{D}_{k+1}$, filtered differential state mean $\boldsymbol{\mu}_{k+1}^{(x)}$ and covariance $\mathbf{P}_{k+1}^{(x)}$

are calculated using equations (3.35)-(3.37).

(e) To compute the filtered algebraic state estimates, the differential state sigma points $\check{\boldsymbol{\mathcal{X}}}^{(x)}_{k+1,i}$ are first obtained using equation (3.42) and replacing $\tilde{\boldsymbol{\mu}}_k$, $\tilde{\mathbf{P}}_k$, λ' and n by $\boldsymbol{\mu}^{(x)}_{k+1}$, $\mathbf{P}^{(x)}_{k+1}$, λ and n_x, respectively.

(f) The algebraic state sigma points $\check{\boldsymbol{\mathcal{X}}}^{(z)}_{k+1,i}$ are generated using the algebraic equation

$$\mathbf{g}(\check{\boldsymbol{\mathcal{X}}}^{(x)}_{k+1,i}, \check{\boldsymbol{\mathcal{X}}}^{(z)}_{k+1,i}) = \mathbf{0}, \quad i = 0, \ldots, 2n_x. \tag{3.55}$$

(g) The filtered algebraic state mean $\boldsymbol{\mu}^{(z)}_{k+1}$, covariance $\mathbf{P}^{(z)}_{k+1}$ and cross-covariance $\mathbf{C}^{(x,z)}_{k+1}$ are computed:

$$\boldsymbol{\mu}^{(z)}_{k+1} = \sum_{i=0}^{2n_x} W_i^{(m)} \check{\boldsymbol{\mathcal{X}}}^{(z)}_{k+1,i}, \tag{3.56}$$

$$\mathbf{P}^{(z)}_{k+1} = \sum_{i=0}^{2n_x} W_i^{(c)} (\check{\boldsymbol{\mathcal{X}}}^{(z)}_{k+1,i} - \boldsymbol{\mu}^{(z)}_{k+1})(\check{\boldsymbol{\mathcal{X}}}^{(z)}_{k+1,i} - \boldsymbol{\mu}^{(z)}_{k+1})^\top + \mathbf{E}_{k+1}, \tag{3.57}$$

$$\mathbf{C}^{(x,z)}_{k+1} = \sum_{i=0}^{2n_x} W_i^{(c)} (\check{\boldsymbol{\mathcal{X}}}^{(x)}_{k+1,i} - \boldsymbol{\mu}^{(x)}_{k+1})(\check{\boldsymbol{\mathcal{X}}}^{(z)}_{k+1,i} - \boldsymbol{\mu}^{(z)}_{k+1})^\top. \tag{3.58}$$

3.2.3 Unscented Rauch-Tung-Striebel smoothing for nonlinear DAEs with stochastic algebraic equations

An approximation to the optimal smoothing solution is now derived for nonlinear descriptor models using the unscented transform, where all states are assumed to be corrupted by zero-mean white Gaussian noise. The smoothing solution is obtained using a separate backward smoothing pass that recursively corrects the forward filtering results.

Since in the algebraic equation the differential states undergo a nonlinear transformation that yields the algebraic states, the proposed algorithm employs the unscented

transformation to propagate the mean and covariance of the smoothed differential state estimates through the nonlinear function $\mathbf{g}$. This provides the smoothed algebraic state estimates, as can be seen in equations (3.79)-(3.82). Furthermore, for the steps involving the propagation of sigma points through the dynamic model, the mean and covariance of both differential and algebraic state estimates are used, as shown in equation (3.64).

The mean and covariance of the filtering distributions, given by

$$p(\mathbf{x}_k|\mathbf{y}_{1:k}) \approx \mathcal{N}(\boldsymbol{\mu}_k^{(x)}, \mathbf{P}_k^{(x)}), \tag{3.59}$$

$$p(\mathbf{z}_k|\mathbf{y}_{1:k}) \approx \mathcal{N}(\boldsymbol{\mu}_k^{(z)}, \mathbf{P}_k^{(z)}), \tag{3.60}$$

shall be assumed computed by the filtering algorithm developed in Section 3.2.2 or an alternative method. A derivation for the RTS smoother that determines the smoothed state estimates of nonlinear DAE systems is given below.

We first consider the joint distribution of $\mathbf{x}_k$, $\mathbf{z}_k$ and $\mathbf{x}_{k+1}$ given $\mathbf{y}_{1:k}$ that may be written as

$$p(\mathbf{x}_k, \mathbf{z}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k}) = p(\mathbf{z}_k|\mathbf{x}_k, \mathbf{x}_{k+1}, \mathbf{y}_{1:k})\ p(\mathbf{x}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k}). \tag{3.61}$$

Re-arranging this joint distribution as

$$p(\mathbf{x}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k}) = \frac{p(\mathbf{x}_k, \mathbf{z}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k})}{p(\mathbf{z}_k|\mathbf{x}_k, \mathbf{x}_{k+1}, \mathbf{y}_{1:k})}, \tag{3.62}$$

we compute a Gaussian approximation using the unscented transform, as follows:

$$\begin{pmatrix} \mathbf{x}_k \\ \mathbf{x}_{k+1} \end{pmatrix} |\mathbf{y}_{1:k} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{\mu}_k^{(x)} \\ \boldsymbol{\mu}_{k+1}^{(x)-} \end{pmatrix}, \begin{pmatrix} \mathbf{P}_k^{(x)} & \mathbf{C}_{k+1}^{(x,x)-} \\ \mathbf{C}_{k+1}^{(x,x)-\top} & \mathbf{P}_{k+1}^{(x)-} \end{pmatrix} \right). \tag{3.63}$$

This can be achieved by using the unscented transform to evaluate the distribution

$$\mathbf{X}_k|\mathbf{y}_{1:k} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{\mu}_k^{(x)} \\ \boldsymbol{\mu}_k^{(z)} \end{pmatrix}, \begin{pmatrix} \mathbf{P}_k^{(x)} & \mathbf{C}_k^{(x,z)} \\ \mathbf{C}_k^{(x,z)\top} & \mathbf{P}_k^{(z)} \end{pmatrix} \right), \tag{3.64}$$

so we may now calculate the joint distribution of $\mathbf{X}_k$ and $\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{z}_k) + \mathbf{q}_k$ as a Gaussian approximation, again using the unscented transform.

By re-writing equation (3.61) as a conditional distribution, we have that

$$p(\mathbf{x}_k|\mathbf{x}_{k+1}, \mathbf{y}_{1:k}) = \frac{p(\mathbf{x}_k, \mathbf{z}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k})}{p(\mathbf{z}_k|\mathbf{x}_k, \mathbf{x}_{k+1}, \mathbf{y}_{1:k})p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})}, \tag{3.65}$$

but using the Markov property yields

$$p(\mathbf{x}_k|\mathbf{x}_{k+1}, \mathbf{y}_{1:K}) = \frac{p(\mathbf{x}_k, \mathbf{z}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:k})}{p(\mathbf{z}_k|\mathbf{x}_k, \mathbf{x}_{k+1}, \mathbf{y}_{1:k})p(\mathbf{x}_{k+1}|\mathbf{y}_{1:k})}. \tag{3.66}$$

Distribution (3.63) is Gaussian and we may therefore exploit the Gaussian computation rules to obtain the Gaussian conditional distribution of equation (3.66), yielding the following approximation:

$$\mathbf{x}_k|\mathbf{x}_{k+1}, \mathbf{y}_{1:K} \sim \mathcal{N}(\boldsymbol{\mu}_{k+1}^{(x)'}, \mathbf{P}_{k+1}^{(x)'}), \tag{3.67}$$

where

$$\mathbf{D}_k = \mathbf{C}_{k+1}(\mathbf{P}_{k+1}^{(x)-})^{-1}, \tag{3.68}$$

$$\boldsymbol{\mu}_{k+1}^{(x)'} = \boldsymbol{\mu}_k^{(x)} + \mathbf{D}_k(\mathbf{x}_{k+1} - \boldsymbol{\mu}_{k+1}^{(x)-}), \tag{3.69}$$

$$\mathbf{P}_{k+1}^{(x)'} = \mathbf{P}_k^{(x)} - \mathbf{D}_k\mathbf{P}_{k+1}^{(x)-}\mathbf{D}_k^\top. \tag{3.70}$$

The joint distribution of $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$ given $\mathbf{y}_{1:K}$ is given by

$$p(\mathbf{x}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:K}) = \frac{p(\mathbf{x}_k, \mathbf{z}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:K})}{p(\mathbf{z}_k|\mathbf{x}_k, \mathbf{x}_{k+1}, \mathbf{y}_{1:K})}. \tag{3.71}$$

We assume knowledge of the smoothing distribution of the following time step, here taken to be Gaussian and given by

$$p(\mathbf{x}_{k+1}|\mathbf{y}_{1:K}) \approx \mathcal{N}(\mathbf{x}_{k+1}|\boldsymbol{\mu}_{k+1}^{(x)s}, \mathbf{P}_{k+1}^{(x)s}). \tag{3.72}$$

The distribution (3.71) then leads to

$$\begin{pmatrix} \mathbf{x}_k \\ \mathbf{x}_{k+1} \end{pmatrix} |\mathbf{y}_{1:K} \sim \mathcal{N}(\boldsymbol{\mu}_{k+1}^{(x)''}, \mathbf{P}_{k+1}^{(x)''}), \tag{3.73}$$

where

$$\boldsymbol{\mu}_{k+1}^{(x)''} = \begin{pmatrix} \boldsymbol{\mu}_k^{(x)} + \mathbf{D}_k(\boldsymbol{\mu}_{k+1}^{(x)s} - \boldsymbol{\mu}_{k+1}^{(x)-}) \\ \boldsymbol{\mu}_{k+1}^{(x)s} \end{pmatrix}, \tag{3.74}$$

$$\mathbf{P}_{k+1}^{(x)''} = \begin{pmatrix} \mathbf{D}_k\mathbf{P}_{k+1}^{(x)s}\mathbf{D}_k^\top + \mathbf{P}_{k+1}^{(x)'} & \mathbf{D}_k\mathbf{P}_{k+1}^{(x)s} \\ \mathbf{P}_{k+1}^{(x)s}\mathbf{D}_k^\top & \mathbf{P}_{k+1}^{(x)s} \end{pmatrix}. \tag{3.75}$$

The differential state smoothing distribution of the time instant k is obtained as a Gaussian approximation by marginalizing over $\mathbf{x}_{k+1}$:

$$\mathbf{x}_k|\mathbf{y}_{1:K} \sim \mathcal{N}(\boldsymbol{\mu}_k^{(x)s}, \mathbf{P}_k^{(x)s}), \tag{3.76}$$

where

$$\boldsymbol{\mu}_k^{(x)s} = \boldsymbol{\mu}_k^{(x)} + \mathbf{D}_k[\boldsymbol{\mu}_{k+1}^{(x)s} - \boldsymbol{\mu}_{k+1}^{(x)-}], \tag{3.77}$$

$$\mathbf{P}_k^{(x)s} = \mathbf{P}_k^{(x)} + \mathbf{D}_k[\mathbf{P}_{k+1}^{(x)s} - \mathbf{P}_{k+1}^{(x)-}]\mathbf{D}_k^\top. \tag{3.78}$$

The joint distribution of $\mathbf{x}_k$ and $\mathbf{z}_k$ given $\mathbf{y}_{1:K}$ is as follows:

$$p(\mathbf{x}_k, \mathbf{z}_k|\mathbf{y}_{1:K}) = \frac{p(\mathbf{x}_k, \mathbf{z}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:K})}{p(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{z}_k, \mathbf{y}_{1:K})}. \tag{3.79}$$

The algebraic state smoothing distribution of time instant k is computed by marginalizing the joint distribution over $\mathbf{x}_k$:

$$p(\mathbf{z}_k|\mathbf{y}_{1:K}) = \int \frac{p(\mathbf{x}_k, \mathbf{z}_k, \mathbf{x}_{k+1}|\mathbf{y}_{1:K})}{p(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{z}_k, \mathbf{y}_{1:K})} d\mathbf{x}_k. \tag{3.80}$$

The generated Gaussian approximation to equation (3.79) is based on the unscented

transform and can be written as

$$\begin{pmatrix} \mathbf{x}_k \\ \mathbf{z}_k \end{pmatrix} |\mathbf{y}_{1:K} \sim \mathcal{N}\left(\begin{pmatrix} \boldsymbol{\mu}_k^{(x)s} \\ \boldsymbol{\mu}_k^{(z)s} \end{pmatrix}, \begin{pmatrix} \mathbf{P}_k^{(x)s} & \mathbf{C}_k^{(x,z)s} \\ \mathbf{C}_k^{(x,z)s\top} & \mathbf{P}_k^{(z)s} \end{pmatrix} \right), \tag{3.81}$$

so that a marginalisation over $\mathbf{x}_k$ approximates the algebraic state smoothing distribution of instant k by the Gaussian distribution

$$\mathbf{z}_k|\mathbf{y}_{1:K} \sim \mathcal{N}(\boldsymbol{\mu}_k^{(z)s}, \mathbf{P}_k^{(z)s}). \tag{3.82}$$

The following URTSS Algorithm 3.4 can be used to compute the smoothed state estimates for every time instant conditional to all measurements spanning time K.

Algorithm 3.4 (*URTSS algorithm for nonlinear DAEs with stochastic algebraic equations*)**.**

1. The sigma points of $\mathbf{X}_k$ are first formed:

$$\begin{aligned} \tilde{\boldsymbol{\mathcal{X}}}_{k,0} &= \tilde{\boldsymbol{\mu}}_k, \\ \tilde{\boldsymbol{\mathcal{X}}}_{k,i} &= \tilde{\boldsymbol{\mu}}_k + (\sqrt{(n+\lambda')\tilde{\mathbf{P}}_k})_i, \ i = 1, \ldots, n, \\ \tilde{\boldsymbol{\mathcal{X}}}_{k,i} &= \tilde{\boldsymbol{\mu}}_k - (\sqrt{(n+\lambda')\tilde{\mathbf{P}}_k})_i, \ i = n+1, \ldots, 2n, \end{aligned} \tag{3.83}$$

   where

$$\tilde{\boldsymbol{\mu}}_k = \begin{pmatrix} \boldsymbol{\mu}_k^{(x)} \\ \boldsymbol{\mu}_k^{(z)} \end{pmatrix}, \tilde{\mathbf{P}}_k = \begin{pmatrix} \mathbf{P}_k^{(x)} & \mathbf{C}_k^{(x,z)} \\ \mathbf{C}_k^{(x,z)\top} & \mathbf{P}_k^{(z)} \end{pmatrix}. \tag{3.84}$$

2. The sigma points are propagated through $\mathbf{f}(\cdot)$ to obtain

$$\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} = \mathbf{f}(\tilde{\boldsymbol{\mathcal{X}}}_{k,i}^{(x)}, \tilde{\boldsymbol{\mathcal{X}}}_{k,i}^{(z)}), i = 0, \ldots, 2n. \tag{3.85}$$

   It is assumed that if a numerical solver is used, the latter can take the sigma points $\tilde{\boldsymbol{\mathcal{X}}}_{k,i}$ as its arguments.

3. The predicted differential state mean $\boldsymbol{\mu}_{k+1}^{(x)-}$, its predicted covariance $\mathbf{P}_{k+1}^{(x)-}$ and the cross-covariance $\mathbf{C}_{k+1}^{(x,x)-}$ are computed as follows:

$$\boldsymbol{\mu}_{k+1}^{(x)-} = \sum_{i=0}^{2n} W_i^{(m)'} \hat{\boldsymbol{\mathcal{X}}}_{k+1,i}, \tag{3.86}$$

$$\mathbf{P}_{k+1}^{(x)-} = \sum_{i=0}^{2n} W_i^{(c)'} (\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \boldsymbol{\mu}_{k+1}^{(x)-})(\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \boldsymbol{\mu}_{k+1}^{(x)-})^\top + \mathbf{Q}_k, \tag{3.87}$$

$$\mathbf{C}_{k+1}^{(x,x)-} = \sum_{i=0}^{2n} W_i^{(c)'} (\tilde{\boldsymbol{\mathcal{X}}}_{k,i}^{(x)} - \boldsymbol{\mu}_k^{(x)})(\hat{\boldsymbol{\mathcal{X}}}_{k+1,i} - \boldsymbol{\mu}_{k+1}^{(x)-})^\top. \tag{3.88}$$

4. The smoother gain $\mathbf{D}_k$ and associated smoothed mean $\boldsymbol{\mu}_k^{(x)s}$ and covariance $\mathbf{P}_k^{(x)s}$ are calculated using the following equations:

$$\mathbf{D}_k = \mathbf{C}_{k+1}^{(x,x)-} (\mathbf{P}_{k+1}^{(x)-})^{-1}, \tag{3.89}$$

$$\boldsymbol{\mu}_k^{(x)s} = \boldsymbol{\mu}_k^{(x)} + \mathbf{D}_k (\boldsymbol{\mu}_{k+1}^{(x)s} - \boldsymbol{\mu}_{k+1}^{(x)-}), \tag{3.90}$$

$$\mathbf{P}_k^{(x)s} = \mathbf{P}_k^{(x)} + \mathbf{D}_k (\mathbf{P}_{k+1}^{(x)s} - \mathbf{P}_{k+1}^{(x)-}) \mathbf{D}_k^\top. \tag{3.91}$$

5. To determine the smoothed algebraic state estimates, the differential state sigma points $\tilde{\boldsymbol{\mathcal{X}}}_{k,i}^{(x)}$ are computed by performing unscented sampling with mean $\boldsymbol{\mu}_k^{(x)s}$ and covariance $\mathbf{P}_k^{(x)s}$.

6. The algebraic state sigma points $\tilde{\boldsymbol{\mathcal{X}}}_{k,i}^{(z)}$ are obtained using the algebraic equation:

$$\mathbf{g}(\tilde{\boldsymbol{\mathcal{X}}}_{k,i}^{(x)}, \tilde{\boldsymbol{\mathcal{X}}}_{k,i}^{(z)}) = \mathbf{0}. \tag{3.92}$$

7. The smoothed algebraic state mean $\boldsymbol{\mu}_k^{(z)s}$ and its covariance $\mathbf{P}_k^{(z)s}$ are calculated:

$$\boldsymbol{\mu}_k^{(z)s} = \sum_{i=0}^{2n_x} W_i^{(m)} \tilde{\boldsymbol{\mathcal{X}}}_{k,i}^{(z)}, \tag{3.93}$$

$$\mathbf{P}_k^{(z)s} = \sum_{i=0}^{2n_x} W_i^{(c)} (\tilde{\boldsymbol{\mathcal{X}}}_{k,i}^{(z)} - \boldsymbol{\mu}_k^{(z)s})(\tilde{\boldsymbol{\mathcal{X}}}_{k,i}^{(z)} - \boldsymbol{\mu}_k^{(z)s})^\top + \mathbf{E}_k. \tag{3.94}$$

The smoothing and filtering distributions are identical at the last time step K and therefore $\boldsymbol{\mu}_K^{(x)s} = \boldsymbol{\mu}_K^{(x)}$, $\boldsymbol{\mu}_K^{(z)s} = \boldsymbol{\mu}_K^{(z)}$, $\mathbf{P}_K^{(x)s} = \mathbf{P}_K^{(x)}$ and $\mathbf{P}_K^{(z)s} = \mathbf{P}_K^{(z)}$. The smoothed estimates are obtained by running the above computations starting from the filtered estimates of the last time instant and proceeding backwards for $k = K-1, \ldots, 0$.

3.2.4 Illustrative example: a nonlinear index two Hessenberg DAE problem

In this section, we demonstrate the effectiveness of the state estimation method developed here for nonlinear DAE systems with stochastic algebraic equations. For this purpose, we consider the following index 2 nonlinear Hessenberg DAE formulation that was presented as an example in [241, 242, 243]:

$$\frac{dx_1(t)}{dt} = tx_2(t)^2 + z(t) + h_1(t), \tag{3.95}$$

$$\frac{dx_2(t)}{dt} = te^{x_1(t)} + tz(t) + h_2(t), \tag{3.96}$$

$$0 = x_1(t) + tx_2(t) + h_3(t), \tag{3.97}$$

where $h_1(t)$, $h_2(t)$ and $h_3(t)$ are nonlinear functions in t as defined in [243]. Differentiating equation (3.97) with respect to time yields an equivalent index 1 DAE with $x_1(t)$ and $x_2(t)$ as differential state variables and $z(t)$ as an algebraic state variable. The formulation was discretised using a forward Euler scheme. For our purpose of investigating state estimation performance, the differential states were assumed to be corrupted with process noise $\mathbf{q}_k$ and the algebraic equation was assumed to be affected by a stochastic uncertainty represented by process noise e_k. The measurement in this example is $z(t)$ and is corrupted by measurement noise r_k. The noise covariance matrices used for the state estimators are $\mathbf{Q}_k = 10^{-6} \times \mathbf{I}_2$, $E_k = 10^{-6}$ and $R_k = 10^{-3}$, where $\mathbf{I}_2$ is a 2×2 identity matrix. State estimation parameters were set as described in [197], with $\alpha = 10^{-3}$ and $\beta = 2$.

Given this descriptor model, a Monte Carlo approach was adopted and 100 reali-

sations of data were generated. Initial conditions for the simulator were drawn from a zero-mean Gaussian distribution having a variance of 10^{-2}, whilst the initial conditions for the filter were $\boldsymbol{\mu}_0^{(x)} = \mathbf{0}$ and $\mathbf{P}_0^{(x)} = 10^{-2} \times \mathbf{I}_2$. Each realisation comprised 100s of data sampled at 1kHz. Figure 3-4 shows a typical simulation result over the initial period for the system under study while Figure 3-5 shows the filtering and smoothing action in terms of the root mean square error performance.

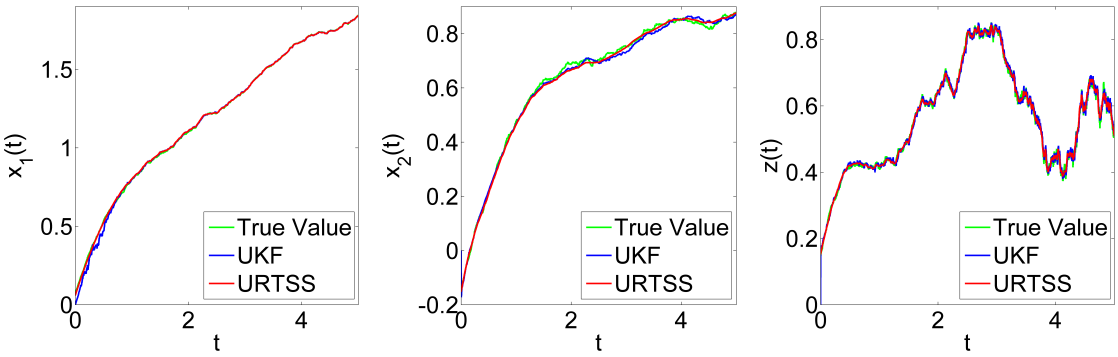


Figure 3-4: True value, UKF estimate and URTSS estimate of (a) $x_1(t)$, (b) $x_2(t)$ and (c) $z(t)$, for the initial period of a single instantiation of the stochastic Hessenberg DAE formulation.

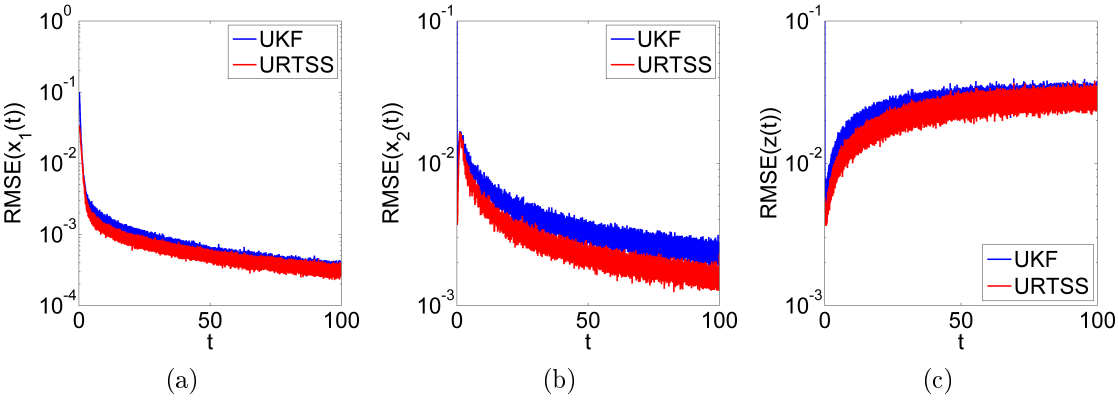


Figure 3-5: State estimation performance. The semi-log RMSE plots are shown for (a) $x_1(t)$, (b) $x_2(t)$ and (c) $z(t)$, following 100 Monte Carlo simulations of the nonlinear DAE model.

The RMSE results for the UKF and URTSS estimates are given in Table 3.3. The smoother estimates exhibit a notable improvement, with a more significant effect observed for the initial time step where URTSS estimates are based on the entire

interval of measurements instead of only prior means. The remaining initial smoother error is then essentially a consequence of the Gaussian approximations employed. When the initial transient is over and the filter may be assumed to have converged, the steady state period ($t \geq 90$) shows the smoother's consistently considerable RMSE enhancement.

Table 3.3: RMSE values of all states for the entire simulation period ($0 \leq t \leq 100$), the initial state ($t = 0$) and the stabilized interval ($90 \leq t \leq 100$), averaged over 100 Monte Carlo runs.

State	UKF	URTSS
RMSE ($\times 10^{-3}$) ($0 \leq t \leq 100$)		
$x_1(t)$	1.36	0.96
$x_2(t)$	4.18	3.03
$z(t)$	25.69	22.99
RMSE ($\times 10^{-3}$) ($t = 0$)		
$x_1(t)$	99.30	33.46
$x_2(t)$	99.49	5.74
$z(t)$	99.46	5.73
RMSE ($\times 10^{-3}$) ($90 \leq t \leq 100$)		
$x_1(t)$	0.34	0.32
$x_2(t)$	2.43	1.63
$z(t)$	30.19	28.89

3.3 Conclusion

In this chapter, the filtering and smoothing problems for temporal nonlinear descriptor systems were addressed for the cases of having differential and stochastic algebraic equations. The unscented transform was employed to determine the algebraic state estimates using the mean and covariance of the differential state estimates. The covariance of the algebraic states was used throughout successive iterations so that all system properties were part of the estimation scheme. The methods also allow the inclusion of both differential and algebraic states in the measurement equation.

Finally, the performance of the proposed filters and smoothers was demonstrated for electrochemical and numerical case studies.

Chapter 4

Estimation of spatio-temporal descriptor systems

Spatio-temporal models naturally describe several complex phenomena, however estimation for the descriptor case remains a challenging task since standard estimation techniques, like Kalman filters, only handle systems that are finite-dimensional and described by ordinary differential equations. Spatio-temporal nonlinear descriptor systems are characterised by a nonlinear partial differential-algebraic equation formulation which is known to describe the physical meaning of the underlying process whilst accommodating spatial heterogeneity. Examples of such physical processes include slender inextensible elastica dynamics in textiles [187], fluid flow [14], electrochemical reactions in fuel cells [184], packed-bed chromatographic adsorption processes [185] and electronic integrated circuit phenomena [186]. This chapter proposes a state estimation method for a class of spatio-temporal nonlinear PDAEs.

Section 4.1 presents a technique which is based on finite-dimensional reduction employing a basis function decomposition method to convert the nonlinear PDAE formulation to a tractable nonlinear DAE form which retains a spatially-continuous representation deemed important for situations involving distributed sensor networks for observing systems such as the ocean [191] and the weather [244]. The remaining problem is to estimate the resulting nonlinear descriptor form, which is tackled using the UKF algorithm derived for nonlinear descriptor systems in Chapter 3. However,

especially for critical real-world spatio-temporal applications, an estimation error performance analysis is needed to quantify the performance limitations and validate imposed performance requirements. A mean square error lower bound for the nonlinear descriptor filtering problem based on the posterior Cramér-Rao inequality is therefore derived in Section 4.2. Section 4.3 provides an application of the proposed framework on a realistic atmospheric boundary layer wind flow estimation problem.

4.1 Generalised state-space representation of nonlinear PDAEs

We consider spatio-temporal nonlinear descriptor systems of equations having the form given by equations (2.11). It is noteworthy that systems characterised by higher-order partial temporal derivatives may be expressed in a first-order partial temporal derivative representation as is usually done for descriptor system representations [28].

A finite-dimensional formulation is obtained by decomposing the spatio-temporal field vector $\boldsymbol{\psi}(\mathbf{s},t)$ using a set of linearly independent basis functions $\mathcal{B} = \{\phi(\mathbf{s}-\boldsymbol{\zeta}_j)\}_{j=1}^{n_\phi}$, where $\boldsymbol{\zeta}_j$ is the centre of the jth basis function and n_ϕ denotes the number of basis functions, each $\phi(\mathbf{s}-\boldsymbol{\zeta}_j) : \Omega \to \mathbb{R}$. Furthermore, let $\boldsymbol{\phi}(\mathbf{s}) = [\phi(\mathbf{s}-\boldsymbol{\zeta}_1)\ \phi(\mathbf{s}-\boldsymbol{\zeta}_2)\ \cdots\ \phi(\mathbf{s}-\boldsymbol{\zeta}_{n_\phi})]^\top$ and $\boldsymbol{\Phi}(\mathbf{s}) = \mathbf{I}_\psi \otimes \boldsymbol{\phi}(\mathbf{s})$, where $\otimes$ denotes the Kronecker product operator of two matrices and $\mathbf{I}_\psi$ is an $n_\psi \times n_\psi$ identity matrix. We may therefore decompose $\boldsymbol{\psi}(\mathbf{s},t)$ by writing

$$\boldsymbol{\psi}(\mathbf{s},t) \approx \boldsymbol{\Phi}^\top(\mathbf{s})\mathbf{X}(t), \tag{4.1}$$

where $\mathbf{X}(t) \in \mathbb{R}^{n_\phi n_\psi}$ and each ith component of $\boldsymbol{\psi}(\mathbf{s},t)$ is a spatio-temporal field given by $\psi_i(\mathbf{s},t) = \sum_{j=1}^{n_\phi} \phi(\mathbf{s}-\boldsymbol{\zeta}_j)X_{i,j}(t) = \boldsymbol{\phi}^\top(\mathbf{s})\mathbf{X}_i(t)$ for $i = 1,2,\ldots,n_\psi$. The vector $\mathbf{X}(t) = (\mathbf{x}^\top(t), \mathbf{z}^\top(t))^\top$ scales the field basis functions $\boldsymbol{\phi}(\mathbf{s})$. Since the observations considered here are discrete in time, the latter is discretised by a first-order Euler method. Define $\mathbf{x}_k := \mathbf{x}(k\Delta_t)$ and $\mathbf{z}_k := \mathbf{z}(k\Delta_t)$ with regular time steps Δ_t and let the subscript $k+1$ denote the index of the future time sample. Considering equation

(2.11a) and substituting the decomposition gives

$$\boldsymbol{\Phi}_b^\top(\mathbf{s})\mathbf{x}_{k+1} = \Delta_t \mathbf{F}\big(D^{n_o}\boldsymbol{\phi}(\mathbf{s}), D^{n_o-1}\boldsymbol{\phi}(\mathbf{s}), \ldots, D\boldsymbol{\phi}(\mathbf{s}), \boldsymbol{\phi}(\mathbf{s}), \mathbf{x}_k, \mathbf{z}_k\big) + \boldsymbol{\Phi}_b^\top(\mathbf{s})\mathbf{x}_k + \boldsymbol{\epsilon}_k(\mathbf{s}), \tag{4.2}$$

where the effects of model uncertainties are approximated by subjecting the spatial field to a disturbance $\boldsymbol{\epsilon}_k(\mathbf{s})$, which represents a zero-mean white Gaussian noise process where $\boldsymbol{\epsilon}_k(\mathbf{s}) \sim \mathcal{N}(\mathbf{0}, \sigma_q^2\mathbf{I})$ and the covariance is defined by

$$\operatorname{cov}(\boldsymbol{\epsilon}_k(\mathbf{s}), \boldsymbol{\epsilon}_{k+\tau}(\boldsymbol{\xi})) = \begin{cases} \sigma_q^2\ \delta(\mathbf{s}-\boldsymbol{\xi}), & \text{if } \tau = 0, \\ 0, & \text{otherwise,} \end{cases} \tag{4.3}$$

for all $\tau \in \mathbb{Z}$, where $\boldsymbol{\xi} \in \Omega$, δ is the Dirac delta function and $\mathcal{N}(\mathbf{0}, \sigma_q^2\mathbf{I})$ denotes the zero-mean normal distribution with covariance $\sigma_q^2\mathbf{I}$. The matrix $\boldsymbol{\Phi}_b = \mathbf{I}_b \otimes \boldsymbol{\phi}(\mathbf{s})$ and $\mathbf{I}_b$ is an $n_b \times n_b$ identity matrix. Pre-multiplying equation (4.2) by $\boldsymbol{\Phi}_b$ and integrating over the spatial domain Ω gives

$$\boldsymbol{\gamma}\mathbf{x}_{k+1} = \Delta_t \int_\Omega \boldsymbol{\Phi}_b(\mathbf{s})\ \mathbf{F}\big(D^{n_o}\boldsymbol{\phi}(\mathbf{s}), D^{n_o-1}\boldsymbol{\phi}(\mathbf{s}), \ldots, D\boldsymbol{\phi}(\mathbf{s}), \boldsymbol{\phi}(\mathbf{s}), \mathbf{x}_k, \mathbf{z}_k\big)d\mathbf{s} + \boldsymbol{\gamma}\mathbf{x}_k + \int_\Omega \boldsymbol{\Phi}_b\boldsymbol{\epsilon}_k(\mathbf{s})d\mathbf{s}, \tag{4.4}$$

where the matrix $\boldsymbol{\gamma} \in \mathbb{R}^{n_b n_\phi \times n_b n_\phi}$ is defined as

$$\boldsymbol{\gamma} \triangleq \int_\Omega \boldsymbol{\Phi}_b(\mathbf{s})\boldsymbol{\Phi}_b^\top(\mathbf{s})d\mathbf{s}. \tag{4.5}$$

Since $\boldsymbol{\gamma}$ is symmetric and positive, then by [245] the matrix is positive definite and therefore invertible. Pre-multiplying equation (4.4) by $\boldsymbol{\gamma}^{-1}$ yields

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{z}_k) + \mathbf{q}_k, \tag{4.6}$$

where

$$\mathbf{f}(\mathbf{x}_k, \mathbf{z}_k) = \Delta_t \boldsymbol{\gamma}^{-1} \int_{\Omega} \boldsymbol{\Phi}_b(\mathbf{s}) \ \mathbf{F}\big(D^{n_o}\boldsymbol{\phi}(\mathbf{s}), D^{n_o-1}\boldsymbol{\phi}(\mathbf{s}), \ldots, D\boldsymbol{\phi}(\mathbf{s}), \boldsymbol{\phi}(\mathbf{s}), \mathbf{x}_k, \mathbf{z}_k\big) d\mathbf{s} + \mathbf{x}_k \tag{4.7}$$

and

$$\mathbf{q}_k = \boldsymbol{\gamma}^{-1} \int_{\Omega} \boldsymbol{\Phi}_b \boldsymbol{\epsilon}_k(\mathbf{s}) d\mathbf{s}. \tag{4.8}$$

By [138], we have that $\mathbf{q}_k$ is a zero-mean, normally distributed white noise process with covariance $\mathbf{Q}_k = \sigma_q^2 \boldsymbol{\gamma}^{-1}$. The terms in $\boldsymbol{\phi}(\mathbf{s})$ and its spatial derivatives get integrated and can either be computed analytically or numerically.

Proceeding similarly for algebraic equation (2.11b), discretising time and substituting the decomposition we obtain

$$\mathbf{0} = \mathbf{g}(\mathbf{x}_k, \mathbf{z}_k) + \mathbf{e}_k, \tag{4.9}$$

where

$$\mathbf{g}(\mathbf{x}_k, \mathbf{z}_k) = \mathbf{G}\big(D^{n_o}\boldsymbol{\phi}(\mathbf{s}), D^{n_o-1}\boldsymbol{\phi}(\mathbf{s}), \ldots, D\boldsymbol{\phi}(\mathbf{s}), \boldsymbol{\phi}(\mathbf{s}), \mathbf{x}_k, \mathbf{z}_k\big), \tag{4.10}$$

and $\mathbf{e}_k$ denotes white Gaussian noise with zero mean and covariance $\mathbf{E}_k = \sigma_e^2 \mathbf{I}$. The discrete-time reduced-order spatio-temporal nonlinear DAE model is then expressed as

$$\mathbf{x}_{k+1} = \mathbf{f}(\mathbf{x}_k, \mathbf{z}_k) + \mathbf{q}_k, \tag{4.11a}$$

$$\mathbf{0} = \mathbf{g}(\mathbf{x}_k, \mathbf{z}_k) + \mathbf{e}_k. \tag{4.11b}$$

4.1.1 The observation process

For general spatio-temporal estimation problems, the required observations are provided by n_s distinct sensors located throughout the spatial domain Ω. Sensors are

generally characterised by a nonlinear function, $\mathbf{H}(\mathbf{s};\mathbf{s}_l) : \mathbb{R}^d \rightarrow \mathbb{R}^{n_s}$. This describes the spatial extent of the sensor over the spatial domain Ω. The general noise-free observation equation is then written as

$$\mathbf{y}(t) = \int_{\Omega} \mathbf{H}(\mathbf{s};\mathbf{s}_l)\boldsymbol{\psi}(\mathbf{s},t)d\mathbf{s}, \tag{4.12}$$

where $\mathbf{s}_l$ is the position of the lth sensor. Substituting the field decomposition yields

$$\mathbf{y}(t) = \int_{\Omega} \mathbf{H}(\mathbf{s};\mathbf{s}_l)\boldsymbol{\Phi}^{\top}(\mathbf{s})d\mathbf{s}\ \mathbf{X}(t). \tag{4.13}$$

Discretising time and approximating the effect of modelling errors stochastically, the output equation is given by

$$\mathbf{y}_k = \mathbf{h}(\mathbf{x}_k,\mathbf{z}_k) + \mathbf{r}_k, \tag{4.14}$$

where

$$\mathbf{h}(\mathbf{x}_k,\mathbf{z}_k) = \int_{\Omega} \mathbf{H}(\mathbf{s};\mathbf{s}_l)\boldsymbol{\Phi}^{\top}(\mathbf{s})d\mathbf{s}\ \mathbf{X}_k, \tag{4.15}$$

and $\mathbf{r}_k$ denotes white Gaussian noise with zero mean and covariance $\mathbf{R}_k = \sigma_r^2\mathbf{I}$. The aim here is to approximate the filtering distribution $p(\mathbf{X}_k|\mathbf{y}_{1:k})$ for time steps k running from 0 to K using a Gaussian distribution.

4.1.2 Basis function selection

The basis function selection approach taken here follows the work reported by Sanner and Slotine [246] and applied to spatio-temporal systems in [139, 149, 183]. The number of sensors and basis functions required to adequately estimate the spatio-temporal field from the observations may be determined by appropriate spectral analysis techniques. A two-dimensional extension of Shannon's sampling theorem suggests how knowledge of the spatial bandwidth of the observed field would provide the minimum number of sensors and basis functions required to cater for all dominant spatial

frequency characteristics within the field [247].

Shannon's sampling theorem requires the field to be spatially band-limited with some cutoff frequency $\boldsymbol{\nu}_c$, also known as the -3 dB point. In the situations considered in this thesis, $\boldsymbol{\nu}_c$ may be determined by performing Fourier analysis of the available data. The distance between adjacent sensors, Δ_y, must then be such that

$$\Delta_y \leq \frac{1}{2\rho_y \boldsymbol{\nu}_c}, \tag{4.16}$$

where $\rho_y \in \mathbb{R} \geq 1$ is an oversampling parameter. This condition must hold in order to prevent spatial aliasing effects during field reconstruction. The reader is referred to [246, 139, 149] for the derivation of the sampling theorem.

In general, the spatial extent of the sensors would effectively attenuate high spatial frequency components, resulting in the state estimation procedure underestimating such frequency content. This motivates the use of higher bandwidth sensors (covering a smaller space), requiring a larger number of sensors in the process. An optimal compromise would therefore be needed. This problem does not feature in our work since for the applications considered, the observations are taken as point measurements over a large spatial domain.

Similar compromise requirements hold for the resolution of the lattice on which basis functions are placed. This must be fine enough to ensure no aliasing occurs throughout the field reconstruction. More specifically, by Shannon's sampling theorem, the distance between basis functions, Δ_ϕ, must satisfy

$$\Delta_\phi \leq \frac{1}{2\rho_\phi \boldsymbol{\nu}_c}, \tag{4.17}$$

where $\rho_\phi \in \mathbb{R} \geq 1$ is an oversampling parameter. The frequency support of the basis functions is also dictated by their width, which may be selected using spectral analysis [246, 139, 149]. In particular, for two-dimensional Gaussian basis functions located

at the origin, we have that

$$\phi(\mathbf{s}) = \exp\left(-\frac{\mathbf{s}^\top \mathbf{s}}{2\sigma_\phi^2}\right). \tag{4.18}$$

Taking the Fourier transform of equation (4.18), we obtain the following [149]:

$$\Upsilon(\boldsymbol{\nu}) = 2\pi\sigma_\phi^2 \exp\left(-2\pi^2\sigma_\phi^2 \boldsymbol{\nu}^\top \boldsymbol{\nu}\right), \tag{4.19}$$

where $\boldsymbol{\nu}$ is the spatial frequency. A 3 dB attenuation at cut-off frequency $\boldsymbol{\nu}_c$ is obtained if the basis function width is chosen using the following equation:

$$\sigma_\phi^2 = \frac{\ln 2}{4\pi^2 \boldsymbol{\nu}_c^\top \boldsymbol{\nu}_c}. \tag{4.20}$$

Such basis function selection ensures the adequate representation of the spatio-temporal field with frequency support $\boldsymbol{\nu}_c$. One problem that often arises, however, is computational cost, which is why an alternative basis function selection method is often necessary. The computational demands of the state estimation process may be reduced by having fewer basis functions, which in turn decreases the spatial bandwidth and yields a higher level of smoothness throughout the approximated field. This design technique requires the user to choose the basis function width first. The spatial cutoff frequency is then calculated by rearranging equation (4.20), as follows:

$$\nu_{c\phi} = \frac{1}{2\pi\sigma_\phi}\sqrt{\ln 2}. \tag{4.21}$$

4.2 State Estimation of nonlinear PDAEs

Nonlinear PDAEs naturally allow for an adequate representation of nonlinear spatio-temporal dynamics which frequently arise in control and engineering applications. They are typically based on the physical description of the underlying processes producing the data and consequently can describe a vast array of natural phenomena. However, their infinite-dimensional nature makes them impractical for standard sig-

nal processing methods and would require a finite-dimensional reduction method to render the model amenable to an estimation scenario.

As was seen in Section 4.1, the basis function decomposition method employed reduces the original infinite-dimensional PDAEs using a projection method where the dynamical system is projected onto a subspace. A finite set of basis functions allows the continuous spatio-temporal field to be represented by a finite-dimensional state vector which facilitates the application of nonlinear state estimation methods, whilst preserving the overall spatio-temporal dynamics. The resulting reduced-order model, given by descriptor equations (4.11) and observation equation (4.14), are readily applicable to Algorithm 3.3 developed in Section 3.2.2.

In a number of critical spatio-temporal applications, however, the estimation process often needs to be analysed for its estimation error performance. The predictive abilities of the derived estimation algorithm might impact decision making in real-world systems and we therefore analyse error performance in terms of the posterior Cramér-Rao bounds that we derive for nonlinear descriptor systems.

4.2.1 Posterior Cramér-Rao bounds for nonlinear descriptor system filtering

Assessing achievable filtering performance is an essential requirement for estimation algorithms. The computation of lower bounds is an important error analysis technique that is indicative of performance limitations and may determine the validity of imposed performance requirements [248, 249]. This subsection therefore derives the posterior Cramér-Rao bounds (PCRBs) for the state estimation of nonlinear descriptor systems.

Let the first and second-order partial derivatives be represented as

$$\delta_{\mathbf{x}_k} = \left[\frac{\partial}{\partial x_k^{(1)}}, \frac{\partial}{\partial x_k^{(2)}}, \ldots, \frac{\partial}{\partial x_k^{(n_b n_\phi)}}\right]^\top, \tag{4.22}$$

$$\delta_{\mathbf{x}_k \mathbf{y}_k} = \delta_{\mathbf{x}_k} \delta_{\mathbf{y}_k}^\top, \tag{4.23}$$

respectively. Assume that the probability density function $p(\mathbf{x}_0)$ is known for initial state $\mathbf{x}_0$. The joint probability density function of $\mathbf{X}_{0:K}$ and $\mathbf{y}_{0:K}$ for time instant k running from 0 to K is given by

$$p(\mathbf{X}_{0:K}, \mathbf{y}_{0:K}) = \left[\prod_{k=0}^{K} p(\mathbf{y}_k|\mathbf{x}_k, \mathbf{z_k})\right] \left[\prod_{k=0}^{K} p(\mathbf{z}_k|\mathbf{x}_k)\right] \left[\prod_{k=0}^{K-1} p(\mathbf{x}_{k+1}|\mathbf{x}_k, \mathbf{z_k})\right] p(\mathbf{x}_0). \quad (4.24)$$

The objective is to obtain the Fisher information submatrices $\mathbf{J}(\mathbf{x}_k)$ and $\mathbf{J}(\mathbf{z}_k)$ for estimating $\mathbf{x}_k$ and $\mathbf{z}_k$, respectively. The matrices $[\mathbf{J}(\mathbf{x}_k)]^{-1}$ and $[\mathbf{J}(\mathbf{z}_k)]^{-1}$ would yield the PCRBs where $\mathbf{P}_k^{(x)} \geq [\mathbf{J}(\mathbf{x}_k)]^{-1}$ and $\mathbf{P}_k^{(z)} \geq [\mathbf{J}(\mathbf{z}_k)]^{-1}$. This gives the lower bound on the mean square error for the estimation of $\mathbf{x}_k$ and $\mathbf{z}_k$, respectively [250, 249]. For brevity, $p(\mathbf{X}_{0:K}, \mathbf{y}_{0:K})$ will be denoted by $p_{0:K}$.

Letting $\tilde{\boldsymbol{\chi}}_{0:K} = (\mathbf{X}_{0:K-1}^\top, \mathbf{x}_K^\top)^\top$ and noting that $\mathbf{X}_{0:K} = (\tilde{\boldsymbol{\chi}}_{0:K}^\top, \mathbf{z}_K^\top)^\top$, we may write

$$\mathbf{J}(\mathbf{X}_{0:K}) = \begin{pmatrix} \mathbf{J}^{(11)} & \mathbf{J}^{(12)} \\ \mathbf{J}^{(21)} & \mathbf{J}^{(22)} \end{pmatrix}, \quad (4.25)$$

where

$$\mathbf{J}^{(11)} = E[-\delta_{\tilde{\boldsymbol{\chi}}_{0:K}\tilde{\boldsymbol{\chi}}_{0:K}} \ln p_{0:K}], \quad (4.26)$$

$$\mathbf{J}^{(21)} = E[-\delta_{\mathbf{z}_K\tilde{\boldsymbol{\chi}}_{0:K}} \ln p_{0:K}] = \mathbf{J}^{(12)\top}, \quad (4.27)$$

$$\mathbf{J}^{(22)} = E[-\delta_{\mathbf{z}_K\mathbf{z}_K} \ln p_{0:K}]. \quad (4.28)$$

Taking the right-lower block matrix of $\mathbf{J}(\mathbf{X}_{0:K})^{-1}$ results in

$$\mathbf{J}(\mathbf{z}_K) = \mathbf{J}^{(22)} - \mathbf{J}^{(21)}{\mathbf{J}^{(11)}}^{-1}\mathbf{J}^{(12)}. \quad (4.29)$$

Since calculating $\mathbf{J}(\mathbf{z}_K)$ requires the computation of the inverse of the large matrices $\mathbf{J}^{(11)}$ or $\mathbf{J}(\mathbf{X}_{0:K})$, we propose a recursive method to improve computational efficiency. The rest of this subsection derives the recursion.

The information matrix $\mathbf{J}(\mathbf{X}_{0:K}, \mathbf{x}_{K+1})$ is obtained by noting that

$$\begin{aligned} p^x_{0:K+1} =& p(\mathbf{X}_{0:K}, \mathbf{x}_{K+1}, \mathbf{y}_{0:K+1}) \\ =& p(\mathbf{x}_{K+1}|\mathbf{X}_K) p(\mathbf{y}_{K+1}|\mathbf{x}_{K+1}) p(\mathbf{X}_{0:K}, \mathbf{y}_{0:K}). \end{aligned} \tag{4.30}$$

Then using equation (4.30), $\mathbf{J}(\mathbf{X}_{0:K}, \mathbf{x}_{K+1})$ may be written in block form as

$$\mathbf{J}(\mathbf{X}_{0:K}, \mathbf{x}_{K+1}) = \begin{pmatrix} \mathbf{J}^{(11)} & \mathbf{J}^{(12)} & \mathbf{0} \\ \mathbf{J}^{(21)} & \mathbf{J}^{(22)'} & \mathbf{J}^{(23)} \\ \mathbf{0} & \mathbf{J}^{(32)} & \mathbf{J}^{(33)} \end{pmatrix}, \tag{4.31}$$

where $\mathbf{0}$ denotes the null matrix of appropriate dimensions and

$$\mathbf{J}^{(22)'} = E[-\delta_{\mathbf{z}_K \mathbf{z}_K} \ln p_{0:K}] + E[-\delta_{\mathbf{z}_K \mathbf{z}_K} \ln p(\mathbf{x}_{K+1}|\mathbf{X}_K)], \tag{4.32}$$

$$\mathbf{J}^{(23)} = E[-\delta_{\mathbf{z}_K \mathbf{x}_{K+1}} \ln p(\mathbf{x}_{K+1}|\mathbf{X}_K)] = \mathbf{J}^{(32)\top}, \tag{4.33}$$

$$\mathbf{J}^{(33)} = E[-\delta_{\mathbf{x}_{K+1} \mathbf{x}_{K+1}} \ln p(\mathbf{x}_{K+1}|\mathbf{X}_K)] + E[-\delta_{\mathbf{x}_{K+1} \mathbf{x}_{K+1}} \ln p(\mathbf{y}_{K+1}|\mathbf{X}_{K+1})]. \tag{4.34}$$

The required information submatrix $J(\mathbf{x}_{k+1})$ may be obtained by calculating the inverse of the right-lower $(n_b n_\phi \times n_b n_\phi)$ submatrix of $\mathbf{J}(\mathbf{X}_{0:K}, \mathbf{x}_{K+1})^{-1}$, as follows:

$$\mathbf{J}(\mathbf{x}_{K+1}) = \mathbf{J}^{(33)} - \begin{pmatrix} \mathbf{0} & \mathbf{J}^{(32)} \end{pmatrix} \begin{pmatrix} \mathbf{J}^{(11)} & \mathbf{J}^{(12)} \\ \mathbf{J}^{(21)} & \mathbf{J}^{(22)'} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{0} \\ \mathbf{J}^{(23)} \end{pmatrix}, \tag{4.35}$$

which may be shown to yield

$$\mathbf{J}(\mathbf{x}_{K+1}) = \mathbf{J}^{(33)} - \mathbf{J}^{(32)} (\mathbf{J}(\mathbf{z}_K) + \mathbf{J}^{(22)''})^{-1} \mathbf{J}^{(23)}, \tag{4.36}$$

where $\mathbf{J}^{(22)''} = E[-\delta_{\mathbf{z}_K \mathbf{z}_K} \ln p(\mathbf{x}_{K+1}|\mathbf{X}_K)]$.

The recursive computation for $\mathbf{J}(\mathbf{z}_{K+1})$ is determined by writing the information

matrix $\mathbf{J}(\mathbf{X}_{0:K+1}) = \mathbf{J}(\mathbf{X}_{0:K}, \mathbf{x}_{K+1}, \mathbf{z}_{K+1})$ in block form as

$$\mathbf{J}(\mathbf{X}_{0:K+1}) = \begin{pmatrix} \mathbf{J}^{(11)} & \mathbf{J}^{(12)} & \mathbf{0} & \mathbf{0} \\ \mathbf{J}^{(21)} & \mathbf{J}^{(22)'} & \mathbf{J}^{(23)} & \mathbf{0} \\ \mathbf{0} & \mathbf{J}^{(32)} & \mathbf{J}^{(33)'} & \mathbf{J}^{(34)} \\ \mathbf{0} & \mathbf{0} & \mathbf{J}^{(43)} & \mathbf{J}^{(44)} \end{pmatrix}, \tag{4.37}$$

where

$$\begin{aligned} \mathbf{J}^{(33)'} &= E[-\delta_{\mathbf{x}_{K+1}\mathbf{x}_{K+1}} \ln p(\mathbf{x}_{K+1}|\mathbf{X}_K)] + E[-\delta_{\mathbf{x}_{K+1}\mathbf{x}_{K+1}} \ln p(\mathbf{y}_{K+1}|\mathbf{X}_{K+1})] \\ &\quad + E[-\delta_{\mathbf{x}_{K+1}\mathbf{x}_{K+1}} \ln p(\mathbf{z}_{K+1}|\mathbf{x}_{K+1})], \qquad (4.38) \\ \mathbf{J}^{(43)} &= E[-\delta_{\mathbf{z}_{K+1}\mathbf{x}_{K+1}} \ln p(\mathbf{y}_{K+1}|\mathbf{X}_{K+1})] + E[-\delta_{\mathbf{z}_{K+1}\mathbf{x}_{K+1}} \ln p(\mathbf{z}_{K+1}|\mathbf{x}_{K+1})] \\ &= \mathbf{J}^{(34)^\top}, \qquad (4.39) \\ \mathbf{J}^{(44)} &= E[-\delta_{\mathbf{z}_{K+1}\mathbf{z}_{K+1}} \ln p(\mathbf{y}_{K+1}|\mathbf{X}_{K+1})] + E[-\delta_{\mathbf{z}_{K+1}\mathbf{z}_{K+1}} \ln p(\mathbf{z}_{K+1}|\mathbf{x}_{K+1})]. \qquad (4.40) \end{aligned}$$

The algebraic state information submatrix $\mathbf{J}(\mathbf{z}_{k+1})$ is hence computed as

$$\mathbf{J}(\mathbf{z}_{K+1}) = \mathbf{J}^{(44)} - \begin{pmatrix} \mathbf{0} & \mathbf{0} & \mathbf{J}^{(43)} \end{pmatrix} \begin{pmatrix} \mathbf{J}^{(11)} & \mathbf{J}^{(12)} & \mathbf{0} \\ \mathbf{J}^{(21)} & \mathbf{J}^{(22)'} & \mathbf{J}^{(23)} \\ \mathbf{0} & \mathbf{J}^{(32)} & \mathbf{J}^{(33)'} \end{pmatrix}^{-1} \begin{pmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{J}^{(34)} \end{pmatrix}, \tag{4.41}$$

which yields

$$\mathbf{J}(\mathbf{z}_{K+1}) = \mathbf{J}^{(44)} - \mathbf{J}^{(43)}(\mathbf{J}(\mathbf{x}_{K+1}) + \mathbf{J}^{(33)''})^{-1}\mathbf{J}^{(34)}, \tag{4.42}$$

where $\mathbf{J}^{(33)''} = E[-\delta_{\mathbf{x}_{K+1}\mathbf{x}_{K+1}} \ln p(\mathbf{z}_{K+1}|\mathbf{x}_{K+1})]$.

The initial information submatrix $\mathbf{J}(\mathbf{x}_0)$ is obtained using the prior probability function $p(\mathbf{x}_0)$:

$$\mathbf{J}(\mathbf{x}_0) = E[-\delta_{\mathbf{x}_0\mathbf{x}_0} \ln p(\mathbf{x}_0)]. \tag{4.43}$$

4.3 Case study: Atmospheric boundary layer flow

We now consider nonlinear PDAEs that may be approximated by the discrete-time nonlinear stochastic descriptor formulation given by equations (4.11), with a specific focus on an important spatio-temporal process: fluid flow. Our motivation is fuelled by the idea of using estimation methods for potential applications in wind gust forecasting and the preview control of wind turbines [3].

Models governed by the Navier-Stokes equations are known to adequately approximate many real-world fluid flows, including atmospheric boundary layer flow [14], and form the basis of this case study. The Navier-Stokes equations are particularly challenging for estimation due to their nonlinear PDAE form and their undetermined nature, the analysis of which is given later on in Chapter 5 where a real-world wind flow estimation application is discussed in detail.

A basis function decomposition approach enables the user to represent the flow field by choosing an appropriate number and placement of basis functions that is independent of the number and placement of observations and which allows a computationally efficient estimation procedure.

With this in mind, the following nonlinear PDAE formulation for fluid flow, expressed by the Navier-Stokes equations for viscous incompressible flow, is considered:

$$\frac{\partial \mathbf{U}(\mathbf{s},t)}{\partial t} = -\nabla P(\mathbf{s},t) - \mathbf{U}(\mathbf{s},t) \cdot \nabla \mathbf{U}(\mathbf{s},t) + \frac{1}{Re}\nabla^2 \mathbf{U}(\mathbf{s},t), \tag{4.44a}$$

$$0 = \nabla \cdot \mathbf{U}(\mathbf{s},t), \tag{4.44b}$$

where $\mathbf{U}(\mathbf{s},t)$ and $P(\mathbf{s},t)$ denote the velocity and pressure fields, respectively, evolving over spatial domain $\Omega \in \mathbb{R}^d$ for d-dimensional flow, with time $t \in \mathbb{R}_+$ and $\mathbf{s} \in \Omega$. The term Re denotes Reynolds number and ∇ denotes the del operator. The boundary condition may be specified as

$$\mathbf{U}(\mathbf{s},t) = \mathbf{U}_\delta(\mathbf{s},t) \text{ with } \mathbf{s} \in \delta\Omega, \tag{4.45}$$

where $\delta\Omega$ is the domain boundary and the initial condition is given by

$$\mathbf{U}(\mathbf{s}, 0) = \mathbf{U}_0. \tag{4.46}$$

By approximating modelling errors stochastically, a tractable, temporally-discrete, spatially-continuous generalised state-space form may be derived by proceeding with basis function decomposition following a pressure Poisson equation description. By letting

$$\mathbf{f}(\mathbf{U}(\mathbf{s}, t), P(\mathbf{s}, t)) = -\nabla P(\mathbf{s}, t) - \mathbf{U}(\mathbf{s}, t) \cdot \nabla \mathbf{U}(\mathbf{s}, t) + \frac{1}{Re} \nabla^2 \mathbf{U}(\mathbf{s}, t), \tag{4.47}$$

$$g(\mathbf{U}(\mathbf{s}, t)) = -\nabla \cdot (\mathbf{U}(\mathbf{s}, t) \cdot \nabla \mathbf{U}(\mathbf{s}, t)), \tag{4.48}$$

the following reduced-order descriptor flow estimation model results:

$$\mathbf{x}_{k+1} = \Delta_t \boldsymbol{\gamma}^{-1} \int_{\Omega} \boldsymbol{\Phi}_d(\mathbf{s}) \mathbf{f}(\boldsymbol{\Phi}_d^{\top}(\mathbf{s}) \mathbf{x}_k, \boldsymbol{\phi}^{\top}(\mathbf{s}) \mathbf{z}_k) d\mathbf{s} + \mathbf{x}_k + \mathbf{q}_k, \tag{4.49a}$$

$$\mathbf{z}_k = \boldsymbol{\eta}^{-1} \int_{\Omega} \boldsymbol{\phi}(\mathbf{s}) g(\boldsymbol{\Phi}_d^{\top}(\mathbf{s}) \mathbf{x}_k) d\mathbf{s} + \tilde{\mathbf{e}}_k, \tag{4.49b}$$

where the velocity and pressure fields are decomposed using a set of Gaussian basis functions described by

$$\phi(\mathbf{s} - \boldsymbol{\zeta}_i) = \exp\left(-\frac{(\mathbf{s} - \boldsymbol{\zeta}_i)^{\top} (\mathbf{s} - \boldsymbol{\zeta}_i)}{2\sigma_{\phi}^2} \right), \tag{4.50}$$

where $\boldsymbol{\zeta}_i$ is the ith basis function centre, $\sigma_{\phi} \in \mathbb{R}$ is a parameter defining the basis function width, $\boldsymbol{\Phi}_b(\mathbf{s}) = \boldsymbol{\Phi}_d(\mathbf{s})$ and $n_b = d$ for the d-dimensional flow problem under consideration. The term $\tilde{\mathbf{e}}_k$ is a zero-mean, normally distributed white noise process with covariance $\tilde{\mathbf{E}}_k = \sigma_{\tilde{e}}^2 \boldsymbol{\eta}^{-1}$ and the matrix $\boldsymbol{\eta} \in \mathbb{R}^{n_{\phi} \times n_{\phi}}$ is defined as

$$\boldsymbol{\eta} \triangleq \int_{\Omega} \boldsymbol{\phi}(\mathbf{s}) (\nabla^2 \boldsymbol{\phi}^{\top}(\mathbf{s})) d\mathbf{s}, \tag{4.51}$$

where ∇^2 denotes the Laplace operator. A full derivation of the reduced-order flow estimation model in strangeness-free nonlinear descriptor form is given in Chapter 5.

4.3.1 Simulation setup

The performance of the estimation framework developed here is evaluated by considering wind flow over a large horizontal plane at a height of 100m above sea level. Rather than estimating the wind field at a single point location, this work has developed a technique to provide the complete picture of the flow field by using sparse flow velocity measurements and a single pressure sensor to provide flow velocity and pressure estimates anywhere throughout the region of interest. This becomes crucial, for instance, when detecting oncoming flow dynamics having length scales smaller than the wind turbine rotor blade diameter, such as wind gusts. Wind field data is generated for estimation purposes using the Simulator for Offshore Wind Farm Applications (SOWFA) [251, 252, 253].

Figure 4-1 shows a typical snapshot of the simulated atmospheric boundary layer together with the output measurement waveforms obtained at a specific location within the 240m square domain. The fluid velocity sensors were set over a 7×7 square grid, with the distance between sensors, $\Delta_y = 40$m. For our purposes of investigating the state estimation framework, we assume free boundary conditions and that the available instrumentation can provide measurements for $\mathbf{U}(\mathbf{s}, t) = (u\ v)^\top$, where $u := u(\mathbf{s}, t)$ and $v := v(\mathbf{s}, t)$ are the horizontal and vertical velocity components, respectively.

4.3.2 Spatial frequency analysis

Basis functions capable of reconstructing the spatio-temporal field were elicited in accordance with the guidelines of Section 4.1.2. A spatial frequency analysis was performed to confirm that the spacing of sensors was adequate to capture the dominant field dynamics. For the synthetic wind data generated, the cutoff frequency of the velocity field, taken to be the -3 dB point, is $f_c^{(u)} \approx f_c^{(v)} \approx 0.01$ cycles/m. From equation (4.16), a maximum sensor separation of 50m is allowed, which confirms that the separation of $\Delta_y = 40$m is sufficient to mitigate any problems related to spatial aliasing of the significant dynamics.

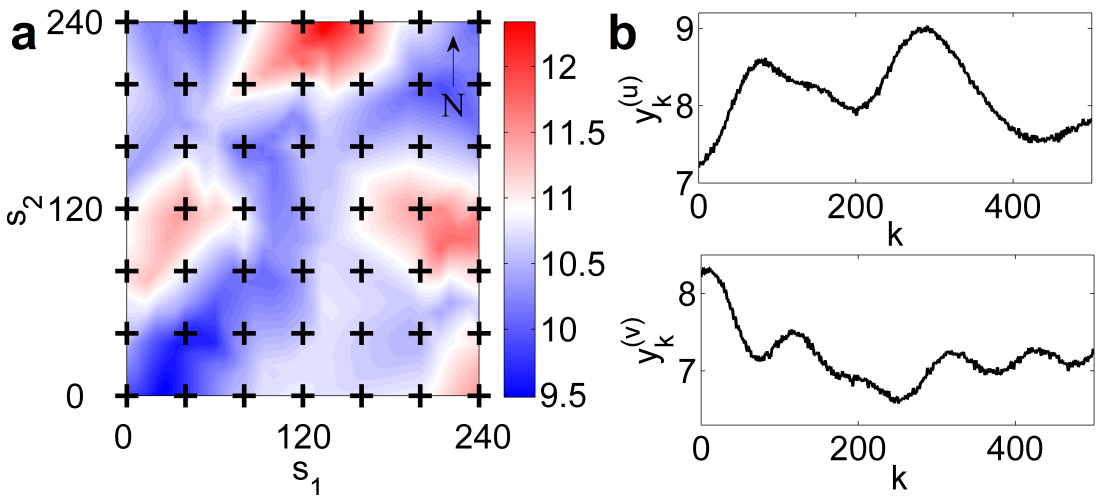


Figure 4-1: (a) A snapshot from a large eddy simulation of the atmospheric boundary layer together with the wind velocity sensor placement within a 240m square domain at a height of 100m above sea level. The contours represent the wind velocity magnitude in ms^{-1} with the prevailing wind direction being from the southwest. (b) Velocity measurement plots as captured by one wind sensor. The terms $y_k^{(u)}$ and $y_k^{(v)}$ denote the sensor readings for the horizontal and vertical velocity components, respectively.

It is equally important to choose a basis function grid layout that adequately approximates the dominant field dynamics. The basis functions can be chosen to represent the full spatial bandwidth (-3 dB point). Alternatively, fewer basis functions can be selected to limit the frequency content of the estimated field in favour of computation time. The velocity and pressure fields cutoff frequency was therefore decreased to $f_{c\phi} = 0.003$ cycles/m. As a consequence of the relatively slow roll-off for Gaussian basis functions, an oversampling parameter of $\rho_\phi = 2.08$ was chosen, giving a total of $n_\phi = 25$ basis functions which were equally spaced over a 5×5 grid laid out in the spatial domain Ω. As a result, $25 \times 3 = 75$ basis functions are required to represent the velocity and pressure fields.

4.3.3 State estimation and results

The state estimation framework implemented follows that proposed in Sections 4.1 and 4.2, with equations (4.11) replaced by equations (4.49). For the situation considered here, the observation equation is linear and given by equation (4.14), where

$\mathbf{h}(\mathbf{x}_k, \mathbf{z}_k) = \mathbf{\Phi}_d^\top(\mathbf{s}_l)\mathbf{x}_k$ and $\mathbf{s}_l$ is the position of the lth sensor. The predicted differential states are hence corrected using the standard Kalman filter update equations instead of executing steps 2(a)-(d) of Algorithm 3.3. The proposed state estimation algorithm was validated using a Monte Carlo approach, where 100 realisations of wind field data were generated. Each realisation estimated $K = 500$ data points (in time) with a sample time of $\Delta_t = 0.1$s. Estimation parameters were set similar to [1], with $\sigma_q^2 = 10^{-1}\text{m}^2\text{s}^{-2}$, $\sigma_{\tilde{e}}^2 = 10^{-4}\text{m}^2\text{s}^{-2}$, $\sigma_r^2 = 10^{-3}\text{m}^2\text{s}^{-2}$, $Re = 10^7$, $\alpha = 1$ and $\beta = 2$. The state estimator received no prior information about initial states.

The accuracy of the resulting state estimates was determined by comparing the estimated field with the generated field using the mean root mean square error (MRMSE), which is often employed in the literature for state estimation of nonlinear systems (e.g. [254, 255]). The MRMSE is defined at each time instant k as

$$\text{MRMSE}_k = \frac{1}{N}\sum_{j=1}^{N}\sqrt{\frac{1}{O}\sum_{o=1}^{O}(\iota_{k,o}^{(j)})^2}, \quad k = 0, 1, 2, \ldots, K, \tag{4.52}$$

where $\iota_{k,o}^{(j)}$ is the estimation error at spatial location o at the kth sampling instant for the jth simulation run. O denotes the number of spatially discrete points (equally spaced on Ω with a spatial discretisation step of Δ_s) and N is the number of simulation runs. For the 240m square domain Ω under study, $O = 37 \times 37 = 1369$ spatially discrete points are represented.

Table 4.1 summarises the average MRMSE values for the whole simulation period ($0 \leq k \leq 500$), with the corresponding MRMSE plots shown in Figure 4-2. Pressure estimation is noted to be accurate despite being measured only at a single point. This stands as evidence of the ability of the derived reduced-order model and proposed nonlinear DAE estimation algorithm to identify dynamic pressure and estimate the whole pressure field.

As a result of basis function decomposition, we attribute part of the estimation error to the loss of high spatial frequency components, as discussed in Section 4.3.2. Figure 4-3 illustrates this band-limiting effect by giving a visual comparison at a single

time instant of the generated fields, estimated fields and corresponding error between both fields. The remaining errors include the Gaussian approximations employed, the model uncertainty and the unmodelled dynamics with respect to the full atmospheric boundary layer flow model [14].

The posterior Cramér-Rao bounds on the estimation error were calculated as proposed in Section 4.2.1. Figure 4-4 shows the estimation variance and PCRB plots for all spatio-temporal fields at a particular space point during one simulation run. A similar pattern was observed for other spatial locations.

Table 4.1: Average MRMSE values for the whole simulation period ($0 \leq k \leq 500$) for 100 simulation runs.

Quantity (units)	Value Range [min, max]	RMSE	RMSE(% of range)
u (ms^{-1})	[3.9251, 9.7919]	0.3332	5.6794%
v (ms^{-1})	[4.6222, 9.6176]	0.3239	6.4840%
P (Pa)	[-27.3478, 29.8723]	4.1704	7.2883%

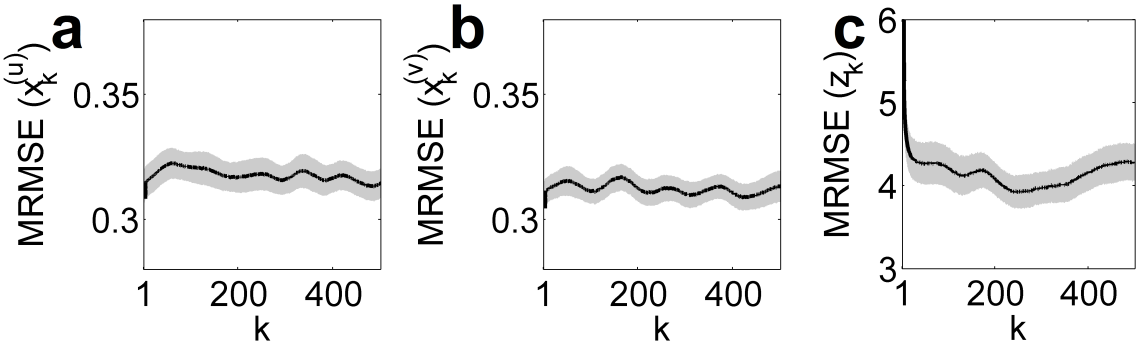


Figure 4-2: Error in the field state estimation. The mean RMSE (solid line) and 95% confidence interval (grey region) plots are shown for (a) horizontal velocity u (in ms^{-1}), (b) vertical velocity v (in ms^{-1}) and (c) pressure P (in Pa), for 100 simulation runs.

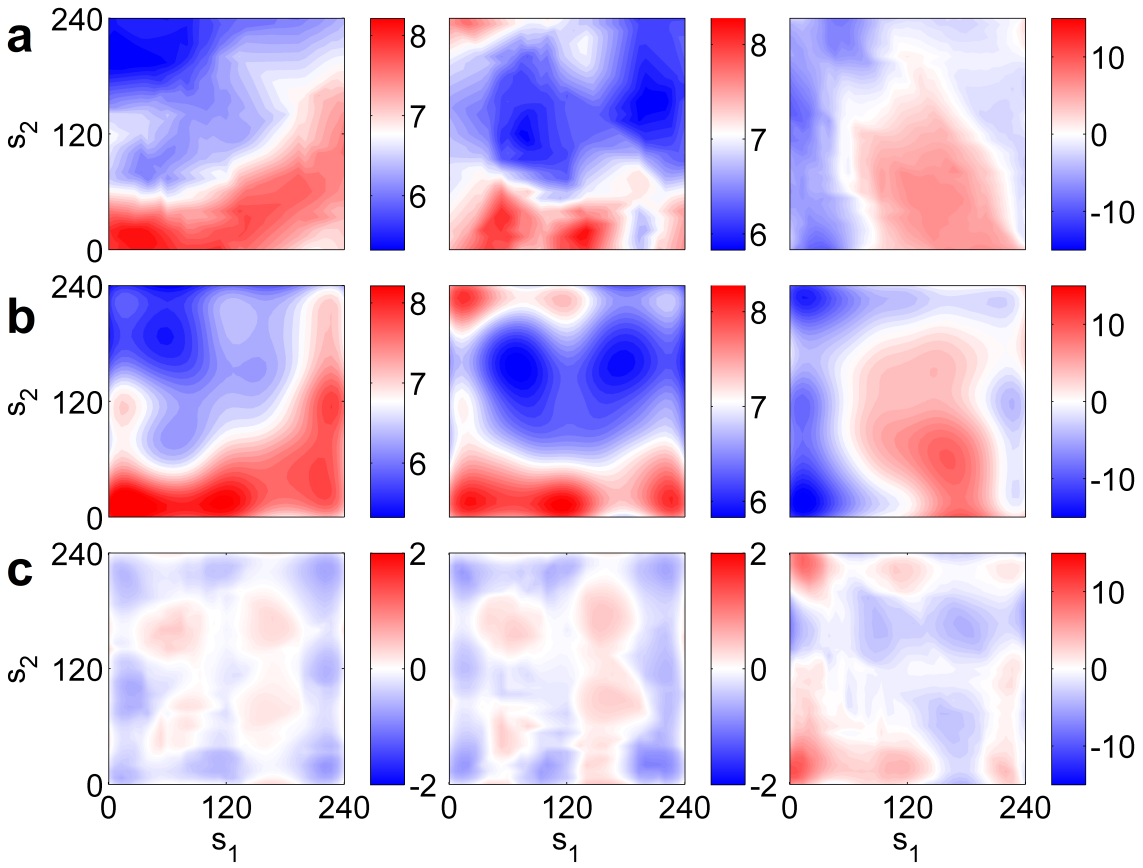


Figure 4-3: The field state estimation is visualised here by showing a single time instant of (a) the generated wind data, (b) the estimated flow fields and (c) the corresponding error between the generated and estimated fields. This is shown for the following fluid flow quantities: horizontal velocity u (in ms^{-1}), vertical velocity v (in ms^{-1}) and pressure P (in Pa) (left to right, respectively).

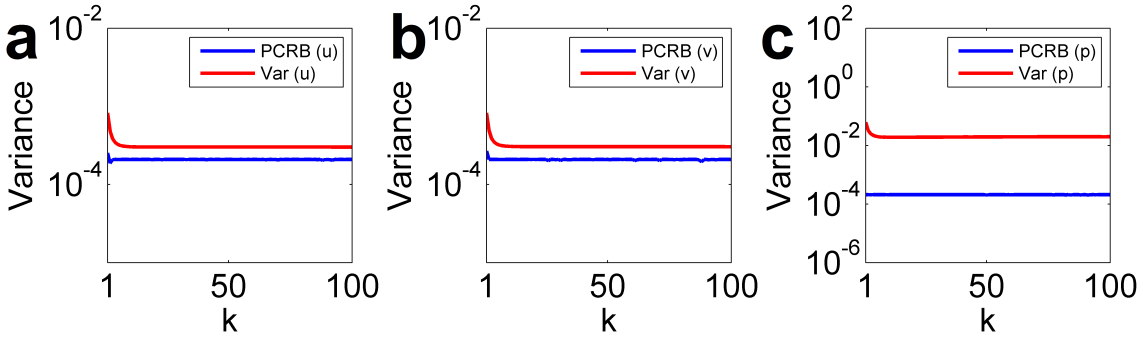


Figure 4-4: Estimation variance and PCRB plots for (a) horizontal velocity u, (b) vertical velocity v and (c) pressure P, plotted on a log scale for a single space point throughout one simulation run.

4.4 Conclusion

The nonlinear PDAE estimation problem has so far not been addressed in the signal processing literature. This chapter developed a novel state estimation scheme for a class of spatio-temporal nonlinear infinite-dimensional PDAEs. By adopting a finite-dimensional reduction approach based on basis function decomposition, a nonlinear PDAE is converted to a nonlinear DAE form for which a new unscented transform-based filtering algorithm is used and associated posterior Cramér-Rao bounds are derived. The performance and viability of the proposed estimation framework is demonstrated for a realistic atmospheric boundary layer wind flow estimation problem where results show consistent accurate estimates of unmeasured flow velocity and pressure.

Chapter 5

Wind field estimation from LIDAR measurements

The ability to harness fluid flow represents a burgeoning source of renewable energy. However, taking wind as an example, fluid flow is also the main disturbance in the control system of wind turbines. This has sparked recent interest in maximising energy production and mitigating structural loads using the preview control of wind turbines and forecasting of wind gusts [3]. Although sampling an oncoming wind field has become possible with recent advances in fluid flow measurement, we are still left with the compelling question of how best to use such limited sparse flow measurements to predict wind gusts and to incorporate such knowledge within a preview control strategy [7, 256, 257]. Furthermore, these measurements are only line-of-sight velocity measurements resulting in what several authors refer to as the Cyclops' dilemma [6, 258], which is the problem encountered by a Cyclops for which three-dimensional vision is impossible due to the availability of only one eye. This requires extra information to resolve wind magnitude and direction. Range weighting is another imperfection influencing LIDAR measurements, where a spatial filter is effectively applied along the laser beam, resulting in wind speeds at locations other than the focal distance to affect the measured value [6]. Overcoming such sources of error require further information that may be obtained from a physical model that captures the spatio-temporal dynamics of the wind. This will inevitably rely upon the

accuracy of wind models employed and calls for wind velocity estimation tools that predict wind turbine gusts using limited spatio-temporal wind velocity measurements [9, 4, 10], thereby mitigating the possible blade damage due to severe wind gusts if the blade pitch is altered in a timely manner [6, 7, 11, 12, 13]. This would of course link measurements to regions of flow which are not directly observed.

Much of the design and control problems treated in the existing wind energy literature take common assumptions that include the steady and uniform flow across the rotor plane [259]. To account for the unsteady nature of fluid flow, the next improvement employs Taylor's frozen turbulence hypothesis [260] that assumes an unchanged spatial structure of wind flow as this advects with a mean velocity [8, 261]. In this work, we seek to provide a more complete picture of the oncoming wind field by estimating wind velocity and pressure over the horizontal plane spanned by the LIDAR beams, rather than estimating wind flow at only single point locations. This becomes a critical requirement, for instance, in the detection of oncoming flow dynamics having length scales smaller than the wind turbine rotor blade diameter, such as wind gusts. The pressure field description is retained since this becomes important for several fluid flow types and situations [262, 263, 264].

To conquer the fact that only limited spatial measurements are available, we propose a model-based estimation framework that presents a reasonable approximation where the field is approximated up to a particular bandwidth. The aforementioned requirements call for a dynamic model that must be amenable to state estimation. We therefore consider models governed by the Navier-Stokes equations which are known to be a good approximation for many real-world fluid flows, including atmospheric boundary layer flow [14], and form the basis of this work. However, the Navier-Stokes equations are in partial differential-algebraic equation form, which presents difficulties for estimation purposes. In addition to being infinite dimensional, the DAE formulation means that it is distinguished from constrained ordinary differential equation systems. In DAE systems, the evolution of algebraic states is not described by differential equations and follow an evolution that is entirely governed by the evolution of differential states, such that all algebraic constraints are satisfied (e.g. pressure in an

incompressible flow) [30]. This key difference means that we will make use of the estimation framework developed for spatio-temporal descriptor systems in Chapter 4. A further important consideration is that DAE systems are generally described by their differentiation index [265]. However, this concept cannot be applied to the Navier-Stokes equations since pressure is only determined up to an additive constant and the equations are hence of an undetermined nature, which Kunkel and Mehrmann [31] describe in terms of a so-called strangeness index. A formal definition of the strangeness index is provided in Appendix A.

The foregoing is particularly challenging since the majority of established estimation techniques are designed for finite-dimensional systems in ODE form. We note that a simplified wind model has been proposed in [215], however this is derived as the spatial discretisation of the linearised incompressible Navier-Stokes equations. A recent work reported in [1] has derived a simplified deterministic state-space model of atmospheric boundary layer flow but is based on spatial discretisation and excludes pressure. In Section 5.1, we therefore derive a generalised state-space flow model in nonlinear, spatially-continuous form. The key to obtaining a strangeness-free DAE of differentiation index 1 is the reformulation of the Navier-Stokes equations using the pressure Poisson equation (PPE) in conjunction with basis function decomposition. The latter enables the user to represent the flow field by choosing an appropriate number and placement of basis functions, or states, that is independent of the number and placement of observations and which allows a computationally efficient estimation procedure. Section 5.2 presents the estimation framework for the incompressible Navier-Stokes equations by obtaining a reduced-order nonlinear descriptor flow estimation model which may be readily applied to the modified DAE form of the discrete-time UKF algorithm derived in Chapter 3. Sections 5.3 and 5.4 demonstrate estimation performance for wind field data generated from large eddy simulations of the atmospheric boundary layer and real-world LIDAR measurement data obtained from a nacelle-mounted LIDAR unit, respectively.

5.1 A generalised state-space model for incompressible flow

We consider wind flow expressed by the Navier-Stokes equations for viscous incompressible flow [266, 267, 215], as given by equations (4.44) and repeated here for completeness' sake:

$$\frac{\partial \mathbf{U}(\mathbf{s},t)}{\partial t} = -\nabla P(\mathbf{s},t) - \mathbf{U}(\mathbf{s},t) \cdot \nabla \mathbf{U}(\mathbf{s},t) + \frac{1}{Re}\nabla^2 \mathbf{U}(\mathbf{s},t), \tag{5.1a}$$

$$0 = \nabla \cdot \mathbf{U}(\mathbf{s},t), \tag{5.1b}$$

where $\mathbf{U}(\mathbf{s},t)$ and $P(\mathbf{s},t)$ denote the velocity and pressure fields, respectively, evolving over spatial domain $\Omega \in \mathbb{R}^d$ for d-dimensional flow, with time $t \in \mathbb{R}_+$ and $\mathbf{s} \in \Omega$. The term Re denotes Reynolds number, the superscript $^\top$ is the transpose operator and ∇ denotes the del operator. The boundary condition may be specified as

$$\mathbf{U}(\mathbf{s},t) = \mathbf{U}_\delta(\mathbf{s},t) \text{ with } \mathbf{s} \in \delta\Omega, \tag{5.2}$$

where $\delta\Omega$ is the domain boundary and the initial condition is given by

$$\mathbf{U}(\mathbf{s},0) = \mathbf{U}_0. \tag{5.3}$$

A derivation of the Navier-Stokes equations is given in [268]. It is noteworthy that equations (5.1) give no explicit equation for the pressure P, with a pressure term appearing only once as a spatial derivative on the right-hand side of equation (5.1a). In fact, P is known as a Lagrange multiplier which enforces the incompressibility of the velocity field $\mathbf{U}$ such that the algebraic equation (5.1b) is always satisfied. This PDAE system is said to have a higher differentiation index (non-decoupled) since no pressure term exists in the algebraic equation. Furthermore, since pressure is only determined up to an additive constant, the system is undetermined and the concept of the differentiation index [265] cannot be readily applied [269]. Whenever undetermined solution components exist, the differentiation index concept for general

nonlinear DAEs is described in terms of a so-called strangeness index [31]. Consequently, following the strangeness index definition given in Appendix A, the above system is, after spatial discretisation, characterised by a unity strangeness index. To render this formulation amenable to an estimation framework, we propose a reformulation into a strangeness-free, spatially-continuous generalised state-space form by employing basis function decomposition following a pressure Poisson equation description. The following preliminaries present the necessary matrices and associated invertibility properties required to obtain the main result of this section. Note that all integral operations presented here are performed element-wise.

Let n_ϕ denote the number of basis functions, each $\phi(\mathbf{s}-\boldsymbol{\zeta}_i) : \Omega \to \mathbb{R}$, where $\boldsymbol{\zeta}_i$ is the centre of the ith basis function. Further, let $\boldsymbol{\phi}(\mathbf{s}) = [\phi(\mathbf{s}-\boldsymbol{\zeta}_1)\ \phi(\mathbf{s}-\boldsymbol{\zeta}_2)\ \cdots\ \phi(\mathbf{s}-\boldsymbol{\zeta}_{n_\phi})]^\top$ and $\boldsymbol{\Phi}_d(\mathbf{s}) = \mathbf{I}_d \otimes \boldsymbol{\phi}(\mathbf{s})$, where $\otimes$ denotes the Kronecker product operator of two matrices and $\mathbf{I}_d$ is a $d \times d$ identity matrix. The matrix $\boldsymbol{\gamma} \in \mathbb{R}^{dn_\phi \times dn_\phi}$ shall be defined as

$$\boldsymbol{\gamma} \triangleq \int_\Omega \boldsymbol{\Phi}_d(\mathbf{s})\boldsymbol{\Phi}_d^\top(\mathbf{s})d\mathbf{s}, \tag{5.4}$$

where $n_b = d$. Also, define the matrix $\boldsymbol{\eta} \in \mathbb{R}^{n_\phi \times n_\phi}$ as

$$\boldsymbol{\eta} \triangleq \int_\Omega \boldsymbol{\phi}(\mathbf{s})(\nabla^2\boldsymbol{\phi}^\top(\mathbf{s}))d\mathbf{s}, \tag{5.5}$$

where ∇^2 denotes the Laplace operator.

We note that since $\boldsymbol{\gamma}$ is symmetric and positive, then by [245] the matrix is positive definite and hence invertible. To show invertibility for the matrix $\boldsymbol{\eta}$, we consider a regular grid of equally-spaced identical basis functions. This yields a Toeplitz-block Toeplitz matrix structure for $\boldsymbol{\eta}$ and its invertibility may be determined by [270, 271].

In order to obtain a strangeness-free nonlinear descriptor system of differentiation index 1, let

$$\mathbf{f}(\mathbf{U}(\mathbf{s},t), P(\mathbf{s},t)) = \frac{\partial \mathbf{U}(\mathbf{s},t)}{\partial t}. \tag{5.6}$$

We decompose the spatio-temporal fields $\mathbf{U}(\mathbf{s},t)$ and $P(\mathbf{s},t)$ using an infinite set of linearly independent basis functions $\mathcal{Z} = \{\phi(\mathbf{s}-\boldsymbol{\zeta}_i)\}_{i=1}^{\infty}$, as follows:

$$\mathbf{U}(\mathbf{s},t) = \boldsymbol{\Phi}_d^\top(\mathbf{s})\mathbf{x}(t), \tag{5.7}$$

$$P(\mathbf{s},t) = \sum_{i=1}^{\infty} \phi(\mathbf{s}-\boldsymbol{\zeta}_i) z_i(t) = \boldsymbol{\phi}^\top(\mathbf{s})\mathbf{z}(t), \tag{5.8}$$

where $\mathbf{x}(t) \in \mathbb{R}^{dn_\phi}$ and $\mathbf{z}(t) \in \mathbb{R}^{n_\phi}$ are the velocity and pressure state vectors, respectively, that scale the field basis functions $\boldsymbol{\phi}(\mathbf{s})$. Substituting the field decomposition in the momentum equation (5.1a) we obtain

$$\boldsymbol{\Phi}_d^\top(\mathbf{s})\frac{\partial \mathbf{x}(t)}{\partial t} = \mathbf{f}(\boldsymbol{\Phi}_d^\top(\mathbf{s})\mathbf{x}(t), \boldsymbol{\phi}^\top(\mathbf{s})\mathbf{z}(t)). \tag{5.9}$$

Pre-multiplying equation (5.9) by $\boldsymbol{\Phi}_d(\mathbf{s})$ and integrating over the spatial domain Ω yields

$$\int_\Omega \boldsymbol{\Phi}_d(\mathbf{s})\boldsymbol{\Phi}_d^\top(\mathbf{s})d\mathbf{s}\ \frac{\partial \mathbf{x}(t)}{\partial t} = \int_\Omega \boldsymbol{\Phi}_d(\mathbf{s})\mathbf{f}(\boldsymbol{\Phi}_d^\top(\mathbf{s})\mathbf{x}(t), \boldsymbol{\phi}^\top(\mathbf{s})\mathbf{z}(t))d\mathbf{s}. \tag{5.10}$$

Substituting $\boldsymbol{\gamma}$ into equation (5.10) and pre-multiplying by $\boldsymbol{\gamma}^{-1}$ we may write

$$\frac{\partial \mathbf{x}(t)}{\partial t} = \boldsymbol{\gamma}^{-1}\int_\Omega \boldsymbol{\Phi}_d(\mathbf{s})\mathbf{f}(\boldsymbol{\Phi}_d^\top(\mathbf{s})\mathbf{x}(t), \boldsymbol{\phi}^\top(\mathbf{s})\mathbf{z}(t))d\mathbf{s}. \tag{5.11}$$

To derive the algebraic equation for $\mathbf{z}(t)$, we first obtain the Poisson equation for pressure by taking the divergence of equation (5.1a) and using the divergence-free condition of equation (5.1b) [272]:

$$\nabla^2 P(\mathbf{s},t) = -\nabla \cdot (\mathbf{U}(\mathbf{s},t) \cdot \nabla \mathbf{U}(\mathbf{s},t)). \tag{5.12}$$

Let

$$g(\mathbf{U}(\mathbf{s},t)) = -\nabla \cdot (\mathbf{U}(\mathbf{s},t) \cdot \nabla \mathbf{U}(\mathbf{s},t)). \tag{5.13}$$

Then substituting the field decomposition into equation (5.12) yields

$$(\nabla^2 \boldsymbol{\phi}^\top(\mathbf{s}))\mathbf{z}(t) = g(\boldsymbol{\Phi}_d^\top(\mathbf{s})\mathbf{x}(t)). \tag{5.14}$$

Pre-multiplying equation (5.14) by $\boldsymbol{\phi}(\mathbf{s})$ and integrating over the spatial domain Ω gives

$$\int_\Omega \boldsymbol{\phi}(\mathbf{s})(\nabla^2 \boldsymbol{\phi}^\top(\mathbf{s}))d\mathbf{s}\ \mathbf{z}(t) = \int_\Omega \boldsymbol{\phi}(\mathbf{s}) g(\boldsymbol{\Phi}_d^\top(\mathbf{s})\mathbf{x}(t))d\mathbf{s}. \tag{5.15}$$

Substituting $\boldsymbol{\eta}$ into equation (5.15) and pre-multiplying by $\boldsymbol{\eta}^{-1}$ we have that

$$\mathbf{z}(t) = \boldsymbol{\eta}^{-1} \int_\Omega \boldsymbol{\phi}(\mathbf{s}) g(\boldsymbol{\Phi}_d^\top(\mathbf{s})\mathbf{x}(t))d\mathbf{s}. \tag{5.16}$$

The final form of the nonlinear descriptor model may therefore be written as

$$\frac{\partial \mathbf{x}(t)}{\partial t} = \boldsymbol{\gamma}^{-1} \int_\Omega \boldsymbol{\Phi}_d(\mathbf{s}) \mathbf{f}(\boldsymbol{\Phi}_d^\top(\mathbf{s})\mathbf{x}(t), \boldsymbol{\phi}^\top(\mathbf{s})\mathbf{z}(t))d\mathbf{s}, \tag{5.17a}$$

$$\mathbf{z}(t) = \boldsymbol{\eta}^{-1} \int_\Omega \boldsymbol{\phi}(\mathbf{s}) g(\boldsymbol{\Phi}_d^\top(\mathbf{s})\mathbf{x}(t))d\mathbf{s}, \tag{5.17b}$$

where both $\mathbf{x}(t)$ and $\mathbf{z}(t)$ are exactly determined, making the formulation strangeness-free. The concept of the differentiation index may now be applied to such system of equations and the formulation is said to have a differentiation index of 1. Note that this representation is still infinite dimensional. In practice, however, a finite-dimensional approximation would be required for estimation purposes, as will be described in the next section.

5.2 Estimation of the incompressible Navier-Stokes equations

In order to render our flow model apt for estimation purposes, we decompose the velocity and pressure fields using a set of Gaussian basis functions given by equation

(4.50). The basis function width and placement are computed using the spatial frequency analysis technique discussed in Section 4.1.2 which considers spatial frequency cutoff as the design parameter describing basis function width and placement. For the flow estimation framework to be computationally tractable and efficient, a reduced-order flow model is derived together with the associated observation process and static pressure estimation schemes.

5.2.1 A reduced-order nonlinear descriptor flow estimation model

Since the observations are discrete in time, we discretise time using a first-order Euler method. This simple approach introduces a discretisation error which is analysed in [273]. However, this error is only incurred in the forward prediction step so it is then corrected throughout the update step of the estimation algorithm. We may write the flow estimation model by first defining $\mathbf{x}_k := \mathbf{x}(k\Delta_t)$ and $\mathbf{z}_k := \mathbf{z}(k\Delta_t)$ with regular time steps Δ_t and denoting the index of the future time sample by the subscript $k+1$. Substituting the decomposition into equation (5.1a) yields

$$\boldsymbol{\Phi}_d^\top(\mathbf{s})\mathbf{x}_{k+1} = \Delta_t \mathbf{f}(\boldsymbol{\Phi}_d^\top(\mathbf{s})\mathbf{x}_k, \boldsymbol{\phi}^\top(\mathbf{s})\mathbf{z}_k) + \boldsymbol{\Phi}_d^\top(\mathbf{s})\mathbf{x}_k + \boldsymbol{\epsilon}_k(\mathbf{s}), \tag{5.18}$$

where to approximate the effects of model reduction and model uncertainties, the spatial field is subjected to a disturbance $\boldsymbol{\epsilon}_k(\mathbf{s})$, which represents a normally distributed zero-mean white noise process where $\boldsymbol{\epsilon}_k(\mathbf{s}) \sim \mathcal{N}(\mathbf{0}, \sigma_q^2\mathbf{I})$ and the covariance is defined by

$$\operatorname{cov}(\boldsymbol{\epsilon}_k(\mathbf{s}), \boldsymbol{\epsilon}_{k+\tau}(\boldsymbol{\xi})) = \begin{cases} \sigma_q^2\ \delta(\mathbf{s}-\boldsymbol{\xi}), & \text{if } \tau = 0, \\ 0, & \text{otherwise,} \end{cases} \tag{5.19}$$

for all $\tau \in \mathbb{Z}$, where $\boldsymbol{\xi} \in \Omega$, $\mathbf{I}$ is an identity matrix of appropriate dimensions, δ is the Dirac delta function and $\mathcal{N}(\mathbf{0}, \sigma_q^2\mathbf{I})$ denotes the zero-mean normal distribution with covariance $\sigma_q^2\mathbf{I}$. Pre-multiplying equation (5.18) by $\boldsymbol{\Phi}_d$, integrating over the spatial

domain Ω and re-arranging gives

$$\mathbf{x}_{k+1} = \mathbf{f}^{(\mathbf{x})}(\mathbf{x}_k, \mathbf{z}_k) + \mathbf{q}_k, \tag{5.20}$$

where

$$\mathbf{f}^{(\mathbf{x})}(\mathbf{x}_k, \mathbf{z}_k) = \Delta_t \boldsymbol{\gamma}^{-1} \int_{\Omega} \boldsymbol{\Phi}_d(\mathbf{s})\ \mathbf{f}(\boldsymbol{\Phi}_d^{\top}(\mathbf{s})\mathbf{x}_k, \boldsymbol{\phi}^{\top}(\mathbf{s})\mathbf{z}_k) d\mathbf{s} + \mathbf{x}_k \tag{5.21}$$

and

$$\mathbf{q}_k = \boldsymbol{\gamma}^{-1} \int_{\Omega} \boldsymbol{\Phi}_d \boldsymbol{\epsilon}_k(\mathbf{s}) d\mathbf{s}. \tag{5.22}$$

By [138], we have that $\mathbf{q}_k$ is a zero-mean, normally distributed white noise process with covariance $\mathbf{Q}_k = \sigma_q^2 \boldsymbol{\gamma}^{-1}$. We note that the terms in $\boldsymbol{\phi}(\mathbf{s})$ and its spatial derivatives get integrated and can either be determined analytically or numerically.

Proceeding similarly for algebraic equation (5.1b), we get

$$\mathbf{z}_k = \mathbf{g}^{(\mathbf{z})}(\mathbf{x}_k) + \tilde{\mathbf{e}}_k, \tag{5.23}$$

where

$$\mathbf{g}^{(\mathbf{z})}(\mathbf{x}_k) = \boldsymbol{\eta}^{-1} \int_{\Omega} \boldsymbol{\phi}(\mathbf{s}) g(\boldsymbol{\Phi}_d^{\top}(\mathbf{s})\mathbf{x}_k) d\mathbf{s}, \tag{5.24}$$

where $\tilde{\mathbf{e}}_k$ is a zero-mean, normally distributed white noise process with covariance $\tilde{\mathbf{E}}_k = \sigma_{\tilde{e}}^2 \boldsymbol{\eta}^{-1}$. The discrete-time reduced-order spatio-temporal nonlinear descriptor model is then given by

$$\mathbf{x}_{k+1} = \mathbf{f}^{(\mathbf{x})}(\mathbf{x}_k, \mathbf{z}_k) + \mathbf{q}_k, \tag{5.25a}$$

$$\mathbf{z}_k = \mathbf{g}^{(\mathbf{z})}(\mathbf{x}_k) + \tilde{\mathbf{e}}_k. \tag{5.25b}$$

We note that the states $\mathbf{x}_k$ and $\mathbf{z}_k$ can be decoupled from the integral computation, thus allowing offline computation and hence more efficient estimation. Following the

dimensional analysis carried out in [1], the wind field dynamics over the horizontal plane are largely independent of changes in height and we will therefore consider only two-dimensional flow ($d = 2$). Taking the first n_ϕ components of $\mathbf{x}_{k+1}$ as an example and denoting this vector by $\mathbf{x}_{k+1}^{(u)}$ (i.e. the state vector describing the horizontal component of velocity) we have that

$$\begin{aligned}\mathbf{x}_{k+1}^{(u)} =&\Delta_t \boldsymbol{\gamma}_0^{-1}\left[\frac{1}{Re}\int_\Omega \boldsymbol{\phi}\frac{\partial^2 \boldsymbol{\phi}^\top}{\partial {s_1}^2}\mathbf{x}_k^{(u)}d\mathbf{s} + \frac{1}{Re}\int_\Omega \boldsymbol{\phi}\frac{\partial^2 \boldsymbol{\phi}^\top}{\partial {s_2}^2}\mathbf{x}_k^{(u)}d\mathbf{s} - \int_\Omega \boldsymbol{\phi}\frac{\partial \boldsymbol{\phi}^\top}{\partial s_1}\mathbf{z}_k d\mathbf{s}\right.\\ &\left. - \int_\Omega \boldsymbol{\phi}\boldsymbol{\phi}^\top \mathbf{x}_k^{(u)}\frac{\partial \boldsymbol{\phi}^\top}{\partial s_1}\mathbf{x}_k^{(u)}d\mathbf{s} - \int_\Omega \boldsymbol{\phi}\boldsymbol{\phi}^\top \mathbf{x}_k^{(v)}\frac{\partial \boldsymbol{\phi}^\top}{\partial s_2}\mathbf{x}_k^{(u)}d\mathbf{s}\right] + \mathbf{x}_k^{(u)} + \mathbf{q}_k^{(u)}, \qquad (5.26)\end{aligned}$$

where $\boldsymbol{\phi} = \boldsymbol{\phi}(\mathbf{s})$, $\mathbf{q}_k^{(u)}$ denotes the first n_ϕ components of $\mathbf{q}_k$, $\mathbf{x}_k^{(v)}$ denotes the last n_ϕ components of $\mathbf{x}_k$ and $\boldsymbol{\gamma}_0$ is given by equation (5.4) with $d = 1$. The states may be decoupled from the integral computation by re-writing equation (5.26) as follows:

$$\begin{aligned}\mathbf{x}_{k+1}^{(u)} =&\Delta_t \boldsymbol{\gamma}_0^{-1}\left[\frac{1}{Re}\int_\Omega \boldsymbol{\phi}\frac{\partial^2 \boldsymbol{\phi}^\top}{\partial {s_1}^2}d\mathbf{s}\ \mathbf{x}_k^{(u)} + \frac{1}{Re}\int_\Omega \boldsymbol{\phi}\frac{\partial^2 \boldsymbol{\phi}^\top}{\partial {s_2}^2}d\mathbf{s}\ \mathbf{x}_k^{(u)}\right.\\ &- \int_\Omega \boldsymbol{\phi}\frac{\partial \boldsymbol{\phi}^\top}{\partial s_1}d\mathbf{s}\ \mathbf{z}_k - \int_\Omega \boldsymbol{\phi}(\boldsymbol{\phi}\otimes\frac{\partial \boldsymbol{\phi}}{\partial s_1})^\top d\mathbf{s}\ (\mathbf{x}_k^{(u)}\otimes\mathbf{x}_k^{(u)})\\ &\left. - \int_\Omega \boldsymbol{\phi}(\boldsymbol{\phi}\otimes\frac{\partial \boldsymbol{\phi}}{\partial s_2})^\top d\mathbf{s}\ (\mathbf{x}_k^{(v)}\otimes\mathbf{x}_k^{(u)})\right] + \mathbf{x}_k^{(u)} + \mathbf{q}_k^{(u)}. \qquad (5.27)\end{aligned}$$

Grouping terms, this equation has the form

$$\mathbf{x}_{k+1}^{(u)} = \mathbf{f}^{(u)}(\mathbf{x}_k, \mathbf{z}_k) + \mathbf{q}_k^{(u)}, \qquad (5.28)$$

where

$$\begin{aligned}
\mathbf{f}^{(u)}(\mathbf{x}_k, \mathbf{z}_k) =& \mathbf{A}_1 \mathbf{x}_k^{(u)} + \mathbf{A}_2 \mathbf{z}_k + \mathbf{A}_3(\mathbf{x}_k^{(u)} \otimes \mathbf{x}_k^{(u)}) + \mathbf{A}_4(\mathbf{x}_k^{(v)} \otimes \mathbf{x}_k^{(u)}), \\
\mathbf{A}_1 =& \frac{1}{Re}\Delta_t \boldsymbol{\gamma}_0^{-1}\left[\int_\Omega \boldsymbol{\phi}\frac{\partial^2 \boldsymbol{\phi}^\top}{\partial {s_1}^2} d\mathbf{s} + \int_\Omega \boldsymbol{\phi}\frac{\partial^2 \boldsymbol{\phi}^\top}{\partial {s_2}^2} d\mathbf{s}\right], \\
\mathbf{A}_2 =& -\Delta_t \boldsymbol{\gamma}_0^{-1}\int_\Omega \boldsymbol{\phi}\frac{\partial \boldsymbol{\phi}^\top}{\partial s_1} d\mathbf{s}, \\
\mathbf{A}_3 =& -\Delta_t \boldsymbol{\gamma}_0^{-1}\int_\Omega \boldsymbol{\phi}(\boldsymbol{\phi} \otimes \frac{\partial \boldsymbol{\phi}}{\partial s_1})^\top d\mathbf{s}, \\
\mathbf{A}_4 =& -\Delta_t \boldsymbol{\gamma}_0^{-1}\int_\Omega \boldsymbol{\phi}(\boldsymbol{\phi} \otimes \frac{\partial \boldsymbol{\phi}}{\partial s_2})^\top d\mathbf{s}.
\end{aligned} \tag{5.29}$$

It is easy to see that the constant matrices $\mathbf{A}_1$, $\mathbf{A}_2$, $\mathbf{A}_3$ and $\mathbf{A}_4$ may be computed offline. By proceeding similarly for any other dimension and component of velocity and pressure, the form of equations (5.25) is obtained.

5.2.2 Static pressure estimation

In the incompressible Navier-Stokes equations (5.1), the total pressure P appears only as a spatial derivative in the momentum equation (5.1a). The static ($P^{(s)}$) and dynamic ($P^{(d)}$) pressure terms that make up $P = P^{(s)} + P^{(d)}$ are characterised by different time constants, with $P^{(s)}$ naturally demonstrating much slower dynamics. This makes P unobservable and consequently, the pressure estimated using equation (5.23) is effectively dynamic pressure. We therefore propose to model static pressure $P^{(s)}$ by a Gaussian random walk model to track its slow variation and take a single pressure measurement $\tilde{y}_k(\mathbf{s}_P) := \tilde{y}(\mathbf{s}_P, k\Delta_t)$ at regular time intervals Δ_t within the spatial domain Ω. We assume that the static pressure is uniform throughout this domain. The full static pressure model is then given by

$$P_{k+1}^{(s)} = P_k^{(s)} + w_k, \tag{5.30}$$

$$\tilde{y}_k(\mathbf{s}_P) - \boldsymbol{\phi}^\top(\mathbf{s}_P)\mathbf{z}_k = P_k^{(s)} + v_k, \tag{5.31}$$

where $\mathbf{s}_P$ is the position of the pressure sensor and $w_k \sim \mathcal{N}(0, \sigma_w^2)$ and $v_k \sim \mathcal{N}(0, \sigma_v^2 + \boldsymbol{\phi}^\top(\mathbf{s}_P)\mathbf{P}_k^{(z)}\boldsymbol{\phi}(\mathbf{s}_P))$ denote additive white Gaussian noise. The term $\boldsymbol{\phi}^\top(\mathbf{s}_P)\mathbf{z}_k$ effectively represents the dynamic pressure estimate so the difference between $\boldsymbol{\phi}^\top(\mathbf{s}_P)\mathbf{z}_k$ and the single pressure measurement $\tilde{y}_k(\mathbf{s}_P)$ is used to estimate the static pressure term using the standard Kalman filter. The covariance term of v_k caters for both pressure sensor noise and dynamic pressure estimation error through σ_v^2 and $\boldsymbol{\phi}^\top(\mathbf{s}_P)\mathbf{P}_k^{(z)}\boldsymbol{\phi}(\mathbf{s}_P)$, respectively.

5.2.3 The observation process

Current LIDAR systems can only detect aerosol speeds in the line-of-sight direction of the laser beam. Measurements are taken at discrete points along the line-of-sight path, so we consider the case where observations are available at regular time intervals Δ_t at n_s points distinctly located throughout the spatial domain $\mathbf{s} \in \Omega$. Define the observation vector as $\mathbf{y}_k(\mathbf{s}_l) := \mathbf{y}(\mathbf{s}_l, k\Delta_t)$. Then the full model observation equation may be written as

$$\mathbf{y}_k(\mathbf{s}_l) = \mathbf{u}_k(\mathbf{s}_l)\sin\theta + \mathbf{v}_k(\mathbf{s}_l)\cos\theta + \mathbf{r}_k(\mathbf{s}_l), \tag{5.32}$$

where $\mathbf{u}_k(\mathbf{s}_l) := \mathbf{u}(\mathbf{s}_l, k\Delta_t)$ and $\mathbf{v}_k(\mathbf{s}_l) := \mathbf{v}(\mathbf{s}_l, k\Delta_t)$ are the horizontal and vertical velocity components, respectively, θ is the LIDAR beam half-angle and $\mathbf{s}_l$ is the lth position. Substituting for field decomposition yields the reduced-order model observation equation given as

$$\mathbf{y}_k(\mathbf{s}_l) = \boldsymbol{\phi}^\top(\mathbf{s}_l)\ (\mathbf{x}_k^{(u)}\sin\theta + \mathbf{x}_k^{(v)}\cos\theta) + \mathbf{r}_k(\mathbf{s}_l). \tag{5.33}$$

5.3 Case study 1: Estimation from large eddy simulation wind field data

Large eddy simulation (LES) wind field data is used to validate the estimator performance. Wind flow over a large horizontal plane is considered and sparse line-of-sight

wind speed measurements and a single pressure measurement are used to yield flow velocity and absolute pressure estimates anywhere in the area of interest.

State estimation is performed using Algorithm 3.3 with the discrete-time reduced-order spatio-temporal nonlinear descriptor model given by equations (5.25) and (5.33). As discussed in [1], the coupling between the wind velocity fields is enhanced by assuming that the instantaneous wind direction is uniform throughout the spatial domain under study. Note that since the observation equation is linear in the states, the predicted differential states are corrected using the standard Kalman filter update equations.

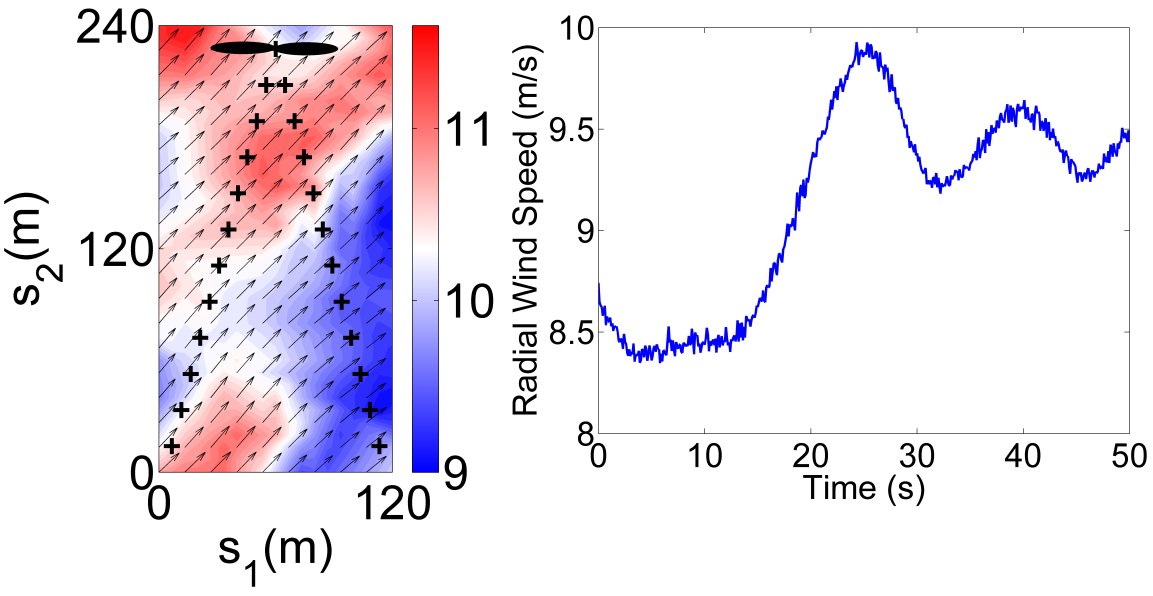
5.3.1 Simulation setup

Realistic wind field data is generated using SOWFA [251, 252, 253], which enables large eddy simulations of the atmospheric boundary layer. The interested reader is referred to [251, 252, 253] for a detailed simulator description.

A typical snapshot of the simulated atmospheric boundary layer is shown in Figure 5-1 together with the output measurement waveform obtained at one of the observation locations within the spatial domain. The LIDAR configuration consists of two beams of 15° half-angle with the line-of-sight wind speed being measured at discrete points along the beams. The LIDAR range considered in this example is 220m and the distance between sample points (black plus signs) is 20m, which is considerably less than the characteristic gust length scales ($\mathcal{O}(10^2)$m) [1]. We shall assume free boundary conditions throughout the estimation process.

5.3.2 Basis function selection and estimation

The basis function selection procedure adopted here follows that described in Section 4.1.2. For the simulated wind data, the cutoff frequency of the velocity field (-3 dB point) is $f_c^{(u)} \approx f_c^{(v)} \approx 0.01$ cycles/m, so that choosing the basis functions for improved computation time, the cutoff frequency for the velocity and pressure fields is set to $f_{c\phi} = 0.003$ cycles/m. An oversampling parameter of $\rho_\phi = 2.08$ was chosen,



(a) (b)

Figure 5-1: (a) A snapshot from a large eddy simulation of the atmospheric boundary layer together with the line-of-sight wind speed observation locations (black plus signs situated over a 15° half-angle LIDAR beam configuration) within a 120m×240m spatial domain at a height of 100m above sea level. The contour represents the wind speed in ms^{-1} with the prevailing wind direction being from the southwest. Note that the rotor blades are shown for clarity, but the turbine dynamics are excluded in this work. (b) Radial wind speed (RWS) measurement plots as captured at one observation location.

requiring $n_\phi = 15$ basis functions which were equally spaced over a 3×5 grid laid out in the spatial domain Ω. Consequently, $15 \times 3 = 45$ basis functions are required to represent the velocity and pressure fields.

A Monte Carlo approach was used to demonstrate the performance of the proposed state estimation algorithm, where 50 realisations of wind field data were generated. Each realisation estimated $K = 500$ data points (in time) with a sample time of $\Delta_t = 0.1\mathrm{s}$. Estimation parameters were set similar to [1], with $\sigma_q^2 = 1\mathrm{m}^2\mathrm{s}^{-2}$, $\sigma_{\tilde{e}}^2 = 10^{-4}\mathrm{m}^2\mathrm{s}^{-2}$, $\sigma_r^2 = 10^{-3}\mathrm{m}^2\mathrm{s}^{-2}$, $\sigma_w^2 = 10^{-5}\mathrm{Pa}^2$, $\sigma_v^2 = 10^{-4}\mathrm{Pa}^2$, $Re = 10^7$, $\alpha = 1$ and $\beta = 2$. No prior information about initial states was passed on to the state estimator and the initial differential state distribution was set to $\mathcal{N}(\mathbf{0}, 100\boldsymbol{\gamma}^{-1})$.

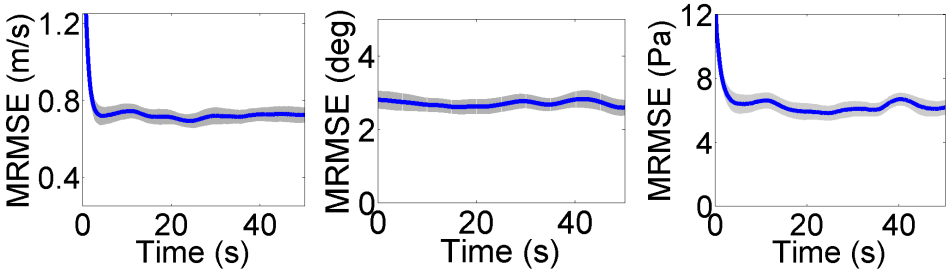
5.3.3 Results and discussion

The mean root mean square error (MRMSE) as defined by equation (4.52) was used to obtain a measure of accuracy for the state estimates by comparing the estimated field with the generated field. For the spatial domain Ω considered, $O = 19 \times 37 = 703$ spatially discrete points are represented.

Table 5.1 summarises the average MRMSE values for the stabilized simulation period ($t \geq 10$s) with Figure 5-2 showing the corresponding MRMSE plots and Figure 5-3 showing a single time instant of the generated fields and estimated fields. Accurate pressure estimation is obtained despite being measured only at a single point in the field. This shows how the derived reduced-order model and proposed nonlinear descriptor estimation algorithm can identify dynamic pressure and correctly estimate the entire pressure field.

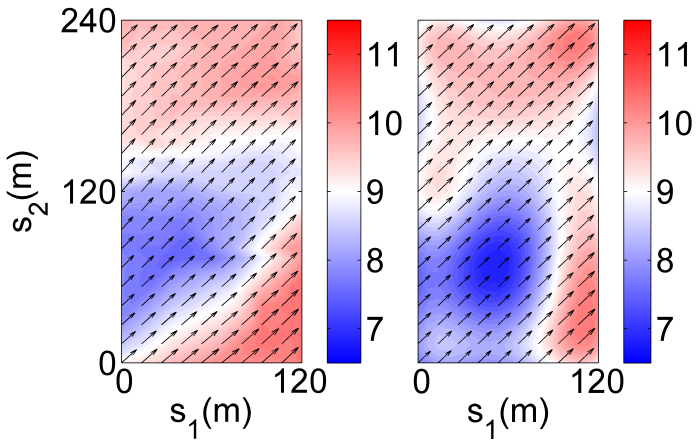
Table 5.1: Average MRMSE values for the stabilized simulation period ($t \geq 10$s) for 50 simulation runs.

Quantity (units)	Value Range [min, max]	RMSE	RMSE(% of range)
Wind speed (ms^{-1})	[6.59, 13.04]	0.72	11.16%
Wind direction (°)	[208.15, 236.66]	2.69	9.44%
Pressure (Pa)	[-27.35, 29.87]	6.17	10.78%

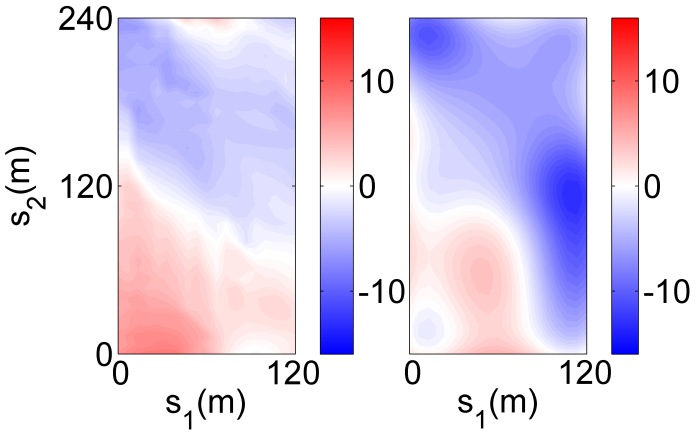


(a) (b) (c)

Figure 5-2: Error in the field state estimation. The mean RMSE (solid line) and 95% confidence interval (shaded area) plots are shown for (a) wind velocity magnitude (m/s), (b) wind direction (°) and (c) pressure (Pa).



(a)



(b)

Figure 5-3: The spatial field estimation is visualised here by showing a single time instant of the generated wind data (left) and estimated flow field (right), shown here for (a) wind velocity magnitude (represented by contours, in m/s) and wind direction (arrows representing direction) and (b) pressure (represented by contours, in Pa).

The contribution to the estimation errors come from mainly three sources: model uncertainty, model reduction/approximation and sparse measurements. The two-dimensional incompressible Navier-Stokes equations are used to approximate the full three-dimensional atmospheric boundary layer flow model [14]. The model reduction scheme employed affects model accuracy and selecting the number of basis functions has to be traded off against the estimation accuracy that can be achieved with limited spatial measurements. Basis function decomposition brings about the loss of high spatial frequency components [149]. The Gaussian approximations made in the UKF-based filtering algorithm also contribute towards the estimation error.

5.4 Case study 2: Estimation from real-world LIDAR measurements

The performance of the proposed estimation framework was further demonstrated for real-world wind speed LIDAR measurements obtained from a five-beam wind turbine-mounted LIDAR unit manufactured by Avent Lidar Technology.

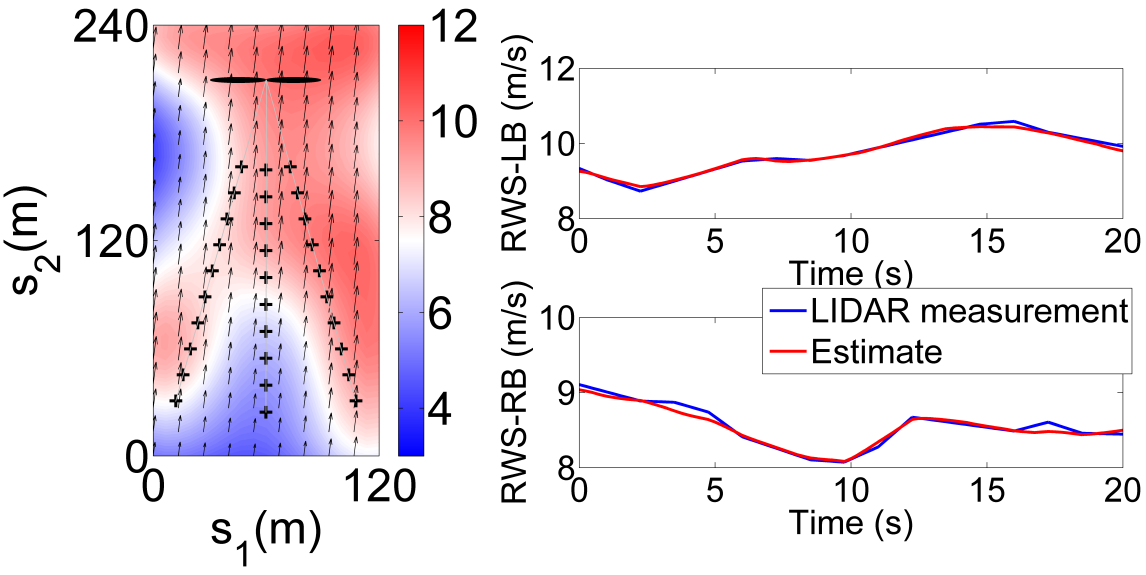
5.4.1 A five-beam wind turbine-mounted LIDAR unit setup

Three of the five LIDAR beams provide measurements on a horizontal plane, with two lateral beams at a 15° half-angle and one central beam. Since the only wind field data available is the radial wind speed (RWS) at these observation locations, the two lateral beams are the ones used for estimation, whilst the central beam is only used for validating estimation performance. The dynamic wind model and estimation scheme developed here allow for wind field reconstruction of the entire region of interest using these LIDAR measurements. A single instance of this reconstruction is given in Figure 5-4(a). The observation points as provided by the LIDAR unit are indicated with black plus signs. The LIDAR range is 185m and the distance between sample points is 15m. Basis function selection and state estimation were performed similarly as in Section 5.3.2, with the time step $\Delta_t = 0.25$s. Numerical tests revealed that the average runtime for the proposed estimation algorithm was 0.083s[1] per single iteration, which comfortably allows for real-time execution.

5.4.2 Results and discussion

A comparison of a typical real-world LIDAR wind speed measurement data set and its estimate is shown in Figure 5-4(b) for the left beam (LB; top plot) and the right beam (RB; bottom plot) at a distance of 80m away from the LIDAR unit. A similar comparison is shown in Figure 5-5 for every observation location located on the central beam in the order of increasing distance from the LIDAR unit (top to bottom plots).

[1]Simulations were carried out on an Intel®Core i5-2450M @ 2.50GHz personal computer with 4GB of RAM.



(a) (b)

Figure 5-4: (a) A single instant from the reconstructed wind field obtained from estimation using the real-world LIDAR wind speed measurements along the two lateral (outer) beams in a 15° half-angle configuration. The wind velocity magnitude is represented by contours (in m/s) and direction arrows represent wind direction. Note that the rotor blades are shown for clarity, but the turbine dynamics are excluded in this work. (b) A typical real-world LIDAR wind speed measurement data set shown in comparison with the estimated radial wind speed (RWS) for the left beam (LB; top plot) and the right beam (RB; bottom plot) at a distance of 80m away from the LIDAR unit.

The accuracy of the state estimates was evaluated by comparing the radial wind speed estimates to the LIDAR wind speed measurements at the central beam, using the MRMSE as defined in equation (4.52), where $O = 10$ is now the number of observed spatial locations (along the central beam) and $N = 30$ is the number of independent LIDAR measurement data sets, each of 20s duration. The MRMSE of the field estimates is shown in Figure 5-6(a). The average MRMSE for the entire duration is 0.8452m/s, which is approximately 12.6% of the wind field RWS range ([5.26 11.95] m/s). This is comparable to the MRMSE value obtained for the synthetic LES data scenario, with a slight increase being potentially a result of more significant uncertainties and measurement imperfections in the real-world scenario, including LIDAR range weighting effects and atmospheric stability.

From Figure 5-5, it is evident that the estimation performance degrades with

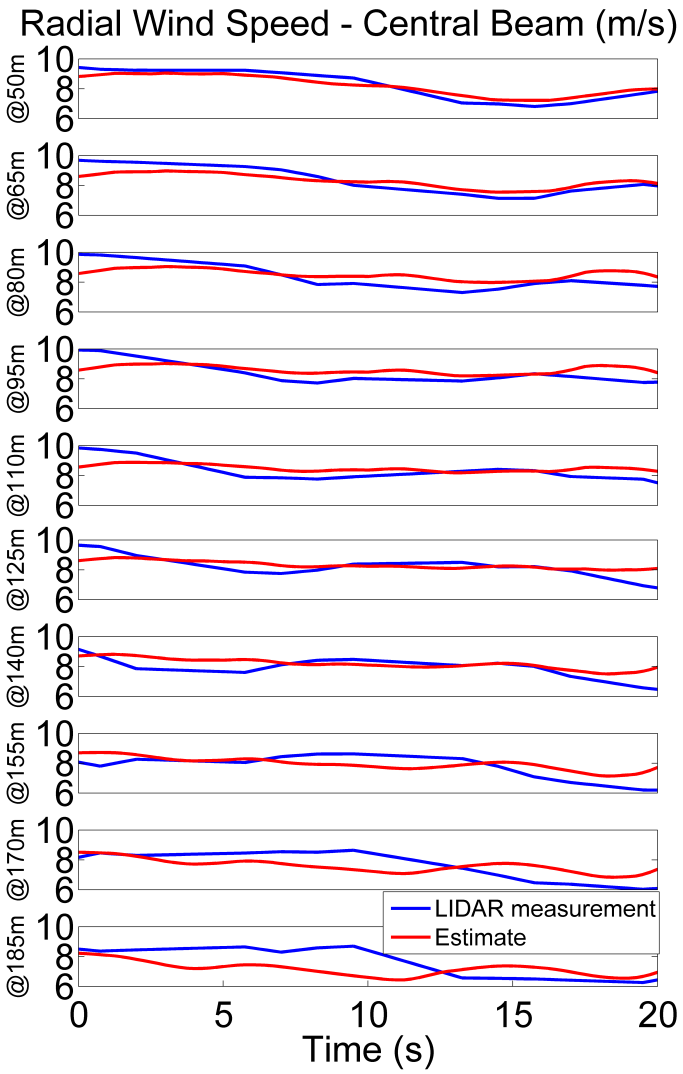
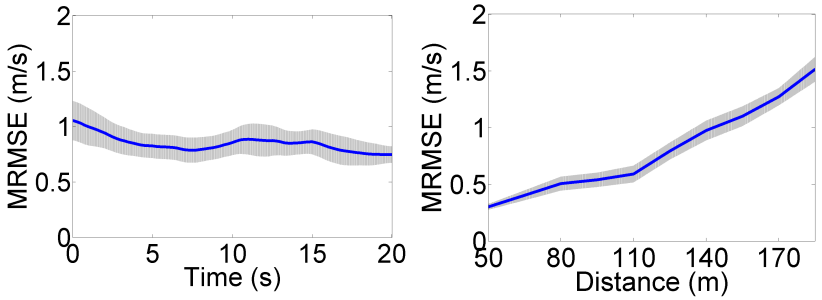


Figure 5-5: A typical comparison of the real-world LIDAR wind speed measurements and the corresponding estimation shown for every observation location along the central beam. Plots are shown in the order of increasing distance from the LIDAR unit, with the top plot showing the estimation at 50m away from the LIDAR equipment.

an increased distance as a result of being further away from the nearest available measurement. Detecting wind gusts coming from the furthest locations might be particularly difficult since some gusts may approach the LIDAR unit without going past any measurement for them to be identifiable. Such estimation performance degradation is further visualised using the MRMSE of the field estimates over space along the central beam, as shown in Figure 5-6(b). The MRMSE is defined at each



(a) (b)

Figure 5-6: Error in the field state estimation at the central beam observation locations. The mean RMSE (solid line) and 95% confidence interval (shaded region) is shown for the radial wind speed (m/s) for 30 independent LIDAR measurement data sets over (a) time and (b) space, along the central beam direction.

spatial location o as

$$\text{MRMSE}_o = \frac{1}{N}\sum_{j=1}^{N}\sqrt{\frac{1}{K+1}\sum_{k=0}^{K}(\iota_{k,o}^{(j)})^2}. \tag{5.34}$$

5.5 Conclusion

This chapter has proposed a spatio-temporal wind flow estimation framework with a reduced-order descriptor fluid flow model and a nonlinear estimator that accurately estimates velocity and in particular pressure given sparse line-of-sight wind speed measurements. By uniquely employing a pressure Poisson equation (PPE) formulation, a spatially-continuous, strangeness-free nonlinear DAE form of the Navier-Stokes equations was obtained. The modified UKF algorithm derived in Chapter 3 was used to estimate the resulting nonlinear descriptor formulation. The estimation framework was successfully validated for both simulated and real-world LIDAR wind measurement data.

Results show that by using only limited radial velocity measurements along two LIDAR beams, estimates of the horizontal and vertical velocity components, together with pressure, is possible. Since determining both velocity components is equivalent to estimating the wind magnitude and direction, this work shows that it is possible to

overcome Cyclops' dilemma using a dual-beam LIDAR setup. It was further shown that the developed method is able to pinpoint the strength, location and direction of incoming gusts, with good accuracy. The derived simplified model enabled estimates to be computed faster than the sample rate of the LIDAR equipment under study, paving the way for use in real-time applications.

Chapter 6

Conclusion

This thesis presents an estimation framework for nonlinear temporal and spatio-temporal descriptor systems. The approach caters for the estimation of infinite-dimensional nonlinear PDAE models, known to easily describe spatially heterogeneous dynamics. The estimation scheme retains a continuous-space representation throughout and makes use of unscented transform-based inference mechanisms. In summary, the proposed methodology for estimating nonlinear spatio-temporal DAE systems consists of:

- reducing the continuous-time, infinite-dimensional nonlinear PDAEs to finite-dimensional discrete-time descriptor form, with the associated state dimensionality decoupled from the measurement process dimensionality;
- adopting a spatial frequency analysis approach for basis function selection;
- estimating the spatio-temporal field using the proposed estimators for descriptor systems from a set of observations.

The methods developed in this work were validated using both simulated data and real-world measurement data sets.

6.1 Summary

Chapter 2 discusses different classes of spatio-temporal models which are commonly used and reveals how the study of nonlinear PDAEs is warranted for the applications we consider in this work. Nonlinear PDAEs retain the physical meaning of the underlying process governing the system whilst catering for spatial heterogeneity. Model reduction methods are also described to show how a spatio-temporal field may be approximated in finite-dimensional form which is readily applicable to the available state estimation tools which are also described in the same chapter. The prevalent state estimation schemes are discussed with particular emphasis to those concerning this work, namely, the UKF and URTSS methods.

In the first part of Chapter 3, the filtering and smoothing problems for temporal nonlinear descriptor systems having deterministic algebraic equations were addressed by presenting a modified UKF algorithm and a modified UKS algorithm following an RTS formulation. The proposed filter and smoother yield the mean and covariance of both differential and algebraic state estimates. The unscented transform is employed to determine the algebraic state estimates using the mean and covariance of the differential state estimates. The performance of the filter and smoother were demonstrated for an electrochemical case study and compared to alternative approaches. As expected, the URTSS was shown to exhibit superior state estimation performance.

The second part of Chapter 3 derives new unscented transform-based filtering and smoothing algorithms for nonlinear DAEs characterised by stochastic differential and algebraic equations. Differential and algebraic state filtering and smoothing distributions are derived as unscented transform-based Gaussian approximations. The resulting approximate optimal filtering and smoothing solutions consist of the mean and covariance of both differential and algebraic state estimates. By using this information throughout successive recursions in the backward smoothing pass of the URTSS, suitable corrections to the forward filtering results were obtained. A simulated filtering and smoothing problem demonstrated the performance of the proposed filter and smoother. All methods developed in this chapter also allow the inclusion

of both differential and algebraic state variables in the measurement equation.

Chapter 4 tackles the nonlinear PDAE estimation problem, which to the best of our knowledge, has not been addressed in the signal processing literature. A novel state estimation scheme for a class of spatio-temporal nonlinear PDAEs is proposed. By employing a finite-dimensional reduction approach based on basis function decomposition, a nonlinear PDAE is converted to a nonlinear DAE form for which the new unscented transform-based filtering algorithm derived in Chapter 3 is used. With flow estimation applications in mind, estimation error performance is deemed critical and the associated posterior Cramér-Rao bounds are therefore derived. The performance and viability of the developed estimation framework is demonstrated for a realistic atmospheric boundary layer wind flow estimation problem where results show consistent accurate estimates of unmeasured flow velocity and pressure.

Finally, Chapter 5 proposes a spatio-temporal wind flow estimation framework with a reduced-order dynamic fluid model and a nonlinear estimator that accurately estimates velocity and in particular pressure given sparse line-of-sight wind speed measurements. By uniquely employing a pressure Poisson equation (PPE) formulation, a spatially-continuous, strangeness-free nonlinear DAE form of the Navier-Stokes equations was obtained. The modified UKF algorithm derived in Chapter 3 was used to estimate the resulting nonlinear descriptor formulation. The estimation framework was successfully validated for both simulated and real-world LIDAR wind measurement data. Results show that by using only limited radial velocity measurements along two LIDAR beams, the wind magnitude and direction, together with pressure, may be estimated with considerable accuracy. This shows that it is possible to overcome Cyclops' dilemma using a dual-beam LIDAR setup. It was further shown that the developed method is able to provide information about the strength, location and direction of incoming gusts, with good accuracy. The derived simplified model enabled estimates to be computed quicker than the sample rate of the LIDAR equipment used in this study, making it a candidate solution in real-time applications.

6.2 Future work

Various new avenues of research may be identified to enhance the proposed state estimation framework and widen the applicability of nonlinear descriptor models to several spatio-temporal phenomena. Such additions include:

- An observability analysis - Although the sensor configurations employed in this work resulted in good estimation performance, a proper observability analysis is still required to have a handle over an adequate sensor layout and the minimum number of sensors required for an acceptable estimation performance. Also, the optimal sensor placement problem in the context of nonlinear spatio-temporal PDAEs remains an open one.

  For the LIDAR-based wind flow estimation problem studied in this work, sparse flow velocity measurements were used to estimate wind velocity and pressure. However, for other potential applications, such as flow control for reduced drag in transport vehicles, using only pressure sensors would be a cheap alternative to estimate wind flow. The coupling between the velocity and pressure fields, together with the associated observability properties of these spatio-temporal fields should be studied in detail for accurate estimation of both velocity and pressure for different observation processes.

- Joint state-parameter estimation - Whenever in addition to the states, a set of parameters are unknown, a joint state-parameter estimation problem arises. The extension of MCMC sampling methods such as the Gibbs sampler [274] and the Metropolis-Hastings algorithm [275], as well as EM methods [74], to nonlinear DAEs and PDAEs would constitute interesting contributions. Joint state-parameter estimation problems are often encountered in interesting applications in fluid flow systems, electrochemical processes and elastic fiber dynamics, to name a few, and novel estimation tools for such systems would need to be developed. Furthermore, although identifiability is well treated for standard ODE models (e.g. [276]), an identifiability analysis for nonlinear DAE and

PDAE models is lacking in the literature, with a notable exception for nonlinear DAE models being the work of Ljung and Glad [277].

- The consideration of boundary conditions - For the spatio-temporal flow estimation problem considered in this thesis, free boundary conditions were assumed for the large spatial domain under study. The assumption of having no boundary condition is made in a wide range of applications, such as the analysis of cloud dynamics in meteorology [128], diffusion of pollutants [278, 279] and ecological models for the spread of invasive species [43]. Nevertheless, for smaller spatial domains, boundary conditions cannot be ignored, as observed in various phenomena and applications in mathematical physics. In view of such conditions for spatio-temporal processes, the estimators must be modified to accommodate the boundary conditions and allow for estimation with known constraints.

- Particle filtering and smoothing for nonlinear descriptor systems - Although Gaussian approximations work well for several filtering problems, filtering and smoothing distributions may sometimes be multi-model, or a number of state components may be discrete. In such circumstances, particle filtering and smoothing can work better. These methods form Monte Carlo approximations to the Bayesian filtering and smoothing equation solutions. Their application to nonlinear DAEs and PDAEs remains largely unexplored ([280] is an exception).

- Spatio-temporal control - The control problem for spatio-temporal descriptor systems has received little attention in systems and control theory. The development of spatio-temporal descriptor models given in this thesis pave the way for robust, optimal and model predictive controllers that may be modified for this class of systems. The control of such spatio-temporal systems may use static or mobile agents and problems such as the planning of trajectories for mobile agents to estimate and control the spatio-temporal fields may be tackled.

Appendix A

Index concepts of nonlinear descriptor systems

This appendix provides a brief description of the main index concepts of nonlinear descriptor systems as applied in this thesis. Such concepts as well as general properties of DAEs are discussed extensively in [31] and references therein.

To describe DAE index concepts, the following general nonlinear DAE is considered:

$$\tilde{\mathbf{F}}(\dot{\mathbf{X}}(t), \mathbf{X}(t), t) = \mathbf{0}, \tag{A.1}$$

where $\tilde{\mathbf{F}} \in \mathbb{R}^{m_s}$, $\mathbf{X}(t) = (\mathbf{x}^\top(t), \mathbf{z}^\top(t))^\top$ and the superscript $^\top$ denotes the transpose operator. DAE models are harder to handle in comparison to state-space models since in general, equation (A.1) is not solvable for $\dot{\mathbf{X}}(t)$. Should this be the case, the DAE may be reformulated into the state-space system given by

$$\dot{\mathbf{X}}(t) = \bar{\mathbf{F}}(\mathbf{X}(t), t), \tag{A.2}$$

for which standard techniques for state-space models may be employed. In order to transform a DAE into an ODE, it is typically required to differentiate

equations a number of times with respect to time. The differentiation index is the number of times a descriptor model needs to be differentiated to obtain a solution for $\dot{\mathbf{X}}(t)$. More formally, the differentiation index is defined as follows [265]:

Definition A.1 (Differentiation index). The differentiation index of the DAE given by equation (A.1) is the least ν_d such that $\dot{\mathbf{X}}(t)$ is uniquely determined as a function of t and $\mathbf{X}(t)$ using the following system of equations:

$$\begin{aligned} \tilde{\mathbf{F}}(\dot{\mathbf{X}}(t), \mathbf{X}(t), t) &= \mathbf{0}, \\ \frac{d}{dt}\tilde{\mathbf{F}}(\dot{\mathbf{X}}(t), \mathbf{X}(t), t) &= \mathbf{0}, \\ &\vdots \\ \frac{d^{\nu_d}}{dt^{\nu_d}}\tilde{\mathbf{F}}(\dot{\mathbf{X}}(t), \mathbf{X}(t), t) &= \mathbf{0}. \end{aligned} \tag{A.3}$$

In order to further examine DAE systems, Kunkel and Mehrmann [281] present a strangeness index, which generalises the differentiation index. Their analysis makes use of successive differentiations of the DAE formulation. For convenience, we shall assume that all functions are sufficiently smooth. A nonlinear derivative array may first be defined as

$$\tilde{\mathbf{F}}_\Lambda(\mathbf{X}^{(\Lambda+1)}(t), \ldots, \dot{\mathbf{X}}(t), \mathbf{X}(t), t) = \mathbf{0}, \tag{A.4}$$

where the original equation is stacked such that all its derivatives up to level Λ are grouped into the following larger formulation:

$$\tilde{\mathbf{F}}_\Lambda(\mathbf{X}^{(\Lambda+1)}(t), \ldots, \dot{\mathbf{X}}(t), \mathbf{X}(t), t) = \begin{bmatrix} \tilde{\mathbf{F}}(\dot{\mathbf{X}}(t), \mathbf{X}(t), t) \\ \frac{d}{dt}\tilde{\mathbf{F}}(\dot{\mathbf{X}}(t), \mathbf{X}(t), t) \\ \vdots \\ \frac{d^\Lambda}{dt^\Lambda}\tilde{\mathbf{F}}(\dot{\mathbf{X}}(t), \mathbf{X}(t), t) \end{bmatrix}. \tag{A.5}$$

Partial derivatives of $\tilde{\mathbf{F}}_\Lambda$ with respect to a selection of variables p_d from $(\mathbf{X}^{(\Lambda+1)}(t),$

$\ldots, \dot{\mathbf{X}}(t), \mathbf{X}(t), t)$ may be represented by, for instance,

$$\tilde{\mathbf{F}}_{\Lambda;\dot{\mathbf{X}},\ldots,\mathbf{X}^{(\Lambda+1)}} = \left[\frac{d}{d\dot{\mathbf{X}}}\tilde{\mathbf{F}}_{\Lambda} \quad \frac{d}{d\ddot{\mathbf{X}}}\tilde{\mathbf{F}}_{\Lambda} \quad \cdots \quad \frac{d}{d\mathbf{X}^{(\Lambda+1)}}\tilde{\mathbf{F}}_{\Lambda}\right]. \tag{A.6}$$

The solution set for the derivative array $\tilde{\mathbf{F}}_{\upsilon}$ for an integer υ is written as

$$\mathbb{L}_{\upsilon} = \{\tilde{\mathbf{z}}_{\upsilon} \in \mathbb{R} \times \mathbb{R}^n \times \cdots \times \mathbb{R}^n | \tilde{\mathbf{F}}_{\upsilon}(\tilde{\mathbf{z}}_{\upsilon}) = \mathbf{0}\}, \tag{A.7}$$

where $n = n_x + n_z$. The following hypothesis given in [281] may now be stated:

Hypothesis A.1. Consider the general DAE equation given by equation (A.1). There exist integers υ, r_s, a_s, d_s and v_s such that $\mathbb{L}_{\upsilon}$ is not empty and the following conditions are true:

1. The set $\mathbb{L}_{\upsilon} \subseteq \mathbb{R}^{(\upsilon+2)n+1}$ makes up a manifold having dimension $(\upsilon+2)n+1-r_s$.

2. The following holds on $\mathbb{L}_{\upsilon}$:

$$\text{rank } \tilde{\mathbf{F}}_{\upsilon;\mathbf{X},\dot{\mathbf{X}},\ldots,\mathbf{X}^{(\upsilon+1)}} = r_s. \tag{A.8}$$

3. The following holds on $\mathbb{L}_{\upsilon}$:

$$\text{corank } \tilde{\mathbf{F}}_{\upsilon;\mathbf{X},\dot{\mathbf{X}},\ldots,\mathbf{X}^{(\upsilon+1)}} - \text{corank } \tilde{\mathbf{F}}_{\upsilon-1;\mathbf{X},\dot{\mathbf{X}},\ldots,\mathbf{X}^{(\upsilon)}} = v_s, \tag{A.9}$$

   where the corank is the corange dimension with the convention that corank $\tilde{\mathbf{F}}_{-1;\mathbf{X}} = 0$.

4. The following holds on $\mathbb{L}_{\upsilon}$:

$$\text{rank } \tilde{\mathbf{F}}_{\upsilon;\dot{\mathbf{X}},\ldots,\mathbf{X}^{(\upsilon+1)}} = r_s - a_s, \tag{A.10}$$

   where the matrix functions $\mathbf{Z}_2$ and $\mathbf{T}_2$ that are smooth, full rank and

defined on $\mathbb{L}_\upsilon$, of size $((\upsilon+1)m_s, a_s)$ and $(n, n-a_s)$, respectively, satisfy

$$\mathbf{Z}_2^\top \tilde{\mathbf{F}}_{\upsilon;\dot{\mathbf{X}},\ldots,\mathbf{X}^{(\upsilon+1)}} = \mathbf{0}, \tag{A.11}$$

$$\text{rank } \mathbf{Z}_2^\top \tilde{\mathbf{F}}_{\upsilon;\mathbf{X}} = a_s, \tag{A.12}$$

$$\mathbf{Z}_2^\top \tilde{\mathbf{F}}_{\upsilon;\mathbf{X}} \mathbf{T}_2 = \mathbf{0}, \tag{A.13}$$

on $\mathbb{L}_\upsilon$.

5. The following holds on $\mathbb{L}_\upsilon$:

$$\text{rank } \tilde{\mathbf{F}}_{\dot{\mathbf{X}}} \mathbf{T}_2 = d_s = m_s - a_s - \upsilon_s. \tag{A.14}$$

The strangeness index of DAE systems as used in this thesis may now be defined as follows [281]:

Definition A.2 (Strangeness index). The strangeness index of the descriptor system given by equation (A.1) is the least υ for which Hypothesis A.1 is true. If $\upsilon = 0$, the DAE is said to be strangeness-free.

Bibliography

[1] P. Towers and B. L. Jones, "Real-time wind field reconstruction from LiDAR measurements using a dynamic wind model and state estimation," *Wind Energy*, vol. 19, no. 1, pp. 133–150, 2016.

[2] N. Fichaux and J. Wilkes, "Oceans of opportunity - harnessing Europe's largest domestic energy resource," *Offshore Report*, The European Wind Energy Association, 2009.

[3] D. Schlipf and L. Pao, "Preview control of wind turbines," in *The Impact of Control Technology, 2nd edition* (T. Samad and A. M. Annaswamy, eds.), IEEE Control Systems Society, 2014.

[4] M. Courtney, R. Wagner, and P. Lindelöw, "Testing and comparison of lidars for profile and turbulence measurements in wind energy," *IOP Conference Series: Earth and Environmental Science*, vol. 1, no. 1, p. 012021, 2008.

[5] T. Mikkelsen, K. H. Hansen, N. Angelou, M. Sjöholm, M. Harris, P. Hadley, R. Scullion, G. Ellis, and G. Vives, "Lidar wind speed measurements from a rotating spinner," in *Proceedings of the European Wind Energy Conference and Exhibition*, 2010.

[6] F. Dunne, E. Simley, and L. Pao, "LIDAR wind speed measurement analysis and feed-forward blade pitch control for load mitigation in wind turbines," tech. rep., NREL Report NREL/SR-5000-52098, 2011.

[7] N. Wang, K. E. Johnson, and A. D. Wright, "FX-RLS-based feedforward control for LIDAR-enabled wind turbine load mitigation," *IEEE Transactions on Control Systems Technology*, vol. 20, no. 5, pp. 1212–1222, 2012.

[8] D. Schlipf, D. J. Schlipf, and M. Kühn, "Nonlinear model predictive control of wind turbines using LIDAR," *Wind Energy*, vol. 16, no. 7, pp. 1107–1129, 2013.

[9] M. Harris, D. J. Bryce, A. S. Coffey, D. A. Smith, J. Birkemeyer, and U. Knopf, "Advance measurement of gusts by laser anemometry," *Jour-*

nal of Wind Engineering and Industrial Aerodynamics, vol. 95, no. 12, pp. 1637–1647, 2007.

[10] N. Angelou, J. Mann, M. Courtney, and M. Sjöholm, "Doppler Lidar mounted on a wind turbine nacelle - upwind deliverable D6.7.1," tech. rep., Risø-R-1757(EN), Risø DTU, 2010.

[11] D. Schlipf, S. Schuler, P. Grau, F. Allgöwer, and M. Kühn, "Look-ahead cyclic pitch control using LiDAR," in *Proceedings of The Science of making Torque from Wind conference*, 2010.

[12] D. Schlipf, T. Fischer, C. E. Carcangiu, M. Rossetti, and E. Bossanyi, "Load analysis of look-ahead collective pitch control using LIDAR," in *Proceedings of the 10th German Wind Energy Conference (DEWEK)*, 2010.

[13] K. A. Kragh, M. H. Hansen, and T. Mikkelsen, "Precision and shortcomings of yaw error estimation using spinner-based light detection and ranging," *Wind Energy*, vol. 16, no. 3, pp. 353–366, 2013.

[14] R. B. Stull, *An Introduction to Boundary Layer Meteorology*, vol. 13. Dordrecht, Netherlands: Kluwer Academic Publishers, 1988.

[15] J. J. Bertin and R. M. Cummings, *Aerodynamics for Engineers, Sixth Edition*. NJ, USA: Prentice Hall, 2013.

[16] R. Wood, "Impact of advanced aerodynamic technology on transportation energy consumption," in *SAE Technical Paper*, 2004.

[17] L. Ljung, *System Identification: Theory for the User, 2nd Edition*. Upper Saddle River, NJ: Prentice Hall, 1999.

[18] D. Luenberger, "Dynamic equations in descriptor form," *IEEE Transactions on Automatic Control*, vol. 22, no. 3, pp. 312–321, 1977.

[19] G. Verghese, B. Levy, and T. Kailath, "A generalized state-space for singular systems," *IEEE Transactions on Automatic Control*, vol. 26, no. 4, pp. 811–831, 1981.

[20] M. Terra, J. Ishihara, and A. Padoan, "Information filtering and array algorithms for descriptor systems subject to parameter uncertainties," *IEEE Transactions on Signal Processing*, vol. 55, no. 1, pp. 1–9, 2007.

[21] R. K. Mandela, R. Rengaswamy, S. Narasimhan, and L. N. Sridhar, "Recursive state estimation techniques for nonlinear differential algebraic systems," *Chemical Engineering Science*, vol. 65, no. 16, pp. 4548 – 4556, 2010.

[22] V. Dovi' and A. Reverberi, "Optimal solutions of processes described by systems of differential-algebraic equations," *Chemical Engineering Science*, vol. 48, no. 14, pp. 2609–2614, 1993.

[23] L. Quartapelle, *Numerical solution of the incompressible Navier-Stokes equations*. Basel, Switzerland: Birkhäuser, 1993.

[24] A. Bloch, M. Reyhanoglu, and N. McClamroch, "Control and stabilization of nonholonomic dynamic systems," *IEEE Transactions on Automatic Control*, vol. 37, no. 11, pp. 1746–1757, 1992.

[25] J. Sjöberg, K. Fujimoto, and T. Glad, "Model reduction of nonlinear differential-algebraic equations," in *Proceedings of 7th IFAC Symposium on Nonlinear Control Systems*, pp. 176–181, 2007.

[26] M. Hasan and M. Azimi-Sadjadi, "Noncausal image modeling using descriptor approach," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol. 42, no. 8, pp. 536–540, 1995.

[27] D. Luenberger, "Singular dynamic leontief systems," *Econometrica*, vol. 45, no. 4, pp. 991–995, 1977.

[28] D. C. Tarraf and H. H. Asada, "A differential-algebraic equation (DAE) formulation of arterial hemodynamics," in *Proceedings of the 22nd Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, vol. 2, pp. 1126–1129, 2000.

[29] L. Petzold, "Differential/algebraic equations are not ODE's," *SIAM Journal on Scientific and Statistical Computing*, vol. 3, no. 3, pp. 367–384, 1982.

[30] S. C. Patwardhan, S. Narasimhan, P. Jagadeesan, B. Gopaluni, and S. L. Shah, "Nonlinear Bayesian state estimation: A review of recent developments," *Control Engineering Practice*, vol. 20, no. 10, pp. 933–953, 2012.

[31] P. Kunkel and V. Mehrmann, *Differential-Algebraic Equations: Analysis and Numerical Solution*. Zurich, Switzerland: European Mathematical Society, 2006.

[32] R. Hoyle, *Pattern Formation*. Cambridge, UK: Cambridge University Press, 2006.

[33] S. Bhatt, P. W. Gething, O. J. Brady, J. P. Messina, A. W. Farlow, C. L. Moyes, J. M. Drake, J. S. Brownstein, A. G. Hoen, O. Sankoh, M. F. Myers, D. B. George, T. Jaenisch, G. R. W. Wint, C. P. Simmons, T. W. Scott, J. J. Farrar, and S. I. Hay, "The global distribution and burden of dengue," *Nature*, vol. 496, no. 7446, pp. 504–507, 2013.

[34] E. Brooks-Pollock, G. O. Roberts, and M. J. Keeling, "A dynamic model of bovine tuberculosis spread and control in Great Britain," *Nature*, vol. 511, pp. 228–231, 2014.

[35] J. de Munck, H. Huizenga, L. Waldorp, and R. Heethaar, "Estimating stationary dipoles from MEG/EEG data contaminated with spatially and temporally correlated background noise," *IEEE Transactions on Signal Processing*, vol. 50, no. 7, pp. 1565–1572, 2002.

[36] A. Matani, Y. Masuda, H. Okubo, and K. Chihara, "Hierarchical clustering and filtering in half-inverse space for MEG and/or EEG hypothesis-free analysis," *IEEE Transactions on Signal Processing*, vol. 51, no. 2, pp. 350–361, 2003.

[37] M. Dewar and V. Kadirkamanathan, "A canonical space-time state space model: State and parameter estimation," *IEEE Transactions on Signal Processing*, vol. 55, no. 10, pp. 4862–4870, 2007.

[38] P. Aram, D. Freestone, M. Dewar, K. Scerri, V. Jirsa, D. Grayden, and V. Kadirkamanathan, "Spatiotemporal multi-resolution approximation of the Amari type neural field model," *NeuroImage*, vol. 66, pp. 88–102, 2013.

[39] P. Aram, V. Kadirkamanathan, and S. Anderson, "Spatiotemporal system identification with continuous spatial maps and sparse estimation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 11, pp. 2978–2983, 2015.

[40] P. Aram, D. Freestone, M. Cook, V. Kadirkamanathan, and D. Grayden, "Model-based estimation of intra-cortical connectivity using electrophysiological data," *NeuroImage*, vol. 118, pp. 563–575, 2015.

[41] M. Pirani, J. Gulliver, G. W. Fuller, and M. Blangiardo, "Bayesian spatiotemporal modelling for the assessment of short-term exposure to particle pollution in urban areas," *Journal of Exposure Science and Environmental Epidemiology*, vol. 24, pp. 319–327, 2014.

[42] H. Moradkhani, S. Sorooshian, H. V. Gupta, and P. R. Houser, "Dual state-parameter estimation of hydrological models using ensemble Kalman filter," *Advances in Water Resources*, vol. 28, no. 2, pp. 135–147, 2005.

[43] M. B. Hooten and C. K. Wikle, "A hierarchical Bayesian non-linear spatiotemporal model for the spread of invasive species with application to the Eurasian Collared-Dove," *Environmental and Ecological Statistics*, vol. 15, no. 1, pp. 59–70, 2008.

[44] M. Bocquet, C. A. Pires, and L. Wu, "Beyond Gaussian statistical modeling in geophysical data assimilation," *Monthly Weather Review*, vol. 138, no. 8, pp. 2997–3023, 2010.

[45] A. Zammit-Mangion, M. Dewar, V. Kadirkamanathan, and G. Sanguinetti, "Point process modelling of the Afghan war diary," *Proceedings*

of the National Academy of Sciences, vol. 109, no. 31, pp. 12414–12419, 2012.

[46] D. Gu and H. Hu, "Spatial Gaussian process regression with mobile sensor networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 8, pp. 1279–1290, 2012.

[47] A. Kokaram and S. Godsill, "MCMC for joint noise reduction and missing data treatment in degraded video," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 189–205, 2002.

[48] N. Cressie and C. K. Wikle, *Statistics for Spatio-Temporal Data*. Hoboken, NJ: Wiley, 2011.

[49] T. Gneiting, M. G. Genton, and P. Guttorp, "Geostatistical space-time models, stationarity, separability, and full symmetry," in *Statistical Methods for Spatio-Temporal Systems* (B. Finkenstaedt, L. Held, and V. Isham, eds.), ch. 4, pp. 151–175, Boca Raton, FL, USA: Chapman and Hall/CRC Press, 2007.

[50] P. Guttorp and P. D. Sampson, "20 methods for estimating heterogeneous spatial covariance functions with environmental applications," in *Environmental Statistics* (G. P. Patil and C. R. Rao, eds.), vol. 12 of *Handbook of Statistics*, pp. 661–689, Amsterdam, Netherlands: Elsevier, 1994.

[51] J. A. Duan, A. E. Gelfand, and C. F. Sirmans, "Modeling space-time data using stochastic differential equations," *Bayesian Analysis*, vol. 4, no. 4, pp. 733–758, 2009.

[52] F. Lindgren, H. Rue, and J. Lindström, "An explicit link between Gaussian fields and Gaussian Markov random fields: the stochastic partial differential equation approach," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 73, no. 4, pp. 423–498, 2011.

[53] G. Storvik, A. Frigessi, and D. Hirst, "Stationary space-time Gaussian fields and their time autoregressive representation," *Statistical Modelling*, vol. 2, no. 2, pp. 139–161, 2002.

[54] D. Dochain, "State and parameter estimation in chemical and biochemical processes: a tutorial," *Journal of Process Control*, vol. 13, no. 8, pp. 801 – 818, 2003.

[55] E. Misawa and J. Hedrick, "Nonlinear observers - a state-of-the-art survey," *Journal of Dynamic Systems, Measurement, and Control*, vol. 111, no. 3, pp. 344–352, 1989.

[56] T. Unny, "Stochastic partial differential equations in groundwater hydrology," *Stochastic Hydrology and Hydraulics*, vol. 3, no. 2, pp. 135–153, 1989.

[57] A. Fassó, M. Cameletti, and O. Nicolis, "Air quality monitoring using heterogeneous networks," *Environmetrics*, vol. 18, no. 3, pp. 245–264, 2007.

[58] A. Jazwinski, *Stochastic processes and filtering theory*. NY, USA: Dover Publications, 1970.

[59] A. Gelb, *Applied optimal estimation*. Cambridge, MA, USA: The MIT Press, 1974.

[60] P. Maybeck, *Stochastic models, estimation and control*. New York, USA: Academic Press, 1979.

[61] H. Sorenson, *Kalman Filtering: Theory and applications*. New York, USA: IEEE Press, 1985.

[62] T. Söderström, *Discrete-time stochastic system*. Advanced textbooks in control and signal processing, Springer-Verlag, 2003.

[63] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman filter - Particle filters for tracking applications*. Boston, MA, USA: Artech House, 2004.

[64] D. Simon, *Optimal state estimation - Kalman, H_∞ and nonlinear approaches*. Hoboken, NJ, USA: John Wiley & Sons., 2006.

[65] S. Särkkä, *Bayesian Filtering and Smoothing*. Cambridge, UK: Cambridge University Press, 2013.

[66] M. Arulampalam, S. Maskell, N. Gordon, and T. Clapp, "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[67] Z. Chen, "Bayesian filtering: From Kalman filters to particle filters, and beyond," tech. rep., Adaptive Syst.Lab., McMaster University, Ontario, Canada, 2003.

[68] J. Ching, J. L. Beck, and K. A. Porter, "Bayesian state and parameter estimation of uncertain dynamical systems," *Probabilistic Engineering Mechanics*, vol. 21, no. 1, pp. 81 – 96, 2006.

[69] F. Daum, "Nonlinear filters: beyond the Kalman filter," *IEEE Aerospace and Electronic Systems Magazine*, vol. 20, no. 8, pp. 57–69, 2005.

[70] J. B. Rawlings and B. R. Bakshi, "Particle filtering and moving horizon estimation," *Computers and Chemical Engineering*, vol. 30, no. 10-12, pp. 1529–1541, 2006.

[71] M. Soroush, "State and parameter estimations and their applications in process control," *Computers and Chemical Engineering*, vol. 23, no. 2, pp. 229 – 245, 1998.

[72] S. A. Billings, *Nonlinear System Identification: NARMAX Methods in the Time, Frequency, and Spatio-Temporal Domains*. West Sussex, UK: John Wiley & Sons., 2013.

[73] C. P. Robert and G. Casella, *Monte Carlo Statistical Methods*. New York, USA: Springer-Verlag, 2004.

[74] G. J. McLachlan and T. Krishnan, *The EM Algorithm and Extensions*. New York, USA: John Wiley & Sons., 1997.

[75] M. Darouach and L. Boutat-Baddas, "Observers for a class of nonlinear singular systems," *IEEE Transactions on Automatic Control*, vol. 53, no. 11, pp. 2627–2633, 2008.

[76] M. Aliyu and E. Boukas, "Kalman filtering for affine nonlinear descriptor systems," *Circuits, Systems, and Signal Processing*, vol. 30, no. 1, pp. 125–142, 2011.

[77] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME Journal of Basic Engineering*, vol. 82, pp. 35–45, 1960.

[78] S. Julier, J. Uhlmann, and H. Durrant-Whyte, "A new method for the nonlinear transformation of means and covariances in filters and estimators," *IEEE Transactions on Automatic Control*, vol. 45, no. 3, pp. 477–482, 2000.

[79] S. Julier and J. Uhlmann, "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, 2004.

[80] H. E. Rauch, F. Tung, and C. T. Striebel, "Maximum likelihood estimates of linear dynamic systems," *AIAA Journal*, vol. 3, no. 8, pp. 1445–1450, 1965.

[81] S. Särkkä, "Unscented Rauch-Tung-Striebel smoother," *IEEE Transactions on Automatic Control*, vol. 53, no. 3, pp. 845–849, 2008.

[82] S. Särkkä, "Continuous-time and continuous-discrete-time unscented Rauch-Tung-Striebel smoothers," *Signal Processing*, vol. 90, no. 1, pp. 225–235, 2010.

[83] I. Alkov and D. Weidemann, "Fault detection with unscented Kalman filter applied to nonlinear differential-algebraic systems," in *Proceedings of the 18th International Conference on Methods and Models in Automation and Robotics*, pp. 166–171, 2013.

[84] S. Pan, H. Su, Z. Liu, and P. Li, "An unscented Kalman filtering approach for nonlinear singular systems," in *Proceedings of the 11th International Conference on Control Automation Robotics Vision*, pp. 485–490, 2010.

[85] V. Becerra, P. Roberts, and G. Griffiths, "Applying the extended Kalman filter to systems described by nonlinear differential-algebraic equations," *Control Engineering Practice*, vol. 9, no. 3, pp. 267–281, 2001.

[86] G. Box and G. Jenkins, *Time Series Analysis: Forecasting and Control*. San Fransisco, CA, USA: Holden-Day, 1970.

[87] R. L. Martin and J. E. Oeppen, "The identification of regional forecasting models using space: Time correlation functions," *Transactions of the Institute of British Geographers*, no. 66, pp. 95–118, 1975.

[88] P. E. Pfeifer and S. J. Deutrch, "A three-stage iterative procedure for space-time modeling," *Technometrics*, vol. 22, no. 1, pp. 35–47, 1980.

[89] J. Besag, "Spatial interaction and statistical analysis of lattice systems," *Journal of the Royal Statistical Society B*, vol. 36, no. 2, pp. 192–236, 1974.

[90] R. J. Bennett, *Spatial Time Series*. London, UK: Pion, 1979.

[91] V. Di Giacinto, "A generalized space-time ARMA model with an application to regional unemployment analysis in Italy," *International Regional Science Review*, vol. 29, no. 2, pp. 159–198, 2006.

[92] P. E. Pfeifer and S. E. Bodily, "A test of space-time ARMA modelling and forecasting of hotel data," *Journal of Forecasting*, vol. 9, no. 3, pp. 255–272, 1990.

[93] L. Anselin, *Spatial econometrics: Models and applications*. Dordrecht, Netherlands: Kluwer Academic, 1988.

[94] N. Cressie, *Statistics for spatial data*. New York, NY, USA: John Wiley & Sons., 1993.

[95] C. A. Glasbey and D. J. Allcroft, "A spatiotemporal auto-regressive moving average model for solar radiation," *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, vol. 57, no. 3, pp. 343–355, 2008.

[96] K. Kaneko, "Overview of coupled map lattices," *Chaos*, vol. 2, no. 3, pp. 279–282, 1992.

[97] K. Kaneko, *Theory and Applications of Coupled Map Lattices*. New York, USA: John Wiley & Sons., 1993.

[98] K. Kaneko, "Spatiotemporal intermittency in coupled map lattices," *Progress of Theoretical Physics*, vol. 74, no. 5, pp. 1033–1044, 1985.

[99] K. Kaneko, "Lyapunov analysis and information flow in coupled map lattices," *Physica D: Nonlinear Phenomena*, vol. 23, no. 1-3, pp. 436–447, 1986.

[100] K. Kaneko, "Pattern dynamics in spatiotemporal chaos: Pattern selection, diffusion of defect and pattern competition intermittency," *Physica D: Nonlinear Phenomena*, vol. 34, pp. 1–41, 1989.

[101] T. Yanagita and K. Kaneko, "Coupled map lattice model for convection," *Physics Letters A*, vol. 175, no. 6, pp. 415 – 420, 1993.

[102] T. Yanagita and K. Kaneko, "Rayleigh-Bénard convection patterns, chaos, spatiotemporal chaos and turbulence," *Physica D: Nonlinear Phenomena*, vol. 82, no. 3, pp. 288 – 313, 1995.

[103] H. Levine and W. Reynolds, "Coupled map lattice techniques for simulating phenomena in reaction-diffusion systems," *Chaos*, vol. 2, no. 3, pp. 337–342, 1992.

[104] T. Yanagita, "Phenomenology of boiling: A coupled map lattice model," *Chaos*, vol. 2, no. 3, pp. 343–350, 1992.

[105] T. Yanagita and K. Kaneko, "Modeling and characterization of cloud dynamics," *Phys. Rev. Lett.*, vol. 78, pp. 4297–4300, 1997.

[106] R. J. Deissler, "Spatially growing waves, intermittency, and convective chaos in an open-flow system," *Physica D: Nonlinear Phenomena*, vol. 25, no. 1-3, pp. 233–260, 1987.

[107] R. J. Deissler and K. Kaneko, "Velocity-dependent lyapunov exponents as a measure of chaos for open-flow systems," *Physics Letters A*, vol. 119, no. 8, pp. 397 – 402, 1987.

[108] K. Kaneko, "Spatial period-doubling in open flow," *Physics Letters A*, vol. 111, no. 7, pp. 321 – 325, 1985.

[109] D. A. Kessler, H. Levine, and W. N. Reynolds, "Coupled-map lattice model for crystal growth," *Phys. Rev. A*, vol. 42, pp. 6125–6128, 1990.

[110] T. Ikegami and K. Kaneko, "Evolution of host-parasitoid network through homeochaotic dynamics," *Chaos*, vol. 2, no. 3, pp. 397–407, 1992.

[111] P. Marcos-Nikolaus, J. M. Martin-González, and R. V. Solé, "Spatial forecasting: Detecting determinism from single snapshots," *International Journal of Bifurcation and Chaos*, vol. 12, no. 2, pp. 369–376, 2002.

[112] R. Sole, J. Bascompte, and J. Valls, "Nonequilibrium dynamics in lattice ecosystems: Chaotic stability and dissipative structures," *Chaos*, vol. 2, no. 3, pp. 387–395, 1992.

[113] C. Price, P. Wambacq, and A. Oosterlinck, "The plastic coupled map lattice: A novel image-processing paradigm," *Chaos*, vol. 2, no. 3, pp. 351–366, 1992.

[114] A. Holden, J. Tucker, H. Zhang, and M. Pole, "Coupled map lattice as computational systems," *Chaos*, vol. 2, no. 3, pp. 367–376, 1992.

[115] M. Shen, G. Chang, S. Wang, and P. Beadle, "Nonlinear dynamics of EEG signal based on coupled network lattice model," in *Advances in Neural Networks - ISNN 2006* (J. Wang, Z. Yi, J. Zurada, B.-L. Lu, and H. Yin, eds.), vol. 3973 of *Lecture Notes in Computer Science*, pp. 560–565, Springer Berlin Heidelberg, 2006.

[116] M. Shen, L. Lin, and G. Chang, "Novel coupled map lattice model for prediction of EEG signal," in *Advances in Neural Networks - ISNN 2008* (F. Sun, J. Zhang, Y. Tan, J. Cao, and W. Yu, eds.), vol. 5263 of *Lecture Notes in Computer Science*, pp. 347–356, Springer Berlin Heidelberg, 2008.

[117] S. A. Billings and D. Coca, "Identification of coupled map lattice models of deterministic distributed parameter systems," *International Journal of Systems Science*, vol. 33, no. 8, pp. 623–634, 2002.

[118] Y. Pan and S. A. Billings, "The identification of complex spatiotemporal patterns using coupled map lattice models," *International Journal of Bifurcation and Chaos*, vol. 18, no. 04, pp. 997–1013, 2008.

[119] N. Parekh, S. Parthasarathy, and S. Sinha, "Global and local control of spatiotemporal chaos in coupled map lattices," *Phys. Rev. Lett.*, vol. 81, pp. 1401–1404, 1998.

[120] L. A. Bunimovich, "Coupled map lattices: one step forward and two steps back," *Physica D: Nonlinear Phenomena*, vol. 86, no. 1-2, pp. 248–255, 1995.

[121] H. Richter, "Coupled map lattices as spatio-temporal fitness functions: Landscape measures and evolutionary optimization," *Physica D: Nonlinear Phenomena*, vol. 237, no. 2, pp. 167–186, 2008.

[122] R. M. May, "Simple mathematical models with very complicated dynamics," *Nature*, vol. 261, pp. 459–467, 1976.

[123] J. Jost and M. P. Joy, "Spectral properties and synchronization in coupled map lattices," *Phys. Rev. E*, vol. 65, no. 1, p. 016201, 2001.

[124] D. Coca and S. Billings, "Identification of coupled map lattice models of complex spatio-temporal patterns," *Physics Letters A*, vol. 287, no. 1-2, pp. 65–73, 2001.

[125] S. A. Billings, L. Z. Guo, and H. L. Wei, "Identification of coupled map lattice models for spatio-temporal patterns using wavelets," *International Journal of Systems Science*, vol. 37, no. 14, pp. 1021–1038, 2006.

[126] D. Coca and S. A. Billings, "Analysis and reconstruction of stochastic coupled map lattice models," *Physics Letters A*, vol. 315, no. 1-2, pp. 61–75, 2003.

[127] L. Z. Guo, S. S. Mei, and S. A. Billings, "Neighbourhood detection and identification of spatio-temporal dynamical systems using a coarse-to-fine approach," *International Journal of Systems Science*, vol. 38, no. 1, pp. 1–15, 2007.

[128] C. K. Wikle, "A kernel-based spectral model for non-Gaussian spatio-temporal processes," *Statistical Modelling*, vol. 2, no. 4, pp. 299–314, 2002.

[129] M. Kot, M. A. Lewis, and P. van den Driessche, "Dispersal data and the spread of invading organisms," *Ecology*, vol. 77, no. 7, pp. 2027–2042, 1996.

[130] M. Kot and W. M. Schaffer, "Discrete-time growth-dispersal models," *Mathematical Biosciences*, vol. 80, no. 1, pp. 109 – 136, 1986.

[131] R. A. Fisher, "The wave of advance of advantageous genes," *Annals of Eugenics*, vol. 7, no. 4, pp. 355–369, 1937.

[132] D. N. Christopher K. Wikle, Ralph F. Milliff and L. M. Berliner, "Spatiotemporal hierarchical Bayesian modeling: Tropical ocean surface winds," *Journal of the American Statistical Association*, vol. 96, no. 454, pp. 382–397, 2001.

[133] K. Xu, C. K. Wikle, and N. I. Fox, "A kernel-based spatio-temporal dynamical model for nowcasting weather radar reflectivities," *Journal of the American Statistical Association*, vol. 100, no. 472, pp. 1133–1144, 2005.

[134] N. Cressie and C. K. Wikle, "Space-time Kalman filter," *Encyclopedia of Environmetrics*, vol. 4, pp. 2045–2049, 2002.

[135] C. Wikle and N. Cressie, "A dimension-reduced approach to space-time Kalman filtering," *Biometrika*, vol. 86, no. 4, pp. 815–829, 1999.

[136] P. E. Brown, G. O. Roberts, K. F. Kåresen, and S. Tonellato, "Blur-generated non-separable space-time models," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 62, no. 4, pp. 847–860, 2000.

[137] P. E. Brown, P. J. Diggle, M. E. Lord, and P. C. Young, "Space-time calibration of radar rainfall data," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 50, no. 2, pp. 221–241, 2001.

[138] M. Dewar, K. Scerri, and V. Kadirkamanathan, "Data-driven spatio-temporal modeling using the integro-difference equation," *IEEE Transactions on Signal Processing*, vol. 57, no. 1, pp. 83–91, 2009.

[139] K. Scerri, M. Dewar, and V. Kadirkamanathan, "Estimation and model selection for an IDE-based spatio-temporal model," *IEEE Transactions on Signal Processing*, vol. 57, no. 2, pp. 482–492, 2009.

[140] C. Wikle and M. Hooten, *Hierarchical Bayesian Spatio-Temporal Models for Population Spread*. Oxford, UK: Oxford University Press, 2005.

[141] M. Kot, "Discrete-time travelling waves: Ecological examples," *Journal of Mathematical Biology*, vol. 30, no. 4, pp. 413–436, 1992.

[142] M. Neubert, M. Kot, and M. Lewis, "Dispersal and pattern formation in a discrete-time predator-prey model," *Theoretical Population Biology*, vol. 48, no. 1, pp. 7 – 43, 1995.

[143] S. A. Billings, L. Z. Guo, and H. L. Wei, "Projecting rates of spread for invasive species," *Risk Analysis*, vol. 24, no. 4, pp. 817–831, 2004.

[144] R. R. Veit and M. A. Lewis, "Dispersal, population growth, and the Allee effect: Dynamics of the house finch invasion of eastern North America," *The American Naturalist*, vol. 148, no. 2, pp. 255–274, 1996.

[145] M.-H. Wang and M. Kot, "Speeds of invasion in a model with strong or weak Allee effects," *Mathematical Biosciences*, vol. 171, no. 1, pp. 83 – 97, 2001.

[146] M. G. Neubert and H. Caswell, "Spatiotemporal hierarchical Bayesian modeling: Tropical ocean surface winds," *Ecology*, vol. 81, no. 6, pp. 1613–1628, 2000.

[147] J. Medlock and M. Kot, "Spreading disease: integro-differential equations old and new," *Mathematical Biosciences*, vol. 184, no. 2, pp. 201–222, 2003.

[148] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. Cambridge, Massachusetts, USA: MIT Press, 2006.

[149] D. Freestone, P. Aram, M. Dewar, K. Scerri, D. Grayden, and V. Kadirkamanathan, "A data-driven framework for neural field modeling," *NeuroImage*, vol. 56, no. 3, pp. 1043–1058, 2011.

[150] M. B. Hooten, C. K. Wikle, R. M. Dorazio, and J. A. Royle, "Hierarchical spatiotemporal matrix models for characterizing invasions," *Biometrics*, vol. 63, no. 2, pp. 558–567, 2007.

[151] K. Scerri, *A Systems Approach to Spatio-Temporal Modelling*. PhD thesis, University of Sheffield, Sheffield, UK, 2010.

[152] A. Zammit-Mangion, G. Sanguinetti, and V. Kadirkamanathan, "A variational approach for the online dual estimation of spatiotemporal systems governed by the IDE," in *Proceedings of the 18th IFAC World Congress*, pp. 3204–3209, 2011.

[153] A. Zammit-Mangion, *Modelling from Spatiotemporal Data: A Dynamic Systems Approach*. PhD thesis, University of Sheffield, Sheffield, UK, 2011.

[154] S. Farlow, *Partial Differential Equations for Scientists and Engineers*. Dover Publications, 1993.

[155] G. Helling, *Partial Differential Equations. An Introduction*. Blaisdell Publishing, 1960.

[156] A. Mitchell and R. Wait, *Partial Differential Equations of Mathematical Physics*. Elsevier Science, 1973.

[157] M. Smith, *Introduction to the Theory of Partial Differential Equations*. Van Nostrand Company, 1967.

[158] A. Bennett, *Inverse Modeling of the Ocean and Atmosphere*. Cambridge, UK: Cambridge University Press, 2002.

[159] E. E. Holmes, M. A. Lewis, J. E. Banks, and R. R. Veit, "Partial differential equations in ecology: spatial interactions and population dynamics," *Ecology*, vol. 75, no. 1, pp. 17–29, 1994.

[160] H. T. Banks and K. Kunisch, *Estimation Techniques for Distributed Parameter Systems*. Boston, USA: Birkhäuser, 1989.

[161] M. I. Asensio and L. Ferragut, "On a wildland fire model with radiation," *International Journal for Numerical Methods in Engineering*, vol. 54, no. 1, pp. 137–157, 2002.

[162] L. Evans, *Partial Differential Equations (Graduate Studies in Mathematics, Vol. 19)*. Providence, RI, USA: AMS, 1998.

[163] J. Hill, *Heat Conduction*. Oxford, UK: Blackwell Scientific, 1987.

[164] L. Debnath, *Nonlinear Partial Differential Equations for Scientists and Engineers*. New York, USA: Birkhauser, 2nd ed., 2005.

[165] J. Logan, *An Introduction to Nonlinear Partial Differential Equations*. Hoboken, NJ, USA: John Wiley & Sons., 2nd ed., 2008.

[166] S. Omatu and J. Seinfeld, *Distributed parameter systems: Theory and Applications*. Oxford, UK: Clarendon Press, 1989.

[167] A. Mitchell and R. Wait, *The Finite Element Methods in Partial Differential Equations*. John Wiley & Sons., 1977.

[168] G. Smith, *Numerical Solution of Partial Differential Equations*. Oxford, UK: Oxford University Press, 1969.

[169] L. Guo and S. A. Billings, "Identification of partial differential equation models for continuous spatio-temporal dynamical systems," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, no. 8, pp. 657–661, 2006.

[170] L. Z. Guo, S. A. Billings, and D. Coca, "Consistent recursive parameter estimation of partial differential equation models," *International Journal of Control*, vol. 82, no. 10, pp. 1946–1954, 2009.

[171] D. Coca and S. A. Billings, "Direct parameter identification of distributed parameter systems," *International Journal of Systems Science*, vol. 31, no. 1, pp. 11–17, 2000.

[172] J. Niedzwecki and P.-Y. Liagre, "System identification of distributed-parameter marine riser models," *Ocean Engineering*, vol. 30, no. 11, pp. 1387 – 1415, 2003.

[173] C. Travis and L. White, "Parameter identification of distributed parameter systems," *Mathematical Biosciences*, vol. 77, no. 1-2, pp. 341–352, 1985.

[174] H. Voss, M. J. Bünner, and M. Abel, "Identification of continuous, spatiotemporal systems," *Phys. Rev. E*, vol. 57, pp. 2820–2823, Mar 1998.

[175] R. A. Carmona, *Stochastic Partial Differential Equations: Six Perspectives*. American Mathematical Society, 1998.

[176] R. C. Dalang and N. E. Frangos, "The stochastic wave equation in two spatial dimensions," *The Annals of Probability*, vol. 26, no. 1, pp. 187–212, 1998.

[177] C. Prévôt and M. Röckner, *A Concise Course on Stochastic Partial Differential Equations*. Berlin, Germany: Springer-Verlag, 2007.

[178] J. B. Walsh, "A stochastic model of neural response," *Advances in Applied Probability*, vol. 13, no. 2, pp. pp. 231–281, 1981.

[179] J. Duan and B. Goldys, "Ergodicity of stochastically forced large scale geophysical flows," *International Journal of Mathematics and Mathematical Sciences*, vol. 28, no. 6, pp. 313–320, 2001.

[180] H. Krim and Y. Bao, "A stochastic diffusion approach to signal denoising," in *Proceedings of the 1999 IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 4, pp. 1773–1776, 1999.

[181] D. Coca and S. A. Billings, "Identification of finite dimensional models of infinite dimensional dynamical systems," *Automatica*, vol. 38, no. 11, pp. 1851 – 1865, 2002.

[182] V. Solo, "Identification of a noisy stochastic heat equation with the EM algorithm," in *Proceedings of the 41st IEEE Conference on Decision and Control*, vol. 4, pp. 4505–4508, 2002.

[183] A. Zammit-Mangion, G. Sanguinetti, and V. Kadirkamanathan, "Variational estimation in spatiotemporal systems from continuous and point-process observations," *IEEE Transactions on Signal Processing*, vol. 60, pp. 3449–3459, July 2012.

[184] K. Chudej, V. Petzet, S. Scherdel, H. J. Pesch, K. Schittkowski, P. Heidebrecht, and K. Sundmacher, "Index analysis of a nonlinear PDAE system describing a molten carbonate fuel cell," *Proceedings in Applied Mathematics and Mechanics*, vol. 3, no. 1, pp. 563–564, 2003.

[185] Y.-I. Lim, S.-C. Chang, and S. B. Jørgensen, "A novel partial differential algebraic equation (PDAE) solver: iterative space-time conservation element/solution element (CE/SE) method," *Computers and Chemical Engineering*, vol. 28, no. 8, pp. 1309–1324, 2004.

[186] G. Alì, A. Bartel, and M. Günther, "Existence and uniqueness for an elliptic PDAE model of integrated circuits," *SIAM Journal on Applied Mathematics*, vol. 70, no. 5, pp. 1587–1610, 2010.

[187] M. Grothaus and N. Marheineke, "On a nonlinear partial differential algebraic system arising in the technical textile industry: analysis and numerics," *IMA Journal of Numerical Analysis*, vol. 36, no. 4, pp. 1783–1803, 2016.

[188] C. Grossmann, H. G. Roos, and M. Stynes, *Numerical Treatment of Partial Differential Equations*. Berlin, Germany: Springer-Verlag, 2007.

[189] E. Hausenblas, "Approximation for semilinear stochastic evolution equations," *Potential Analysis*, vol. 18, no. 2, pp. 141–186, 2003.

[190] M. Demetriou, A. Paskaleva, O. Vayena, and H. Doumanidis, "Scanning actuator guidance scheme in a 1-D thermal manufacturing process," *IEEE Transactions on Control Systems Technology*, vol. 11, no. 5, pp. 757–764, 2003.

[191] N. Leonard, D. Paley, F. Lekien, R. Sepulchre, D. Fratantoni, and R. Davis, "Collective motion, sensor networks, and ocean sampling," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 48–74, 2007.

[192] H. Sorenson, "Least-squares estimation: from Gauss to Kalman," *IEEE Spectrum*, vol. 7, no. 7, pp. 63–68, 1970.

[193] D. Luenberger, "Observers for multivariable systems," *IEEE Transactions on Automatic Control*, vol. 11, no. 2, pp. 190–197, 1966.

[194] J. R. Stroud, P. Müller, and B. Sansó, "Dynamic models for spatiotemporal data," *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, vol. 63, no. 4, pp. 673–689, 2001.

[195] Y. Ho and R. Lee, "A Bayesian approach to problems in stochastic estimation and control," *IEEE Transactions on Automatic Control*, vol. 9, no. 4, pp. 333–339, 1964.

[196] R. C. K. Lee, *Optimal Estimation, Identification and Control*. Cambridge, MA, USA: MIT Press, 1964.

[197] E. A. Wan and R. Van der Merwe, "The unscented Kalman filter," in *Kalman Filtering and Neural Networks* (S. Haykin, ed.), ch. 7, pp. 221–280, New York, NY, USA: John Wiley & Sons., 2001.

[198] M. S. Grewal and A. P. Andrews, *Kalman Filtering, Theory and Practice Using MATLAB*. New Jersey, USA: Wiley, 2001.

[199] K. R. Muske and T. F. Edgar, "Nonlinear process control," ch. Nonlinear state estimation, pp. 311–370, Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1997.

[200] J. Sarmavuori and S. Särkkä, "Fourier-Hermite Kalman filter," *IEEE Transactions on Automatic Control*, vol. 57, no. 6, pp. 1511–1515, 2012.

[201] A. Romanenko, L. O. Santos, and P. A. F. N. A. Afonso, "Unscented Kalman filtering of a simulated pH system," *Industrial Engineering Chemistry Research*, vol. 43, no. 23, pp. 7531–7538, 2004.

[202] A. Romanenko and J. A. Castro, "The unscented filter as an alternative to the EKF for nonlinear state estimation: a simulation case study," *Computers Chemical Engineering*, vol. 28, no. 3, pp. 347 – 355, 2004.

[203] P. Vachhani, S. Narasimhan, and R. Rengaswamy, "Robust and reliable estimation via unscented recursive nonlinear dynamic data reconciliation," *Journal of Process Control*, vol. 16, no. 10, pp. 1075–1086, 2006.

[204] E. Wan and R. Van der Merwe, "The unscented Kalman filter for nonlinear estimation," in *The IEEE Adaptive Systems for Signal Processing, Communications, and Control Symposium*, pp. 153–158, 2000.

[205] S. Julier, J. Uhlmann, and H. Durrant-Whyte, "A new approach for filtering nonlinear systems," in *Proceedings of the American Control Conference*, vol. 3, pp. 1628–1632, 1995.

[206] G. Burgers, P. Leeuwen, and G. Evensen, "Analysis scheme in the ensemble Kalman filter," *Monthly Weather Review*, vol. 126, pp. 1719–1724, 1998.

[207] G. Evensen, "The ensemble Kalman filter: theoretical formulation and practical implementation," *Ocean Dynamics*, vol. 53, no. 4, pp. 343–367, 2003.

[208] O. Cappe, S. Godsill, and E. Moulines, "An overview of existing methods and recent advances in sequential Monte Carlo," *Proceedings of the IEEE*, vol. 95, no. 5, pp. 899–924, 2007.

[209] D. G. Robertson, J. H. Lee, and J. B. Rawlings, "A moving horizon-based approach for least-squares estimation," *AIChE Journal*, vol. 42, no. 8, pp. 2209–2224, 1996.

[210] C. C. Qu and J. Hahn, "Computation of arrival cost for moving horizon estimation via unscented Kalman filtering," *Journal of Process Control*, vol. 19, no. 2, pp. 358 – 363, 2009.

[211] M. Tenny and J. Rawlings, "Efficient moving horizon estimation and nonlinear model predictive control," in *Proceedings of the 2002 American Control Conference*, vol. 6, pp. 4475–4480, 2002.

[212] D. Fraser and J. Potter, "The optimum linear smoother as a combination of two optimum linear filters," *IEEE Transactions on Automatic Control*, vol. 14, no. 4, pp. 387–390, 1969.

[213] G. Kitagawa, "The two-filter formula for smoothing and an implementation of the Gaussian-sum smoother," *Annals of the Institute of Statistical Mathematics*, vol. 46, no. 4, pp. 605–623, 1994.

[214] M. Briers, A. Doucet, and S. Maskell, "Smoothing algorithms for state–space models," *Annals of the Institute of Statistical Mathematics*, vol. 62, no. 1, p. 61, 2009.

[215] M. Soleimanzadeh, R. Wisniewski, and A. Brand, "State-space representation of the wind flow model in wind farms," *Wind Energy*, vol. 17, no. 4, pp. 627–639, 2014.

[216] J. Mercieca, P. Aram, and V. Kadirkamanathan, "Unscented Rauch-Tung-Striebel smoothing for nonlinear descriptor systems," in *Proceedings of the 2015 European Control Conference (ECC)*, pp. 491–496, 2015.

[217] L. Dai, "Filtering and LQG problems for discrete-time stochastic singular systems," *IEEE Transactions on Automatic Control*, vol. 34, no. 10, pp. 1105–1108, 1989.

[218] R. Nikoukhah, A. Willsky, and B. Levy, "Kalman filtering and Riccati equations for descriptor systems," *IEEE Transactions on Automatic Control*, vol. 37, no. 9, pp. 1325–1342, 1992.

[219] M. Darouach and M. Boutayeb, "Design of observers for descriptor systems," *IEEE Transactions on Automatic Control*, vol. 40, no. 7, pp. 1323–1327, 1995.

[220] M. Hou and P. Muller, "Observer design for descriptor systems," *IEEE Transactions on Automatic Control*, vol. 44, no. 1, pp. 164–169, 1999.

[221] D. N. Shields, "Observer design and detection for nonlinear descriptor systems," *International Journal of Control*, vol. 67, no. 2, pp. 153–168, 1997.

[222] Z. Gao and D. W. C. Ho, "State/noise estimator for descriptor systems with application to sensor fault diagnosis," *IEEE Transactions on Signal Processing*, vol. 54, no. 4, pp. 1316–1326, 2006.

[223] D. Koenig, "Observer design for unknown input nonlinear descriptor systems via convex optimization," *IEEE Transactions on Automatic Control*, vol. 51, no. 6, pp. 1047–1052, 2006.

[224] G. Lu and D. W. C. Ho, "Full-order and reduced-order observers for Lipschitz descriptor systems: the unified LMI approach," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 53, no. 7, pp. 563–567, 2006.

[225] C. Yang, Q. Zhang, and T. Chai, "Observer design for a class of nonlinear descriptor systems," in *Proceedings of the 48th IEEE Conference on Decision and Control held jointly with the 2009 28th Chinese Control Conference*, pp. 8232–8237, 2009.

[226] C. Yang, Q. Zhang, and T. Chai, "Nonlinear observers for a class of nonlinear descriptor systems," *Optimal Control Applications and Methods*, vol. 34, no. 3, pp. 348–363, 2013.

[227] G. Zimmer and J. Meier, "On observing nonlinear descriptor systems," *Systems and Control Letters*, vol. 32, no. 1, pp. 43 – 48, 1997.

[228] W. C. Rheinboldt, "Differential-algebraic systems as differential equations on manifolds," *Mathematics of Computation*, vol. 43, no. 168, pp. 473–482, 1984.

[229] M. Boutayeb and M. Darouach, "Observers design for nonlinear descriptor systems," in *Proceedings of the 34th IEEE Conference on Decision and Control*, vol. 3, pp. 2369–2374, 1995.

[230] N. Kidane, Y. Yamashita, and H. Nishitani, "Observer based I/O-linearizing control of high index DAE systems," in *Proceedings of the American Control Conference*, vol. 4, pp. 3537–3542, 2003.

[231] S. Kaprielian and J. Turi, "An observer for a nonlinear descriptor system," in *Proceedings of the 31st IEEE Conference on Decision and Control*, vol. 1, pp. 975–976, 1992.

[232] J. Åslund and E. Frisk, "An observer for non-linear differential-algebraic systems," *Automatica*, vol. 42, no. 6, pp. 959 – 965, 2006.

[233] R. Nikoukhah, "A new methodology for observer design and implementation," *IEEE Transactions on Automatic Control*, vol. 43, no. 2, pp. 229–234, 1998.

[234] J. S. Albuquerque and L. Biegler, "Decomposition algorithms for on-line estimation with nonlinear DAE models," *Computers and Chemical Engineering*, vol. 21, no. 3, pp. 283–299, 1997.

[235] R. K. Mandela, R. Rengaswamy, and S. Narasimhan, "Nonlinear state estimation of differential algebraic systems," in *Proceedings of the 7th IFAC International Symposium on Advanced Control of Chemical Processes*, pp. 792–797, 2009.

[236] J. Purohit, S. C. Patwardhan, and S. Mahajani, "Predictive control of a reactive distillation column using multi-rate DAE EKF," in *10th IFAC International Symposium on Dynamics and Control of Process Systems*, pp. 75–80, 2013.

[237] S. Julier, "The scaled unscented transformation," in *Proceedings of the American Control Conference*, vol. 6, pp. 4555–4559, 2002.

[238] J. K. Uhlmann, "Simultaneous map building and localization for real time applications," *transfer thesis, University of Oxford, Oxford, UK*, 1994.

[239] S. Julier and J. Uhlmann, "A general method for approximating nonlinear transformations of probability distributions," tech. rep., University of Oxford, Oxford, UK, 1996.

[240] E. Çelik, E. Karaduman, and M. Bayram, "Numerical method to solve chemical differential-algebraic equations," *International Journal of Quantum Chemistry*, vol. 89, no. 5, pp. 447–451, 2002.

[241] M. Hosseini, "Adomian decomposition method for solution of nonlinear differential algebraic equations," *Applied Mathematics and Computation*, vol. 181, no. 2, pp. 1737–1744, 2006.

[242] F. Soltanian, S. Karbassi, and M. Hosseini, "Application of He's variational iteration method for solution of differential-algebraic equations," *Chaos, Solitons and Fractals*, vol. 41, no. 1, pp. 436–445, 2009.

[243] C.-S. Liu, "Solving nonlinear differential algebraic equations by an implicit $GL(n, \mathbb{R})$ lie-group method," *Journal of Applied Mathematics*, vol. 2013, ID 987905, 2013.

[244] H. Choi, *Adaptive Sampling and Forecasting with Mobile Sensor Networks*. PhD thesis, MIT, Cambridge, MA, USA, 2009.

[245] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics (SIAM), 2000.

[246] R. Sanner and J.-J. Slotine, "Gaussian networks for direct adaptive control," *IEEE Transactions on Neural Networks*, vol. 3, no. 6, pp. 837–863, 1992.

[247] D. P. Petersen and D. Middleton, "Sampling and reconstruction of wave-number-limited functions in N-dimensional euclidean spaces," *Information and Control*, vol. 5, no. 4, pp. 279 – 323, 1962.

[248] B. Bobrovsky and M. Zakai, "A lower bound on the estimation error for Markov processes," *IEEE Transactions on Automatic Control*, vol. 20, no. 6, pp. 785–788, 1975.

[249] P. Tichavsky, C. H. Muravchik, and A. Nehorai, "Posterior Cramér-Rao bounds for discrete-time nonlinear filtering," *IEEE Transactions on Signal Processing*, vol. 46, no. 5, pp. 1386–1396, 1998.

[250] P. Tichavsky, "Posterior Cramér-Rao bound for adaptive harmonic retrieval," *IEEE Transactions on Signal Processing*, vol. 43, no. 5, pp. 1299–1302, 1995.

[251] M. J. Churchfield, P. J. Moriarty, G. Vijayakumar, and J. Brasseur, "Wind energy-related atmospheric boundary-layer large-eddy simulation using openFOAM," in *Proceedings of the 19th Symposium on Boundary Layers and Turbulence*, 2010.

[252] M. J. Churchfield, S. Lee, J. Michalakes, and P. J. Moriarty, "A numerical study of the effects of atmospheric and wake turbulence on wind turbine dynamics," *Journal of Turbulence*, vol. 13, no. 14, pp. 1–32, 2012.

[253] M. Churchfield, S. Lee, P. Moriarty, L. Martinez, S. Leonardi, G. Vijayakumar, and J. Brasseur, "A large-eddy simulation of wind-plant aerodynamics," in *Proceedings of the 50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, 2012.

[254] K. K. Kottakki, S. Bhartiya, and M. Bhushan, "State estimation of nonlinear dynamical systems using nonlinear update based unscented Gaussian sum filter," *Journal of Process Control*, vol. 24, no. 9, pp. 1425–1443, 2014.

[255] Y. Guo and B. Huang, "State estimation incorporating infrequent, delayed and integral measurements," *Automatica*, vol. 58, pp. 32–38, 2015.

[256] N. Wang, K. E. Johnson, and A. D. Wright, "Comparison of strategies for enhancing energy capture and reducing loads using LIDAR and feedforward control," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1129–1142, 2013.

[257] Z. Ma, Z. Yan, M. L. Shaltout, and D. Chen, "Optimal real-time control of wind turbine during partial load operation," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 6, pp. 2216–2226, 2015.

[258] T. Mikkelsen, N. Angelou, K. Hansen, M. Sjöholm, M. Harris, C. Slinger, P. Hadley, R. Scullion, G. Ellis, and G. Vives, "A spinner-integrated wind lidar for enhanced wind turbine control," *Wind Energy*, vol. 16, no. 4, pp. 625–643, 2013.

[259] L. Y. Pao and K. Johnson, "A tutorial on the dynamics and control of wind turbines and wind farms," in *Proceedings of the American Control Conference*, pp. 2076–2089, 2009.

[260] G. I. Taylor, "The spectrum of turbulence," *Proceedings of the Royal Society of London. Series A - Mathematical and Physical Sciences*, vol. 164, no. 919, pp. 476–490, 1938.

[261] S. Raach, D. Schlipf, F. Haizmann, and P. W. Cheng, "Three dimensional dynamic model based wind field reconstruction from Lidar data," *Journal of Physics: Conference Series*, vol. 524, no. 1, p. 012005, 2014.

[262] B. Galletti, C. H. Bruneau, L. Zannetti, and A. Iollo, "Low-order modelling of laminar flow regimes past a confined square cylinder," *Journal of Fluid Mechanics*, vol. 503, pp. 161–170, 2004.

[263] B. R. Noack, P. Papas, and P. A. Monkewitz, "The need for a pressure-term representation in empirical Galerkin models of incompressible shear flows," *Journal of Fluid Mechanics*, vol. 523, pp. 339–365, 2005.

[264] Y. Koveos and A. Tzes, "Resonant fluid actuator: Modeling, identification, and control," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 3, pp. 852–860, 2013.

[265] K. E. Brenan, S. L. Campbell, and L. R. Petzold, *Numerical Solution of Initial-Value Problems in Differential-Algebraic Equations*. Philadelphia, PA, USA: SIAM, 1996.

[266] J. Ainslie, "Calculating the flowfield in the wake of wind turbines," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 27, no. 1-3, pp. 213–224, 1988.

[267] G. Strang, *Computational Science and Engineering*. Wellesley, MA, USA: Wellesley-Cambridge Press, 2007.

[268] A. Chorin and J. Marsden, *A mathematical introduction to fluid mechanics, Third edition*. New York, NY, USA: Springer-Verlag, 2000.

[269] J. Weickert, *Applications of the theory of differential-algebraic equations to partial differential equations of fluid dynamics*. PhD thesis, Technische Universität Chemnitz-Zwickau, Chemnitz, Germany, 1997.

[270] I. Gohberg and G. Heinig, "Inversion of finite Toeplitz matrices consisting of elements of a noncommutative algebra," *Rev. Roumaine Math. Pures Appl.*, vol. 19, no. 5, pp. 623–663, 1974.

[271] A. Ben-Artzi and T. Shalom, "On inversion of block toeplitz matrices," *Integral Equations and Operator Theory*, vol. 8, no. 6, pp. 751–779, 1985.

[272] D. Rempfer, "On boundary conditions for incompressible Navier-Stokes problems," *Applied Mechanics Reviews*, vol. 59, no. 3, pp. 107–125, 2006.

[273] J. C. Butcher, *Numerical Methods for Ordinary Differential Equations, Second Edition*. Chichester, West Sussex, England: John Wiley and Sons Ltd., 2008.

[274] A. E. Gelfand, S. E. Hills, A. Racine-Poon, and A. F. M. Smith, "Illustration of Bayesian inference in normal data models using Gibbs sampling," *Journal of the American Statistical Association*, vol. 85, no. 412, pp. 972–985, 1990.

[275] W. K. Hastings, "Monte Carlo sampling methods using Markov chains and their applications," *Biometrika*, vol. 57, no. 1, pp. 97–109, 1970.

[276] E. Walter, *Identifiability of State Space Models with Applications to Transformation Systems*. Berlin, Germany: Springer-Verlag, 1982.

[277] L. Ljung and T. Glad, "On global identifiability for arbitrary model parametrizations," *Automatica*, vol. 30, no. 2, pp. 265–276, 1994.

[278] S. K. Sahu, A. E. Gelfand, and D. M. Holland, "Spatio-temporal modeling of fine particulate matter," *Journal of Agricultural, Biological, and Environmental Statistics*, vol. 11, no. 1, pp. 61–86, 2006.

[279] H.-C. Huang and N.-J. Hsu, "Modeling transport effects on ground-level ozone using a non-stationary space-time model," *Environmetrics*, vol. 15, no. 3, pp. 251–268, 2004.

[280] M. Gerdin and J. Sjöberg, "Nonlinear stochastic differential-algebraic equations with application to particle filtering," in *Proceedings of the 45th IEEE Conference on Decision and Control*, pp. 6630–6635, 2006.

[281] P. Kunkel and V. Mehrmann, "Analysis of over- and underdetermined nonlinear differential-algebraic systems with application to nonlinear control problems," *Mathematics of Control, Signals and Systems*, vol. 14, no. 3, pp. 233–256, 2001.