Enhancements of online Bayesian filtering algorithms for efficient monitoring and improved uncertainty quantification in complex nonlinear dynamical systems

Audrey Olivier

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

Columbia University
2017

This page is intentionally left blank.

This page is intentionally left blank.

ABSTRACT

Enhancements of online Bayesian filtering algorithms for efficient monitoring and improved uncertainty quantification in complex nonlinear dynamical systems

Audrey Olivier

Recent years have seen a concurrent development of new sensor technologies and high-fidelity modeling capabilities. At the junction of these two topics lies an interesting opportunity for real-time system monitoring and damage assessment of structures. During monitoring, measurements from a structure are used to learn the parameters and equations characterizing a physics-based model of the system; thus enabling damage identification. Since monitored quantities are physical, these methods offer precious insight into the damage state of the structure (localization, type of damage and its extent). Furthermore, one obtains a model of the structure in its current condition, an essential element in predicting the future behavior of the structure and enabling adequate decision-making procedures.

This dissertation focuses more specifically on solving some of the challenges associated with the use of online Bayesian learning algorithms, also called sequential filtering algorithms, for damage detection and characterization in nonlinear structural systems. A major challenge regarding online Bayesian filtering algorithms lies in achieving good accuracy for large dimensional systems and complex nonlinear non-Gaussian systems, where non-Gaussianity can arise for instance in systems which are not globally identifiable. In the first part of this dissertation, we show that one can derive algorithmic enhancements of filtering techniques, mainly based on innovative ways to reduce the dimensionality of the problem at hand, and thus obtain a good trade-off between accuracy and computational complexity of the learning algorithms. For instance, for particle filtering techniques (sampling-based algorithms) subjected to the so-called curse of dimensionality, the concept of Rao-Blackwellisation can be used to greatly reduce the dimension of the sampling

space. On the other hand, one can also build upon nonlinear Kalman filtering techniques, which are very computationally efficient, and expand their capabilities to non-Gaussian distributions.

Another challenge associated with structural health monitoring is the amount of uncertainties and variabilities inherently present in the system, measurements and/or inputs. The second part of this dissertation aims at demonstrating that online Bayesian filtering algorithms are very well-suited for SHM applications due to their ability to accurately quantify and take into account these uncertainties in the learning process. First, these algorithms are well-suited to address ill-conditioned problems, where not all parameters can be learnt from the available noisy data, a problem which frequently arises when considering large dimensional nonlinear systems. Then, in the case of unknown stochastic inputs, a method is derived to take into account in this sequential filtering framework unmeasured stationary excitations whose spectral properties are known but uncertain.

Contents

This page is intentionally left blank.

List of Tables

This page is intentionally left blank.

List of Figures

This page is intentionally left blank.

Acknowledgements

I would like to express my sincere appreciation to all the people who have made this work possible. My doctoral studies have been a great life experience and a fulfilling academic journey, and I am grateful to all the people around me who have helped make these four years such a pleasant episode of my life.

First and foremost I would like to thank my Ph.D. advisor Prof. Andrew Smyth, for his constructive academic advice, his continuous support over these four years and most importantly his precious mentorship. I also would like to thank Faculty members and colleagues from whom I have learnt so much during my studies at Columbia through many lectures and talks. A special thanks to Faculty, colleagues and staff from the CEEM Department who are doing a great job at making this Department a very enjoyable secondary home for their Ph.D. students. I also want to acknowledge the support of the U.S. National Science Foundation, which partially supported my research under grants No. CMMI-1100321 and CMMI-1563364.

I also would like to thank Faculty from École Centrale de Nantes, who have prepared me extremely well for my subsequent graduate studies. More specifically, I would like to thank Prof. Frederic Dorel, who first mentioned to me the possibility of studying at Columbia University and has been very supportive during this little adventure.

Of course, none of this would have been possible without the constant support of friends and family. My parents have been extremely supportive of my choices, even though I know that my

decision to fly so far away from home was not easy on them. They have passed on to me their strength, their open mindedness, and their passion for teaching and mentoring, and for this I will forever be grateful. Merci les loulous! I also would like to thank my brothers and friends back home, who have not given up on me despite my poor handling of these long-distance relationships.

I also would like to express my gratitude to all the friends and colleagues I have met here in New York, through Columbia, International House or random encounters, who have helped me feel at home in 10027. Finally, thank you to Giuseppe, who started this adventure with me five years ago, and who has continuously challenged me to give my best, while also reminding me that life is not all about a degree. Grazie di tutto!

Introduction

In the last decade, structural health monitoring (SHM) has become an important area of research as it provides the means for improved and responsible management of our aging infrastructure, leading to potential life-safety and economic benefits. In March 2017 the American Society of Civil Engineers (ASCE) released the 2017 Infrastructure Report Card [American Society of Civil Engineers, 2017], grading 16 categories of infrastructure with an average grade of D+, highlighting the issues arising with our aging infrastructure and need for better maintenance. SHM provides the potential to detect and localize potential damage before it reaches a critical level, thus avoiding dramatic collapse. Furthermore it provides some information about the remaining lifetime of a structure, thus helping the owner in his decision making process regarding maintenance and repair.

In [Farrar and Worden, 2007], damage is defined as "changes to the material and/or geometric properties of [a system], including changes to the boundary conditions and system connectivity, which adversely affect the system's performance". Since damage is a change in the properties of the system, this notion only makes sense if one is comparing two different states of the structure: the initial state (usually assumed healthy) and an unknown state, which one wants to identify as either healthy or damaged.

In practice, damage identification has been performed for a long time using either visual inspection, which is subjective and is not able to detect all types of damage; or by Non-Destructive Evaluation (NDE), which is costly, time-consuming and often provides localized information, thus

requires a priori knowledge of the damaged region. Lately, research efforts have been directed towards vibration-based methods, which use the fact that damage will affect the dynamic response of a structure. Thus by measuring and analyzing the response of the structure at different locations one should be able to detect and localize damage.

Structural Health Monitoring applications face many well-known challenges. For instance, damage is often a local phenomenon that might not significantly affect some of the global responses typically extracted from the data, thus creating a need to exploit measurements from various types of sensors along the structure of interest (data fusion). Also, consideration of nonlinear effects in SHM is crucial since 1) damage can cause a structure that initially behaved linearly to exhibit non-linear behavior (e.g., opening/closing of a crack in a beam), and 2) many structures actually behave non-linearly even within their undamaged state. For instance, a structure might be designed to respond nonlinearly to high intensity excitations (e.g., earthquakes), events that are of particular interest for SHM purposes, and thus accurate damage identification methods must be used in order to avoid misinterpreting this healthy nonlinear behavior as damage. Finally, environmental and operational conditions (temperature, humidity etc) are often variable and complex to quantify, which creates many uncertainties in the models to be considered; measurements are usually corrupted with noise, and many excitations (wind, seismic ground motions, traffic...) are random and sometimes hard to measure.

The SHM process can be decomposed into four steps, [Farrar and Worden, 2007]:

- **Operational evaluation**: provide a justification for performing SHM on a given system, determine the operational and environmental conditions in which the system operates...
- **Data acquisition, normalization and cleansing**: decision on sensor number and placement, acquire data and process the data...
- **Feature extraction and information condensation**: extract from the data some features that are sensitive to damage (for instance, identify parameters of the model, identify fre-

quencies and mode shapes for linear systems, etc)

- **Statistical model development for feature discrimination**: quantify the damage state of the structure.

The feature extraction part of the aforementioned process, which relates to detecting the presence of damage through analysis of the damage sensitive features, has received a lot of attention in the literature. However, in order to provide the owner of a structure with valuable information and enable educated decision making procedures, the full damage state of the structure should be evaluated. This can be performed by answering the following questions (hierarchy of damage, [Rytter, 1993]).

1. Existence: Is the system damaged?

2. Location: Where is the damage?

3. Type: What kind of damage is present?

4. Extent: How severe is it?

5. Prognosis: How much useful life remains?

One possible way to look at SHM is to define it as a statistical pattern recognition problem [Farrar and Worden, 2007]. This idea has received tremendous attention in the literature and many statistical pattern recognition schemes have been proposed to detect and even localize damage, both for linear and nonlinear structures, see e.g. [Balsamo et al., 2014, Bornn et al., 2010, Nair et al., 2006]. These algorithms are very well-suited to classify healthy vs. damaged conditions of a structure as well as locate the damage, while taking into account operational and environmental variabilities. Typically, algorithms along these lines are used in an unsupervised learning mode; they make use of a data-based (possibly non-physical) model to represent the behavior of the system in its healthy state (autoregressive-moving-average ARMA models, support vector machines...), and extract data features that would be sensitive to the presence of damage. Then data

coming from a structure in an unknown condition can be classified as healthy vs. damaged using statistical classification tools, such as outlier rejection algorithms. This type of method requires extensive training in order to learn the parameters of a statistical model that is representative of the healthy structure and encompasses environmental variabilities. This requires having access to large amounts of data from the healthy structure, which might not be trivial for new structures or structures not often excited. More importantly, non-physical data-based models provide limited insight into the actual behavior of the system in its current, potentially damaged, condition (type/extent of the damage). To get an idea of the type and extent of the damage, as well as information on future behavior of the structure (prognosis step), coupling measured data with analytical or physics-based models can prove more useful.

In the case of linear structures, model-based methods typically consist of estimating natural frequencies and mode shapes of vibration using measured data (e.g., modal identification through Frequency Domain Decomposition [Brincker et al., 2001], Observer Kalman filter Identification coupled with Eigensystem Realization Algorithm [Mosquera et al., 2012], state-space model estimation via Subspace Identification [Van Overschee and De Moor, 2006]). The physical parameters (mass M and stiffness K matrices for instance) can also be recovered from data (see e.g. [Mukhopadhyay et al., 2014]), or an FEM model can be updated to fit the measured data, see e.g. [Jang and Smyth, 2017], thus providing good insight into the damage state of the system, as well as a model of the structure in its current condition, essential for prognosis purposes.

Regarding nonlinear systems, [Worden et al., 2008] provides an early review of some of the methods that can be used to detect damage in systems that behave linearly in their healthy state and for which damage causes nonlinear responses. One of the methods considered is the system identification approach, which is the focus of this dissertation, where the basic idea is to obtain an estimate of the equations of motion of the system in its unknown condition using measured data. Again, this allows the user to build a model of the structure in its current unknown condition, thus

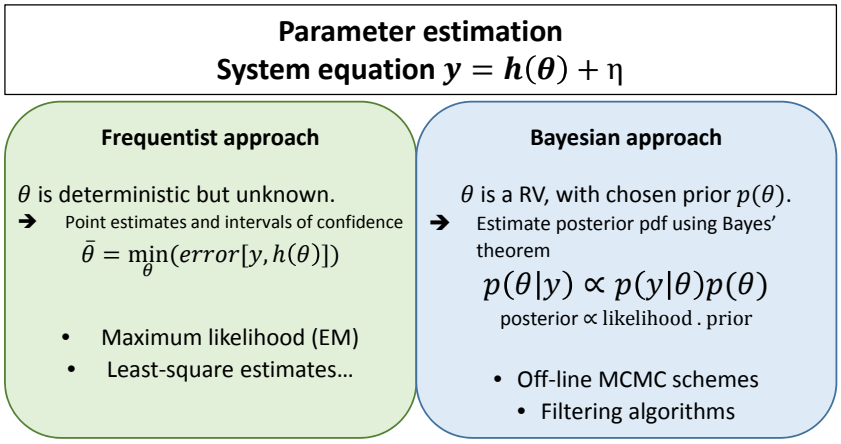


Figure 1: Frequentist vs. Bayesian approaches to parameter estimation

providing deep knowledge about the damage state and allowing prediction of future behavior. As previously mentioned, one of the major challenges of SHM is the presence of many uncertainties in the model, its inputs as well as the measurements. Thus it is imperative to adopt system identification algorithms that can quantify and accurately take into account all these uncertainties.

For this reason, the Bayesian framework has been recognized to be particularly well-suited for SHM and damage detection purposes, see e.g. [Green and Worden, 2015]. Bayesian techniques (see Fig. 1) consider parameters θ (and states) as random variables (RVs) that are assigned a prior probability density function (pdf) that reflects knowledge of this parameter prior to observing data; this prior pdf can be chosen to be quite uninformative if little information is accessible prior to monitoring. Then measurements y are used to update this prior pdf using Bayes' theorem, and thus yield the posterior pdf of the parameters knowing the measurements:

$$\overbrace{p(\theta|y)}^{\text{posterior}} = \frac{\overbrace{p(y|\theta)}^{\text{likelihood}}\overbrace{p(\theta)}^{\text{prior}}}{\underbrace{p(y)}_{\text{evidence}}} = \frac{p(y|\theta)p(\theta)}{\int p(y|\theta)p(\theta)d\theta}$$

Uncertainties are inherently taken into account in this framework when computing the likelihood function $p(y|\theta)$.

A crucial advantage of the Bayesian framework lies in the fact that it is able to tackle ill-

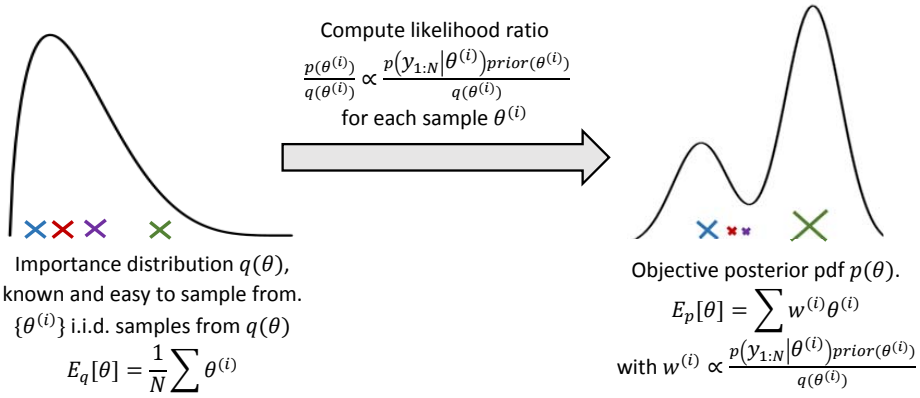


Figure 2: The principle of importance sampling, used in MCMC and particle filtering schemes: independent and identically distributed (i.i.d.) samples are drawn from the chosen importance distribution q and weighted proportionally to their likelihood ratio, yielding a set of weighted samples from which one can compute statistics of the objective distribution p.

conditioned problems where some or all parameters cannot be uniquely identified based on available measurements [Muto and Beck, 2008], a topic that is discussed in chapter 5 of this dissertation. Lately, research efforts have been mostly directed towards Markov Chain Monte Carlo (MCMC) techniques, which make use of the importance sampling principle (Fig. 2) to efficiently explore the parameter space (see e.g. [Smith, 2014] for an introduction to frequentist vs. Bayesian approaches and MCMC techniques). MCMC techniques build Markov chains whose stationary distribution (also called steady-state distribution) is the posterior pdf, thus sampling from the chain after it has converged to its stationary distribution yields an estimate of the target posterior. MCMC techniques are off-line methods; they necessitate running the entire forward problem with different values of the parameter vector θ, each of them being selected based on the previous value. Furthermore, they are quite computationally expensive for medium to large dimensional problems. Ongoing research focuses on deriving novel algorithms that more efficiently explore the parameter space (Data Annealing principle, see [Green, 2015], sampling from a sequence of intermediate distributions that converge to the posterior, see e.g. [Beck and Au, 2002, Beck and Zuev, 2013, Ching and Chen, 2007]).

This thesis focuses instead on online Bayesian inference algorithms, which can be used to monitor both the dynamic states x^{dyn} (displacements, velocities...) and parameters θ of the model (damping, stiffness...), using a physics-based state-space representation of the system and a time-series of measurements $y_{1:N}$. Monitoring physical quantities provides valuable insight into the full damage state of the structure, and possibly a model of the structure in its damaged condition - essential for prognosis. A small multi-degree-of-freedom (MDOF) example on simulated data is provided for demonstration purposes in chapter 1, detailing how online Bayesian filtering algorithms can be used to identify and characterize damage and how probabilistic results can be further utilized in the decision-making procedure.

Online Bayesian inference algorithms for nonlinear systems, also called nonlinear filtering algorithms, can be divided into two categories: particle filtering schemes (PF) and nonlinear Kalman filter type algorithms, such as the widely used unscented Kalman filter (UKF). The latter is computationally efficient, a crucial advantage for real-time monitoring applications, however it is based on a Gaussianity assumption of the posterior pdfs and an approximate moment propagation scheme, which may lead to inaccurate results for certain systems. The PF on the other hand is a sampling-weighting based algorithm, which yields the true posterior pdf, even for highly nonlinear non-Gaussian systems. However, along with other sampling techniques, the PF suffers from the so-called curse of dimensionality, meaning that the number of particles required for the approximation, and thus its overall computational complexity, increases exponentially with the dimension of the system studied. Thus, one of the main challenges regarding filtering algorithms is to achieve a satisfactory trade-off between accuracy of the posterior pdfs and acceptable computational complexity. This trade-off can be achieved using algorithmic enhancements of the PFs and nonlinear Kalman filters, which is the subject of part I of this dissertation.

First, with regard to the PF, its basic formulation is reviewed in depth in chapter 2, and a novel algorithm based on the concept of Rao-Blackwellisation is introduced in order to greatly

reduce the dimension in which particle approximation is performed, thus mitigating the curse of dimensionality issue (the state vector $x \in \mathbb{R}^{n_x}$ is judiciously partitioned in two subvectors $a \in \mathbb{R}^{n_a}$ and $b \in \mathbb{R}^{n_b}$ in order to reduce the computational time (CT) from $CT \propto exp(n_x) = exp(n_a + n_b)$ to $CT \propto n_a \cdot exp(n_b)$). Another way to increase accuracy and efficiency of online Bayesian algorithms for large dimensional systems is to build upon nonlinear Kalman filtering techniques, which are computationally efficient, and expand their capabilities to non-Gaussian pdfs. More specifically, after an in depth review of nonlinear Kalman filtering schemes in chapter 3, we show how one can couple non-Gaussian filters and higher order moment propagation schemes to enhance the capabilities of nonlinear Kalman filters to different underlying probability distributions. Then in chapter 4, the capabilities of nonlinear Kalman filters for parameter estimation in larger dimensional systems, represented by FE models with tens to hundreds of DOFs, are demonstrated. A novel algorithmic enhancement, based on the principle of marginalization applied to the unscented transform, is derived in order to render the UKF more robust and less computationally expensive when used to perform parameter identification within the finite element analysis framework.

As previously mentioned, another compelling property of the Bayesian framework is its ability to take into account uncertainties in the system, measurements and inputs and propagate them to the filtering posterior pdfs of the quantities of interest. Studying this treatment of uncertainties with online filtering algorithms is the focus of the second part of this dissertation. In chapter 5, the behavior of these algorithms is studied with respect to systems that exhibit challenging identifiability properties, e.g., locally identifiable systems for which the inverse problems possesses several solutions, leading multi-modal posterior pdfs, or systems that are unidentifaible due to a large noise to signal ratio, a topic of high relevance for monitoring of nonlinear systems whose nonlinearities might not always be fully excited during small intensity events. Another advantage of the state-space Bayesian framework used in online filtering is that one can easily quantify separately uncertainties due to model vs. measurement errors, as well as random inputs that might be

hard to measure at each degree-of-freedom (DOF) of a structure. In chapter 6, a method is presented to take into account unmeasured stationary inputs in this framework, with a focus on wind excitations.

Most of the findings reported in this thesis have already been presented in peer-reviewed journals, [Olivier and Smyth, 2017a,b,c] are published (or accepted for publication) and [Olivier and Smyth, under review] is currently under review. The present dissertation is structured as follows. The formulation of Bayesian sequential filtering is first presented in chapter 1. Then we introduce our work on algorithmic enhancements of particle filtering schemes (chapter 2) and nonlinear Kalman filters (chapters 3 and 4) and demonstrate the superiority of these novel formulations on both mathematical benchmark problems and structural systems. In the second part of the dissertation, the problem of identifiability for parameter estimation is studied, and some of the novel algorithms presented in Part I are used to accurately and efficiently estimate parameters for systems that exhibit challenging identifiability properties. Finally in the last chapter, the issue of unmeasured inputs is tackled for stationary stochastic inputs whose spectral properties are known but also uncertain.

This page is intentionally left blank.

Part I

Trade-off between accuracy and computational requirement of online filtering algorithms

Chapter 1

Bayesian filtering for damage identification: formulation and case study

1.1 Formulation of Bayesian filtering for dynamical systems

1.1.1 System equations

Many dynamical models, including structural systems, can be written as state-space models and discretized in time to follow a process equation of the form $x_{k+1}^{dyn} = f(x_k^{dyn}, \theta, e_k)$, where x^{dyn} represent the dynamic states (e.g. displacements, velocities), θ the vector of static parameters (e.g. stiffness, damping) that characterize the system and e_k a known forcing function. In Bayesian inference, one wants to estimate the states x^{dyn} and parameters θ from a series of measurements $y_{1:N}$, which can be contaminated by noise. Thus, the full system equations can be written as:

$$x_k^{dyn} = f\left(x_{k-1}^{dyn}, \theta, e_{k-1}\right) + v_{k-1} \text{ (propagation equation)} \tag{1.1a}$$

$$y_k = h(x_k^{dyn}, \theta) + \eta_k \text{ (measurement equation)} \tag{1.1b}$$

As previously explained, taking into account uncertainties in the inputs, measurements and/or model is crucial for SHM applications. In the framework described above, uncertainties are introduced through the terms v_{k-1} and η_k, called system and measurement noise respectively. In order to derive a sequential filtering formulation, these noise terms are assumed to follow certain properties:

$$E[v_k] = 0, \quad E[\eta_k] = 0 \tag{1.2a}$$

$$E[v_k v_l^T] = Q\delta_{kl}, \quad E[\eta_k \eta_l^T] = R\delta_{kl}, \quad E[v_{k-1,k}\eta_k^T] = 0 \tag{1.2b}$$

i.e., the process and measurement noise are white, zero-mean and uncorrelated to each other. Very often, the noise terms are assumed to be additive, as shown in Eq. (1.1), however, extension of filtering methods to non-additive noise can be achieved by adding the noise terms in the state vector and considering f and h as functions of both the states and the noise. Throughout this thesis, the covariance terms Q and R will be assumed known. However, the parameters of these noise terms could also be learnt using Bayesian inference (see e.g. [Kontoroupi and Smyth, 2016] for learning parameters of additive noise terms with the UKF and [Özkan et al., 2013] for formulation with the PF).

Finally, up until chapter 6, the excitation time series $e_{1:k}$ will also be assumed known (or measured) at each degree-of-freedom (DOF) of the system. Relaxation of this strong assumption will be the focus of chapter 6. Thus in the current formulation, inputs can be introduced in the formulation as either 1) known time series $e_{1:k}$ or 2) white noise zero-mean inputs with known covariance (process noise term).

1.1.2 Online learning of static parameters

To perform online estimation of the full posterior pdf (i.e., combined inference of dynamic states $x_{0:k}^{dyn}$ and static parameters θ), several methods can be used. A very common approach is to perform

joint state/parameter filtering, that is, the state vector is augmented with the static parameters that are assigned the propagation equation $\theta_k = \theta_{k-1}$. The state vector then becomes $x_k = \begin{Bmatrix} x_k^{dyn} \\ \theta \end{Bmatrix}$, and at each time step the full posterior pdf $p\left(x_k^{dyn}, \theta | y_{1:k}\right) = p\left(x_k | y_{1:k}\right)$ is inferred. This method is used in the remainder of the dissertation.

Another approach commonly used consists in separating estimation of the dynamic states and identification of the parameters. In [Wan and van der Merwe, 2000], an on-line dual approach is presented, where two filtering algorithms (UKFs) are run in parallel, one for estimation of the states, the other for estimation of the parameters, and both make use of the current estimates from the concurrent filter to propagate to the next time step. Other dual approaches can also be derived by combining a state filtering scheme with any parameter learning algorithm. For instance, in [Poyiadjis et al., 2006] on-line estimation of both states and parameters is performed by combining a particle filter for state estimation and an on-line maximum likelihood estimation procedure for the parameters, while in [Lindsten, 2013], an off-line procedure is presented, combining a particle smoother for the states with an expectation-maximization (EM) algorithm for the parameters.

1.1.3 Formulation of sequential filtering algorithms

In this joint state/parameter filtering framework, i.e., $x_k = \begin{Bmatrix} x_k^{dyn} \\ \theta \end{Bmatrix}$, the system can also be described in a probabilistic format as a prior $p(x_0) = p(x_0^{dyn}|\theta)p(\theta)$ and a sequence of conditional distributions:

- transition density function $p\left(x_k | x_{k-1}\right)$
- likelihood $p\left(y_k | x_k\right)$

The system described by Eq. (1.1) with noise properties as in Eqs. (1.2) is Markovian (see representation in Fig. 1.1), which means that x_k given x_{k-1} is independent of other past states and past

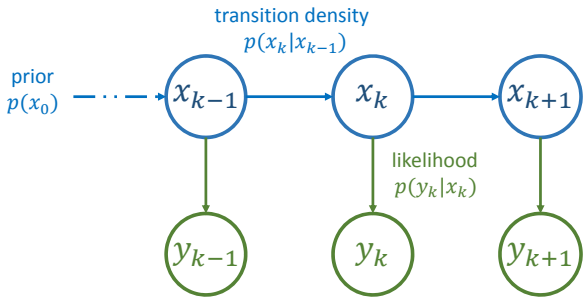


Figure 1.1: Markov chain

observations, and y_k given x_k is also independent of past states and observations, i.e.:

$$p\left(x_k|x_{0:k-1}, y_{1:k-1}\right) = p\left(x_k|x_{k-1}\right) \tag{1.3a}$$

$$p\left(y_k|x_{0:k}, y_{1:k-1}\right) = p\left(y_k|x_k\right) \tag{1.3b}$$

Using these Markovian properties of the system, one can derive a sequential filtering procedure to infer at each time step k the filtering posterior pdf of the states knowing all the past observations, i.e. $p\left(x_k|y_{1:k}\right)$. The Bayesian filtering formulation can be written at each time step k as a two-stage approach:

- Propagation step (Chapman-Kolmogorov equation)

$$p\left(x_k|y_{1:k-1}\right) = \int p\left(x_k|x_{k-1}\right) p\left(x_{k-1}|y_{1:k-1}\right) dx_{k-1} \tag{1.4a}$$

- Update step: make use of the new measurement y_k and Bayes' theorem to derive the posterior distribution

$$p\left(x_k|y_{1:k}\right) = \frac{p\left(y_k|x_k\right) p\left(x_k|y_{1:k-1}\right)}{p\left(y_k|y_{1:k-1}\right)} \tag{1.4b}$$

When the propagation and measurement functions f and h (defined in Eq. 1.1) are both linear in the augmented state x, and both noise terms are Gaussian, posterior pdfs are exactly Gaussian and exact inference can be performed using the well-known Kalman filter (KF). However, when joint state-parameter estimation is performed, functions f, h will be at least quadratic in the aug-

mented state $x_k = \begin{Bmatrix} x_k^{dyn} \\ \theta \end{Bmatrix}$ (e.g., stiffness $\times$ displacement, or damping $\times$ velocity terms). Thus for the problems at hand, f, h will always be nonlinear functions, which renders the posterior pdfs intractable. Two main algorithms exist to estimate the posterior pdfs when functions f, h are nonlinear and noise terms are possibly non-Gaussian: particle filtering methods and nonlinear Kalman filters. In particle filtering (PF), pdfs are represented using a set of samples called particles, which are propagated forward in time and weighted according to their proximity to the measurements. The posterior pdf at time step k estimated using a particle filter is then written as:

$$p\left(x_k \mid y_{1:k}\right) \approx \sum_{i=1}^{n_p} w_k^{(i)} \delta\left(x_k - x_k^{(i)}\right) \tag{1.5}$$

where $\{x_k^{(i)}, w_k^{(i)}\}_{i=1:n_p}$ are the weighted particles, and δ the delta dirac function. The PF can thus handle highly nonlinear, non-Gaussian systems, as long as the number of particles n_p is large enough. As will be discussed in chapter 2, the number of particles should grow exponentially with the dimension of the problem, rendering the PF impractical for large dimensional problems unless algorithmic enhancements, such as Rao-Blackwellisation, are used.

On the other hand, nonlinear Kalman filters make a Gaussianity assumption on the posterior pdfs, thus only knowledge of their first and second order moments is required. This assumption greatly reduces computational requirements compared to a PF, but it might be inexact for some systems. An in-depth review of these algorithms and more particularly the unscented Kalman filter (UKF) is presented in chapter 3, along with algorithmic enhancements enabling 1) employment of non-Gaussian distributions, which are further proved to be very well-suited for structural systems that exhibit challenging identifiability properties, as discussed in chapter 5, and 2) application to relatively large dimensional systems, represented by finite element models with 10s to 100s of DOFs (chapter 4).

The present study focuses on the application of such algorithms to damage detection and characterization tasks for structural systems. Various examples of identification in structural systems

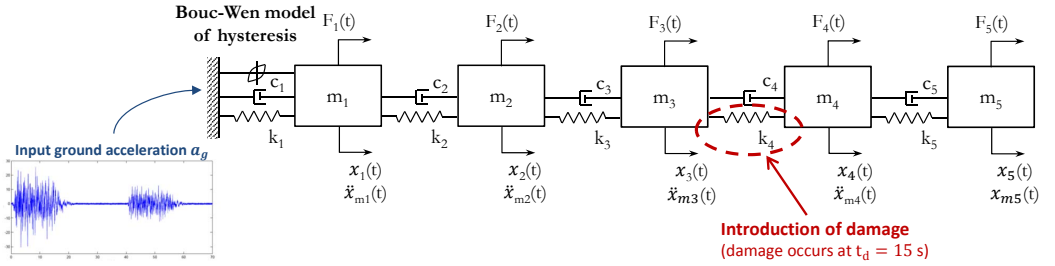
using these filtering algorithms can be found in the literature, see e.g. [Chatzi and Smyth, 2009] for identification of a 3 degree-of-freedom (DOF) highly nonlinear system using both UKF and PF schemes, [Chatzis et al., 2015a] for an experimental validation using the UKF, [Nasrellah and Manohar, 2010] for identification of parameters in vehicle-structure interaction problems using the PF, or [Astroza et al., 2015] for material parameter identification through a combination of finite element model and UKF. Applicability of such algorithms to the damage detection and characterization task is shown thereafter using a small case study on simulated data.

1.2 Online filtering for damage detection: motivating example

1.2.1 Presentation of the problem of interest

In this section we show how performing online filtering on a physics-based model of a system can be used for damage detection and characterization purposes. In the following simulated data is generated from a 5-DOF system, shown in Fig. 1.2. In its healthy condition, the structure already shows nonlinear behavior at the base, modeled by a Bouc-Wen model of hysteresis. The rest of the structure is linear when the structure is undamaged. We assume that we can measure the response of the structure at each DOF, thus the parameter vector is identifiable.

In SHM, one collects data from the structure in an unknown condition and tries to determine if this condition is healthy or damaged by comparing with the assumed/known healthy condition. Possible errors in damage detection are of two types (following definitions from [Farrar and Worden, 2007]): false-positive (detecting damage when the structure is actually undamaged) and false-negative (failing to detect damage when damage is truly present in the structure). Thus we will consider three unknown conditions for this structure, described in Fig. 1.2, and see if we can accurately classify healthy vs. damaged conditions.

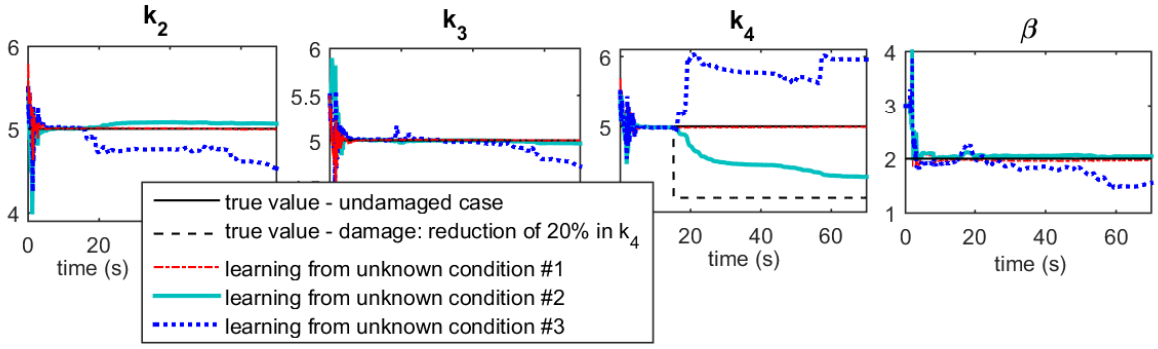


Known healthy condition: DOF 1 is highly nonlinear (Bouc-Wen model of hysteresis), all other DOFs are linear
Unknown condition #1: structure is healthy
Unknown condition #2: structure is damaged, decrease of 20% in k_4
Unknown condition #3: structure is damaged, restoring force F_{34} becomes nonlinear: Duffing oscillator
$F_{34} = k_{4,n}(x_4 - x_3) + \alpha(x_4 - x_3)^3$

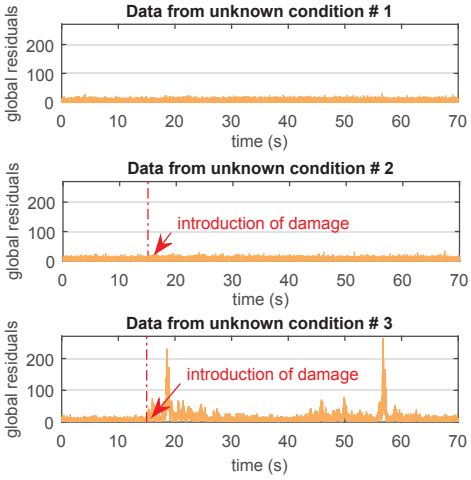
Figure 1.2: Motivating example: system of interest and damage conditions

1.2.2 Use of filtering algorithms for online damage identification and characterization

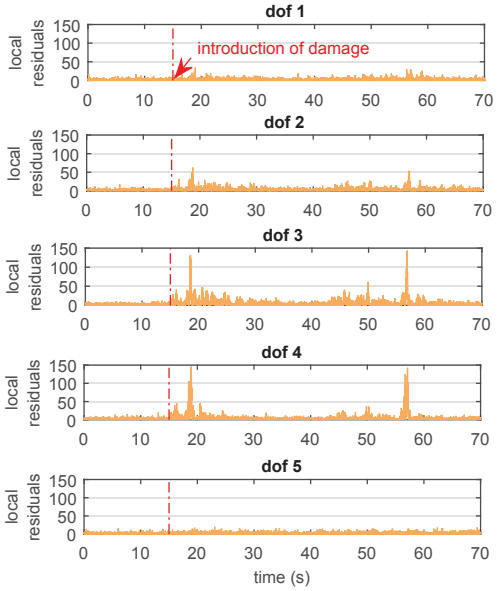
We show with this example how model-based methods can be used to detect, localize and identify damage. This type of method requires a priori knowledge of the structure itself in order to build a physics-based representation of the structure in its assumed healthy state and tune some of the model parameters. In the following case, we assume that reliable estimates of the floor masses were obtained through prior experience or identified from prior vibration data. The method then uses a filtering algorithm (in this particular example, the UKF) to monitor in real time the behavior of the states and unknown parameters (stiffness, damping...). If and when damage occurs, this will translate into a change in these parameters, which we hope to detect quickly by monitoring their behavior in an online fashion. Fig. 1.3a shows filtering of the parameters (i.e., estimation of $E[\theta|y_{1:k}]$ at each time step k) for the 3 unknown cases. For the unknown condition #1 (red dash-dot line on Fig. 1.3a), the algorithm converges to the true value (undamaged) of the parameters in few seconds and then remains at that value. This was expected since unknown condition #1 is in fact the healthy structure. For unknown condition #2 (green solid line), the parameters converge to their true (undamaged) value in few seconds. At $t = 15$ seconds, the value for k_4 starts to decrease,



(a) Convergence of some parameters (i.e., evolution in time of $E\left[\theta|y_{1:k}\right]$)



(b) Behavior of the global residuals for all 3 unknown conditions (see Eq. (1.6a))



(c) Behavior of the local residuals for each DOF (see Eq. (1.6b)), for unknown condition #3

Figure 1.3: Motivating example: online monitoring of structures using the UKF for joint estimation of parameters and states, both the behavior of the parameters θ and the residuals $y_{k,meas} - E\left[y_k|y_{1:k-1}\right]$ should be monitored to obtain a complete picture of the damage state (existence and location)

which was also expected since in unknown case #2, the stiffness k_4 is reduced of 20% at $t_d = 15s$. The remaining parameters remain unchanged. Thus, in this case, monitoring the behavior of the parameters enables one to detect, localize and identify damage.

For unknown condition #3 however, looking at Fig. 1.3a, one can detect that damage has

occurred at $t = 15s$ since the value of all the parameters starts to vary at that time, however localization of damage is impossible (by looking only at convergence of the parameters). In this case, damage cannot be represented only by a change in the parameters since a new nonlinearity appears in the system, a new model (different equations of motion) should be used to fit the data. The algorithm tries to find the parameters of the undamaged model that best match the measurements coming from this new structure/model. Even with this new set of parameters, the undamaged model will most likely not be able to accurately predict the behavior (and thus the measurements) of the damaged model. We can use this idea to better localize damage, by looking at the residuals $y_{k,meas} - E\left[y_k|y_{1:k-1}\right]$, where $y_{k,meas}$ are the actual measurements from the structure and $E\left[y_k|y_{1:k-1}\right]$ are outputs of the filtering algorithm.

If a model is able to accurately represent the behavior of the structure, the residuals should be small (close to 0). If damage occurs, one expects those residuals to increase. Fig. 1.3b shows, for the 3 unknown cases, a plot of the normalized residuals (normalization is required here for the purposes of comparison since both displacements and accelerations are measured), defined as:

$$\text{global residuals: } \left(y_{k,meas} - E\left[y_k|y_{1:k-1}\right]\right)^T P_k^{-1} \left(y_{k,meas} - E\left[y_k|y_{1:k-1}\right]\right), \tag{1.6a}$$

$$\text{local residuals for measurement } j\text{: } \frac{\left(y_{k,meas}^{[j]} - E\left[y_k^{[j]}|y_{1:k-1}\right]\right)^2}{var(y_k^{[j]}|y_{1:k-1})} \tag{1.6b}$$

where P_k is the covariance of the predicted measurement $y_k|y_{1:k-1}$. Fig. 1.3b shows a plot of this quantity for the 3 unknown cases. We can see that for unknown cases #1 and #2 the residuals are consistently small, meaning that our model is able to predict the behavior of the system. This was expected since the linear model is indeed the true model for these 2 cases, and the UKF converges to the right parameters (or close enough) in both cases. However, data from unknown condition #3 exhibits larger residuals after damage is introduced in the system, reaching values one order of magnitude higher than in the undamaged case. Looking at the residuals of each DOF separately (1.3c), one can see that the residuals are the highest for DOFs 3 and 4, which is to be expected

since the new nonlinearity (damage) appears in F_{34}, i.e., restoring force between DOFs 3 and 4.

Thus by monitoring both the behavior of the parameters and the residuals we were able to accurately detect and localize damage, in an on-line fashion, for the three unknown cases. Also, by performing damage detection using a physics-based model of the structure we obtained an insight on the type of damage introduced in the system (change in stiffness vs. creation of a new nonlinearity).

1.2.3 Introduction of the filtering estimates into the decision making process

Another advantage of this Bayesian approach is that results are obtained in a probabilistic format, and can thus be easily integrated into a fast decision making process. Several quantities can be defined to assess in a quantitative fashion the structure's integrity in real time. For some systems, estimates of the parameters can be directly used, for instance through the definition of a damage indicator D (for example, the model strain energy damage indicator for linear systems, [Shi et al., 1998]). In such cases, if one has access to the pdf of the parameter vector (or its moments), then one can compute the pdf or moments of this damage indicator at each time step k and thus make an informed decision, comparing values with available values in the literature for instance. For nonlinear systems however, it might be better to use performance-based methods to assess the structure's integrity. For instance, when designing for seismic events, thresholds are set on maximum interstory drifts and acceleration. One could use similar techniques to check in an online fashion if these quantities of interest reach critical values once the structure is damaged. Again if the pdfs of the states are known, one could estimate the probability that the quantities of interest reach the critical level, i.e., probability that the structure is damaged, in the sense that it does not meet its design criteria any more.

Knowing the posterior pdfs of the parameters, one can also predict future behavior and perform further analysis of the structure in its damaged condition (prognosis step) in an off-line fashion,

using uncertainty propagation schemes to take into account uncertainties in the parameters. In some cases (creation of a new nonlinearity for instance, unknown condition #3 in the previous example), additional work would be needed to identify the form of the equations representing the damaged structure. Selection of a model can be performed for instance using Bayesian model assessment, see for instance early work on the subject from Beck et. al [Beck and Yuen, 2004] and [Muto and Beck, 2008], and [Kontoroupi and Smyth, 2017] for an online implementation using the UKF.

1.3 Concluding remarks

With this small case study on simulated data we show how online filtering algorithms can be used to monitor the behavior of some physical quantities of interest in a structural system and identify potential damage. Making use of probabilistic methods such as Bayesian filtering yields results in a probabilistic format, which can further be integrated in decision making tools. For SHM applications, both accuracy and computational requirements of the system identification tools considered are relevant, since one might need to assess a structure's integrity rapidly after an event, of even in real time if possible. As previously mentioned, the first part of this dissertation focuses on the derivation of algorithmic enhancements of existing filtering algorithms to accurately estimate parameters and states in highly nonlinear problems, while reducing computational requirements for medium and large dimensional systems. The second part of the dissertation studies how these approaches can be used on challenging systems such as unidentifiable systems or systems subjected to unmeasured excitations. We start in the following chapter with a review of particle filtering and possible enhancements using the concept of Rao-Blackwellisation.

This page is intentionally left blank.

Chapter 2

Enhancement of particle filtering schemes through Rao-Blackwellisation

2.1 The particle filter: strengths and weaknesses

2.1.1 The particle filter

Particle filtering schemes use Monte Carlo approximations of the posterior pdf, i.e., it is represented by a finite number of weighted samples (particles) as:

$$p\left(x_k \mid y_{1:k}\right) \approx \sum_{i=1}^{n_p} w_k^{(i)} \delta\left(x_k - x_k^{(i)}\right) \tag{2.1}$$

and expectations of functions of x_k are computed by sample averages:

$$E\left[\varphi(x_k) \mid y_{1:k}\right] = \int \varphi(x_k) p\left(x_k | y_{1:k}\right) dx_k \tag{2.2a}$$

$$\approx \sum_{i=1}^{n_p} w_k^{(i)} \varphi\left(x_k^{(i)}\right) \tag{2.2b}$$

with n_p the number of particles used for the approximation. We will often be interested in the expectation of the state, thus $\varphi\left(x_k\right) = x_k$, and its covariance so $\varphi\left(x_k\right) = \left(x_k - E\left(x_k\right)\right)\left(x_k - E\left(x_k\right)\right)^T$.

However one can also compute higher order moments in the same fashion.

Only a limited review of the theory of particle filtering is provided here. For more details on the theory of particle filtering one can refer, for example, to [Cappé et al., 2007, Doucet and Johansen, 2011, Gustafsson, 2010]. The main steps of the *bootstrap particle filter*, the simplest version of the particle filter, can be summarized as in Fig. 2.1. Briefly: from the posterior pdf at time step $k-1$, $p(x_{k-1}|y_{1:k-1})$, one samples particles, propagates them to time step k via the propagation equation and then weights them according to their proximity to the actual measurements. However, as this is done sequentially over a large number of steps, one usually ends up with a set of particles among which only a few have significant weights (impoverishment). The worst case scenario is when only one particle has weight one, which is called collapse of the particle filter (the posterior pdf becomes a single dirac function). To overcome this issue one adds a resampling scheme to the algorithm, the goal being to duplicate particles with high weight while getting rid of particles with low weights. In this way at each time step the system is "reset", and one focuses on regions of high likelihood.

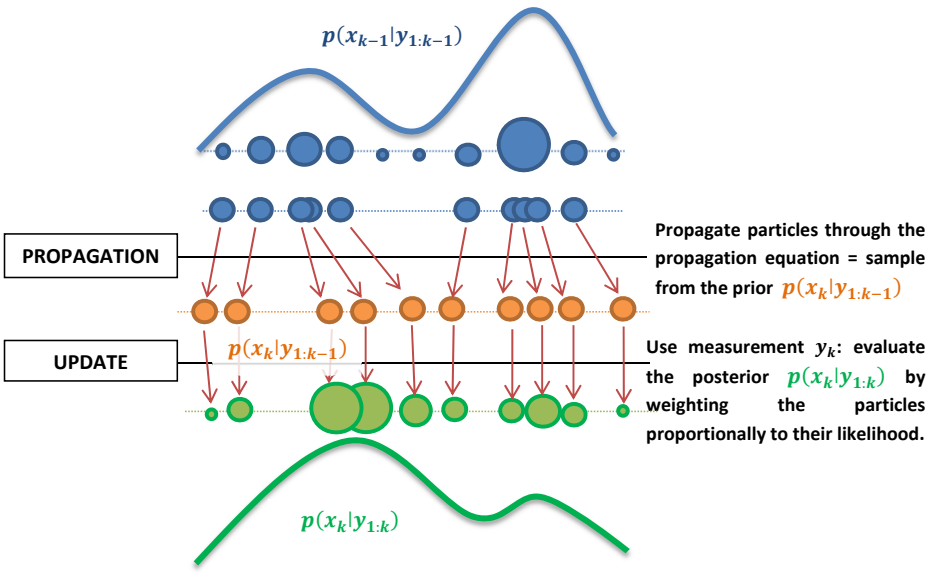


Figure 2.1: Two-stage approach of the bootstrap particle filter, for one time step k

In a more general context, the particle filter can also be cast in the Sequential Importance Sampling framework, as explained in [Doucet and Johansen, 2011]. Importance sampling relates to the fact that, as the posterior pdf at time step k is unknown, one instead samples particles from an importance distribution $\pi(x_k \mid x_{0:k-1}, y_{1:k})$ (sometimes also called proposal distribution) from which samples can be easily drawn. The discrepancy between the true posterior pdf and the importance distribution is taken into account by weighting the particles as $w_k^{(i)} = w_{k-1}^{(i)} \frac{p\left(x_k^{(i)} | x_{k-1}^{(i)}\right) p\left(y_k | x_k^{(i)}\right)}{\pi(x_k^{(i)} | x_{0:k-1}^{(i)}, y_{1:k})}$ (e.g. [Särkkä, 2013]). A resampling scheme is then performed to mitigate the sampling degeneracy issue, as previously mentioned. The algorithm of the PF, as seen from an importance sampling point of view, is presented in Alg. 1. In the *bootstrap particle filter* previously mentioned, the importance density function is chosen as the state-transition density

$$\pi\left(x_k^{(i)} \middle| x_{0:k-1}^{(i)}, y_{1:k}\right) = p\left(x_k^{(i)} \middle| x_{k-1}^{(i)}\right) \tag{2.3a}$$

from which samples are easily drawn. The recursive formula used to compute the weights is then greatly simplified:

$$w_k^{(i)} = w_{k-1}^{(i)} p\left(y_k \middle| x_k^{(i)}\right) \tag{2.3b}$$

In the remainder of the paper, we will always use this standard importance density function.

Convergence results [Crisan and Doucet, 2002] show that, under some assumptions regarding the resampling step, the bootstrap particle filter converges to the true posterior pdf when the number of particles goes to infinity. Also, it has been shown [Crisan and Doucet, 2002] that the error between the true and particle estimation of the pdf does not increase in time as long as the system is *quickly mixing*, which means, roughly, that the state vector at time step k is more or less independent from x_{k-l}, for a relatively small value of the lag l. When static parameters are added to the state vector, the system becomes slowly mixing ($\theta_{k+1} = \theta_k$), and the sample impoverishment issue is exacerbated. This is a major drawback of the PF for SHM purposes since many static parameters are added in the state vector to detect damage. Another issue that will arise when doing joint

Algorithm 1: Particle filter with resampling step (Sequential Importance Resampling).

Initialization. **for** $i = 1, \cdots, n_p$ **do**
  sample $x_0^{(i)} \sim p(x_0)$;
  assign initial importance weights $w_0^{(i)} = \frac{1}{n_p}$
end
for $k = 1, \cdots, N$ **do**
  for $i = 1, \cdots, n_p$ **do**
    Propagate: draw $\tilde{x}_k^{(i)} \sim \pi\left(\cdot | x_{0:k-1}^{(i)}, y_{1:k}\right)$;
    Compute weight: $\tilde{w}_k^{(i)} = w_{k-1}^{(i)} \frac{p\left(y_k | \tilde{x}_k^{(i)}\right) p\left(\tilde{x}_k^{(i)} | x_{k-1}^{(i)}\right)}{\pi\left(\tilde{x}_k^{(i)} | x_{0:k-1}^{(i)}, y_{1:k}\right)}$;
  end
  Normalize weights $w_k^{(i)} = \tilde{w}_k^{(i)} / \sum_{j=1}^{n_p} \tilde{w}_k^{(j)}$, $i = 1, \cdots, n_p$;
  Resampling. **if** $ESS = \frac{1}{\sum_{i=1}^{n_p} (w_k^{(i)})^2} < \frac{n_p}{2}$ **then**
    Resample (select n_p particles, each particle $x_{k-1}^{(i)}$ being drawn with probability $w_k^{(i)}$, then set $w_k^{(i)} = \frac{1}{n_p}$, $i = 1, \cdots, n_p$);
  else
    set $x_k^{(i)} = \tilde{x}_k^{(i)}$;
  end
  Compute moments of the state at time step k $E[\varphi(x_k)|y_{1:k}] = \sum_{i=1}^{n_p} w_k^{(i)} \varphi\left(\tilde{x}_k^{(i)}\right)$.
end

estimation is the dimension of the state vector. Indeed if many parameters need to be identified, the size of the state vector will quickly increase, rendering the use of the PF quite cumbersome.

2.1.2 The PF in high dimensional systems

Particle filters have so far been applied mostly to low-dimensional state-space models, as they rapidly tend to fail when the number of states increases. Many authors have examined in detail what obstacles arise when one tries to apply a PF to a very high dimensional problem.

[Bui Quang et al., 2010, Rebeschini and van Handel, 2015] examine the behavior of the mean square error (MSE) between the true value of integral Eq. (2.2a) and its filter approximation Eq.

(2.2b). More precisely it can be shown that the error associated with the weighting step of the particle filter is inversely proportional to the following quantity:

$$\int_{R^d} p(y_k \mid x_k)\, p(x_k \mid y_{1:k-1})\, dx_k \tag{2.4}$$

If the particles sampled from the prior $p(x_k \mid y_{1:k-1})$ fall in the tail of the likelihood $p(y_k \mid x_k)$, many weights will be zero and the particle filter will most likely degenerate and produce poor results.

On the other hand, [Bengtsson et al., 2008, Snyder, 2011, Snyder et al., 2008] examine more precisely the collapse issue (i.e., when only one particle gets all the weight), and its dependence on the dimension of the system. They show for a simple example that to avoid collapse, the number of particles should grow exponentially with the number of measurements, indirectly related to the dimension of the state vector.

Recall that the main objective here is to estimate at each time step k the posterior distribution $p(x_{0:k} | y_{1:k})$, where x_k is a state-vector of possibly very large size, using a weighted average over the so-called particles. It is well known that it is challenging to estimate a probability density function in high dimensional spaces from a finite number of data points: as explained in [Silverman, 1986], the number of required data points increases exponentially with the dimension. When the dimension of the space increases, the data becomes very sparse, which explains why the number of required data points necessary to estimate a density distribution increases exponentially and not linearly with the dimension of the space (curse of dimensionality). Numerically, the PF will collapse during the weighting step, because in high dimensions it is very probable that the prior and the posterior will be mutually singular, thus a particle sampled from the prior will have very low probability under the posterior distribution (it will be assigned a very low weight).

2.1.3 Literature review: increasing efficiency of the PF

Various methods can be used to improve efficiency of the particle filter and mitigate the degeneracy issue observed for medium and large dimensional systems. Improvements can usually be classified into two categories. The first type of algorithms provide improvements of the resampling step that aim at re-introducing diversity amongst the samples (e.g., adding an MCMC step, [Doucet et al., 2001], fit a density (kernel density, mixture of Gaussians) to the particle density estimate and sample from it, incorporate a mutation operator, [Chatzi and Smyth, 2013], a method particularly efficient when static parameters are introduced in the state vector). The second category of algorithms aims at increasing efficiency through a better choice of importance density $\pi(\cdot|x_{0:k}, y_{1:k})$ from which samples are drawn. Indeed, the bootstrap particle filter previously mentioned uses the transition density function as proposal, however particles sampled from the prior might have low probability under the posterior, thus leading to many particles having negligible weights. To overcome this issue, one may use a proposal distribution that already takes into account the future measurement, thus driving the particles into regions of high likelihood: see e.g. [Snyder et al., 2015] for a discussion on the efficiency of the so-called optimal proposal, defined as $\pi(x_k|x_{0:k-1}, y_{1:k}) = p(x_k|x_{k-1}, y_k)$, [van Leeuwen, 2009] for a presentation of particle filters using the ensemble Kalman filter EnKF as proposal distribution, auxiliary particle filters and other variants. Algorithms that have also received interest lately are the unscented particle filter [Rui and Chen, 2001, van der Merwe et al., 2001], and sigma-point particle filter [van der Merwe, 2004]; these importance sampling schemes use the unscented Kalman filter, which we thoroughly review in the next chapter, to generate a suitable proposal distribution and thus increase efficiency of the particle filter. In [van der Merwe, 2004], this idea is combined with a Gaussian mixture fit to reduce computational cost (Gaussian mixture sigma-point particle filter GMSPPF).

Another simple way to mitigate the sampling depletion problem is to modify the noise levels, making them appear larger in the filter than they actually are in the data [Gustafsson, 2010], method

called *jittering* or *roughening* in the literature. This is particularly interesting in the case where static parameters are included in the state vector: these are treated as time-varying parameters by adding artificial noise v^θ, i.e. $\theta_k = \theta_{k-1} + v^\theta$ (see e.g. [Gustafsson and Hriljac, 2003] that uses a decreasing roughening method, i.e. the noise variance of the noise term v^θ decreases over time, or [Chatzi and Smyth, 2009] for a successful application for parameter identification on a structural system). Actually, this method is also used in other nonlinear filtering techniques that also suffer lack of robustness when learning static parameters: in [Chui and Chen, 2009] it is shown for systems linear in the states x^{dyn} that adding artificial noise is required to learn the static parameters when using an EKF; in [Wan and van der Merwe, 2000] artificial noise is added to learn parameters using both joint and dual UKFs. A drawback of this method however is that it increases the sampling variance, and that the noise level should be tuned carefully, which is not trivial, especially for large dimensional systems.

Another method to increase efficiency of the particle filter that has received a lot of attention lately in the particle filtering community, and will be the subject of the following sections of this dissertation chapter, is the so-called Rao-Blackwellised, or marginalized, particle filter. The main idea behind Rao-Blackwellisation is to use the structure of the equations (more precisely conditional linearities) to marginalize out some of the states and thus use the particle filter in a smaller dimensional space while the remaining states are inferred using a bank of Kalman filters.

The concept of Rao-Blackwellisation has already been introduced for structural systems in [Sajeeb et al., 2009a,b]. The authors Sajeeb et al. show that the Rao-Blackwellised particle filter for Conditionally Linear Gaussian systems cannot usually be directly applied to joint state/parameter estimation in structural systems since equations are coupled. In their first paper, this issue is overcome by substructuring the system into several linear and nonlinear sets. However this method requires that such substructures exist, which will not be the case if all parameters are unknown (which is the case treated here), because all equations will be nonlinear through multiplication of

an unknown parameter with an unknown state (e.g., kx and $c\dot{x}$). In their second paper, Sajeeb et al. instead obtain the conditional pdf of all the states using a bank of Kalman filters, conditioned on few states previously propagated with the particle filter. They show that increasing the analyticity of the filtering algorithm (by using Kalman filtering as much as possible) reduces the variance of the estimate, a conclusion that we will also observe with the enhanced algorithms derived thereafter. Finally in a third paper, [Sajeeb et al., 2010], the same authors propose a semi-analytical particle filter that consists in locally linearizing the equations to obtain an ensemble of linearized systems that can then be solved using Kalman filtering and further be exploited to obtain samples for the particle filter. All these methods perform considerably better than the generic particle filter as they provide more accurate results and lower variance estimates.

In sections 2.2 and 2.3, we first present two Rao-Blackwellized algorithms that are in many ways related to the algorithms presented in [Sajeeb et al., 2009a,b]. However they differ on one important point: in those two sections we are trying to directly apply and improve upon algorithms that already exist and are extensively used in the particle filtering literature in other fields (tracking, navigation...): namely the marginalized particle filter for mixed linear/nonlinear state-space models (called MLN-RBPFin the remainder of the paper) developed in [Schön et al., 2005] and Storvik's algorithm, developed in [Storvik, 2002] and extensively used for static parameter estimation. In this way, we hope to present algorithms that are easily generalizable. More importantly, we present in the last section of this chapter a new algorithm that combines the Marginalized PF with second order Kalman filter (EKF^2) propagation and update steps. This enhanced algorithm is particularly well-suited for monitoring of structural systems, since high nonlinearities are often localized while the remaining equations are bilinear in the dynamic states and static parameters. Using this property of structural systems, a second order Taylor Series expansion enables us to find exactly the mean and covariance of many of the states and parameters, conditioned on the few highly nonlinear ones. In this way, we are able to greatly reduce the dimension of the sampling

space of the particle filter, which would hopefully enable us to use this algorithm for medium size systems.

2.2 Reducing the size of the sampling space via marginalization

2.2.1 The concept of Rao-Blackwellisation

The main idea in Rao-Blackwellisation [Doucet et al., 2000] is to partition the state vector as $\{x_k\} = \begin{Bmatrix} u_k \\ z_k \end{Bmatrix}$ [1] and the full posterior distribution as:

$$p(x_{0:k}|y_{1:k}) = p(u_{0:k}, z_{0:k}|y_{1:k}) = \underbrace{p(z_{0:k}|u_{0:k}, y_{1:k})}_{\text{analytically tractable}} \overbrace{p(u_{0:k}|y_{1:k})}^{\text{inferred with MC sampling}} \tag{2.5}$$

where $p(z_{0:k}|u_{0:k}, y_{1:k})$ can be computed exactly. This is easily done for example for Conditionally Linear Gaussian (CLG) models (Equation 2.6), where the state vector $\{x_k\}$ can be divided into two parts $\begin{Bmatrix} u_k \\ z_k \end{Bmatrix}$, where z_k is conditionally Gaussian given the nonlinear variable u_k:

$$\begin{aligned} u_k &\sim p(u_k|u_{k-1}) \\ z_k &= G(u_{k-1})z_{k-1} + \nu_{k-1} \\ y_k &= H(u_k)z_k + \eta_k \end{aligned} \tag{2.6}$$

with $\eta_k \sim \mathcal{N}(\cdot; 0, R)$ and $\nu_{k-1} \sim \mathcal{N}(\cdot; 0, Q)$. It is important to notice that in CLG models, the propagation equation of the nonlinear part u does not depend explicitly on the linear part z.

In this case, assuming a Gaussian prior for the linear part z, z conditioned on u can be inferred exactly using a Kalman filter. Thus only the nonlinear part u will be inferred using solely sequential importance sampling (particle filter). At each time step, the filtering density for u_k is a particle

[1] Note that this decomposition might be different from a dynamic states vs. static parameters partition of the augmented state vector.

approximation while it is a mixture of Gaussians for z_k:

$$p(u_k, z_k \mid y_{1:k}) = \sum_{i=1}^{n_p} w_k^{(i)} \mathcal{N}\left(z_k; z_{k|k}^{(i)}, P_{k|k}^{(i)}\right) \delta\left(u_k - u_k^{(i)}\right) \tag{2.7}$$

with $\left\{z_{k|k}^{(i)}, P_{k|k}^{(i)}\right\}$ being the mean and covariance of Kalman filter conditioned on the history of the corresponding particle $\left\{u_0^{(i)}, \cdots, u_k^{(i)}\right\}$. Regarding the algorithm, at each time step k, for each particle $\cdot^{(i)}$, the propagation and update steps combine particle filter and Kalman filter equations, see for instance [Särkkä, 2013]:

- Propagate the nonlinear part as in the PF: $u_k^{(i)} \sim p\left(u_k | u_{k-1}^{(i)}\right)$
- Propagate the linear part, conditioned on the sampled nonlinear part $u_{k-1}^{(i)}$, as in the KF[2]:

$$\left[z_{k|k-1}, P_{k|k-1}\right]^{(i)} = KFpropagate_{\left(G(u_{k-1}^{(i)})\right)} \left\{z_{k-1|k-1}, P_{k-1|k-1}\right\}^{(i)}$$

- Weight the particles as in the PF: $w_k^{(i)} = w_{k-1}^{(i)} p\left(y_k | u_{0:k}^{(i)}, y_{1:k}\right)$
- Update the linear part as in the KF[2]:

$$\left[z_{k|k}, P_{k|k}\right]^{(i)} = KFupdate_{\left(y_k; H(u_k^{(i)})\right)} \left\{z_{k|k-1}, P_{k|k-1}\right\}^{(i)}$$

where $p\left(y_k | u_{0:k}^{(i)}, y_{1:k}\right)$ and $p\left(u_k | u_{k-1}^{(i)}\right)$ are Gaussian pdfs, thus easy to evaluate and sample from.

2.2.2 Rao-Blackwellisation for mixed linear-nonlinear systems

However, as explained in [Sajeeb et al., 2009a], due to the fact that equations for structural systems are coupled, it is usually not possible to write our state-space model as in Equation 2.6 (i.e., our systems are not Conditionally Linear Gaussian).

[2]Notations: the function *KFpropagate* takes as input $\left\{z_{k-1|k-1}, P_{k-1|k-1}\right\}^{(i)}$ (i.e., posterior at time step $k-1$), and outputs $\left\{z_{k|k-1}, P_{k|k-1}\right\}^{(i)}$ (i.e., prior at time step k), using the propagation equations of the KF to the equation $z_k = G(u_{k-1}^{(i)}) z_{k-1} + v_{k-1}$. Similarly for the measurement update *KFupdate*. The actual equations contained in the functions *KFpropagate* and *KFupdate* are explicitly written in Appendix A. In the remainder of the chapter, we will write for simplicity G_{k-1} for $G(u_{k-1})$ and so on.

Recently, a new marginalized particle filter was derived in [Schön et al., 2005] that can be applied to models where the propagation of the nonlinear part u also depends explicitly on the linear part z. Those models are called, following [Schön et al., 2005], mixed linear/nonlinear Gaussian models and can be written as:

$$u_k = F_{k-1} z_{k-1} + f_{k-1} + v^u_{k-1} \tag{2.8a}$$

$$z_k = G_{k-1} z_{k-1} + g_{k-1} + v^z_{k-1} \tag{2.8b}$$

$$y_k = H_k z_k + h_k + \eta_k \tag{2.8c}$$

F_{k-1} , G_{k-1} and H_k are matrices and f_{k-1}, g_{k-1} and h_k are vectors that can integrate terms nonlinear in $u_{k,k-1}$. Eq. (2.8c) is the measurement equation for both the linear and nonlinear parts of the state vector, using y_k as a measurement. Eqs. 2.8a and (2.8b) are propagation equations for the nonlinear and linear parts respectively. However, the nice property that is used in algorithm 2 is that Eq. 2.8a can also be seen as a linear measurement equation for the linear part z, using u_k as a measurement. Thus, knowing u_k, one can use the Kalman filter update equations to update the mean and covariance of the linear part, and then propagate it using Eq. 2.8b. In this way, the density $p(z_k | u_{0:k}, y_{1:k})$ is estimated using the Kalman filter (which gives the exact posterior density for a linear system with Gaussian noise) while the density $p(u_{0:k} | y_{1:k})$ is estimated using the PF, which will now be used in a lower dimensional space, and the full posterior pdf can be written as in Eq. (2.7). The full algorithm is presented in Alg. 2, in the case of diagonal covariance matrices. The reader is referred to [Schön et al., 2005] for full proofs and derivation of the algorithm.

2.2.3 Marginalization applied to parameter learning

In this section, we consider systems in which the vector z of states/parameters that appear linearly in the equations is composed solely of static parameters, for which the propagation equation is

Algorithm 2: Marginalized particle filter for mixed linear/nonlinear state-space models, adapted from [Schön et al., 2005]

Initialization; **for** $i = 1, \cdots, n_p$ **do**

Sample $u_0^{(i)} \sim p(u_0)$ for the non linear states and assign equal weights $w_0^{(i)} = \frac{1}{n_p}$;

Assign for each particle $z_{0|0}^{(i)} = E[p_0(z)]$ and $P_{0|0}^{(i)} = cov(p_0(z))$;

end

for $k = 1, \cdots, N$ **do**

for $i = 1, \cdots, n_p$ **do**

Propagate particles (nonlinear states) with the PF (refers to Eq. (2.8a))

$$\begin{aligned} u_k^{(i)} &\sim \mathcal{N}(\cdot; \mu_k, \Sigma_k) \\ \mu_k &= F_{k-1} z_{k-1|k-1}^{(i)} + f_{k-1} \\ \Sigma_k &= Q_k + F_{k-1} P_{k-1|k-1}^{(i)} (F_{k-1})^T \end{aligned} \tag{2.9}$$

Update 1 + Propagate: update mean & covariance of the linear subvector z_k with the KF update equations (refers to Eq. (2.8a) as a measurement equation for z and particle $u_k^{(i)}$ as the "measurement"); then propagate its mean & covariance (refers to Eq. (2.8b)) [2]:

$$\begin{aligned} \left[\tilde{z}_{k-1|k-1}, \tilde{P}_{k-1|k-1}\right]^{(i)} &= KFupdate_{(u_k^{(i)}; F(u_{k-1}^{(i)}), f(u_{k-1}^{(i)}))} \left\{z_{k-1|k-1}, P_{k-1|k-1}\right\}^{(i)} \\ \left[z_{k|k-1}, P_{k|k-1}\right]^{(i)} &= KFpropagate_{(G(u_{k-1}^{(i)}), g(u_{k-1}^{(i)}))} \left\{\tilde{z}_{k-1|k-1}, \tilde{P}_{k-1|k-1}\right\}^{(i)} \end{aligned} \tag{2.10}$$

Assign weights to each particle using the measurements y_k

$$\begin{aligned} w_k^{(i)} &= w_{k-1}^{(i)} p(y_k | u_k^{(i)}) = w_{k-1}^{(i)} \mathcal{N}(y_k; \hat{\mu}_k, S_k) \\ \hat{\mu}_k &= h_k + H_k z_{k|k-1}^{(i)} \\ S_k &= R_k + H_k P_{k|k-1} (H_k)^T \end{aligned} \tag{2.11}$$

Update 2: update the linear states using measurements y_k, using again the KF update equations (refers to Eq. (2.8c)) [2]

$$\left[z_{k|k}, P_{k|k}\right]^{(i)} = KFupdate_{(y_k; H(u_k^{(i)}), h(u_k^{(i)}))} \left\{z_{k|k-1}, P_{k|k-1}\right\}^{(i)} \tag{2.12}$$

end

Resample if needed and compute the quantities of interest (moments of the states):

$$E[u_k] = \sum_{i=1}^{n_p} w_k^{(i)} u_k^{(i)} \qquad Cov(u_k) = \sum_{i=1}^{n_p} w_k^{(i)} \left(u_k^{(i)} - E[u_k]\right) \left(u_k^{(i)} - E[u_k]\right)^T \tag{2.13}$$

$$E[z_k] = \sum_{i=1}^{n_p} w_k^{(i)} z_{k|k}^{(i)} \qquad Cov(z_k) = \sum_{i=1}^{n_p} w_k^{(i)} \left(P_{k|k}^{(i)} \left(z_{k|k}^{(i)} - E[z_k]\right) \left(z_{k|k}^{(i)} - E[z_k]\right)^T\right) \tag{2.14}$$

end

simply $z_k = z_{k-1}$. Thus the system equations can be written as:

$$u_k = F_{k-1} z_{k-1} + f_{k-1} + \nu_{k-1} \tag{2.15a}$$

$$z_k = z_{k-1} \tag{2.15b}$$

$$y_k = H_k z_k + h_k + \eta_k \tag{2.15c}$$

with F_{k-1}, H_k, f_{k-1} and h_k matrices and vectors that can contain nonlinear terms in $u_{k-1,k}$. This special case is of interest for structural systems since the stiffness and damping parameters c, k usually appear linearly in the equations of motion. It is also important to notice that the vector u is composed of both the dynamic states and the parameters that appear nonlinearly in the equations, if any. In our numerical example in section 2.3, the Bouc-Wen parameter n appears nonlinearly in the equations and thus $u_k = \begin{Bmatrix} x_k \\ n \end{Bmatrix}$, while all other parameters appear linearly in the equations and thus form the vector z.

For this type of system, two solvers that make use of the concept of Rao-Blackwellisation can be used: the MLN-RBPFand Storvik's algorithm.

Using the marginalized particle filter for mixed linear/nonlinear Gaussian systems

In [Schön and Gustafsson, 2003], the method described in section 2.2.2 is applied to parameter estimation in the case where the parameters are conditionally Gaussian given the dynamic states (or vice versa). Such a system can be written in the form Eq. 2.15, which is the same as Eq. 2.8 with $G = I$ and $g = 0$, thus the MLN-RBPFcan be directly applied. We furthermore write $z_{k|k}^{(i)} = \mu_k^{(i)}$ and $P_{k|k}^{(i)} = C_k^{(i)}$ in Equation 2.7.

Using Storvik's algorithm

In on-line parameter estimation, a similar concept is used in [Storvik, 2002] to marginalize static parameters out from the full pdf, while using the PF to recover the dynamic states. Storvik's algo-

rithm can be applied to systems in which the pdf of the static parameters $p(\theta|x_{0:k}, y_{1:k})$ depends only on some low dimensional sufficient statistics s_k that can be easily updated at each time step, i.e.:

$$p(\theta|x_{0:k}, y_{1:k}) = p(\theta|s_k) \tag{2.16a}$$

$$s_k = update(s_{k-1}, x_k, y_k) \tag{2.16b}$$

where $p(\theta|s_k)$ is known and easy to sample from. In this way, a new parameter vector can be simulated at each time step without using previous values of the parameter vector, which helps prevent the problem of impoverishment. Going back to notations in Equation 2.15, if the static parameters in vector z appear linearly in the propagation equation, the pdf $p\left(z|s_k^{(i)}\right)$ is Gaussian, thus the sufficient statistics are its mean vector and covariance matrix $s_k^{(i)} = \{\mu_k^{(i)}, C_k^{(i)}\}$, and the update function consists of a Kalman filter update, in a similar fashion as for the MLN-RBPF. For each time step k and for each particle $\cdot^{(i)}$, Storvik's algorithm can then be decomposed into 2 steps: (1) sample a realization $z^{(i)}$ from $p\left(z|u_{0:k-1}^{(i)}, y_{1:k-1}\right) = p\left(z|s_{k-1}^{(i)}\right)$, (2) knowing this realization $z^{(i)}$, run a PF for u_k, i.e. sample from $p\left(u_k|u_{k-1}^{(i)}, z^{(i)}\right)$ and weight proportionally to $p\left(y_k|u_k^{(i)}, z^{(i)}\right)$. The detailed algorithm is given in Alg. 3.

It can be noted here that even though both Storvik's algorithm and the MLN-RBPFare based on the same decomposition of the full pdf, there exists an important difference between the two algorithms. In Storvik's algorithm, a realization of $z|u_{0:k}^{(i)}, y_{1:k}$ is simulated at each time step, while in the MLN-RBPF, all the equations are re-written in terms of $\{\mu_k^{(i)}, C_k^{(i)}\}$, where $p\left(z|u_{0:k}^{(i)}, y_{1:k}\right) = \mathcal{N}(z; \mu_k^{(i)}, C_k^{(i)})$, and no realization is ever generated from this pdf.

Again, Storvik's algorithm involves, for each particle, an update of the sufficient statistics of the static parameters, conditioned on sampled particle $u_k^{(i)}$ and measurement y_k at time step k:

$$s_k^{(i)} = update\left(s_{k-1}^{(i)}, u_k^{(i)}, y_k\right) \tag{2.17}$$

Examples for systems where the parameters appear linearly in the propagation equation are pre-

Algorithm 3: Storvik's algorithm, using standard proposal distributions (importance densities) for both dynamic states u and static parameters z

Initialize u_0 as in PF, also initialize s_0;
for $k = 1 : N$ **do**
  for $i = 1 : n_p$ **do**
    sample $z^{(i)}$ from $p\left(z|s_{k-1}^{(i)}\right)$;
    sample $u_k^{(i)}$ from $p\left(u_k|u_{k-1}^{(i)}, z^{(i)}\right)$;
    evaluate importance weights via $w_k^{(i)} = w_{k-1}^{(i)} p\left(y_k|u_k^{(i)}, z^{(i)}\right)$;
  end
  resample particles;
  for $i = 1 : n_p$ **do**
    update sufficient statistics of the static parameters via

$$s_k^{(i)} = update\left(s_{k-1}^{(i)}, u_k^{(i)}, y_k\right)$$

  end
end

sented in [Storvik, 2002]. However, the systems of interest for SHM purposes are more complex since the same static parameters will appear in both the propagation and the measurement equations (Equation 2.15), a case not often considered in the literature.

The way we developed the update equation (2.17) in the case where the same parameters appear in both the propagation and measurement equation is as follows:

- write the measurement y_k as a function of vector $u_{k-1}^{(i)}$ and static parameters z, by combining the propagation and measurement equations:

$$y_k = g(u_{k-1}^{(i)}, z, \nu_{k-1}, \eta_k) \tag{2.18a}$$

$$= h(f(u_{k-1}^{(i)}, z, \nu_{k-1}), z, \eta_k) \tag{2.18b}$$

  If the same parameters appear linearly in both f and h, they will appear quadratically in g

- using also the propagation equation one can write:

$$u_k^{(i)} = f(u_{k-1}^{(i)}, z, \nu_{k-1}) \tag{2.19a}$$

$$y_k = g(u_{k-1}^{(i)}, z, \nu_{k-1}, \eta_k) \tag{2.19b}$$

- the first equation of this system is linear in z so one can directly apply the Kalman filter update equations to update z, conditioned on $u_k^{(i)}$. On the contrary g is quadratic in z thus one needs to linearize it as $g \simeq \tilde{g} = G_{z^{(i)}}$ where $G_{z^{(i)}}$ is the Jacobian of g with respect to z computed using realizations $u_{k-1}^{(i)}$ and $z_{k-1}^{(i)}$ (same procedure as the one used in the Extended Kalman filter, see note in appendix A for more details on the EKF)
- now one can write the following system, linear in z:

$$u_k^{(i)} = f(u_{k-1}^{(i)}, z, \nu_{k-1}) \tag{2.20a}$$

$$y_k = \tilde{g}(u_{k-1}^{(i)}, z, \nu_{k-1}, \eta_k) \tag{2.20b}$$

  thus at each time step, for each particle, one can apply the KF or EKF update equations to update the sufficient statistics $s_k^{(i)}$, knowing both $u_k^{(i)}$ and y_k.

Several comments can be made on this update algorithm. First, one can see that the way we tackle the measurements y_k by creating a new function g is similar to what is done in the dual UKF (presented in [Wan and van der Merwe, 2000]). However, in the dual UKF this step is done only once per time step, thus the correlation between dynamic states and static parameters is lost. Here instead, this update is performed for each particle; thus this correlation is preserved. The second comment is that linearizing g can create some error. However as will be shown later in the numerical example, in the systems that are of interest for us this error will be negligible, due to the fact that the propagation equation is discretized with small time steps.

Posterior pdf of the marginalized static parameters

Using those two algorithms (MLN-RBPFand Storvik's algorithm), the posterior filtering distribution of the states and parameters can be written as:

$$p\left(u_k, z_k \mid y_{1:k}\right) = \sum_{i=1}^{n_p} w_k^{(i)} \mathcal{N}\left(z_k; \mu_k^{(i)}, C_k^{(i)}\right) \delta\left(u_k - u_k^{(i)}\right) \tag{2.21}$$

where z is a vector composed of static parameters that appear linearly in the equations. Then the expected value of these marginalized static parameters is computed as:

$$E\left[z \mid y_{1:k}\right] \approx \sum_{i=1}^{n_p} w^{(i)} E\left[z | u_{0:k}^{(i)}, y_{1:k}\right] = \sum_{i=1}^{n_p} w^{(i)} \mu^{(i)} \tag{2.22}$$

i.e., the posterior $z|y_{1:k}$ is a mixture of n_p Gaussians and the expected value of z is the weighted sum of the means.

2.2.4 Rao-Blackwellisation outputs smaller variance estimates

Rao-Blackwellisation algorithms output estimates with lower variance than the standard particle filter estimates. This can be understood intuitively as the dimension of the space where importance sampling is performed is reduced, while in the remaining dimensions inference is performed analytically using an optimal filter (here, the Kalman filter for linear Gaussian systems). [Doucet et al., 2000], which first presented this concept of Rao-Blackwellisation for particle filters, provides mathematical proof of this property.

In practice, this means that methods that use Rao-Blackwellisation should produce more consistent results over several runs. Indeed, as we will see at the end of this paper, our new algorithm that uses two levels of marginalization produces parameter estimates with lower sampling variance.

2.3 Numerical Validation

2.3.1 Structural System

These algorithms were tested on a highly nonlinear 3-DOF dynamic system that was already presented in [Chatzi and Smyth, 2009]. The Bouc-Wen model of hysteresis is used for the first degree of freedom to model high nonlinearities at the base of the structure, as shown in Fig. 2.2. According to this model, the restoring force is:

$$F(t) = k_1 r_1(t) \tag{2.23}$$

$r_1(t)$ is the hysteretic displacement and follows a highly-nonlinear equation of motion:

$$\dot{r_1}(t) = \dot{x_1}(t) - \beta \left|\dot{x_1}(t)\right| \left|r_1(t)\right|^{n-1} r_1(t) - \gamma \dot{x_1}(t) \left|r_1(t)\right|^n \tag{2.24}$$

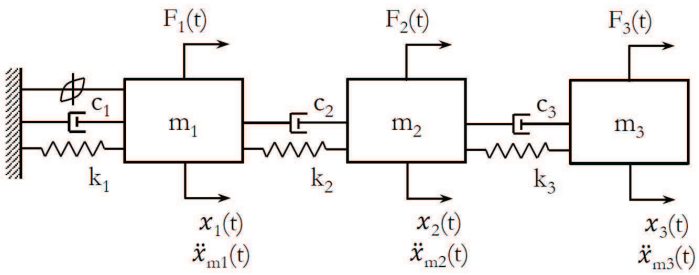


Figure 2.2: 3-DOF structural system considered in simulations, adapted from [Chatzi and Smyth, 2009]

The dynamic equations of motion are then as follows:

$$\begin{bmatrix} m_1 & 0 & 0 \\ 0 & m_2 & 0 \\ 0 & 0 & m_3 \end{bmatrix} \begin{Bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \ddot{x}_3 \end{Bmatrix} + \begin{bmatrix} c_1+c_2 & -c_2 & 0 \\ -c_2 & c_2+c_3 & -c_3 \\ 0 & -c_3 & c_3 \end{bmatrix} \begin{Bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \end{Bmatrix} + \begin{bmatrix} k_1 & k_2 & -k_2 & 0 \\ 0 & -k_2 & k_2+k_3 & -k_3 \\ 0 & 0 & -k_3 & k_3 \end{bmatrix} \begin{Bmatrix} r_1 \\ x_1 \\ x_2 \\ x_3 \end{Bmatrix} = -\ddot{u}_g \begin{Bmatrix} m_1 \\ m_2 \\ m_3 \end{Bmatrix} \tag{2.25}$$

This system equation can then be discretized using a forward Euler scheme (integration time step $dt = 0.005s$ in our case) and cast in state space form, using the state vector:

$$x = \begin{bmatrix} x_1 & x_2 & x_3 & \dot{x}_1 & \dot{x}_2 & \dot{x}_3 & r_1 \end{bmatrix}^T \in \mathbb{R}^7 \tag{2.26a}$$

Assuming $m_1 = m_2 = m_3 = 1$, the system is parameterized by:

$$\theta = \begin{bmatrix} k_1 & k_2 & k_3 & c_1 & c_2 & c_3 & \beta & \gamma & n \end{bmatrix}^T \in \mathbb{R}^9 \tag{2.26b}$$

We simulate data by running the problem with zero initial conditions, using a 4th order Runge-Kutta scheme, using as true parameter vector:

$$\theta_{true} = \begin{bmatrix} 8 & 8 & 8 & 0.25 & 0.25 & 0.25 & 2 & 1 & 2 \end{bmatrix}^T \tag{2.27}$$

We also generate measurements (accelerations at three DOFs), to which we add 10% RMS Gaussian noise. We also add 10% RMS noise to the excitation $\ddot{u}_g$, to represent possible error made when measuring the excitation.

In those equations, the nonlinearities appear in two forms:

- the Bouc-Wen model itself,
- the multiplication of a unknown parameter with an unknown state (e.g., kx or $c\dot{x}$), in both the process and the measurement equations.

Also, one can see that, except for the Bouc-Wen parameters β, γ and n that do not appear in the measurement equations, the same parameters (stiffness and damping) appear in both the process

and the measurement equations. This is very common for this type of structural system, but not commonly considered in the particle filtering literature. Indeed problems solved using particle filters usually exhibit unknown static parameters in only one of the two equations, or different parameters in the two equations, which simplifies the parameter learning part. Finally as explained in [Sajeeb et al., 2009a], in such structural systems the equations are coupled, thus one cannot easily extract a nonlinear subspace that would follow a Markov process, i.e., this system is not Conditionally Linear Gaussian. However, we can see that except for the Bouc-Wen parameter n, all the parameters appear linearly in all of the equations. This means that we can use a marginalized particle filter for mixed linear/nonlinear states (MLN-RBPF), or Storvik's algorithm with our new update equations, marginalizing out the linear parameters.

2.3.2 Quantities of interest computed for comparison of performance between several methods

From a SHM perspective, accurate estimation of the structural parameters is crucial. When using simulated data, the true value of the parameter vector θ_{true} is known and can be easily compared with the identified value, which can be defined as the mean value of the posterior pdf at the end of the filtering process, i.e., $\theta_{id} = E[\theta|y_{1:N}]$. However, in order to not overemphasized the effect of the one final point, we decided here to average over the last few time steps, i.e.:

$$\theta_{id} = \frac{1}{l} \sum_{k=N-l+1}^{N} E[\theta|y_{1:k}] \tag{2.28}$$

with $l = 80$, i.e., we average the results over the last $0.40s$, which represents roughly a quarter of a loop of the hysteresis variable. To observe convergence over time of the parameters, one can plot the evolution over time of the mean value of the posterior pdf, i.e., plot $\theta_k = E[\theta|y_{1:k}]$ as a function of the time step k.

Also, to validate the method we use the identified parameter θ_{id} and re-run the equations of

motion, using zero initial conditions and the MATLAB ode45 solver. This will output a time series $\{x_{val,k}\}_{k=1:N}$, and we can then plot the hysteresis loop and compare with the true one. Both the true loop and the validation one are generated using the clean excitation, as we want to compare the behaviors of the true and identified system to the true excitation. To look at errors on the dynamic states, we will compute the coefficient of variation of the root mean squared error, used to compare two time series and defined as:

$$CV(RMSE)_{states} = \frac{\sqrt{\frac{1}{N}\sum_{k=1}^{T}(x_k - x_{true,k})^2}}{\frac{1}{N}\sum_{k=1}^{N} x_{true,k}} \tag{2.29}$$

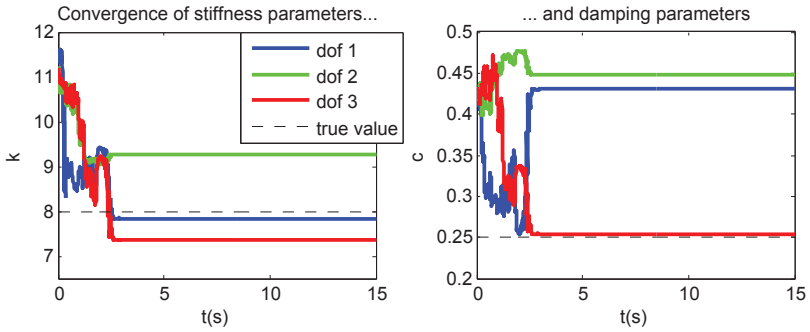
where x_k can be either the time series obtained during online learning, i.e. $E[x_k|y_{1:k}]$, or the one obtained after validation with the last parameter value, i.e., $x_{val,k}$.

2.3.3 Degeneracy of the bootstrap particle filter

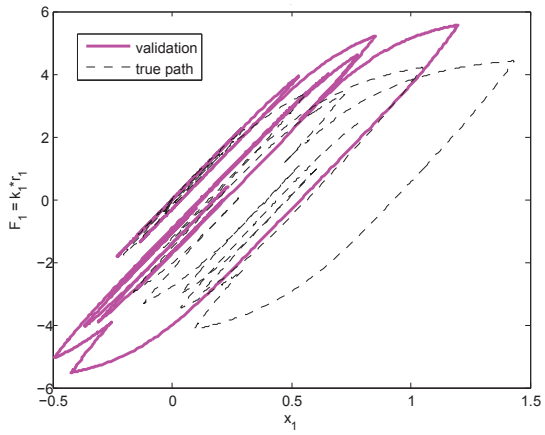
First we ran the problem with the generic bootstrap particle filter using 500 particles, no noise added to the static parameters. We chose lognormal priors for damping and stiffness parameters (admissible region for those parameters is $\mathbb{R}^+$), a uniform prior $\mathcal{U}(1,5)$ for n and a uniform prior over an admissible region of $\{\beta,\gamma\}$, which is $\{\beta \geq 0, \beta \leq 5, |\gamma| \leq \beta\}$.

We used diagonal covariance matrices for both the process and measurement equations. For the measurement equations, we used the same noise level as the one actually added to the measurements and excitation. Idem for the process noise corresponding to the velocity states. For the displacements we use a very small variance. Because we have simulated data, we also know that our Bouc-Wen model of hysteresis is exact, thus there should not be noise in this equation either. However, because in real life it is not possible to assume that the system is perfect, we used a non-zero variance for this equation ($\sigma_r = dt \times$ 5% RMS noise on $\dot{x}$) to see how the particle filter reacts to non-zero noise.

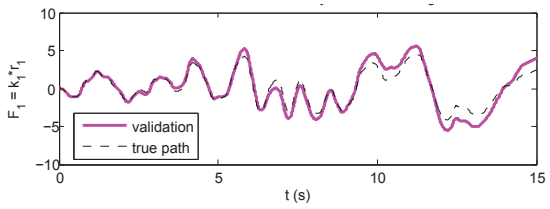
Figures 2.3a and 2.3b show the convergence results and validation hysteresis loop respectively.



(a) Convergence of some parameters



(b) Validation: hysteresis loop



(c) Validation: time series of the hysteretic restoring force

Figure 2.3: Bootstrap particle filter: convergence results, highlighting the particle degeneracy phenomenon observed when using a generic PF

We can see that the particle filter degenerates quickly, then the particles are stuck on one value and cannot learn anymore from the incoming data. Then the final parameter vector obtained is not able to reproduce the characteristics of the system, as can be seen by plotting the validation hysteresis loop.

2.3.4 Rao-Blackwellisation

Equations for the MLN-RBPF

To use Rao-Blackwellisation for this system, we define the linear and nonlinear vectors as:

$$u_k = \begin{bmatrix} x_1 & x_2 & x_3 & \dot{x}_1 & \dot{x}_2 & \dot{x}_3 & r_1 & n \end{bmatrix}_k^T \in \mathbb{R}^8 \tag{2.30a}$$

$$z_k = \begin{bmatrix} k_1 & k_2 & k_3 & c_1 & c_2 & c_3 & \beta & \gamma \end{bmatrix}_k^T \in \mathbb{R}^8 \tag{2.30b}$$

The static parameters that appear linearly in the equations form the vector z_k, while the vector u_k contains the dynamic states and the nonlinear Bouc Wen parameter n. Then we can use the MLN-RBPFpresented in Alg. 2 for a system of the form Equation 2.15. More precisely, the matrices and vectors F, f, H and h of the process and measurement equations respectively can be computed as follows (using notation $x_{ij} = x_i - x_j$):

$$\begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ r_1 \\ n \end{Bmatrix}_{[k]} = \overbrace{\begin{Bmatrix} x_1 + dt\cdot\dot{x}_1 \\ x_2 + dt\cdot\dot{x}_2 \\ x_n + dt\cdot\dot{x}_3 \\ \dot{x}_1 - dt\cdot\ddot{u}_g \\ \dot{x}_2 - dt\cdot\ddot{u}_g \\ \dot{x}_3 - dt\cdot\ddot{u}_g \\ r_1 + dt\cdot\dot{x}_1 \\ n \end{Bmatrix}_{[k-1]}}^{f(u_{k-1})} + \underbrace{dt \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -r_1 & x_{21} & 0 & -\dot{x}_1 & \dot{x}_{21} & 0 & 0 & 0 \\ 0 & x_{12} & x_{32} & 0 & \dot{x}_{12} & \dot{x}_{32} & 0 & 0 \\ 0 & 0 & x_{23} & 0 & 0 & \dot{x}_{23} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -|\dot{x}_1||r_1|^{n-1} r_1 & -\dot{x}_1 |r_1|^n \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}_{[k-1]}}_{F(u_{k-1})} \begin{Bmatrix} k_1 \\ k_2 \\ k_3 \\ c_1 \\ c_2 \\ c_3 \\ \beta \\ \gamma \end{Bmatrix} \tag{2.31a}$$

$$y_k = \begin{Bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \\ \ddot{x}_3 \end{Bmatrix}_{[k]} = \overbrace{\begin{Bmatrix} -\ddot{u}_g \\ -\ddot{u}_g \\ -\ddot{u}_g \end{Bmatrix}_{[k]}}^{h(u_k)} + \underbrace{\begin{bmatrix} -r_1 & x_{21} & 0 & -\dot{x}_1 & \dot{x}_{21} & 0 & 0 & 0 \\ 0 & x_{12} & x_{32} & 0 & \dot{x}_{12} & \dot{x}_{32} & 0 & 0 \\ 0 & 0 & x_{23} & 0 & 0 & \dot{x}_{23} & 0 & 0 \end{bmatrix}_{[k]}}_{H(u_k)} z_k \tag{2.31b}$$

Equations for Storvik's algorithm

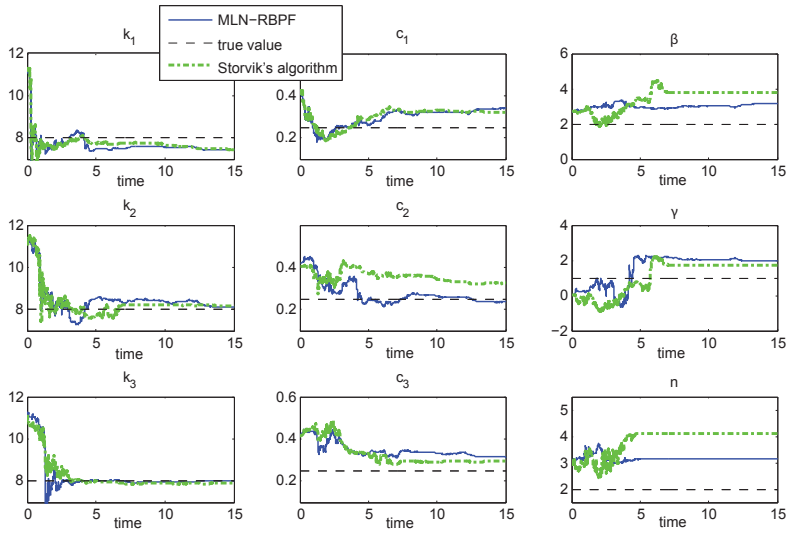
To use Storvik's algorithm, since the same parameters appear in both the process and measurement equations, we need to write the measurements at step k as function of z and u_{k-1}, then compute its Jacobian to use the EKF (method presented in section 2.2.3).

Performance of the marginalized algorithms

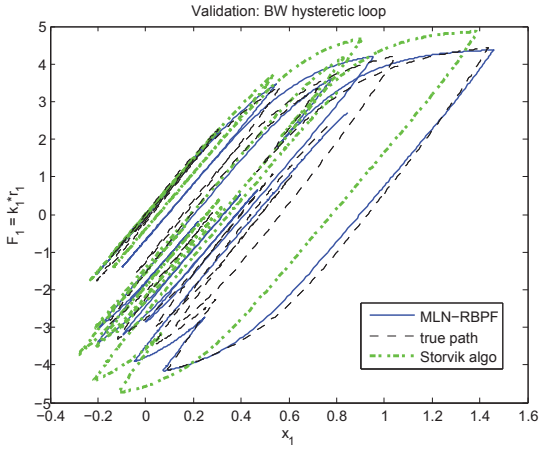
Fig. 2.4 shows the behavior of the MLN-RBPFand Storvik's algorithm for the same problem, again with 500 particles and the same prior for the parameters. We can see that the PF no longer collapses, and the results are much better for the stiffness and damping parameters. It also seems that the MLN-RBPFperforms a little bit better than Storvik's algorithm.

However, we still have problems learning the Bouc-Wen parameters β, γ and n. Let's recall that the propagation equation for static parameters is $\theta_{k+1} = \theta_k$, thus static parameters are not learnt during propagation. They can only be learnt via their correlation with other states and through the measurement update. However the 3 parameters β, γ and n do not appear directly in the measurement equation h, they only influence the likelihood indirectly through the hysteresis variable r_1 that appears in the equation of the first measurement $y_1 = \ddot{x}_1$. This renders learning of these 3 parameters more difficult[3], thus it seems that either more particles or an improved algorithm would be needed to obtain good estimates of those Bouc-Wen parameters.

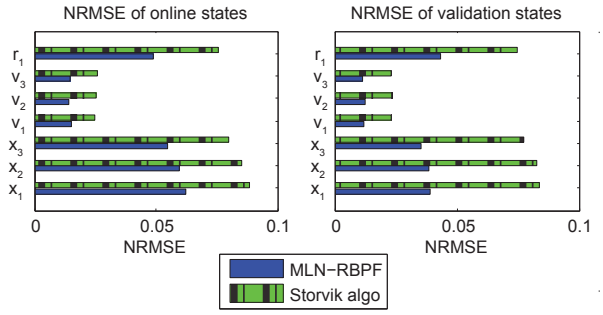
[3]To see more clearly this sensitivity of the likelihood to the parameters, we also run a maximum likelihood estimator (described in [Poyiadjis et al., 2006]), i.e., while the states are recovered through the PF, the estimated static parameters are computed at each time step by maximization of the likelihood $p(y_{1:k}|\theta)$. Our numerical experiments show that if the acceleration at DOF 1 is not measured, the gradients $\frac{\partial p(y_{1:k}|\theta)}{\partial \beta,\gamma,n}$ are zero at each time step, i.e., one cannot learn those parameters via on-line likelihood maximization. Indeed this can be easily explained since if $\ddot{x}_1$ is not measured, neither β,γ,n nor r_1 appear explicitly in the measurement equation h. Thus the parameters β,γ,n do not have any influence on the likelihood at the current time step, which explains why the gradients are zero.



(a) Convergence of static parameters



(b) Validation: hysteresis loop



(c) Normalized MSE of the dynamic states

Figure 2.4: Performance of the MLN-RBPF(blue) vs. Storvik's algorithm (green)

2.4 Improvements on learning the Bouc Wen parameters

2.4.1 Second Order EKF for second level of marginalization

As mentioned previously, the nonlinearities in the problem studied, and more generally in problems of interest for SHM purposes, come from two sources:

- multiplication of an unknown state with an unknown parameter (bilinear nonlinearities in dynamic states and parameters)
- Bouc Wen model of hysteresis (localized high nonlinearity)

In many structural systems of interest for SHM, high nonlinearities are localized, thus only few states are actually involved in highly nonlinear equations. Thus, conditioned on those few states and parameters (in our example, conditioned on $\dot{x}_1, r_1, n$) the system is bilinear in the states and parameters. Recalling basics of nonlinear Kalman filtering (extended Kalman filtering for instance), we know that it is possible to find exactly the mean and covariance of a multivariate Gaussian random vector that undergoes a quadratic transformation, through a second order Taylor Series Expansion of this transformation (for a quadratic, or bilinear, transformation, the 3rd order and higher derivatives are 0, thus there is no error term in the second order Taylor series expansion). Using this concept in nonlinear Kalman filtering gives rise to the so-called second order extended Kalman filter, whose detailed equations are given in Appendix A (see for instance [Särkkä, 2013] for a detailed derivation of those equations). This filter has been gaining some interest lately, due to its exactness for quadratic transformations over other filters. For instance, Table 2.1 shows the mean and covariance of a transformed Gaussian random variable (prior is $x \sim \mathcal{N}(\cdot; 0, I_{n_x})$), for a quadratic transformation $z = g(x) = x^T x$, obtained with different methods. The theoretical distribution of the transformed variable z is $\chi^2(n_x)$, with mean n_x and variance $2n_x$. Table 2.1, adapted from [Gustafsson and Hendeby, 2012], shows that using a second order Taylor series expansion (T2) gives the correct mean and covariance, while the first order Taylor expansion (T1) and even the unscented transform (UT), used in the UKF, do not output exact results.

Table 2.1: Nonlinear approximation of $x^T x$, x is n_x-dimensional for $x \sim \mathcal{N}(\cdot; 0, I_{n_x})$, adapted from [Gustafsson and Hendeby, 2012]

	theory	T1	T2	UT
mean	n_x	0	n_x	n_x
covariance	$2n_x$	0	$2n_x$	$2n_x^2$

Thus here let's consider in our numerical example the partition of the augmented state vector

$$x^{bl} = \begin{bmatrix} x_1 & x_2 & x_3 & \dot{x}_2 & \dot{x}_3 & k_1 & k_2 & k_3 & c_1 & c_2 & c_3 & \beta & \gamma \end{bmatrix}^T \in R^{13} \tag{2.32a}$$

$$x^{n} = \begin{bmatrix} \dot{x}_1 & r_1 & n \end{bmatrix}^T \in R^{3} \tag{2.32b}$$

The vector x^{bl} undergoes only quadratic transformations, thus knowing the measurements y_k and the nonlinear states x^n one can find its mean and covariance using a EKF^2. Plugging this idea into the Rao-Blackwellised concept can be done by running a marginalized particle filter for mixed linear-nonlinear Gaussian systems, using second order extended Kalman filter propagation and update equations instead of the normal Kalman filter ones. More precisely, in Alg. 2, use $u = x^n$, $z = x^{bl}$, and replace the $KFpropagate$ and $KFupdate$ functions with the $EKF^2propagate$ and $EKF^2update$ ones respectively, detailed in Appendix A. This marginalized particle filter with EKF^2 updates will be referred in the remainder of the paper as MPF-EKF^2.

It is very important to recognize that no linearization of a high nonlinear equation is performed here. The posterior pdf of any state or parameter that undergoes a transformation that is neither quadratic nor linear is inferred using the particle filter. Any linearization of a high nonlinearity would create some error; here we only use the second order EKF inference for quadratic functions, and simple EKF for linear functions. However, there is a Gaussian approximation done for the quadratically transformed variables, i.e., we assume that conditioned on the measurements y_k and the highly nonlinear variables x^n, the pdf of x^{bl} is Gaussian, which is not exact. However, this approximation is done for each particle, not for the overall pdf, thus the overall pdf can still be non-Gaussian (mixture of Gaussians). This small assumption enables us to compute analytically the mean and covariance of x^{bl} conditioned on y_k and x^n exactly, thus outputting better results with the same number of particles, as will be shown in the following section.

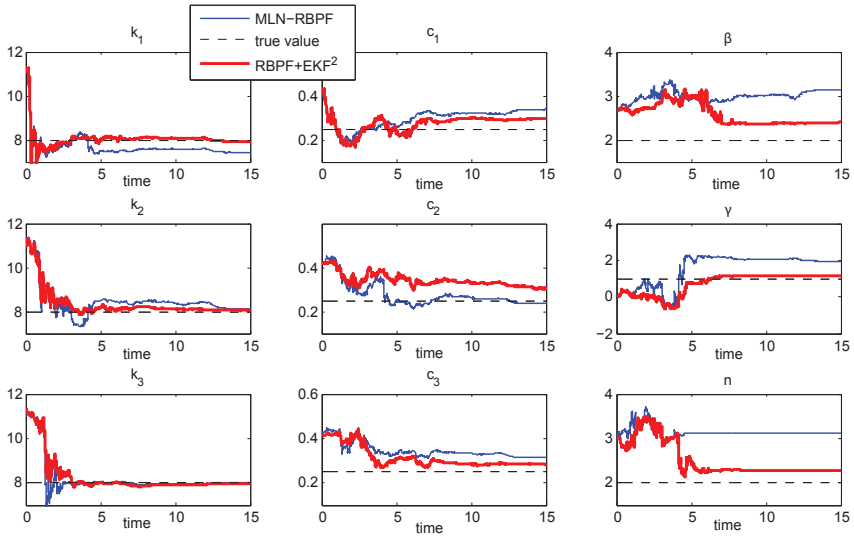
Another important observation is that with this last algorithm, the number of states that are inferred only via particle approximation (here $x^n \in \mathbb{R}^3$) does no longer scale with the dimension

of the problem as it was for the MLN-RBPFand Storvik's algorithm. This will generalize to larger structures as long as the high nonlinearities are localized, which is often the case for structural systems. Thus, this provides some optimism for the use of this last algorithm on higher dimensional systems.

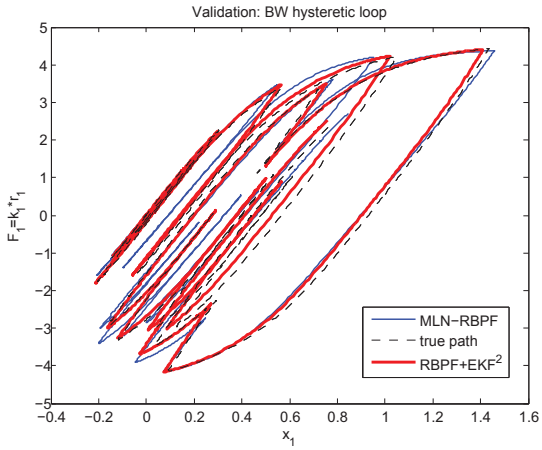
2.4.2 Results and discussion

Results of this new algorithm vs. the MLN-RBPFare shown in Fig. 2.5. Again we can see that this new algorithm performs very well for the stiffness and damping parameters. Results for the Bouc-Wen parameters vary from one run to another. To look more precisely at results for the static parameters, the three solvers were run 50 times each; then for each run the identified parameter θ_{val} was computed (as explained in section 2.3.2). Then we looked at the statistics (mean and standard deviation) of this identified parameter vector. Results are shown on Fig. 2.6: a cross is drawn for each parameter estimate, its position indicates the mean value over 50 runs, its size indicates the standard deviation over 50 runs in each direction. First we can observe that the standard deviation for damping and stiffness parameters is much lower for our new algorithm, meaning that each run outputs almost the same results, and those results are very good for the stiffness, as well as c_1 and c_3. We are overestimating c_2 with all three algorithms, however with our last algorithm we obtained about 20% error on average over the damping parameters, which is generally acceptable for damping parameters that are usually harder to recover than stiffness parameters. Concerning the Bouc-Wen model parameters, we can see that on average γ and n are pretty well recovered with our new algorithm. However the parameter β is not well recovered. This is a trend that we have observed on all runs (even with other filters), i.e., the β parameter seems to be the hardest to recover. Note however that this would most likely vary with the excitation, response levels, frequency content e.t.c.

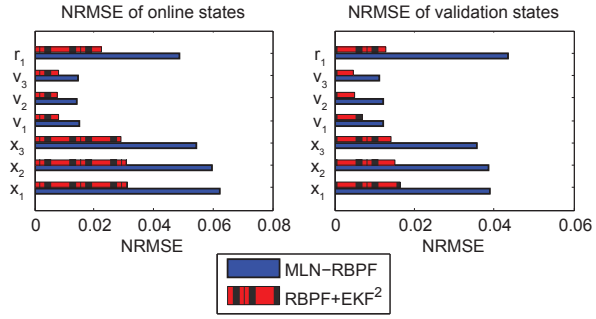
We can also observe in Fig. 2.5 that our new algorithm performs better when looking at the



(a) Convergence of some parameters



(b) Validation: hysteresis loop



(c) Normalized MSE of the dynamic states

Figure 2.5: Performance of the MLN-RBPF(blue) vs. the novel MPF-EKF^2(red)

NRMSE of the dynamic states (i.e., error between inferred hidden states and the true states).

Also, even though results on the damping parameters are acceptable (a little bit less than 20% error on average with our last algorithm), we can observe some bias in the identified parameter, especially for parameter c_2. Indeed Fig. 2.6 shows that our last algorithm consistently outputs overestimated values of the damping parameters, especially for c_2. Also, the parameters β and

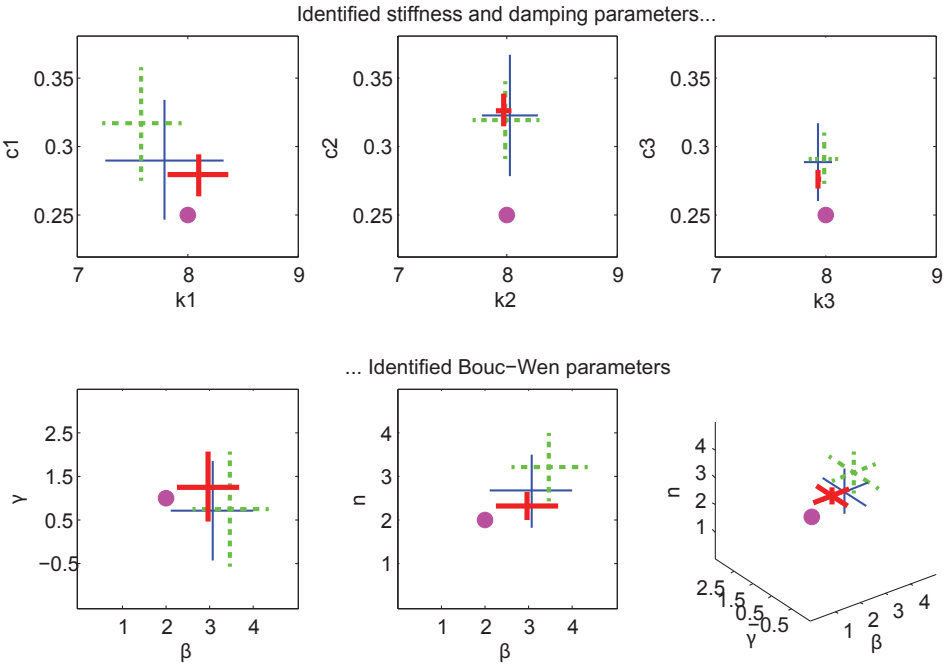


Figure 2.6: Performance of the MLN-RBPF(blue) vs. Storvik's algorithm vs. the MPF-EKF^2(red) on the final identification of the static parameters, i.e., identifying θ_{id}. A cross is drawn for each parameter estimate, its position indicates the mean value over 50 runs, its size indicates the s.t.d. over 50 runs in each direction.

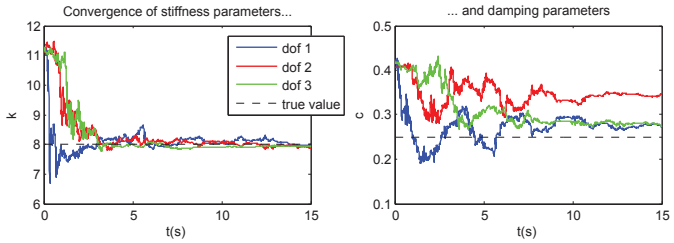
γ, which govern the shape of the hysteresis loop are usually a little bit overestimated. However, when we look at the validation loop, it can be very well recovered, meaning that the error on some parameters compensates the error on others. For example, Fig. 2.7 shows the results for one run of our new algorithm, where the validation loop is very good, which is pretty hard to achieve for such hysteretic systems. However, the identified parameter vector shows clear bias for c_2 and β:

$$\theta_{val} = \begin{bmatrix} 7.973 & 7.912 & 7.943 & 0.277 & 0.342 & 0.278 & 3.418 & 0.960 & 2.488 \end{bmatrix}^T \tag{2.33}$$

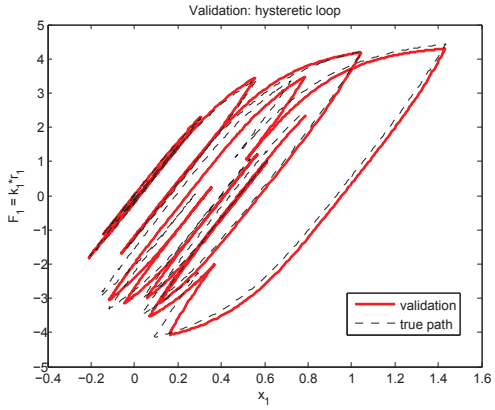
to be compared to the true values:

$$\theta_{true} = \begin{bmatrix} 8 & 8 & 8 & 0.25 & 0.25 & 0.25 & 2 & 1 & 2 \end{bmatrix}^T \tag{2.34}$$

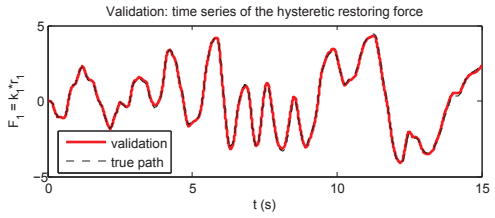
We believe that those errors could be reduced if we could use a better discretization scheme than the forward-Euler. Actually, since the underlying process is continuous, one could use a Continuous-



(a) Convergence of some parameters



(b) Validation: hysteresis loop



(c) Validation: time series of the hysteresis variable

Figure 2.7: A single run with the MPF-EKF^2

Discrete Sequential Importance Resampling algorithm, as derived in [Sottinen and Särkkä, 2008]. Furthermore, in this same paper Rao-Blackwellisation algorithms are also derived for continuous-discrete systems (i.e., continuous process equation and discrete measurements, as is actually the case for us), so those algorithms might be adapted and applied to our systems of interest.

2.5 Concluding remarks

In this chapter we have tackled state estimation and parameter learning in structural systems using particle filtering algorithms. These algorithms are known to yield the true posterior pdf of the state vector, even for highly nonlinear, non-Gaussian systems and are thus very attractive theoretically for monitoring of highly nonlinear structural systems. However, in practice, particle filtering schemes are quite cumbersome to run, if not impossible, on large dimensional systems, since they tend to collapse into a single particle. This problem is also exacerbated when static parameters are added to the state vector, a strong disadvantage for damage detection applications.

As shown in this chapter, enhanced particle filters can be derived using the concept of Rao-Blackwellisation, yielding algorithms that are particularly well-suited for structural systems with localized nonlinearities. Several methods can be thought of to further improve upon these algorithms, such as using an optimal proposal, or an auxiliary particle filter, method already used in [Carvalho et al., 2010] as an improvement upon Storvik's algorithm for parameter learning. Another idea that has appeared in the literature is that of a *block particle filter* derived in [Rebeschini and van Handel, 2015]. This algorithm uses the fact that high dimensional systems usually represent phenomena or structures that are spatially distributed over relatively large distances (for us, numerous degrees of freedom, not all connected to each other), thus one can assume that the dynamics at one degree of freedom are mostly dependent on measurements performed in its vicinity. This translates mathematically into a *decay of correlation* as the actual spatial distance between the degrees of freedom increases. In [Rebeschini and van Handel, 2015], it is explained that using this property can provide a mechanism to overcome the curse of dimensionality; however, the problem of static parameters has not been tackled in this context and should be studied carefully.

As previously mentioned, the filtering task can also be performed using nonlinear Kalman filter type algorithms, which are known to be much more computationally efficient than particle filtering schemes. However, these algorithms are based on a Gaussianity assumption of the posterior pdfs,

an assumption that might be hard to validate for nonlinear systems and might lead to erroneous results. As we will see in the next chapters, the validity of this Gaussianity assumption will highly depend on the distribution of the noise terms, as well as the identifiability properties of the system. The following chapter reviews in detail the assumptions made in the widely used unscented Kalman filter, and inspects this Gaussianity assumption and order of accuracy of the so-called unscented transform for several nonlinear systems whose parameter vector is uniquely identifiable knowing the given measurements. The validity of the Gaussianity assumption and accuracy of the UKF can be assessed by comparing UKF estimates with estimates yielded by a particle filter, known to yield the true posterior pdf if enough particles are used for the approximation.

This page is intentionally left blank.

Chapter 3

Generalization of nonlinear Kalman filters to non-Gaussian distributions

3.1 Nonlinear Kalman filtering

3.1.1 Kalman filtering recursions in the Gaussian setting

In particle filtering, a possibly large amount of particles are propagated in time and weighted using Bayes' theorem, yielding an exact algorithm with a high computational complexity. On the contrary, nonlinear Kalman filters make a Gaussianity assumption on the posterior pdfs, thus only the mean and covariance of the pdfs must be inferred, which greatly simplifies the filtering equations Eq. 1.4.

As explained in chapter 1, during the measurement update step (Eq. (1.4b)) of the filtering scheme, one makes use of Bayes' theorem to condition upon the new measurement y_k. If a Gaussianity assumption is used for the pdfs, this update step can be written in closed form using properties of the Gaussian distribution, more specifically the closure under conditioning property, leading to the well-known Kalman filter update equations (Eqs. (3.2b) and (3.2c)). More explicitly, in

Gaussian Kalman filtering the distributions

$$p\left(x_k, y_k | y_{1:k-1}\right) \tag{3.1a}$$

$$p\left(x_k | y_{1:k}\right) \tag{3.1b}$$

are all assumed to be Gaussian. At time step $k-1$ the state is represented by a Gaussian distribution with mean $x_{k-1|k-1}$ and covariance $P_{k-1|k-1}$. Each recursion of the nonlinear Gaussian Kalman filter (i.e., equations 1.4 in the filtering formulation) can then be written as:

- Start with posterior from time step $k-1$:

$$x_{k-1} | y_{1:k-1} \sim \mathcal{N}\left(\cdot; x_{k-1|k-1}, P_{k-1|k-1}\right)$$

- **Prediction step** at time step k: infer the moments of the prior pdf $p\left(x_k | y_{1:k-1}\right)$

$$\begin{aligned} x_{k|k-1} &= E\left[x_k | y_{1:k-1}\right] \\ P_{k|k-1} &= Cov\left(x_k | y_{1:k-1}\right) \end{aligned} \tag{3.2a}$$

- **Update step** at time step k: infer the moments of the posterior pdf $p\left(x_k | y_{1:k}\right)$ as

$$\begin{aligned} x_{k|k} &= x_{k|k-1} + K_k(y_k - y_{k|k-1}) \\ P_{k|k} &= P_{k|k-1} - K_k S_k K_k^T \end{aligned} \tag{3.2b}$$

  where:

$$\begin{aligned} y_{k|k-1} &= E\left[y_k | y_{1:k-1}\right], \\ S_k &= Cov\left(y_k | y_{1:k-1}\right), \\ \Psi_k &= Cov\left(x_k, y_k | y_{1:k-1}\right), \\ K_k &= \Psi_k S_k^{-1} \quad \text{(Kalman gain)} \end{aligned} \tag{3.2c}$$

  where $Cov(X)$ represents the covariance of random variable X, and $Cov(X,Z)$ the cross-covariance between two jointly distributed RVs X and Z.

In the Gaussian linear case (both f and h are linear, noise terms are Gaussian), the Gaussian assumption is exact since linear transformations preserve Gaussianity, thus the well-known Kalman filter equations yield the true posterior distributions in closed form. However in general, the propagation and observation equations f and h are not linear, thus the pdfs in Eqs. (3.1) are not truly Gaussian and cannot be computed in closed-form. The second approximation used in nonlinear Kalman filtering is then to estimate the first and second order moments of those pdfs and build a Gaussian approximation using those moments. Three main methods exist to approximate those moments (depicted in the same order on Fig. 3.1):

- Monte Carlo approximation (which yields the ensemble Kalman filter EnKF, where particles are propagated and updated using the Kalman gain $K_k = Cov\left(x_k, y_k | y_{1:k-1}\right) Cov\left(y_k, y_k | y_{1:k-1}\right)$, where the covariance terms are computed as sample covariances over the particles, see e.g. [Gillijns et al., 2006, Roth et al., 2015] for an introduction),
- linearization using Taylor Series (TS) expansion (yields the extended Kalman filter EKF),
- the unscented transform (yields the unscented Kalman filter UKF).

The UKF usually shows a good trade off between accuracy and computational time. We review more explicitly its order of accuracy in the following section.

3.1.2 Approximating the moments of a random variable with the unscented transform

Moment propagation in the KF recursions

From the formulation in Eqs. (3.2) one can see that the KF recursions require estimation of the mean and covariance of $p\left(x_k, y_k | y_{1:k-1}\right)$. With the system described by Eq. (1.1) with noise prop-

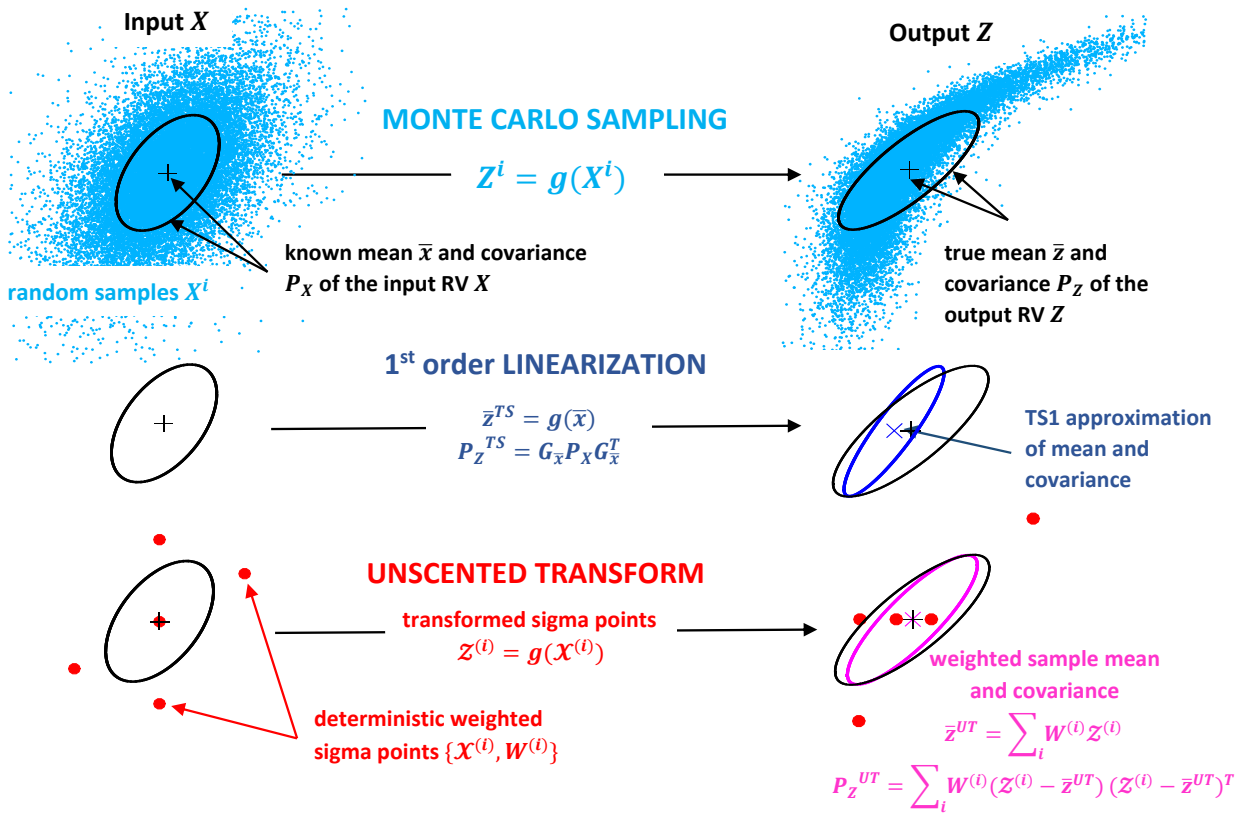


Figure 3.1: Mean and covariance propagation: actual (Monte Carlo sampling), first order linearization TS1, unscented transform UT, adapted from [Wan and van der Merwe, 2000].

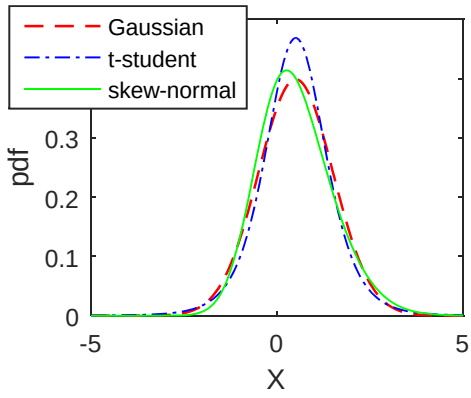
erties as in Eqs. (1.2), one can show using the Markovian properties of the system:

$$p(x_k|y_{1:k-1}) = \int \overbrace{p(x_k|x_{k-1})}^{\text{relates to process equation f}} \underbrace{p(x_{k-1}|y_{1:k-1})}_{\text{posterior at step k-1}} dx_{k-1} \tag{3.3a}$$

$$p(y_k|y_{1:k-1}) = \int \overbrace{p(y_k|x_k)}^{\text{relates to measurement equation h}} \underbrace{p(x_k|y_{1:k-1})}_{\text{prior at step k}} dx_k \tag{3.3b}$$

So in the prediction step, the posterior at time step $k-1$ $x_{k-1}|y_{1:k-1}$ undergoes the transformation f to output the new random variable (RV) $x_k|y_{1:k-1}$, i.e. the prior at time step k. In the update step, one computes the pdf of the posterior $x_k|y_{1:k}$ as the conditional probability of the prior $x_k|y_{1:k-1}$ knowing the actual measurement y_k. This requires knowledge of the distribution of $y_k|y_{1:k-1}$, obtained by transformation of the RV $x_k|y_{1:k-1}$ through h, possibly nonlinear. Thus for both the propagation and update steps, we are looking for the pdf, or moments, of a RV that undergoes a transformation. In the following, we will consider an input RV X, which undergoes a potentially nonlinear transformation g to output a new RV $Z = g(X)$.

In linearization one tries to approximate the nonlinear function g through its Taylor Series expansion, then uses this linear approximation to find the mean and covariance of the new random variable Z (leads to the EKF). The unscented transform (UT) [Julier and Uhlmann, 1996] on the contrary estimates directly the mean and covariance of Z using a set of transformed points (see Fig. 3.1 for a schematic representation of linearization vs. UT, adapted from [Wan and van der Merwe, 2000]). Those sigma points are deterministically chosen so that some of their moments match those of the input RV X. Then those points are transformed using g, and the statistics of the new random variable Z are computed using weighted averages over the transformed set of points. The accuracy of the UT depends on how many moments of the input RV X are captured by the sigma points set.



Distribution of input RV X $E[X]=\frac{1}{2}, var(X)=1$	Moments of output RV $Z=g(X)=X^2$	
	$E[Z]$	$var(Z)$
Gaussian $N(\cdot;\frac{1}{2},1)$	5/4	3
t-Student $St(\cdot;\frac{1}{2},\frac{2}{3},6)$	5/4	6
Skew-normal $SN(\cdot;-\frac{1}{2},2,\sqrt{\frac{\pi/4}{1-\pi/4}})$	5/4	$1+\pi$

Figure 3.2: Influence of higher order moments of the input RV X on the covariance of the output RV $Z=g(X)=X^2$; the plot shows the pdfs of the input RV X (three pdfs that exhibit different 3rd and 4th order moments); the table shows the true value of the mean and variance of the output RV Z, computed exactly using Eqs. (3.4), the mean value is the same in all 3 cases, however the covariance changes drastically depending on higher order moments of the distribution of the input RV X.

Sigma point set	N_{sig}	accuracy/comments
Simplex set [Julier, 2003, Julier and Uhlmann, 2002]	n_X+2	captures the mean and covariance of the input RV, no assumption is made on higher order moments, thus it is accurate to the second order for computation of the mean of the output RV, and to the first order for its covariance (see discussion in section 3.1.2).
Symmetric set [Julier and Uhlmann, 1997]	$2n_X+1$	captures the mean and covariance of the input RV, the 3rd order for a symmetric distribution, and minimize the error on the marginal 4th order moments of a Gaussian RV (if parameter $\kappa=3-n_X$). It yields 2nd order accuracy for the mean, 1st order accuracy for the covariance.
4th order Gaussian, [Julier and Uhlmann, 1997]	$2n_X^2+1$	captures up to the 4th order moments of a Gaussian input RV, thus it is accurate to the second order for the computation of both the mean and covariance of the output RV (only if the input RV is Gaussian!).
A note on scaling: because of possible issues that can appear in high dimensions with certain sets of points, [Julier, 2002] presents a scaling transformation to minimize the effect of high order terms in high dimensions and incorporate partial information about the 4th order terms. This transformation was initially applied to the simplex set but is very often used on the symmetric set of points. Roughly, scaling transforms a set of points $\mathcal{X}^{(i)}$ as: $\mathcal{X}^{(i)}_{scaled}=\mathcal{X}^{(0)}+\alpha(\mathcal{X}^{(i)}-\mathcal{X}^{(0)})$ If parameter α is small, higher order moments $E[(X-E[X])^n]$, $n\geq 3$ become close to 0. An additional parameter β is introduced to reduce error on the kurtosis; $\beta=2$ is optimal for a Gaussian distribution.		

Table 3.1: Description and discussion on accuracy of different sigma points sets to be used in the UT. The sets are plotted on Fig. 3.3.

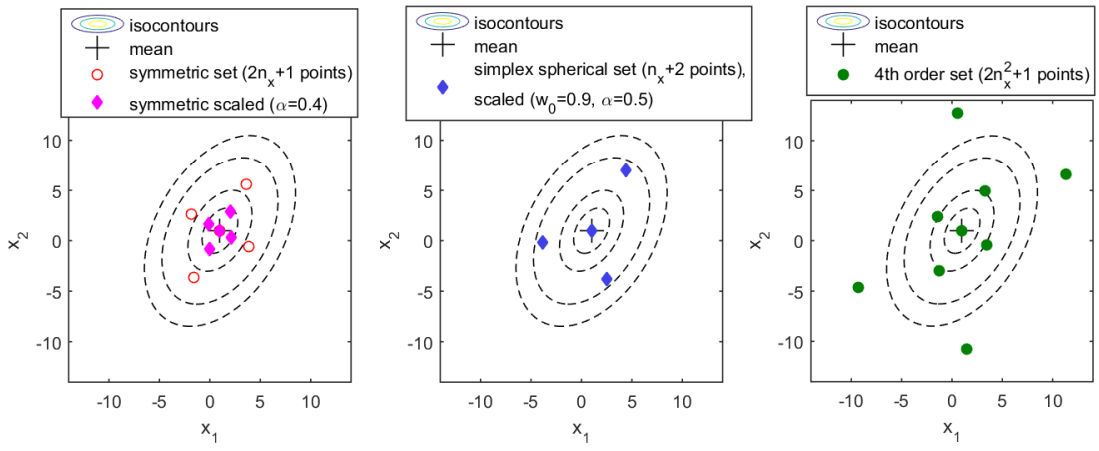


Figure 3.3: Location of sigma points for a Gaussian input RV X, comparison of different sigma point sets described in Table 3.1.

Theoretical aspects: influence of higher order moments

For any function g, the second order approximations of the mean and covariance of $Z = g(X)$ can be written as (in the univariate case, following for instance [Ang and Tang, 2007]):

$$E[Z] \simeq g(\mu_X) + \frac{1}{2}\sigma_X^2 \frac{d^2 g}{dX^2} \tag{3.4a}$$

$$Var(Z) \simeq \sigma_X^2 \left(\frac{dg}{dX}\right)^2 - \frac{1}{4}\sigma_X^2 \left(\frac{d^2 g}{dX^2}\right)^2 + E\left[(X-\mu_X)^3\right] \frac{dg}{dX}\frac{d^2 g}{dX^2} + \frac{1}{4}E\left[(X-\mu_X)^4\right]\left(\frac{d^2 g}{dX^2}\right)^2 \tag{3.4b}$$

These are second order approximations in the sense that they keep only terms that contain up to the second order derivatives of g. We can observe that the second order approximation of the variance $Var(Z)$ involves higher order (third and fourth) moments of the input RV X. Fig. 3.2 gives a sense of how much higher order moments of X influence the covariance of the output RV Z with a simple example.

Going back to the UT, a set of sigma points is chosen to match some of the moments of X. Several sigma point sets have been derived in the literature. It is shown in [Julier, 2003, Julier and Uhlmann, 2002] that $N_{sig} = n_X + 2$ points (where n_X is the dimension of the state vector X) are

enough to capture the first and second order moments of the input RV X. Thus this set achieves a second order accuracy on the mean (Eq. (3.4a)), but only a first order accuracy on the covariance, since third and fourth order moments of X are not matched (it does not achieve Eq. (3.4b)). Other sigma points sets, like the widely used symmetric set [Julier and Uhlmann, 1996] or the fourth order set [Julier and Uhlmann, 1997] differ on how they account for higher order moments of the input RV X, and will lead to different accuracy on the estimated moments of the output RV Z. In [Julier, 2002], a scaling transformation is derived that leads to a new way to take into account third and higher order moments. Table 3.1 summarizes these sigma points sets and their accuracy.

Two examples

The moment propagation schemes previously described differ in the accuracy of the mean and covariance calculation. We will explain this more precisely in two examples.

Example 1. A quadratic transformation. In [Julier and Uhlmann, 1996] the example of a univariate RV $X \sim \mathcal{N}(\cdot; m, P)$ that undergoes the quadratic transformation $g(X) = X^2$ is considered. It is shown that the UKF with a symmetric set of sigma points leads to accurate results on both the mean and covariance, contrary to the first order EKF. In [Gustafsson and Hendeby, 2012], a similar example is considered where a n_X-dimensional RV $X \sim \mathcal{N}(\cdot; 0, I)$ undergoes the quadratic transformation $g(X) = X^T X$. We consider here X in dimensions 2 and 3 and look at the estimates obtained with the UT using different sigma point sets. Table (3.2) shows that only the second order Taylor series and the fourth order UT yield correct results on both mean and covariance in both cases ($n_X = 2, 3$). This was expected since we have already mentioned that the UT using either the simplex set or the symmetric set, scaled or not, yield estimates with only first order accuracy on the covariance. Since here g is quadratic, one needs a second order accuracy to output the accurate results for the covariance.

Example 2. Highly nonlinear transformation of a bivariate random variable. The sec-

	$n_X = 2$ mean	$n_X = 2$ variance	$n_X = 3$ mean	$n_X = 3$ variance
true values	2	4	3	6
first order Taylor Series	0 ✗	0 ✗	0 ✗	0 ✗
second order Taylor Series	2 ✓	4 ✓	3 ✓	6 ✓
UT spherical simplex ($N_{sig} = n_X + 2$, $w_0 = 0.5$)	2 ✓	4 ✓	3 ✓	9 ✗
UT spherical simplex, scaled ($N_{sig} = n_X + 2$, $w_0 = 0.5$, $\alpha = 1e-3$, $\beta = 2$)	2 ✓	8 ✗	3 ✓	18 ✗
UT symmetric set ($N_{sig} = 2n_X + 1$, $\kappa = 3 - n_X$)	2 ✓	2 ✗	3 ✓	0 ✗
UT symmetric set, scaled ($N_{sig} = 2n_X + 1$, $\kappa = 0$, $\alpha = 1e-3$, $\beta = 2$)	2 ✓	8 ✗	3 ✓	18 ✗
UT 4th order ($N_{sig} = 2n_X^2 + 1$, $\kappa = 0$, $\alpha = 1e-3$, $\beta = 2$)	2 ✓	4 ✓	3 ✓	6 ✓

Table 3.2: Comparison of different methods to estimate mean and variance of the output RV $Z = X^T X$, with $X \sim \mathcal{N}(\cdot; 0, I)$ (example 1). Since g is quadratic, only the second order Taylor series and the 4th order UT, which both captures up to 4th order moments of a Gaussian, are accuracte in estimating the covariance of Z. Other sigma point sets are accurate only for estimation of the mean.

ond example presented here consists of a 2-dimensional variable $X = \begin{Bmatrix} X_1 \\ X_2 \end{Bmatrix}$ that undergoes the following nonlinear transformation:

$$g(X) = \begin{Bmatrix} g_1(X_1, X_2) \\ g_2(X_1, X_2) \end{Bmatrix} = \begin{Bmatrix} 8X_2 \\ X_1 - 2|X_1||X_2|^{3-1}X_2 - 1X_1|X_2|^3 \end{Bmatrix} = \begin{Bmatrix} Z_1 \\ Z_2 \end{Bmatrix} \tag{3.5}$$

This function is typical of a hysteretic system with Bouc-Wen model of hysteresis ($X_1 \leftarrow \dot{x}$ and $X_2 \leftarrow r$). It is non-differentiable at $X_1 = 0$ and $X_2 = 0$. Fig. 3.4 shows the mean and covariance of the output RV Z. Both the symmetric set (not scaled) and the fourth order sets perform well. The scaled symmetric set ($2n_X + 1$ points) and the scaled simplex set ($n_X + 2$ points) perform exactly the same. This can be expected since scaling (with small α, 10^{-3} here) reduces the influence of third and higher order moments (i.e., it sets the higher order moments of X to 0). Thus we expect the same behavior from any scaled set of points (with small α), regardless of how many points it contains (as long as it still matches the mean and covariance of X).

3.1.3 The UKF for uncertainty quantification

We have seen that in theory, the same order of accuracy is not as easily achieved for the estimation of the covariance as for the estimation of the mean when using the UT for propagation of moments, which might in turn affect the behavior of the UKF. This issue is not usually tackled in the literature since no true paths are easily available for assessing the estimation performance of the covariance. Furthermore, values for the covariance are usually of less interest. For SHM purposes however,

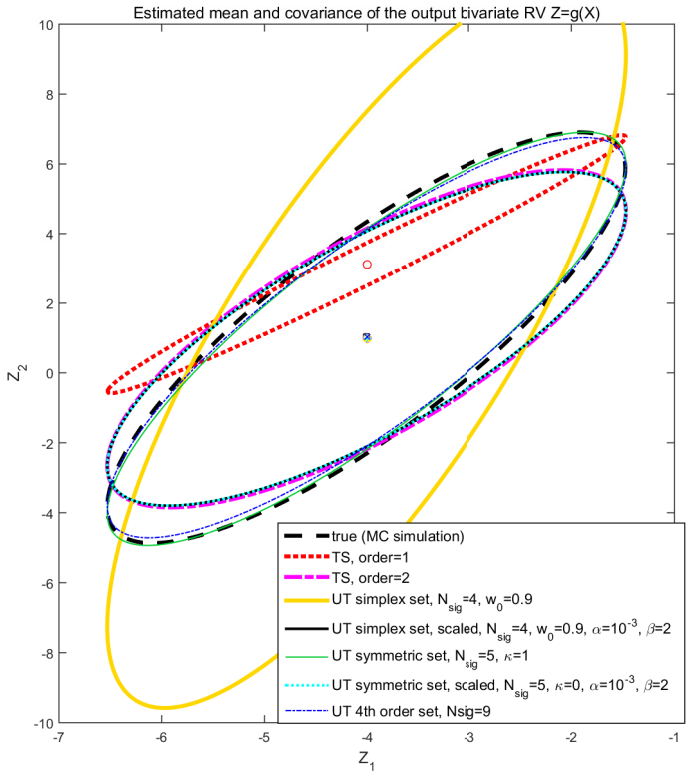


Figure 3.4: Computation of mean and covariance of a transformed bivariate random variable (example 2). Comparison between Taylor Series linearizations, and several unscented sigma points sets. The "true" covariance was obtained by MC simulation. All methods perform well in recovering the variance of $Z_1 = g_1(X) = 8X_2$. This seems logical since g_1 is linear in this case, and all methods are known to perform well for linear functions. On the contrary, for estimation of $var(Z_2)$ and $cov(Z_1, Z_2)$, where $Z_2 = g_2(X)$ and g_2 is highly nonlinear, the TS1 and unscaled simplex methods perform poorly. Scaled transformations, both with the symmetric set and the simplex set, perform exactly the same. The unscaled symmetric set and the 4th order sets perform the best in this case.

knowledge about the covariance can be of crucial interest since it enables the user to quantify uncertainty on the states/parameters, and thus on the presence of damage, which would further allow one to integrate these techniques in more reliable probabilistic decision-making tools, as presented in the motivating example of chapter 1.

In the following section we will look at 3 different single degree-of-freedom (SDOF) systems, with different nonlinearities, that are relevant in the context of structural systems. We assess performance of the covariance estimates by comparing results from UKFs to results from particle filter schemes, known to give an estimation of the true posterior pdf as $n_p \to \infty$.

3.2 Assessment of UKF estimates for identifiable nonlinear systems

3.2.1 Analysis of covariance: methodology

For each problem we run a PF along with several UKFs, using different sigma point sets. As previously stated, the particle filter yields at each time step k the true posterior pdf, thus the true mean and covariance of $p(x_k|y_{1:k})$, with the condition that $n_p \to \infty$. Since this is not achievable in practice, we decided here to run the PF several times with a large number of particles, then to present statistics of the results. More precisely, for each run of the PF, using n_p particles, we can compute at each time step k the moments of the posterior pdf $p(x_k|y_{1:k})$ as approximations over particles $\{x_k^{(i)}, w_k^{(i)}\}_{i=1:n_p}$. For instance for the mean and covariance:

$$E[x_k|y_{1:k}] = \sum_{i=1}^{n_p} w_k^{(i)} x_k^{(i)} \tag{3.6a}$$

$$Cov(x_k|y_{1:k}) = \sum_{i=1}^{n_p} w_k^{(i)} (x_k^{(i)} - E[x_k|y_{1:k}])(x_k^{(i)} - E[x_k|y_{1:k}])^T \tag{3.6b}$$

This gives an approximation of mean and covariance over one run of the PF. Performing R runs we can take statistics of each quantity, i.e.

$$E[q] = \frac{1}{R}\sum_{r=1}^{R} q^{(r)} \text{ and } Std(q) = \sqrt{\frac{1}{R-1}\sum_{r=1}^{R}(q^{(r)} - E[q])^2} \tag{3.6c}$$

where q is any scalar quantity (in our numerical examples we look at mean, standard deviation std. dev., skewness parameter and kurtosis of the marginal pdfs of the states/parameters, respectively noted $E[\cdot], \sigma(\cdot), \gamma_1(\cdot), Kurt(\cdot)$ in the plots.) $q^{(r)}$ is a realization of this quantity obtained at the r-th run of the PF, and we plot for each quantity of interest a shaded area for the intervals

$$[E[q] - Std(q), E[q] + Std(q)] \tag{3.6d}$$

We then expect the true value of the quantity to be contained inside this interval (i.e., inside the shaded area in the plots). In this way we take into account the uncertainty inherently present with a PF, since we don't have an infinite number of particles.

For the UKF, it directly outputs mean and covariance of the state that can be directly plotted along with the results of the PF. The idea here is to compare those results with the ones given by the PF to see if the Gaussianity assumption leads to correct results on both the mean and the covariance.

3.2.2 Single-degree-of-freedom examples: results and discussion

We study in this section three SDOF mechanical systems that present different types of nonlinearities in both the propagation and measurement equations:

- a pendulum,
- a Duffing oscillator,
- a Bouc-Wen model of hysteresis.

For each case, the actual noise added to the synthetic data is Gaussian and additive, usually 10% root-mean-square (RMS), and the prior used for all algorithms is also Gaussian. Figures 3.5, 3.6

and 3.7 present more precisely each system, its equations and filtering results for some chosen states and/or parameters. For all three problems, the system is observable and parameters are identifiable, i.e., states/parameters can be theoretically identified from a series of measurements in the absence of noise/uncertainties. We then study the ability of filtering algorithms to 1) accurately estimate the states and parameters and 2) take into account and quantify uncertainties.

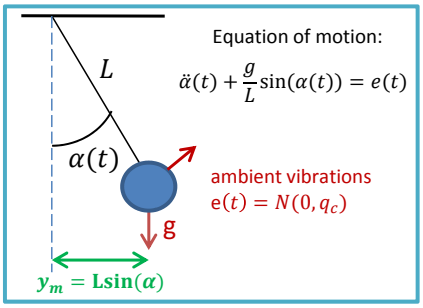
Several general conclusions can be drawn from these three examples. Looking at results from the PF (up to fourth order moments, see Figs. 3.6b, 3.7b, 3.7c), it looks like after convergence of the parameters, the posterior pdfs are actually very close to Gaussian. As a consequence it is not surprising to see that the UKF usually performs very well, after convergence of the parameters is achieved. In particular the UKF with the symmetric point set performs very well in estimating the true value of the parameter, and in recovering both the mean and covariance of the states. However before convergence of the parameters is achieved, a clear discrepancy is observed between results of the particle filter and the UKFs regarding the variance of the states/parameters, which also coincides with values of the higher order moments that do not match those of a Gaussian distribution. This can be very well explained by the theory previously studied (variance depends on higher order moments, thus if the Gaussian approximation is not correct, the output covariance from the UKF will be inaccurate).

Regarding comparison of the different sigma point sets, the unscaled symmetric set ($2n_X + 1$ points) and fourth order set ($2n_X^2 + 1$ points) perform in a very similar fashion (see Fig. 3.6b), thus there is probably no need to use the higher order set. The scaled simplex set ($n_X + 2$ points) usually performs well, but it still looks less robust since it is shown to sometimes fail while the unscaled symmetric set performed well, regardless of the chosen prior. And finally, we observe again that the scaled symmetric and scaled simplex sets (with scaling parameters α small, $\simeq 10^{-3}$) output the exact same results. This was also expected, since scaling reduce higher order moments of the input RX X to 0, regardless of the number of points in the set.

Specific comments on Fig. 3.5. Values used were $n_p = 8000$ particles to run the generic bootstrap PF, averaged over $R = 10$ runs. For this example the shaded area is very narrow, implying that each run of the PF yields very consistent estimates of the quantities of interest, that fit well the true path (parameter is well identified). Regarding the UKFs, except for the simplex set that behaves poorly, all the UKFs perform well since they are able to recover the expected value and covariance (slightly overestimated) of the true posterior pdf, given by the PF.

Specific comments on Fig. 3.6. At the beginning of the filtering process, before convergence is achieved, results of the PF clearly shows higher order moments (third and fourth) that differ from moments of a Gaussian distribution. At the same time UKF algorithms do not give estimates of the mean and the covariance in accordance with results from the PF, which yields the true pdf. For two algorithms, namely the sigma point set (unscaled) and the fourth order set (here the principle of marginalization is applied to run this fourth order set, see section 4.3.1 in the following chapter), this does not influence convergence, however for the simplex set and the scaled unscented set, convergence to the true value is not achieved. For the two algorithms that achieve convergence, the estimate of the covariance at convergence is in agreement with the estimate given by the PF. It has to be noted that behavior of the UKFs depend on the prior chosen for the states/parameters: in some cases all UKFs yield good results. The symmetric, unscaled, UKF set seems to be the most robust (in the sense that it very rarely fails).

Specific comments on Fig. 3.7. The hysteresis variable shows higher values for the third and fourth order moments at the beginning of the time series than the velocity state, indicating that its pdf is quite non-Gaussian. Again, when the Gaussianity assumption does not hold, large discrepancies between the variance estimates from the PF and the UKFs can be observed. However, after some time the pdf tends toward Gaussianity and all UKFs are capable of recovering mean and covariance of the pdf of the states. Same conclusions for the static parameters.



Static parameter to identify: $\theta = L$.

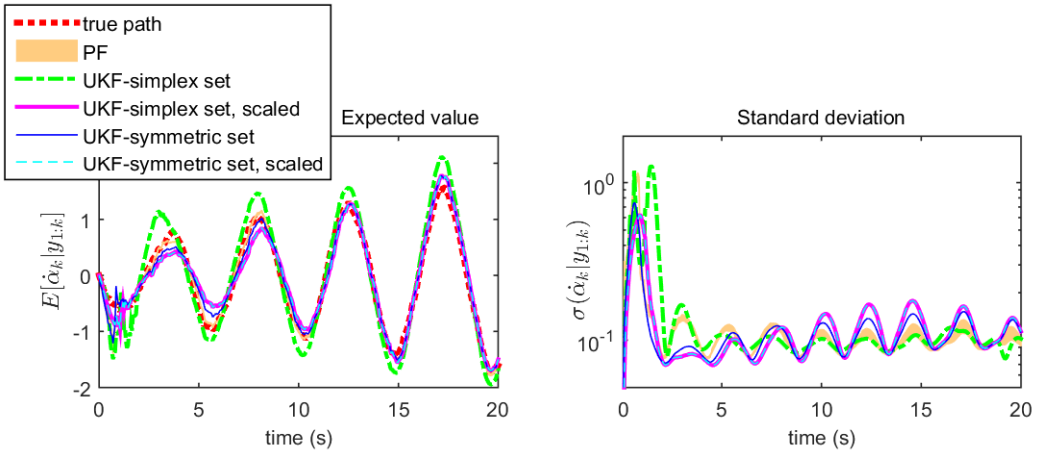
State-space formulation for filters:

State vector: $x_k = \begin{Bmatrix} \alpha \\ \dot{\alpha} \\ L \end{Bmatrix}_k$

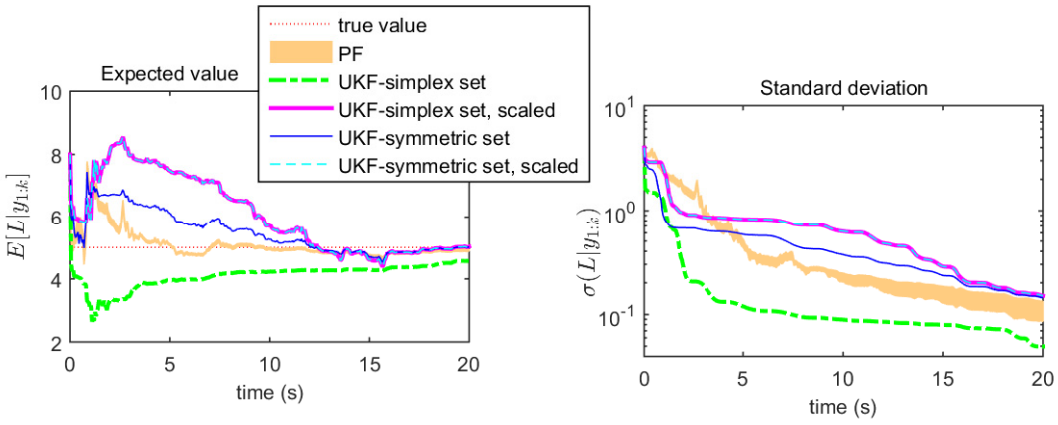
Propagation equation: $\begin{Bmatrix} x_{1,k} \\ x_{2,k} \\ x_{3,k} \end{Bmatrix} = \begin{Bmatrix} x_{1,k-1} + dt \cdot x_{2,k-1} \\ x_{2,k-1} - dt \cdot \frac{g}{x_{3,k-1}} \sin(x_{1,k-1}) + dt \cdot e_k \\ x_{3,k-1} \end{Bmatrix}$

Measurement equation: $y_{m,k} = x_{3,k}\sin(x_{1,k})$

(a) System #1: SDOF pendulum, example adapted from [Särkkä, 2013]

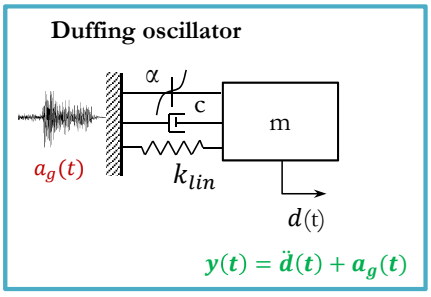


(b) Evolution in time of mean and std. dev. of $p(\dot{\alpha}_k|y_{1:k})$ (filtering of angular velocity $\dot{\alpha}$)



(c) Evolution in time of mean and std. dev. of $p(L|y_{1:k})$ (filtering of static parameter $\theta = L$)

Figure 3.5: Comparison of estimates (mean and std. dev.) from UKFs and PF schemes for the pendulum system.

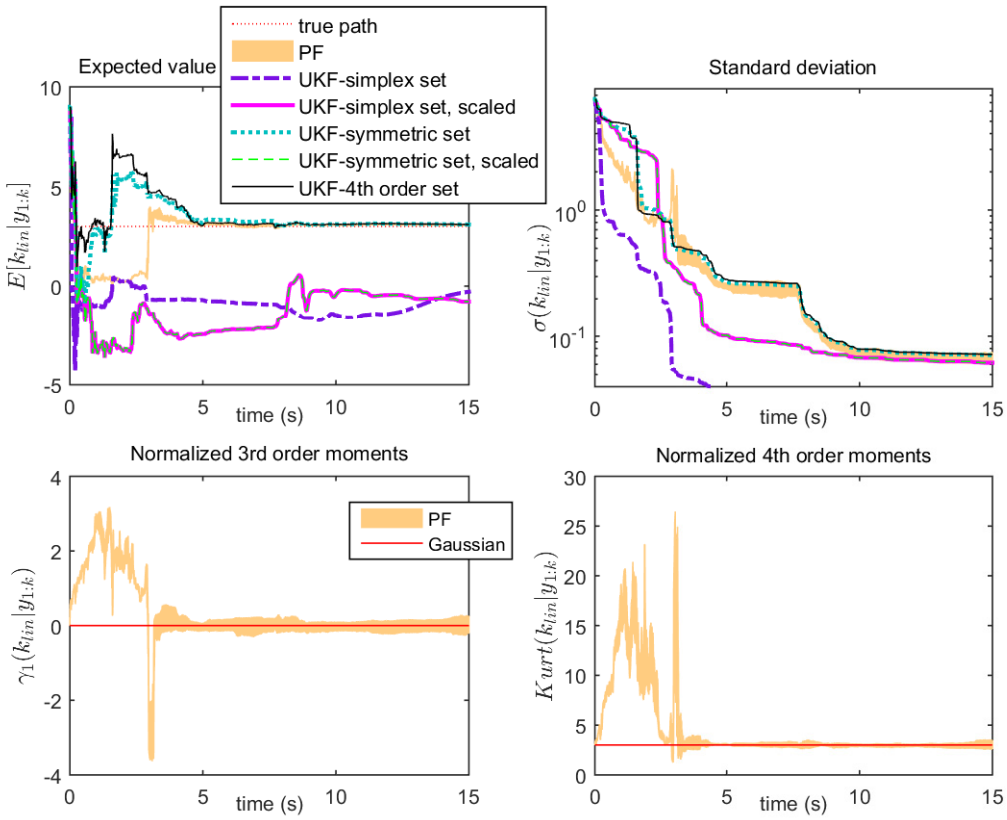


Equation of motion:

$$m\ddot{d} + c\dot{d} + k_{lin}d + \alpha d^3 = -ma_g$$

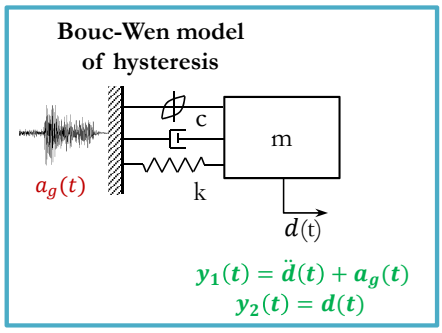
State vector $= [d \quad \dot{d} \quad k_{lin} \quad c \quad \alpha]^T$
$m = 1$ is known

(a) System #2: SDOF Duffing oscillator



(b) Evolution in time of the moments (expectation, std. dev., skewness, kurtosis) of the posterior pdf $p(k_{lin}|y_{1:k})$ of the linear stiffness parameter

Figure 3.6: Comparison of estimates (1st to 4th order moments) from UKFs and PF schemes for the SDOF Duffing system.



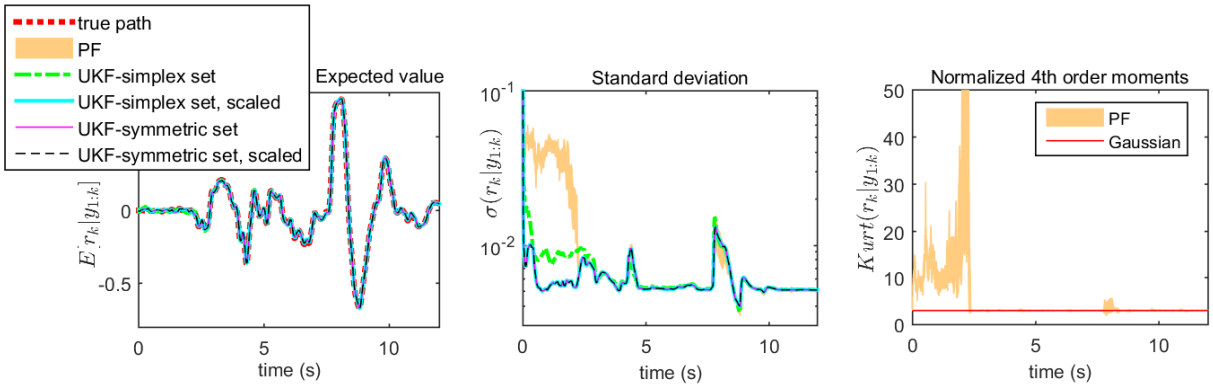
Equations of motion:

$$m\ddot{d} + c\dot{d} + kd = -ma_g$$

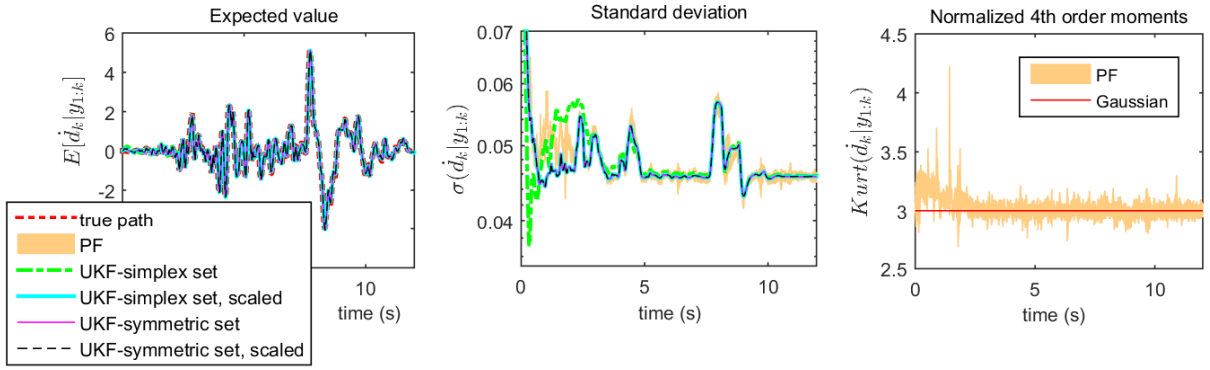
$$\dot{r} = \dot{d} - \beta|\dot{d}||r|^{n-1} - \gamma\dot{d}|r|^n$$

State vector $x_k = [d \quad \dot{d} \quad r \quad k \quad c \quad \beta]^T$
$m = 1, n = 3$ and $\gamma = 0$ are known

(a) System #3: SDOF Bouc-Wen model of hysteresis



(b) Evolution in time of moments (mean, std. dev., kurtosis) of $p(r_k|y_{1:k})$ (filtering of the highly nonlinear hysteresis variable r)...



(c) ... and of $p(\dot{d}_k|y_{1:k})$ (filtering of the velocity $\dot{d}$)

Figure 3.7: Comparison of moment estimates from UKFs and PF schemes for the SDOF Bouc Wen model of hysteresis system.

3.2.3 Increasing the dimension of the system

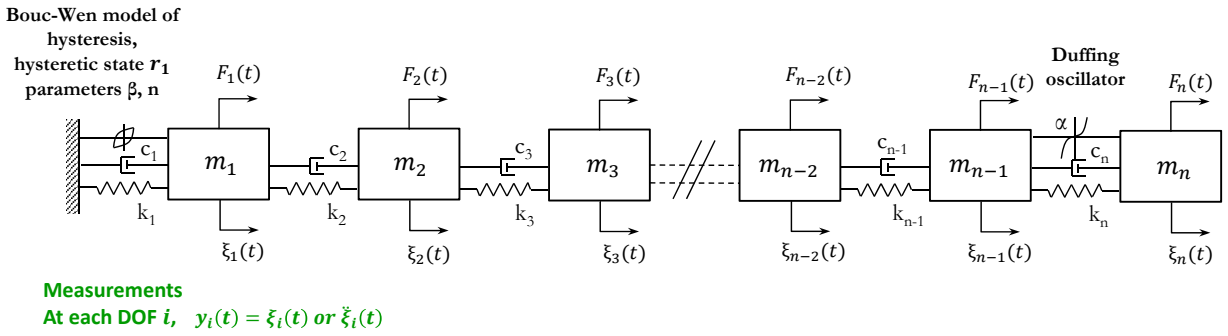
In the previous section, small scale systems were studied in order to enable running a particle filter and thus obtain the true posterior pdf of the states/parameters. In this section, we show that the UKF is indeed capable of handling a much larger system, for which the particle filter would be much too cumbersome to run. The chosen system is a 30-DOF structure with 7 nonlinearities (one Bouc-Wen model of hysteresis and 6 Duffing oscillators throughout the height of the structure), schematically represented in Fig. 3.8a. Measurements (displacements or accelerations) are available from all floors, thus rendering the system identifiable. Fig. 3.8b shows filtering of some of the states (top row) and evolution in time of the error between the posterior mean and the true value of the parameters (second row), demonstrating that the UKF still yields very acceptable results when the dimension of the system is largely augmented. Running a UKF on this 30-DOF system, which includes 61 dynamic states and 68 static parameters, takes about 18 minutes on a work computer, thus enabling efficient damage assessment of such a structure after a large amplitude event for instance.

3.3 Influence of noise

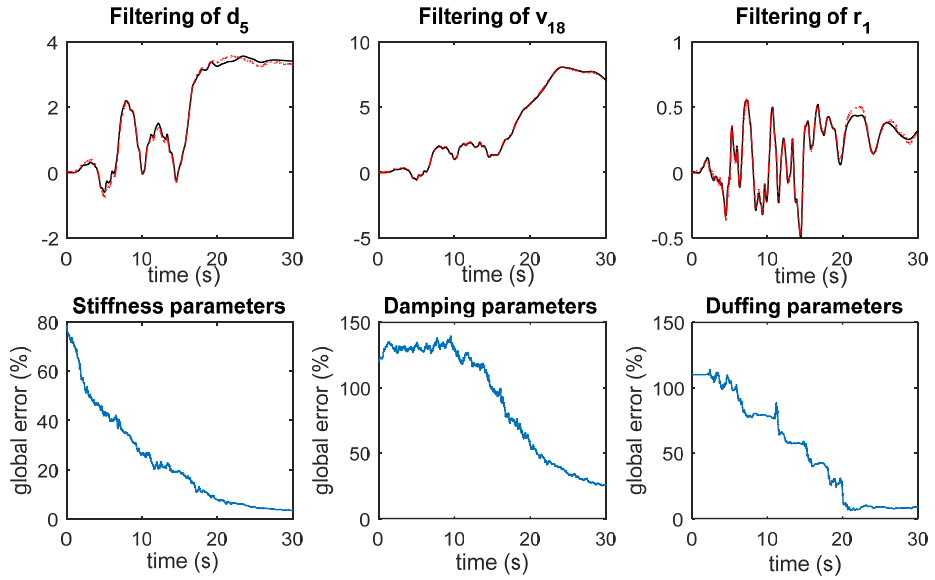
3.3.1 What happens when the true noise is non-Gaussian?

All the examples considered previously used additive Gaussian noise, however in real-life systems this might not be the case. Sensors might be broken, and more importantly many excitations, which cannot be directly measured, could not be represented as Gaussian noise.

Revisiting system #2 (Fig. 3.6), but now considering the case where the noise added to the simulated data is corrupted by outliers and can be described as follows (as done in [Roth et al.,



(a) Schematic representation of a n-DOF system



(b) Top row: filtering of some states (evolution in time of $E\left[x_y^{dyn}|y_{1:k}\right]$); bottom row: evolution in time of the average error for some static parameters

Figure 3.8: Filtering of a 30-DOF identifiable system with a UKF

2013]):

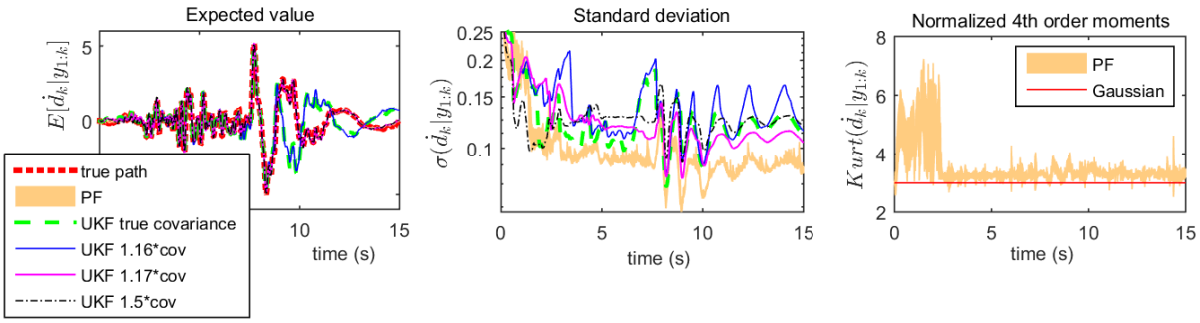
$$v_k \sim \begin{cases} \mathcal{N}(\cdot;0,25Q_{exc}) & \text{with probability } 10\% \\ \mathcal{N}(\cdot;0,Q_{exc}) & \text{with probability } 90\% \end{cases} \tag{3.7a}$$

for the noise added to the excitation, and

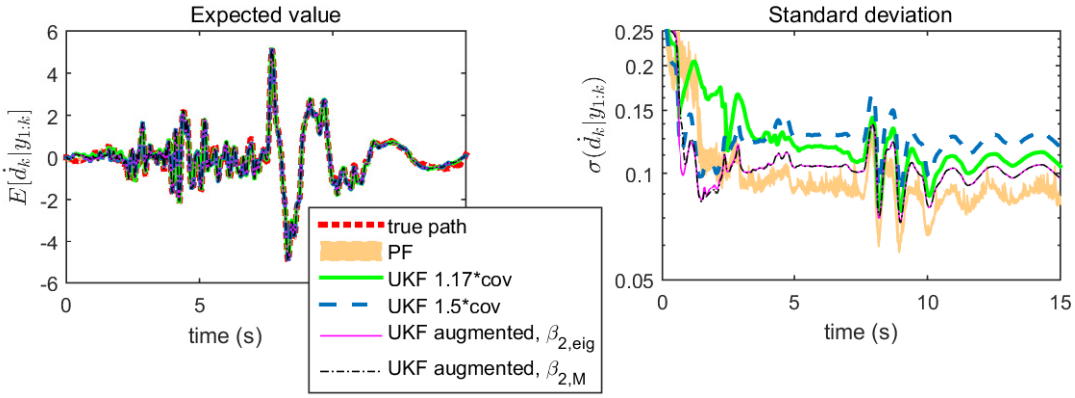
$$\eta_k \sim \begin{cases} \mathcal{N}(\cdot; 0, 25R_{meas}) & \text{with probability } 10\% \\ \mathcal{N}(\cdot; 0, R_{meas}) & \text{with probability } 90\% \end{cases} \tag{3.7b}$$

for the measurement noise, with Q_{exc} and R_{meas} the covariance of the noise added to the excitation and the measurement time histories in the previous example (10% RMS, Gaussian). Adding outliers will result in noise that show higher order moments than those of a Gaussian distribution, especially higher fourth order moments (kurtosis). Eqs. (3.7) actually represent mixtures of Gaussians, thus the variance and fourth order moments of these two noise distributions can be computed exactly.

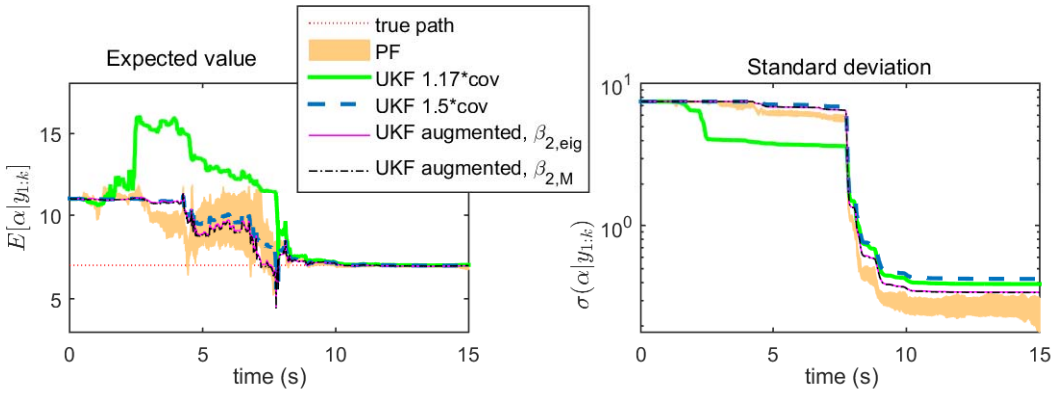
In this experiment, we added noise as described in Eqs. (3.7) to the true data, then we tried to solve this problem with both the PF (the Gaussian mixture noise is taken into account through the use of an indicator value that determines which mixture to consider for each particle, see [Särkkä, 2013]) and the UKF. Fig. 3.9a shows the true moments (PF) of the velocity $\dot{d}_k|y_{1:k}$, as well as the behavior of the UKF when the unscaled symmetric set, is used. First we can observe that even at convergence, the posterior of state $\dot{d}$ is not exactly Gaussian anymore, since the normalized fourth order moments are (slightly) higher than 3. Regarding performance of the UKF, if the covariance parameters Q and R of the filtering algorithm are chosen as the true covariance of the noise added to the data, the UKF fails (green curve: the expected value does not match the true path, because the parameters do not converge to their true value). This is because the algorithm is unable to process the higher order moments present in the noise/state. A well-known trick in that case is to increase the covariance of the noise. Here we gradually augment both Q and R; we can see that we need to augment Q and R by 17% (pink line) to obtain convergence to the true values/path. However using these parameters, the covariance of the state is overestimated, more so if we go on increasing Q and R (black line).



(a) Filtering of velocity $\dot{d}$ with a generic UKF, when noise added to the data has outliers



(b) Filtering of the velocity $\dot{d}$ and ...



(c) ... the static parameter α (nonlinear stiffness) when the new sigma point set that takes into account higher order moments of the noise is used in the UKF.

Figure 3.9: Filtering of some states/parameters when noise has outliers: using a generic sigma point set for the noise vs. using a sigma point set that captures higher kurtosis of the noise terms.

3.3.2 A new sigma point set to take into account higher order moments of the noise

We derive in this section a new sigma point set that enables one to take into account higher fourth order moments *of the noise*, assuming that these statistics are known. In the generic UT, the parameter κ is used to tune the fourth order moments of the sigma point set to a Gaussian pdf (0 excess kurtosis). The main idea here is to use three different values of κ: one for the state that is still assumed to be Gaussian, thus the linear Kalman filter update can be used, and one for each of the noise terms, which can show different kurtosis behavior.

We use in this section the augmented version of the UKF, as presented in [Wan and van der Merwe, 2000], i.e. we augment the state vector with the noise terms and we consider that f and h are functions of the dynamic states, the static parameters and the noise terms:

$$x_k = f\left(x_{k-1}, e_k, v_k\right) \text{ (propagation equation)} \tag{3.8a}$$

$$y_k = h\left(x_k, \eta_k\right) \text{ (observation equation)} \tag{3.8b}$$

Thus the noise need not be additive anymore, however we still keep the assumptions described in Eq. (1.2), especially v_k and η_k are uncorrelated to each other and uncorrelated to the states/parameters at each time step.

The dimension of the augmented state is now $n_{aug} = n_x + n_v + n_\eta$. The mean and covariance of the augmented state are $\bar{x}_{aug}$ and P_{aug} respectively. Now consider a set of $N_{sig} = 2n_{aug} + 1$ sigma points $\mathcal{X}^{(i)}$ and their respective weights $W^{(i)}$ defined as follows:

$$\mathcal{X}^{(0)} = \bar{x}_{aug}, \qquad W^{(0)}, \tag{3.9a}$$

$$\mathcal{X}^{(i,n_{aug}+i)} = \bar{x}_{aug} \pm \sqrt{n_{aug} + \kappa_1}\sqrt{P_{aug}}_i, \qquad W^{(1)}, \text{ for } i = 1 : n_x, \tag{3.9b}$$

$$\mathcal{X}^{(n_x+i,n_{aug}+n_x+i)} = \bar{x}_{aug} \pm \sqrt{n_{aug} + \kappa_2}\sqrt{P_{aug}}_i, \qquad W^{(2)}, \text{ for } i = 1 : n_v, \tag{3.9c}$$

$$\mathcal{X}^{(n_x+n_v+i,n_{aug}+n_x+n_v+i)} = \bar{x}_{aug} \pm \sqrt{n_{aug} + \kappa_3}\sqrt{P_{aug}}_i, \qquad W^{(3)}, \text{ for } i = 1 : n_\eta \tag{3.9d}$$

where $\sqrt{P_{aug}}$ is such that $\sqrt{P_{aug}}\sqrt{P_{aug}}^T = P_{aug}$ and $\sqrt{P_{aug}}_i$ is the i^{th} column of $\sqrt{P_{aug}}$. We want this set to capture the mean and covariance of the augmented state, which can be written as:

$$\sum_{l=1}^{N_{sig}} W^{(l)} \mathcal{X}^{(l)} = \bar{x}_{aug} = \begin{bmatrix} x_{k|k} \\ 0 \\ 0 \end{bmatrix} \tag{3.10a}$$

$$\sum_{l=1}^{N_{sig}} W^{(l)} (\mathcal{X}^{(l)} - \bar{x}_{aug})(\mathcal{X}^{(l)} - \bar{x}_{aug})^T = P_{aug} = \begin{bmatrix} P_{k|k} & 0 & 0 \\ 0 & Q & 0 \\ 0 & 0 & R \end{bmatrix} \tag{3.10b}$$

which gives the following constraints over the weights:

$$W^{(0)} = 1 - \frac{n_x}{n_{aug} + \kappa_1} - \frac{n_v}{n_{aug} + \kappa_2} - \frac{n_\eta}{n_{aug} + \kappa_3} \tag{3.11a}$$

$$W^{(1)} = \frac{1}{2(n_{aug} + \kappa_1)}, W^{(2)} = \frac{1}{2(n_{aug} + \kappa_2)}, W^{(3)} = \frac{1}{2(n_{aug} + \kappa_3)} \tag{3.11b}$$

To find the value of the $\kappa_{1,2,3}$, we use information on the fourth order moment of the state x and the noise terms v, η. Higher order unscented transforms have been derived in the literature (e.g., [Liu et al., 2014, Ponomareva et al., 2010, Tenne and Singh, 2003]) that consider higher order moments of the marginal distributions, and sometimes an average over all dimensions. However there exist other measures of multivariate kurtosis that are widely used for hypothesis testing (Gaussian testing of a data set), and take into account, to a certain extent, correlation between dimensions in the definition of the kurtosis parameter. One widely known measure of kurtosis for a d-variate vector X with mean $E[X]$ and covariance Σ is defined in [Mardia, 1970] as:

$$\beta_{2,M} = E\left[\{(X - E[X])^T \Sigma^{-1} (X - E[X])\}^2\right] \tag{3.12}$$

which equals to $d(d+2)$ if X is Gaussian. We then compute the values of $\kappa_{1,2,3}$ so that the set of sigma points captures the value of $\beta_{2,M}$, assumed known, for the states, the process noise and the measurement noise separately. For the states, we assume its posterior pdf will be Gaussian, so $\beta_{2,M}(X) = n_x(n_x+2)$. For the noise vectors, $\beta_{2,M}$ is assumed known (if the pdf of the noise is known it can be computed exactly, or it should be approximated somehow). The value of $\beta_{2,M}$, as

well as the values of Q and R, can vary with time. Using the fact that the states and noise terms are uncorrelated, one can uncouple computations of the three κ parameters. More precisely, $\kappa_{1,2,3}$ are computed using the following equations:

$$(n_{aug}+\kappa_1)\sum_{i=1}^{n_x}\{\sqrt{P_{xi}}^T P_x^{-1}\sqrt{P_{xi}}\}^2 = \beta_{2,M}(x) = n_x(n_x+2) \tag{3.13a}$$

$$(n_{aug}+\kappa_2)\sum_{i=1}^{n_v}\{\sqrt{Q_i}^T Q^{-1}\sqrt{Q_i}\}^2 = \beta_{2,M}(v) \tag{3.13b}$$

$$(n_{aug}+\kappa_3)\sum_{i=1}^{n_\eta}\{\sqrt{R_i}^T R^{-1}\sqrt{R_i}\}^2 = \beta_{2,M}(\eta) \tag{3.13c}$$

Another possibility is to use the measure of multivariate kurtosis defined in [Srivastava, 1984], by:

$$\beta_{2,S} = \frac{1}{d}\sum_{i=1}^{d}\frac{E\left[(Y_i-\theta_i)^4\right]}{\lambda_i^2} \tag{3.14}$$

where Y_i, θ_i are the projections of X and $E[X]$ on the i^{th} eigenvector of $\Sigma = cov(X)$, and λ_i the corresponding eigenvalue. For a Gaussian RV X, $\beta_{2,S} = 3$. One can also derive a set of sigma points based on this measure of kurtosis: sigma points defined in Eqs. (3.9) and weights defined in Eqs. (3.11) remain the same, but Eq. (3.14) is used to compute $\kappa_{1,2,3}$. This involves a projection onto the eigenvectors of the covariance matrix P_{aug}, which can be more easily obtained if $\sqrt{P_{aug}}$ is defined using the eigenvalue decomposition of P_{aug}, instead of the usual Cholesky decomposition usually used in the UKF.

Results using this new set of points for system #2 with outliers in the noise are shown in Figs. 3.9b and 3.9c. We can see that this new set of points yields more accurate estimates of the variance of both the static and the dynamic states.

It is important to highlight here that such a sigma point set can handle only a simplified measure of multivariate kurtosis *of the noise*, the Gaussian assumption on the states must be conserved in order to use the linear Kalman filtering equations. This assumption might be inappropriate if the noise is more complex and the functions f and h highly nonlinear. In the following, we present a

novel framework to derive nonlinear Kalman filters that can handle more complex distributions for both the noise and the states.

3.4 A framework to derive higher order nonlinear Kalman filter schemes

The previous sections highlighted some of the limitations of both the UKF and the PF. On one hand, the UKF makes a Gaussianity assumption for the state, which has two drawbacks: 1) one cannot estimate higher order moments of the posterior pdf and 2) this assumption can lead to inaccurate estimates of the covariance for highly nonlinear non Gaussian systems. On the other hand the PF is, theoretically, capable of handling higher order moments, however a very large number of particles is required if one wants to obtain accurate estimates of the moments (large sampling variance).

In this section we present a framework that would enable one to derive higher order Kalman filter algorithms. The main idea is to propagate and update the parameters of a distribution in a similar fashion as done with the Gaussian distribution in the UKF. This distribution can be chosen so that its higher order moments can vary and be propagated/updated consistently at each time step. As will be explained in more detail later, this requires both a moment propagation scheme (UT for instance) that propagates higher order moments through nonlinear transformations, as well as a measurement update set of equations that updates higher order moments. In this fashion one could have access to higher order moments of the posterior pdfs and handle more complex non-Gaussian noise, while avoiding the sampling variance issue observed when using particle approximations of the pdfs.

3.4.1 Kalman filtering using non-Gaussian probability distributions

As previously mentioned, Gaussian Kalman filter type algorithms make two approximations:

- use an approximation (MC, linearization, or UT) to estimate the moments of RVs that undergo nonlinear transformations,
- assume Gaussianity of the pdfs, and thus use the linear Kalman filter measurement update equations.

Lately, research has focused on two possible improvements of the Kalman filter schemes: higher order sigma point sets for the UKF, and the derivation of linear Kalman filters that assume a different underlying distribution than the Gaussian one.

Higher order UKFs

The unscented transform can be used to capture moments of a non Gaussian RV, for example non zero skewness terms, as presented in [Julier, 1998]. However when this sigma point set was inserted in a UKF, results did not show much improvements from a typical UKF with a symmetric point set. The authors explained that this could be partially due to the linear measurement update. Indeed the measurement update equations are linear and strictly involve mean and covariance of the RV, thus higher order moments are not used and propagated during the measurement update. In [Liu et al., 2014] this problem is tackled by propagating the higher order moments through a recursive estimation model consistent with the optimal Kalman filtering framework. However, from a Bayesian estimation perspective, the linear measurement update was derived using a Gaussian assumption for the RVs, thus using it to propagate higher order moments (which are fixed in a Gaussian distribution) shows a kind of contradiction in the method.

The method we propose instead is to combine higher order unscented transforms (or Monte Carlo approximations) with update equations that propagate higher order moments; in this way the higher order statistics are taken into account consistently at each step of the algorithm.

Linear filters that make use of a different probability distribution

To present filters that use different underlying distributions than the Gaussian one, it is easier to start with linear filters. Indeed, using the same (Bayesian) reasoning as for the Gaussian Kalman filter, one can derive linear filters using different distributions than the Gaussian one, as long as these distributions possess the three following properties:

- closure under linear transformation (and closure under marginalization), to propagate the distribution through linear f and h,
- closure under conditioning, to derive measurement update equations,
- closure under addition of independent RVs in this family (if the noise is additive), or more generally closure under joint distribution of independent RVs in this family, to incorporate noise terms in the equations.

Again, the closure under conditioning enables one to write the measurement update equations, i.e., compute the distribution of $p\left(x_k|y_{1:k}\right)$ knowing the joint pdf $p\left(x_k, y_k|y_{1:k-1}\right)$ by conditioning upon the measured value y_k.

Using different distributions enables one to take into account different distributions for the noise (skewed, higher kurtosis...), and/or a different prior distribution $p(x_0)$ to start the algorithm. Two of these linear filters derived in the literature are:

- the t-Student linear filter,
- the closed skew normal (CSN) linear filter.

The linear t-student filter was derived in [Roth et al., 2013]. The multivariate t-student distribution is symmetric (skewness is zero), but it has one additional parameter ν_{stud} (compared to the Gaussian distribution) which permits one to tune the kurtosis. However, the kurtosis should be the same in all dimensions (excess kurtosis = $\frac{6}{\nu_{stud}-4}$), which might not be representative of many multivariate distributions, especially if the dimensions represent different types of variables (static parameters vs. dynamic states for instance). Furthermore, in the derivation of the filter, the noise

distributions should also possess the same parameter ν_{stud}, and thus the same marginal kurtosis, as the state vector, which again might be very restrictive.

Another linear filter was derived in [Naveau et al., 2005] using a closed-skew normal distribution (CSN), which is an extension of the Gaussian distribution allowing for non-zero skewness. It is parameterized by 5 vector/matrix parameters, which allows for different marginal third order moments in different directions, as shown in Fig. 3.10b for a 2-dimensional CSN random variable.

Using the CSN distribution as the underlying distribution for the pdfs, one can derive a nonlinear filter, using one of the three approximations previously mentioned for the Gaussian case. An extension to nonlinear systems is presented in [Rezaie and Eidsvik, 2014] that uses a MC approximation to propagate the system through a nonlinear process equation, however the measurement equation h is linear, thus the linear filter update part can be used directly. In the following we consider a nonlinear extension of this filter based on the unscented transform, and is capable of handling both f and h nonlinear.

3.4.2 The unscented CSN filter

The closed skew normal distribution

Let's consider a RV $X \in \mathbb{R}^{n_X}$ that follows a CSN distribution, i.e., $X \sim CSN_{n_X,q}(\mu, \Sigma, \Gamma, \nu, \Delta)$. Its pdf is written as follows:

$$p(X) = \left[\Phi_q(0; \nu, \Delta + \Gamma\Sigma\Gamma^T)\right]^{-1} \Phi_q(\Gamma(X-\mu); \nu, \Delta)\phi_n(X; \mu, \Sigma) \tag{3.15}$$

where Φ_q and ϕ_q are the cdf and pdf respectively of the q-variate Gaussian distribution, $\mu \in \mathbb{R}^{n_X \times 1}$, $\Sigma \in \mathbb{R}^{n_X \times n_X}$, $\Gamma \in \mathbb{R}^{q \times n_X}$, $\nu \in \mathbb{R}^{q \times 1}$ and $\Delta \in \mathbb{R}^{q \times q}$, and q is a positive integer. The matrices Σ and Δ should be symmetric, positive definite. For this distribution, μ and Σ do not equate to the mean and covariance of the distribution. A CSN distribution with either $q = 0$ or $\Gamma = 0$ is Gaussian. For derivation of properties, moment generating function etc, see for instance [González-Farías et al.,

2004].

Moment matching with the CSN distribution

The moments of a distribution can be computed using the moment generating function (c.g.f.):

$$M_X(s) = \frac{\Phi_q(\Gamma\Sigma s; \nu, \Delta + \Gamma\Sigma\Gamma^T)}{\Phi_q(0; \nu, \Delta + \Gamma\Sigma\Gamma^T)} exp\{s^T\mu + \frac{1}{2}s^T\Sigma s\}, \text{ with } s \in \mathbb{R}^{n_X} \tag{3.16}$$

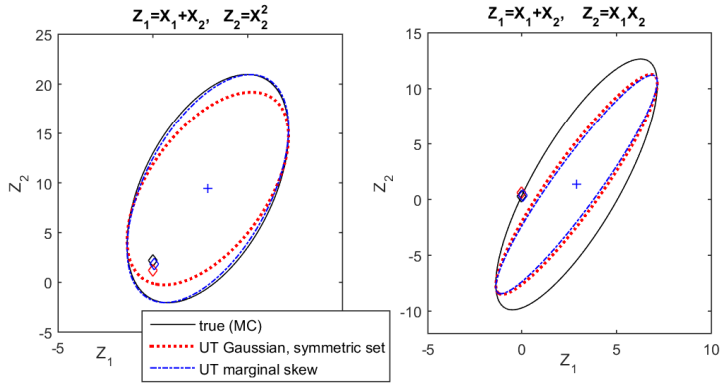
Moments of a CSN distribution are difficult to compute in closed form, because there usually exist no closed form expressions for ϕ_q, Φ_q or their respective derivatives, except for the cases $q = 1$, $q = 2$.

In a nonlinear Kalman filter, one also needs to estimate the parameters of the distribution knowing its moments (moment matching). Contrary to the Gaussian distribution where moment matching is trivial ($\mu = E[X]$ and $\Sigma = Cov(X)$), moment matching in the CSN setting is difficult to achieve (requires some kind of nonlinear optimization solver) since the relationship between parameters of the distribution and its moments is very complex, except for the case $q = 1, \nu = 0$ (see appendix B.1). In the following example we restrict ourselves to CSN distributions with $q = 1, \nu = 0$ to simplify moment matching.

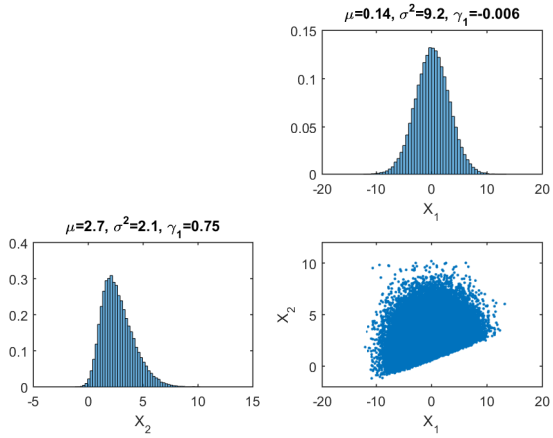
Sigma point set that captures up to the third order moment of a RV

In this section, we assume that the CSN distribution and the moments of an input RV X are known. We want to find a sigma point set that captures up to the third order moments of X, and thus obtain more accurate estimates of the moments of $Z = g(X)$. We will see that depending on the type of nonlinearity present in the function g, one might need to capture all skew terms of X (marginal skew terms and cross skew terms), or use marginalization to capture some of them.

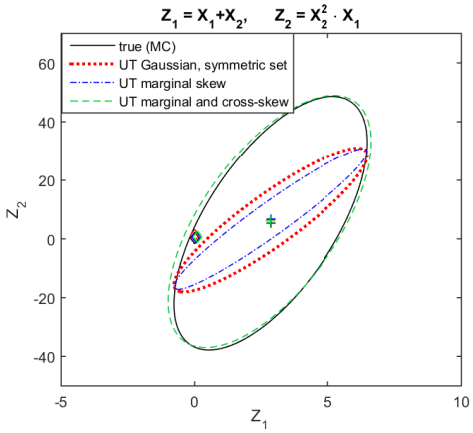
In [Julier, 1998], a set of points that captures the mean, covariance and skewness of a distribution is derived. It is composed of three shape sets, each of them capturing one skew term $E\left[X_jX_jX_j\right]$, $E\left[X_jX_jX_k\right]$ and $E\left[X_jX_kX_l\right]$. Capturing those three terms becomes very cumbersome



(a) Symmetric set vs. novel set that captures marginal skew



(b) Distribution of the input RV X (bivariate CSN), obtained by MC sampling



(c) Comparison of several sigma point sets for a cubic transformation g, the green line (best performance) is obtained using a set that takes into account cross-skew terms of X

Figure 3.10: Performance of several sigma point sets on estimating mean (cross marker), covariance (contour line) and marginal skewness (diamond marker), for several transformations g of a bivariate CSN X

if the state vector is large, many sigma points will be needed (N_{sig} scales with n_X^3). However one can use a simplification of this set, where only the marginal skew term $S_j = E\left[X_j X_j X_j\right]$ is captured in each dimension, which requires only $2n_X + 1$ sigma points, as for the symmetric set (using solely the first shape set from [Julier, 1998]). A sigma point set that captures the known mean $E[X] = m$, covariance P_{XX}, marginal skew terms S^X, and minimize the error on the average kurtosis, is derived in appendix C.1.

Figure 3.10a shows results obtained with this new sigma point set, for two different nonlinearities. In both cases g_1 is linear in X ($Z_1 = X_1 + X_2$), and g_2 is nonlinear. For the left plot, $Z_2 = X_2^2$, and we can see that our set performs better than the symmetric set in recovering the covariance of the output set. Indeed looking at Fig. 3.10b, we can see that the input X_2 is clearly skewed, however the symmetric set does not take this into account, creating error in the estimation of the output covariance. Our set however takes into account both skewness and kurtosis terms of the input RV X, and thus outputs a more accurate estimate of the covariance. Also, first order accuracy is achieved in estimating the skew of Z, and it can be observed that again our estimate of the skew (diamond shape) is better than the one obtained with a symmetric set. However, looking at the right plot of Fig. 3.10a, it can be observed that for a nonlinearity of the type $Z_2 = X_1 X_2$ this new set performs in the same way as the symmetric set. This is probably due to the fact that this nonlinearity involves cross terms, and thus the cross skew terms of X should also be captured to obtain accurate results.

The most straightforward way to catch all the skew terms of the input RV X would be to use the full set presented in [Julier, 1998] (number of sigma points scales with n_X^3). However in systems of interest for SHM, nonlinearities usually involve cross terms such as kx or $c\dot{x}$, thus we mainly want to catch cross skew terms involving multiplication of a state and a parameter. We then used a different approach to bypass this problem, using marginalization of some states or parameters (method adapted from [Morelande and Moran, 2007], see section 4.3.1 in the following chapter). Fig. 3.10c shows results on a cubic transformation of the form $\begin{bmatrix} Z_1 \\ Z_2 \end{bmatrix} = \begin{bmatrix} X_1 + X_2 \\ X_1 X_2^2 \end{bmatrix}$. We see in the figure that this new method (in green) is more accurate than either the symmetric set of the simple skew set in estimating the covariance of the output RV Z.

The unscented CSN filter, additive noise case

In this section we present a nonlinear filter that aims at propagating the mean, covariance and third order moments of a distribution, using as baseline pdf the CSN distribution. The noise is assumed additive, however non-additive noise can also be considered as previously mentioned. The algorithm would follow the following steps:

- Start with knowledge of the posterior pdf at time step $k-1$, $p(x_{k-1}|y_{1:k-1}) \sim CSN_{n_x,1}(\Theta_{k-1})$, and its moments
- **Propagation step:** estimate the prior pdf $p(x_k|y_{1:k-1})$
  - Compute sigma points that capture the mean, covariance and skewness of the input RV $x_{k-1}|y_{1:k-1}$
  - Propagate them through f
  - Compute the moments (or cumulants) of the output RV $f(x_{k-1}|y_{1:k-1})$
  - Compute the moments of the prior pdf $p(f(x_{k-1}|y_{1:k-1}) + \nu_{k-1})$, using additivity and independence of the noise and thus additivity of the cumulants
- **Update step:** estimate the posterior pdf $p(x_k|y_{1:k})$
  - Compute sigma points that capture the mean, covariance and skewness of the input RV $x_k|y_{1:k-1}$
  - Propagate them through h
  - Compute the moments (or cumulants) of the output RV $h(x_{k-1}|y_{1:k-1})$ and the cross-moments between the input and output RVs
  - Use the additive property of the measurement noise η_k to compute the moments of the joint pdf $p(x_k, y_k|y_{1:k-1}) = p(x_k|y_{1:k-1}, h(x_k|y_{1:k-1}) + \eta_k)$
  - Fit a CSN ($q = 1,2$) distribution to the joint pdf $p(x_k, y_k|y_{1:k-1}) \sim CSN_{n_x+n_y,q}(\Theta_{xy})$, by matching its mean, covariance and third order moments

– Use the closure under conditioning property of the CSN distribution (see appendix B.2) to condition upon the observed value y_k and thus find the parameters Θ_k of the posterior distribution $p(x_k|y_{1:k}) \sim CSN_{n_x,q}(\Theta_k)$ as:

$$\mu_k = \mu_x + \Sigma_{xy}\Sigma_{yy}^{-1}(y_k - \mu_y), \tag{3.17a}$$

$$\Sigma_k = \Sigma_{xx} - \Sigma_{xy}\Sigma_{yy}^{-1}\Sigma_{yx}, \tag{3.17b}$$

$$\Gamma_k = \Gamma_x, \tag{3.17c}$$

$$\nu_k = \nu - (\Gamma_y + \Gamma_x\Sigma_{xy}\Sigma_{yy}^{-1})(y_k - \mu_y), \tag{3.17d}$$

$$\Delta_k = \Delta \tag{3.17e}$$

The steps followed in this algorithm are actually similar to the ones in the well-known Gaussian unscented Kalman filter, with the main difference that third order moments are taken into account, and the measurement updates equations are modified. However, two steps are much more complicated to derive in this CSN framework (compared to the Gaussian framework):

1. compute a sigma point set that captures up to the third order moments of the input RV,
2. fit a CSN ($q = 1, 2$) to the known moments of a RV.

Numerical example

We looked at one small dimensional problem to test our algorithm. The benchmark logistic maps problem is modified to include skewed process and measurement noise with the following system equations:

$$x_k = (1 - x_{k-1})x_{k-1}\theta + \nu_k \tag{3.18a}$$

$$y_k = x_k + \eta_k \tag{3.18b}$$

where $\theta = 3.91$ is a static parameter, ν_k is skewed (mean 0, variance $7 \cdot 10^{-14}$, skewness parameter $\gamma_1 = 0.6$) and η_k is also skewed (mean 0, variance $8 \cdot 10^{-5}$, skewness parameter $\gamma_1 = -0.9$), and

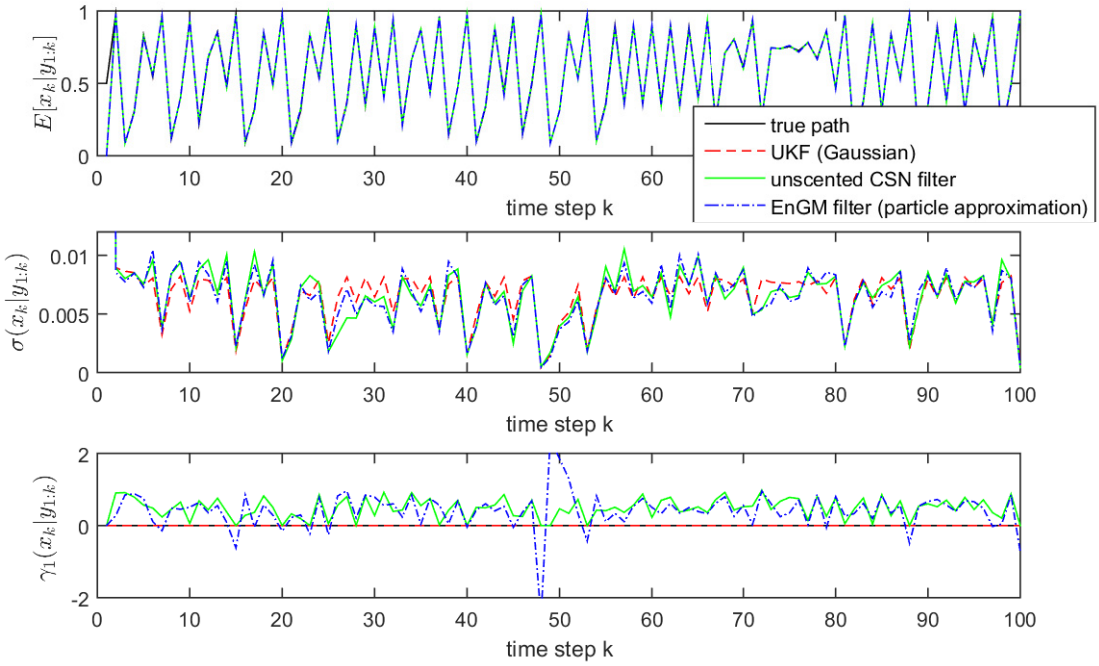


Figure 3.11: Inference of the dynamic state in a non Gaussian system: comparison of several nonlinear Kalman filter algorithms.

are sampled from a CSN distribution when we generate simulated data.

Here we consider the simple case where θ is known and we are just following the behavior of the state x. Results are shown in figure 3.11, where the normal UKF, the unscented CSN filter and the ensemble GM filter (described in the next section, has higher order of accuracy) are compared. We can observe that the unscented CSN filter and the higher order EnGM filter both behave similarly in recovering the mean, variance and skewness parameter of the state, which shows that the CSN indeed performs well in this case, and that accurately keeping track of the skewness influences learning of the covariance. One can observe though that at $k \simeq 50$, the skewness reaches a relatively high value, and the CSN filter cannot approximate it correctly, showing that the CSN distribution with $q = 1, \nu = 0$ is not flexible enough to represent highly skewed distributions.

3.4.3 The ensemble Gaussian Mixture filter

The CSN distribution considered previously is still similar to a Gaussian distribution in the sense that it can represent only unimodal distributions, and its high order statistics are fixed. For more complex distributions, one should consider particle filtering. However, the particle filter is known to undergo a degeneracy issue that can lead to dramatic collapse during the weighting/resampling step, issue that is exacerbated when static parameters are added to the system. Here we present an nonlinear Kalman filter type algorithm, which uses as distribution baseline a mixture of Gaussians (GM), and MC approximation to propagate the distributions through nonlinear functions. This idea of fitting a mixture of Gaussians to a particle approximation has already been seen in the literature, for instance in the GMSPPF previously mentioned in the particle filtering review section. The GMSPPF makes use of the unscented transform to derive a proposal distribution from which particles are drawn, then weighted according to the importance sampling principle. On the contrary the algorithm presented thereafter does not rely on the unscented transform, it uses MC simulation to propagate through both f and h.

Derivation of the algorithm

As previously mentioned, to derive a nonlinear Kalman filter type algorithm, one needs two components:

- a baseline distribution that captures the needed moments and is closed under conditioning (for measurement update),
- a choice of approximation to propagate the distribution and its moments through a nonlinear function.

The GM distribution is closed under conditioning (see appendix B.3), and its parameters are usually learnt through maximum likelihood (EM algorithm), thus a MC approximation seems appropriate here.

The algorithm follows the following steps:

- Start with knowledge of the posterior distribution at time step $k-1$

$$p(x_{k-1}|y_{1:k-1}) \sim GM(\pi_{k-1}^{(l)}, \mu_{k-1}^{(l)}, \Sigma_{k-1}^{(l)})$$

  ($l = 1 : N_{mix}$, number of Gaussians in the mixture), and its moments

- **Propagation step:** estimate the prior pdf $p(x_k|y_{1:k-1})$

  - Sample from both the input RV $x_{k-1}|y_{1:k-1}$ and the noise ν_{k-1}
  - Propagate each sample through $f(x_{k-1}|y_{1:k-1}) + \nu_k$

- **Update step:** estimate the posterior pdf $p(x_k|y_{1:k})$

  - Sample from the noise η_k
  - Propagate each sample through $h(x_k|y_{1:k-1}) + \eta_k$
  - Fit a GM distribution (through EM algorithm) to the samples to estimate the joint pdf $p(x_k, y_k|y_{1:k-1}) \sim GM(\pi_{xy}^{(l)}, \mu_{xy}^{(l)}, \Sigma_{xy}^{(l)})$
  - Use the closure under conditioning property of the GM distribution to condition upon the observed value y_k and thus find the parameters $\pi_k^{(l)}, \mu_k^{(l)}, \Sigma_k^{(l)}$ of the posterior distribution $p(x_k|y_{1:k}) \sim GM(\pi_k^{(l)}, \mu_k^{(l)}, \Sigma_k^{(l)})$ as:

$$\pi_k^{(l)} = \frac{\pi_{xy}^{(l)} \mathcal{N}\left(y_k; \mu_y^{(l)}, \Sigma_{yy}^{(l)}\right)}{\sum_{j=1}^{L} \pi_{xy}^{(j)} \mathcal{N}\left(y_k; \mu_y^{(j)}, \Sigma_{yy}^{(j)}\right)} \tag{3.19a}$$

$$\mu_k^{(l)} = \mu_x^{(l)} + \Sigma_{xy}^{(l)} (\Sigma_{yy}^{(l)})^{-1} (y_k - \mu_y^{(l)}) \tag{3.19b}$$

$$\Sigma_k^{(l)} = \Sigma_{xx}^{(l)} - \Sigma_{xy}^{(l)} (\Sigma_{yy}^{(l)})^{-1} \Sigma_{yx}^{(l)} \tag{3.19c}$$

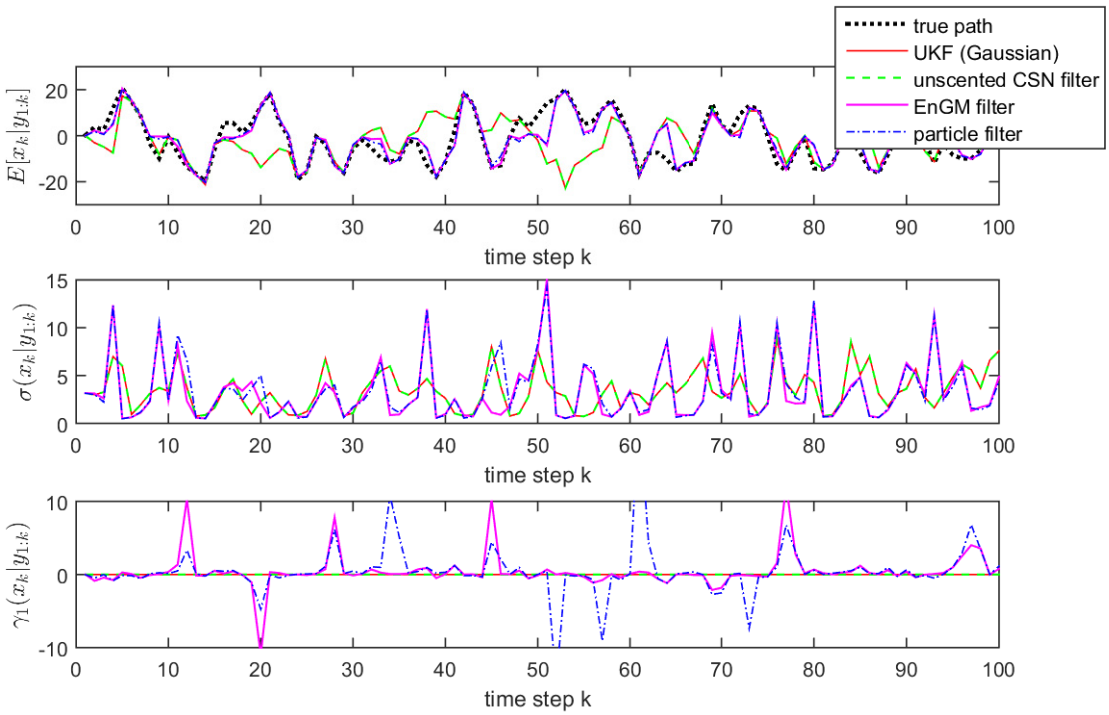


Figure 3.12: Results of several Kalman filter type filters and particle filter for a benchmark highly nonlinear 1-dimensional system.

Results on a highly nonlinear problem

This algorithm is tested on a highly nonlinear benchmark problem from the particle filtering literature:

$$x_k = \frac{x_{k-1}}{2} + 25\frac{x_{k-1}}{1+x_{k-1}^2} + 8cos(1.2k) + v_k \tag{3.20a}$$

$$y_k = \frac{x_k^2}{20} + \eta_k \tag{3.20b}$$

where $v_k \sim \mathcal{N}(\cdot;0,1)$ and $\eta_k \sim \mathcal{N}(\cdot;0,10)$. Even though the noise terms are Gaussian, this problem is known to lead to multimodal posterior distributions for certain time steps, because of the high nonlinearities in both equations.

Results comparing several Kalman filter type filters and particle filters are shown in Fig. 3.12. One can see that both the particle filter and our novel EnGM filter perform similarly in recovering

both expectation and covariance of the state, and that their approximation of the mean is close to the true state. In contrast the UKF and CSN filters perform poorly in recovering the mean, thus showing that this highly nonlinear system requires a high order of accuracy (MC sampling) and a more flexible underlying distribution (possibly multimodal). One can observe some discrepancies in the high valued skewness parameters between the particle filter and the EnGM filter, however the sampling variance is quite high for these high valued skewnesses (i.e., two consecutive runs of the same algorithm might give significantly different results), due to the fact that sample estimates of high order moments are quite sensitive to outliers and thus have usually higher sampling variance. One could potentially look at more robust higher order statistics (e.g. [Welling, 2005]) to get a better idea of the higher order statistics. Also, a mixture of 3 Gaussians was used here, more Gaussians might be needed to accurately represent the distribution at those time steps when the skewness is large.

The idea behind this filter is mainly to use a flexible distribution, and accurate approximation of the transformed distributions, as in the PF, but still using closed form expressions for the measurement update. In this way, we might be able to alleviate the degeneracy problem seen in a particle filter type algorithm. Also, particles are resampled at each time step from the approximated posterior pdf, and thus alleviating the sample impoverishment issue for slowly mixing systems (especially when static parameters are added to the state vector). The performance of this algorithm should be tested on a high dimensional system to study its behavior and possible improvements compared to a generic particle filter.

3.5 Concluding remarks

In this chapter, a detailed review of the UKF and its assumptions was performed. Also, the accuracy of the UT (in the UKF framework) and the Gaussianity assumption were inspected for small

structural systems. It thus appears that the UKF, when using a symmetric sigma point set, yields very accurate estimates of the posterior mean and covariance of the states and parameters compared to a PF, as long as 1) the noise terms are actually Gaussian and 2) the systems studied are identifiable. In these cases, the Gaussianity assumption made in the UKF is quite correct, and the UKF yields acceptable results with a much reduced computational time compared to a PF.

For cases where the noise terms are known to be non-Gaussian, but the posterior pdf of the state itself is still expected to be close to Gaussian, one can use an enhanced sigma point set to take into account higher order moments *of the noise* when propagating moments, while still using the linear Kalman filter measurement update equations (derived using a Gaussianity assumption on the states).

For cases where the Gaussianity assumption *on the state* might be incorrect, it is possible to use the nonlinear Kalman filtering framework to derive enhanced Kalman filtering schemes that use as a baseline a different distribution than the Gaussian one, if possible a more flexible distribution (e.g., multi-modal, or with adjustable higher order moments). In this chapter, this framework was solely tested on two highly nonlinear, non-Gaussian, mathematical benchmark problems. For structural systems, non-Gaussianity can arise due to a lack of identifiability in the system (the data is not informative enough to accurately learn all the parameters). Using the framework derived therein, a Gaussian mixture unscented Kalman filter can be derived to tackle this type of challenging problems, with a reduced computational time compared to a PF. This work is the focus of a subsequent chapter (chapter 5).

The systems considered in the first two chapters of this dissertation contained relatively few DOFs; they could be used as reduced order models to represent the main dynamics of a larger system. However, in some cases it might be useful to perform identification on larger high-fidelity representations such as finite element models that are used a lot in practice in the fields of civil, mechanical, aerospace or even bio-mechanical engineering. Increasing the number of DOFs in

the system will greatly increase the computational burden of filtering algorithms. In the following chapter we present a novel method, also based on the principle of marginalization, to help alleviate this problem when making use of the UKF for parameter identification.

Chapter 4

A marginalized unscented Kalman filtering approach for efficient parameter estimation in finite element models

4.1 Literature review: dimension reduction in Gaussian filters

As explained in the introduction of this dissertation, the on-line formulation of filtering algorithms presents the advantage that process noise, which might encompass stochastic excitations and modeling errors, can be easily taken into account. However, it comes at the expense that uncertainty around both the dynamic states x^{dyn} and the parameters θ must be quantified and propagated. For the UKF more specifically, this means that the number of sigma points, and thus the number of function evaluations, will be proportional to the size of the augmented state vector $n_x = n_{x^{dyn}} + n_\theta$. This renders the use of the UKF cumbersome for systems where the number of DOFs is large such as finite element (FE) models, systems that are of great interest in the civil and mechanical engineering communities where FE analysis (FEA) is used a lot in practice. This dimensionality issue

also arises in other fields, for instance state estimation in ocean or atmospheric data assimilation (see e.g. [Ambadan and Tang, 2009]).

To mitigate this issue, methods have been derived to decrease the number of sigma points of the UKF, and consequential its overall computational burden. Reduced-order (unscented) Kalman filters (see e.g. [Farrell and Ioannou, 2001, Solonen et al., 2016]) make use of a reduced-order model that capture the "most important" components of the system dynamics, which can be done for instance using a Principal Component Analysis (PCA). The applicability of this method and its potential for computational reduction largely depend on the existence of a low-dimensional space that can represent the system states. This method might thus be quite problem dependent, especially for highly nonlinear structural systems, where the state dynamics will depend not only on the type of nonlinearities present but also on some characteristics of the excitation (frequency, amplitude and so on) that might vary over time. On the contrary, reduced-rank filtering algorithms take into account these variations over time by re-computing the low-dimensional subspace at each time step. Several algorithmic variants exist, the main idea being to compute a low-rank estimate of the full covariance matrix at each time step by solving an eigenvalue problem (see e.g. [Fisher, 2002] for computation of a subspace through the so-called Hessian singular vectors, [Brasseur and Verron, 2006] which reviews developments of such algorithms in the context of the extended Kalman filter, [Padilla and Rowley, 2010] for derivation of a low-rank unscented Kalman filter based on the singular value decomposition of the propagation error matrix). For such algorithms, the question arises of the best choice of truncation threshold in the eigenvalue/singular value decomposition, which, based on a few numerical tests we have performed with this type of algorithm on small structural systems, might also be largely dependent on the system studied and quite difficult to tune.

Most of the literature previously cited deals with state estimation in the context of data assimilation of atmospheric or ocean systems. More closely related to our systems of interest, a reduced-

rank UKF is used for parameter estimation of a FE model inspired from cardiac biomechanics in [Moireau and Chapelle, 2011, Xi et al., 2011]. The size of the low-rank subspace is chosen as the size of the parameter space, thus reducing enormously the number of sigma points in the analysis. It has to be noted however that in one case, the system was quasi-static, thus the uncertainty around the dynamic states did not propagate forward in time, explaining why only uncertainty around the parameters should be considered (more about this will be discussed later in this chapter). In the other paper, an additional Luenberger filter was integrated to the analysis to provide at each time step an improved estimate of the dynamic states, thus reducing uncertainty around the states and enabling the user to use a reduced sigma point set to solely quantify uncertainties around the parameters. The use of the Luenberger state observer was detailed when considering velocity measurements; its applicability to more general measurement types (including acceleration measurements, for which the measurement equation is a function of both the parameters and the states, or combinations of several types of measurements) should be tackled to understand if this method could be generalized to different dynamical systems.

Finally, in [Astroza et al., 2015] parameter estimation in a distributed plasticity FE model is considered using a slightly different approach where the UKF is used in the parameter space only. Not including the dynamic states as part of the state vector implies that 1) there is no process noise with respect to the dynamic states, and thus no explicit way to take into account uncertainty in the excitation and FE modeling errors, and 2) the full analysis with a new sampled value of θ must be run at each time step, from t_0 to t_k, since dynamic states are not updated when new measurements are available.

In the following we present an alternate method to reduce the number of sigma points when performing parameter estimation in dynamical systems represented by a FE model. As for a generic UKF, the augmented state approach is used, thus enabling introduction of process noise with respect to the dynamic states (uncertainty in the excitation for instance). Contrary to the reduced-

order and reduced-rank filters previously described, the method proposed therein does not require truncation of the dynamics, instead it uses the structure of the equations to decompose uncertainties around the dynamic states vs. parameters and uses different moment propagation schemes for each of them. Fundamentally this method relies on the concept of marginalization, which was first applied to increase efficiency of the unscented transform in [Morelande and Moran, 2007]. The method is shown to be generalizable to many types of FE models, and is derived in a way that optimally makes use of linear vs. nonlinear FE analysis, as implemented in generic FEA software, the goal being to enable in the future integration of such a learning algorithm in a generic FE software. In the following section an overview of the method is outlined to explain the fundamental differences between this new approach and the generic UKF as well as its potential for computational time reduction when performing parameter estimation in FE models.

4.2 Marginalization for improved efficiency of parameter estimation in FE models

Integration of an external FEA software with a generic UKF can be easily implemented as shown in Fig. 4.1, due to the fact that the unscented transform is a non-intrusive moment propagation scheme (it requires only function evaluations, no derivatives). However, it will be extremely computationally cumbersome for FE models with many DOFs. In this section it is shown how marginalization can be applied to FE models to reduce computational cost.

Recall that in the UKF, the unscented transform is used to propagate moments through the propagation and measurement equations f, h. Let us go back to a generic setting, where an input RV X with known moments undergoes a nonlinear transformation g to yield the output RV $Z = g(X)$ whose moments we want to estimate. The principle of marginalization consists of partitioning the input state vector as $X = \begin{Bmatrix} a \\ b \end{Bmatrix}$ and decompose computations of output expectations $E[Z] =$

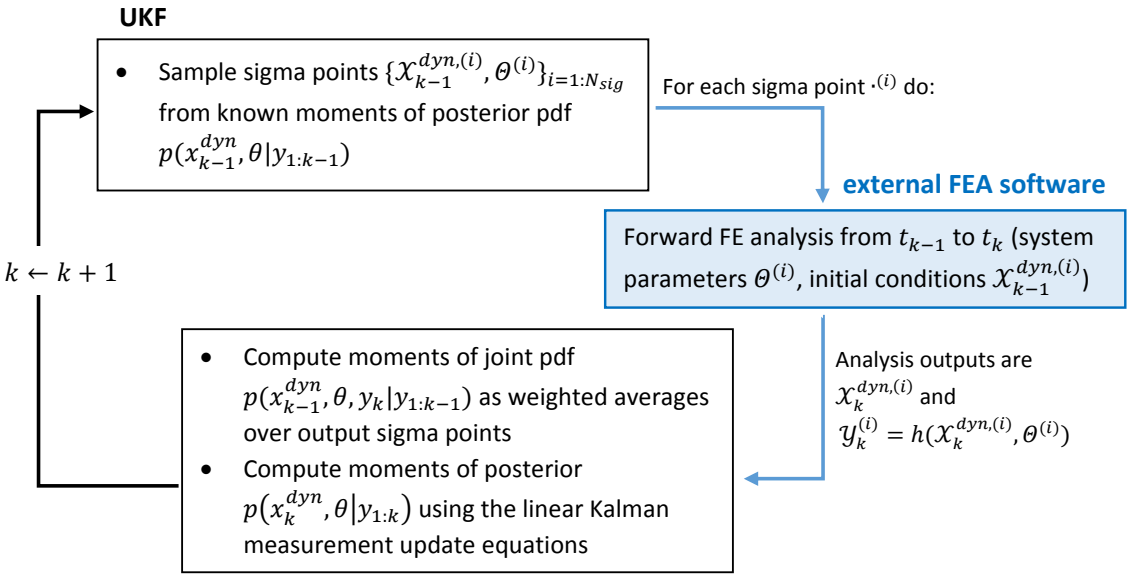


Figure 4.1: Integration of a UKF and external FEA software for forward propagation. The static parameters θ could include material properties, values defining the geometry of the problem and so on.

$E[g(X)]$ (and other moments) as follows:

$$\begin{aligned} E_{p(a,b)}[g(a,b)] &= \int_{a,b} g(a,b)p(a,b)dbda \\ &= \int_a \underbrace{\int_{b|a} g(a,b)p(b|a)db}_{\text{can be computed analytically}} \; p(a)da \end{aligned} \tag{4.1}$$

As explained in [Morelande and Moran, 2007], if the nonlinear function g is conditionally linear in b, i.e., $g(X) = \varphi(a) + G(a) \cdot b$, and the input X is assumed to be Gaussian - assumption that agrees with the nonlinear Kalman filtering framework used in the UKF - then integration over $p(b|a)$ can be performed analytically (full derivation and equations will be given later in this chapter). The UT is then used solely to perform integration over $p(a)$, and the number of sigma points is reduced from $N_{sig} \propto n_x = n_a + n_b$ to $N_{sig} \propto n_a$. In this setting, computation of moments will require function evaluations $g(a,b)$ as in the generic UKF as well as evaluations of the matrix $G(a)$, which happens to be the Jacobian of the transformation g with respect to subvector b. Thus, on condition that 1)

the Jacobian $G(a)$ can be evaluated in an efficient fashion and 2) the reduction in the number of sigma points is relatively consequent (i.e., $n_a \ll n_x$), such a marginalization procedure can lead to a substantial reduction in the filter's computational time.

First, it is shown on a simple example how this procedure can be applied to joint state/parameter estimation in linear dynamical systems. It is important to mention here that the term *linear system* therein refers to a system that behaves linearly, according to the principles of linear vs. nonlinear dynamics. However when performing joint/state parameter estimation on such linear system, the equations f, h become nonlinear functions of the augmented state, due to multiplicative terms such as stiffness $\times$ displacement. In the remainder of the chapter particular attention is thus given to the usage of terms *linear/nonlinear system* vs. *linear/nonlinear functions*. A generic linear system, discretized using a simple forward Euler scheme (more appropriate discretization schemes will be discussed later on), can be represented using the following system equations:

$$\begin{Bmatrix} \mathtt{u}_k \\ \mathtt{v}_k \end{Bmatrix} = \begin{Bmatrix} \mathtt{u}_{k-1} \\ \mathtt{v}_{k-1} \end{Bmatrix} + dt \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix} \begin{Bmatrix} \mathtt{u}_{k-1} \\ \mathtt{v}_{k-1} \end{Bmatrix} + dt \begin{Bmatrix} 0 \\ M^{-1}\mathrm{P} \end{Bmatrix} \tag{4.2a}$$

$$y_k = \begin{bmatrix} H_1 & H_2 \end{bmatrix} \begin{Bmatrix} \mathtt{u}_k \\ \mathtt{v}_k \end{Bmatrix} \tag{4.2b}$$

where $\mathtt{u}, \mathtt{v}$ represent the vectors of displacements and velocities respectively (the vector of accelerations will be referred to as $\mathtt{a}$). For a linear system, the system matrices and force vector M, C, K and P do not depend on the dynamic states $x^{dyn} = \begin{Bmatrix} \mathtt{u} \\ \mathtt{v} \end{Bmatrix}$, however they are definitely a function of the static parameters θ. Such a measurement equation as (4.2b), where matrices $H_{1,2}$ may be a function of the parameters θ encompasses displacement, velocity, acceleration or even strain measurements. In the previous equations one can see that, knowing the parameters θ, both the propagation and measurement equations are linear in the dynamic states $x^{dyn} = \begin{Bmatrix} \mathtt{u} \\ \mathtt{v} \end{Bmatrix}$, and that the Jacobian matrices with respect to the dynamic state vector are simply $J_f = \begin{bmatrix} I & dtI \\ -dtM^{-1}K & I - dtM^{-1}C \end{bmatrix}$ and $J_h = \begin{bmatrix} H_1 & H_2 \end{bmatrix}$ for the propagation and measurement

equations respectively. Thus, the marginalization procedure previously described can be used, introducing the partitioning $a \leftarrow \theta$ and $b \leftarrow x^{dyn}$. Using this specific partitioning, the computations of the Jacobians $G = J_{f,h}$ can be computed very efficiently since they are quite simple functions of the system matrices M, C, K, which are necessarily computed by the FEA software during propagation, regardless of the marginalization procedure. Thus this Jacobian computation will not add large computational requirements to the UKF. However, if the number of parameters is much smaller than the number of dynamic states, which would quite often be the case in large FE models, then this marginalization procedure would lead to a large decrease in the number of sigma points, and thus in the number of function evaluations (where function evaluation in this context means running the FEA forward in time). In the following sections it is shown that this process is generalizable to other discretization schemes, for instance the Newmark-beta method, largely employed in FEA due to its unconditional stability (for a given choice of discretization parameters), and the performance of the algorithm is studied on three numerical examples.

Contrary to linear systems for which the restoring force $\mathrm{F} = K\mathtt{u} + C\mathtt{v}$ depends linearly on the states $\mathtt{u}, \mathtt{v}$; nonlinear systems are characterized by a restoring force F that is a nonlinear function of the dynamic states. Very often, the stiffness term K is a nonlinear function of the displacement $\mathtt{u}$, as can be the force term P (e.g., follower force). In this context, the propagation equation cannot be written directly as a linear function of the dynamic states, and integration over $p(b|a) = p(x^{dyn}|\theta)$ cannot be performed exactly. Possibly, a different partitioning of the augmented state vector could be considered, however this solution would undoubtedly be problem dependent and thus difficult to generalize. Instead one can maintain the partitioning previously described, i.e., $a \leftarrow \theta$, $b \leftarrow x^{dyn}$, and approximate the integral $\int_{b|a}$ using a first order Taylor series expansion (method used in a EKF), which will require computation of the Jacobian of the propagation equation with respect to the dynamic states. For instance, in the case of a forward Euler discretization, the propagation

equation and its Jacobian can be computed as follows:

$$\begin{bmatrix} \mathrm{u} \\ \mathrm{v} \end{bmatrix}_{k+1} = \begin{bmatrix} \mathrm{u} \\ \mathrm{v} \end{bmatrix}_{k} + dt \cdot \begin{bmatrix} \mathrm{v}_k \\ M^{-1}\mathrm{P}(\mathrm{u}_k) - M^{-1}\mathrm{F}(\mathrm{u}_k) - M^{-1}C\mathrm{v}_k \end{bmatrix} \tag{4.3a}$$

$$J_f = \begin{bmatrix} I & dt \cdot I \\ -dt \cdot M^{-1}K_t & I - dt \cdot M^{-1}C \end{bmatrix} \tag{4.3b}$$

$$\mathrm{F}(\mathrm{u}_k) \text{ nonlinear restoring force (nonlinear stiffness term)} \tag{4.3c}$$

$$K_t = \frac{\partial\left(\mathrm{F}(\mathrm{u}) - \mathrm{P}(\mathrm{u})\right)}{\partial \mathrm{u}} \text{ tangent stiffness matrix} \tag{4.3d}$$

Again, such an approach would be computationally efficient only if 1) the size of the parameter vector θ is much smaller than the size of the augmented state vector, and 2) the Jacobian can be computed efficiently. Looking back at Eq. (4.3b), one can observe that this Jacobian is again a function of the linear system matrices M, C, as well the so-called tangent stiffness matrix K_t, which is nothing else than the derivatives of the nonlinear stiffness/external force terms with respect to the dynamic states. Efficient calculation of the tangent stiffness matrix is already integrated in any FEA software and is usually performed analytically, since FEA software usually solve for nonlinear problems using a Newton scheme, i.e., subsequent linearizations of the nonlinear problem, which requires multiple computations of the tangent stiffness matrix, and thus an efficient implementation. Furthermore, as for the linear problem, these matrices must be computed during the propagation step, regardless of the marginalization procedure, thus computing the Jacobian would not increase enormously the computational burden per time step. This algorithm is studied more in depth in section 4.4, using more suited discretization schemes, and its potential for parameter identification in nonlinear systems is demonstrated on two examples.

To summarize, the algorithms proposed in this chapter are based on a decomposition of the augmented state vector as dynamic states vs. static parameters to enable treatment of parameter uncertainties using the highly accurate UT, while using an EKF to take into account uncertainties around the dynamic states. This approach is computationally efficient over a generic UKF for

systems that include many dynamic states, as will often be the case for FE models, due to the fact that efficient computations of Jacobians required in the EKF can be related to computations of the system matrices, including the tangent stiffness for nonlinear systems, which are readily implemented in any generic FEA software. It is good to notice however that the EKF is known to be less accurate than the UKF, especially for highly nonlinear systems. Thus the limits of such an algorithm will also be discussed at the end of the chapter.

4.3 Implementation for linear dynamical systems

4.3.1 Marginalized UT for conditionally linear transformations

In this first section, the case of linear dynamical systems is studied. It is shown that applying this principle of marginalization in this case leads to a more efficient algorithm, without loss of accuracy compared to a generic UKF, i.e., no additional approximation is made compared to a generic UKF.

First, the formulation of the marginalized unscented transform (MUT), which was first presented in [Morelande and Moran, 2007], is detailed considering the case of conditionally linear transformations, i.e., the input RV, assumed Gaussian, is partitioned as $X = \begin{Bmatrix} a \\ b \end{Bmatrix}$ so that the transformation g is linear in subvector b:

$$g(a,b) = \varphi(a) + G(a)b \tag{4.4}$$

In this context, the Jacobian of the transformation with respect to subvector b is simply $\frac{\partial g}{\partial b} = G(a)$ and is independent of b itself. Computation of output expectations $E[Z] = E[g(X)]$ are decomposed as follows:

$$E_{p(a,b)}[g(a,b)] = \int_a \underbrace{\int_{b|a} g(a,b)p(b|a)db}_{\text{computed analytically}} p(a)da \tag{4.5}$$

Using the closure under conditioning property of the Gaussian distribution, moments of the Gaussian RV $b|a$ are given as:

$$E\left[b|a\right] = \mu_b + C_{ba}(P_a)^{-1}(a-\mu_a) \tag{4.6a}$$

$$Cov\left(b|a\right) = P_b - C_{ba}(P_a)^{-1}C_{ab} \tag{4.6b}$$

where

$$X = \begin{Bmatrix} a \\ b \end{Bmatrix} \sim \mathcal{N}\left(\cdot; \begin{bmatrix} \mu_a \\ \mu_b \end{bmatrix}, \begin{bmatrix} P_a & C_{ab} \\ C_{ba} & P_b \end{bmatrix}\right)$$

Moments of the conditional output $Z|a$ can also be calculated analytically using the closure under affine transformations property of the Gaussian distribution.

$$E\left[Z|a\right] = \varphi(a) + G(a)E\left[b|a\right] = g(a, E\left[b|a\right]) \tag{4.7a}$$

$$Cov\left(Z|a\right) = G(a)Cov\left(b|a\right)G(a)^T \tag{4.7b}$$

$$Cov\left(X,Z|a\right) = \begin{bmatrix} 0 \\ Cov\left(b|a\right)G(a)^T \end{bmatrix} \tag{4.7c}$$

Integration over the probability $p(a)$ is performed using a sigma point approximation. Any sigma point set can be used for this purpose; in the remainder of the chapter the symmetric set is used due to its known robustness in tackling joint state/parameter estimation in dynamical systems, as demonstrated in the previous chapter of this dissertation. The number of sigma points scales with the dimension of subvector a, and are sampled to match its marginal mean and covariance μ_a and P_a. Integration over $p(b|a)$ and $p(a)$ must be integrated together in order to compute the moments of the output RV Z, which is done as follows:

- Compute a set of $N_{sig} = 2n_a + 1$ sigma points that match moments of subvector a: $\mathcal{S} = \{\mathcal{A}^{(i)}, W^{(i)}\}$
- For each sigma point $\cdot^{(i)}$, compute the moments of the conditional RV $b|\mathcal{A}^{(i)}$, i.e., $\mathcal{B}^{(i)} = E\left[b|\mathcal{A}^{(i)}\right]$ and $\Gamma = Cov\left(b|\mathcal{A}^{(i)}\right)$ using Eqs. (4.6).

- Propagate all sigma points through the transformation g

$$\mathcal{Z}^{(i)} = g\left(\mathcal{A}^{(i)}, \mathcal{B}^{(i)}\right)$$

- Compute mean of output Z using the law of total expectation

$$\begin{aligned} E[Z] &= E_a[E[Z|a]] \\ &= \sum_{i=1}^{N_{sig}} W_m^{(i)} \mathcal{Z}^{(i)} \end{aligned} \tag{4.8a}$$

- Compute covariance of output Z using the law of total variance[1]

$$\begin{aligned} Cov(Z) &= Var_a(E[Z|a]) + E_a(var(Z|a)) \\ &= \sum_{i=1}^{N_{sig}} W_c^{(i)} (\mathcal{Z}^{(i)} - E[Z])(\mathcal{Z}^{(i)} - E[Z])^T + \sum_{i=1}^{N_{sig}} W_m^{(i)} G(\mathcal{A}^{(i)}) \Gamma G(\mathcal{A}^{(i)})^T \end{aligned} \tag{4.8b}$$

- One can also compute the cross-covariance terms needed in the measurement update step using the law of total covariance[1]

$$Cov(X,Z) = \sum_{i=1}^{N_{sig}} W_c^{(i)} \left(\begin{bmatrix} \mathcal{A}^{(i)} \\ \mathcal{B}^{(i)} \end{bmatrix} - \begin{bmatrix} \mu_a \\ \mu_b \end{bmatrix} \right) (\mathcal{Z}^{(i)} - E[Z])^T + \sum_{i=1}^{N_{sig}} W_m^{(i)} \begin{bmatrix} 0 \\ \Gamma G(\mathcal{A}^{(i)})^T \end{bmatrix} \tag{4.8c}$$

The key differences between the UT and the MUT and summarized in Fig. 4.2.

It is good to point out at this stage that this marginalization procedure can be successfully applied in different scenarios and that its scope is not limited to the Gaussian setting. Actually, this marginalization procedure is used several times throughout this thesis. For instance, it was used in chapter 3 to derive and use higher order sigma point sets that are challenging to use on large dimensional systems. More specifically, it was used successfully when applying the 4th order Gaussian sigma point set to identify the parameters of the Duffing oscillator (Fig. 3.6, $n_X = 5$).

[1] For certain sigma point sets, the weights used for computation of expectation vs. covariance terms are different (noted $W_m^{(i)}$ vs. $W_c^{(i)}$ therein). When performing integration over $p(a)$, careful attention must be given to the usage of the correct type of weights, as shown in Eqs. 4.8.

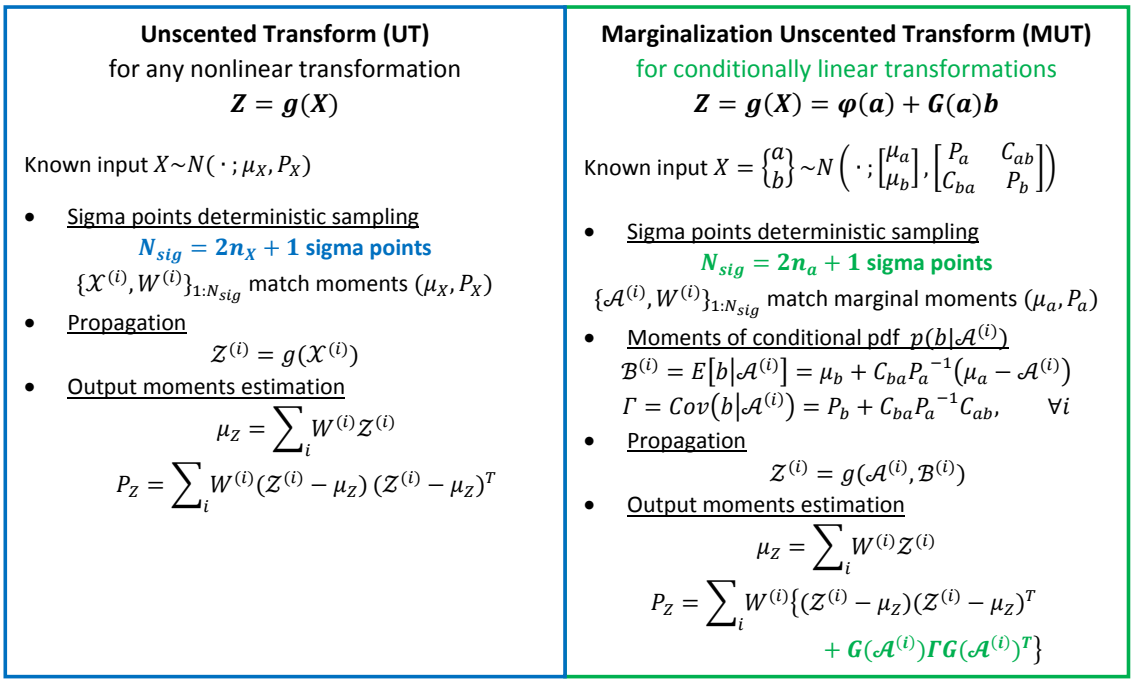


Figure 4.2: Principles of the unscented transform vs. marginalization unscented transform.

Computation of the sigma points and weights for this set requires solving a nonlinear optimization scheme, problem that is easier to solve in smaller dimension (2 or 3). For this Duffing oscillator, the 3 static parameters appear linearly in the equations, thus marginalization can be applied and n_a is reduced to 2, thus the 4th order set is easily applicable. This marginalization method can also be extended to other distributions that are closed under conditioning and affine transformations, such as the CSN distribution. The approach is similar, however Eqs. (4.6), (4.7) are modified according to the CSN distribution ($b|a$ and $Z|a$ follow CSN distributions). Also, 3rd and higher order moments can be computed, using the law of total cumulance when applying the sigma point approximation. This method was again successfully applied in section 3.4.2 when deriving a new sigma point set to take into account marginal and cross skew terms in the UT. Finally, this approach is also used in chapter 6, section 6.3.3 in order to reduce the number of sigma points and thus the computational complexity of the UKF when integrating a shaping filter to the filtering framework and thus tackle unmeasured excitations.

4.3.2 Integration in the UKF framework for joint state/parameter estimation of linear dynamical systems

From a slightly different point of view, the concept of marginalization has been employed to increase efficiency of *particle filtering* schemes for quite some time, as presented in chapter 2 of this dissertation. However, it has been much less used in the literature with respect to the nonlinear Kalman filtering framework. This concept of marginalized unscented transform for conditionally linear equations and its implications for the UKF was first presented in [Morelande and Moran, 2007], then recently used in e.g. [Chang, 2014a,b, Liu and Chang, 2015] to reduce computational time. As previously mentioned, it was also used for various purposes in this present work. With respect to computational time reduction, it is important to notice at this stage that making use of marginalization does not necessarily lead to reductions in the computational requirements of the UKF. Indeed, as can be seen in the details provided in the last section, using the MUT instead of the generic UT leads to a reduction in the number of sigma points, but it increases the number of computations per sigma point through the addition of another sum term in the covariance computation. Thus an overall reduction in computational time is contingent upon 1) a *significant* reduction in the number of sigma points, i.e., $n_a \ll n_X$ and 2) being able to easily write the function g under its linear conditional form, i.e., being able to compute the Jacobian $J = G$ efficiently. In the next section it is demonstrated that using marginalization for state/parameter estimation in linear FE models leads to significant reductions in computational time *under the condition that* the number of parameters to be estimated does not scale with the number of dynamic states (number of integration points in the FE model).

Integration of the MUT in the UKF framework is straightforward and simply consists in replacing the generic UT by the MUT for either/both of the propagation and measurement update steps. For state/parameter estimation of linear dynamical systems, one can show (see next two sections) that, for certain discretization schemes, a possible partitioning of the augmented state vector is

$a \leftarrow \theta$ and $b \leftarrow x^{dyn}$, and the MUT can then replace the UT in both the propagation and measurements update steps. Detailed algorithms for these two steps are given in Algs. 4 and 5 for this specific partitioning. In these algorithms, propagation of the dynamic state to the next time step simply consists in running the forward FE analysis, for a given set of parameters and initial conditions, i.e., a given sigma point $\{\Theta^{(i)}, \mathcal{X}_{k-1}^{(i)}\}$. Calculation of the Jacobians J_f and J_h depend on 1) the chosen discretization scheme and 2) the type of problem considered, parabolic vs. hyperbolic. In the following sections, these Jacobian calculations are detailed for both types of problems, using discretization schemes widely used in FE analysis due to their capacity at unconditional stability.

4.3.3 Application to linear parabolic systems

A linear parabolic system can be represented by the following dynamic equation:

$$M\mathrm{v} + K\mathrm{u} = \mathrm{P} \tag{4.14}$$

Heat conduction can be described as a parabolic system, where u represents the vector of temperatures at the FE model nodes. Following [Hughes, 2000], a parabolic problem can be discretized in time using a discretization scheme of the generalized trapezoidal family, with a choice of $\alpha \in [0,1]$:

$$M\mathrm{v}_{k+1} + K\mathrm{u}_{k+1} = \mathrm{P}_{k+1} \tag{4.15a}$$

$$\mathrm{u}_{k+1} = \mathrm{u}_k + dt \cdot \mathrm{v}_{k+\alpha} \tag{4.15b}$$

$$\mathrm{v}_{k+\alpha} = (1-\alpha)\mathrm{v}_k + \alpha \mathrm{v}_{k+1} \tag{4.15c}$$

For $\alpha = \frac{1}{2}$, the method is known as the trapezoidal rule and is unconditionally stable, i.e., the method is stable for any value of the time step dt. This method can also be implemented in the following form [Hughes, 2000]:

$$(M + \alpha dtK)\mathrm{u}_{k+1} = (M - (1-\alpha)dtK)\mathrm{u}_k + dt(\alpha \mathrm{P}_{k+1} + (1-\alpha)\mathrm{P}_k) \tag{4.16}$$

Algorithm 4: PROPAGATION STEP OF MARGINALIZED UKF

Data: Gaussian posterior at time step $k-1$ of joint pdf

$$p\left(x_{k-1}^{dyn}, \theta | y_{1:k-1}\right) \sim \mathcal{N}\left(\cdot; \begin{Bmatrix} x_{k-1|k-1}^{dyn} \\ \theta_{k-1|k-1} \end{Bmatrix}, \begin{bmatrix} P_{k-1|k-1}^{x} & C_{k-1|k-1}^{x\theta} \\ C_{k-1|k-1}^{\theta x} & P_{k-1|k-1}^{\theta} \end{bmatrix}\right)$$

Result: Gaussian prior at time step k,

$$p\left(x_{k}^{dyn}, \theta | y_{1:k-1}\right) \sim \mathcal{N}\left(\cdot; \begin{Bmatrix} x_{k|k-1}^{dyn} \\ \theta_{k|k-1} \end{Bmatrix}, \begin{bmatrix} P_{k|k-1}^{x} & C_{k|k-1}^{x\theta} \\ C_{k|k-1}^{\theta x} & P_{k|k-1}^{\theta} \end{bmatrix}\right)$$

sample $N_{sig} = 2n_\theta + 1$ sigma points from the subvector $\theta | y_{1:k-1} \sim \mathcal{N}\left(\cdot; \theta_{k-1|k-1}, P_{k-1|k-1}^{\theta}\right)$;

for each sampled sigma point $\Theta^{(i)}$, compute the corresponding dynamic state vector as the mean of the conditional pdf $p(x_{k-1}^{dyn} | \Theta^{(i)}, y_{1:k-1})$, i.e.,

$$\mathcal{X}_{k-1}^{dyn,(i)} = x_{k-1|k-1}^{dyn} + C_{k-1|k-1}^{x\theta}(P_{k-1|k-1}^{\theta})^{-1}(\Theta^{(i)} - \theta_{k-1|k-1}) \tag{4.9a}$$

for all sigma points $\cdot^{(i)}$, the state conditional covariance is identical and equal to

$$\Gamma_{x|\theta} = P_{k-1|k-1}^{x} - C_{k-1|k-1}^{x\theta}(P_{k-1|k-1}^{\theta})^{-1} C_{k-1|k-1}^{\theta x} \tag{4.9b}$$

for $i = 1 : N_{sig}$ **do**

propagate dynamic state to next time step: $\mathcal{X}_{k}^{dyn,(i)} = f(\mathcal{X}_{k-1}^{dyn,(i)}, \Theta^{(i)})$;

compute the Jacobian of the transformation f: $J_f(\Theta^{(i)})$

end

compute moments of prior $p(x_k^{dyn}, \theta | y_{1:k-1})$ as averages over the sigma points

$$\theta_{k|k-1} = \sum_{i=1}^{N_{sig}} W_m^{(i)} \Theta^{(i)} \tag{4.10a}$$

$$x_{k|k-1}^{dyn} = \sum_{i=1}^{N_{sig}} W_m^{(i)} \mathcal{X}_k^{dyn,(i)} \tag{4.10b}$$

$$P_{k|k-1}^{\theta} = \sum_{i=1}^{N_{sig}} W_c^{(i)} \left(\Theta^{(i)} - \theta_{k|k-1}\right)\left(\Theta^{(i)} - \theta_{k|k-1}\right)^T \tag{4.10c}$$

$$C_{k|k-1}^{\theta x} = \sum_{i=1}^{N_{sig}} W_c^{(i)} \left(\Theta^{(i)} - \theta_{k|k-1}\right)\left(\mathcal{X}_k^{dyn,(i)} - x_{k|k-1}^{dyn}\right)^T \tag{4.10d}$$

$$P_{k|k-1}^{x} = \sum_{i=1}^{N_{sig}} \left\{ W_c^{(i)} \left(\mathcal{X}_k^{dyn,(i)} - x_{k|k-1}^{dyn}\right)\left(\mathcal{X}_k^{dyn,(i)} - x_{k|k-1}^{dyn}\right)^T + \underbrace{W_m^{(i)} J_f \Gamma_{x|\theta} J_f^T}_{\text{additional term (marginalization)}} \right\} \tag{4.10e}$$

Algorithm 5: MEASUREMENT UPDATE STEP OF MARGINALIZED UKF

Data: Gaussian prior at time step $k-1$ of joint pdf $p\left(x_k^{dyn}, \theta | y_{1:k-1}\right)$

Result: Gaussian posterior at time step k, $p\left(x_k^{dyn}, \theta | y_{1:k}\right) \sim \mathcal{N}\left(\cdot; \begin{Bmatrix} x_{k|k}^{dyn} \\ \theta_{k|k} \end{Bmatrix}, \begin{bmatrix} P_{k|k}^{x} & C_{k|k}^{x\theta} \\ C_{k|k}^{\theta x} & P_{k|k}^{\theta} \end{bmatrix}\right)$

sample $N_{sig} = 2n_\theta + 1$ sigma points from the subvector $\theta | y_{1:k-1} \sim \mathcal{N}\left(\cdot; \theta_{k-1|k-1}, P_{k-1|k-1}^{\theta}\right)$;

for each sampled sigma point $\Theta^{(i)}$, compute the corresponding dynamic state vector as the mean of the conditional pdf $p(x_k^{dyn} | \Theta^{(i)}, y_{1:k-1})$, i.e.,

$$\mathcal{X}_k^{dyn,(i)} = x_{k|k-1}^{dyn} + C_{k|k-1}^{x\theta} (P_{k|k-1}^{\theta})^{-1} (\Theta^{(i)} - \theta_{k|k-1}) \tag{4.11a}$$

for all sigma points $\cdot^{(i)}$, the state conditional covariance is identical and equal to

$$\Gamma_{x|\theta} = P_{k|k-1}^{x} - C_{k|k-1}^{x\theta} (P_{k|k-1}^{\theta})^{-1} C_{k|k-1}^{\theta x} \tag{4.11b}$$

for $i = 1 : N_{sig}$ **do**

  compute predicted measurement: $\mathcal{Y}_k^{(i)} = h(\mathcal{X}_k^{dyn,(i)}, \Theta^{(i)})$;

  compute the Jacobian of the transformation h: $J_h(\Theta^{(i)})$

end

compute moments of joint pdf $p(x_k^{dyn}, \theta, y_k | y_{1:k-1})$ as averages over the sigma points

$$y_{k|k-1} = \sum_{i=1}^{N_{sig}} W_m^{(i)} \mathcal{Y}_k^{dyn,(i)} \tag{4.12a}$$

$$S_k = \sum_{i=1}^{N_{sig}} \left\{ W_c^{(i)} \left(\mathcal{Y}_k^{dyn,(i)} - y_{k|k-1}\right) \left(\mathcal{Y}_k^{dyn,(i)} - y_{k|k-1}\right)^T + \underbrace{W_m^{(i)} J_h \Gamma_{x|\theta} J_h^T}_{\text{additional term (marginalization)}} \right\} \tag{4.12b}$$

$$C_{k|k-1}^{\theta y} = \sum_{i=1}^{N_{sig}} W_c^{(i)} \left(\Theta^{(i)} - \theta_{k|k-1}\right) \left(\mathcal{Y}_k^{dyn,(i)} - y_{k|k-1}\right)^T \tag{4.12c}$$

$$C_{k|k-1}^{xy} = \sum_{i=1}^{N_{sig}} \left\{ W_c^{(i)} \left(\mathcal{X}_k^{dyn,(i)} - x_{k|k-1}^{dyn}\right) \left(\mathcal{Y}_k^{dyn,(i)} - y_{k|k-1}\right)^T + \underbrace{W_m^{(i)} \Gamma_{x|\theta} J_h^T}_{\text{additional term (marginalization)}} \right\} \tag{4.12d}$$

$$\Psi_k = \begin{bmatrix} C_{k|k-1}^{xy} \\ C_{k|k-1}^{\theta y} \end{bmatrix} \tag{4.12e}$$

Perform measurement update to compute posterior moments at time step k, see Eqs. (3.2b) and (3.2c).

$$\{x_{k|k}, P_{k|k}\} = \text{KalmanMeasurementUpdate}(x_{k|k-1}, P_{k|k-1}, y_{k|k-1}, S_k, \Psi_k) \tag{4.13}$$

which will be further used to derive the Jacobian matrix for use in the marginalization UKF (MUKF). It is further assumed that the measurement equation is linear in the state vector, i.e.,

$$y_k = H\mathrm{u}_k \tag{4.17}$$

where the matrix H can be nonlinearly dependent on the parameters θ.

In the context of the UKF, the dynamic states x^{dyn} would be composed solely of the vector u, to which the vector of unknown parameters θ is added to obtain the augmented state vector x. As previously mentioned, integration over the pdf $p(a) = p(\theta)$ is performed using the UT. Thus let us assume that θ is known, set equal to $\Theta^{(i)}$ (sampled sigma point). For this specific value of θ, the system matrices M, K and P can be computed; also, an initial conditions vector $\mathcal{X}_{k-1}^{dyn,(i)}$ is sampled, as the mean of the conditional pdf $p(x_{k-1}^{dyn}|\Theta^{(i)}, y_{1:k-1})$ (first two steps of algorithm 4). Then the system described by this sampled sigma point can be propagated forward in time from t_{k-1} to t_k, by simply running the dynamic FE analysis. In order to further calculate the moments of the joint prior pdf $p(x_k, \theta|y_{1:k})$ (Eqs. (4.10) in Alg. 4), one also needs to compute the Jacobian of the propagation equation with respect to the dynamic states, i.e., $J_f = \frac{\partial \mathrm{u}_{k+1}}{\partial \mathrm{u}_k}$, which can be easily computed using Eq. (4.16):

$$J_f = (M + \alpha dtK)^{-1}(M - (1-\alpha)dtK) \tag{4.18}$$

Since the system matrices M, K have already been computed to propagate the system forward in time, computing the Jacobian does not require excessive additional computational time.

Regarding the measurement step, fully detailed in Alg. 5, one needs to re-sample the $N_{sig} = 2n_\theta + 1$ sigma points, matching moments of the prior pdf $p(\theta|y_{1:k-1})$, and compute the corresponding state sigma points $\mathcal{X}_k^{dyn,(i)}$ as the mean of the conditional pdfs $p(x_k^{dyn}|\Theta^{(i)}, y_{1:k-1})$. It is crucial here to notice that these sigma points are different from the ones sampled at the propagation step, since these new sigma points make use of the prior moments $p(x_k, \theta|y_{1:k-1})$ obtained at

the end of the propagation step. Again, knowledge of the Jacobian of the measurement equation is required to compute moments of the joint pdf $p(x_k^{dyn}, \theta, y_k|y_{1:k-1})$ (Eqs. (4.12)), needed to perform the measurement update. This Jacobian is readily available as $J_h = \frac{\partial y_k}{\partial u_k} = H$, which again can be function of the parameter θ (evaluated at sigma point $\Theta^{(i)}$).

In a numerical example the performance of this marginalized UKF is compared to the generic UKF, both in terms of running time and accuracy in their estimation of the moments of the posterior pdf $p(\theta|y_{1:N})$. The problem studied is a 2-dimensional heat conduction problem, adapted from a static problem defined in [Fish and Belytschko, 2007]. The system is excited by a homogeneous temperature source, which varies over time as a sine function; temperature measurements are assumed to be available at two locations on the plate (Fig. 4.3a). Data for this problem is generated using a fine mesh (294 DOFs); 10% root-mean-square (RMS) noise is added to both the measurement vector and the excitation. When using the UKF or MUKF for identification, the statistics of these noise terms are assumed known, and the augmented state approach is used to deal with the process noise. [2]

An inverse problem is run employing both the generic UKF and the marginalized UKF to learn the unknown material conductivity κ, which appears in the stiffness matrix K, always using the same measurements time series as an input. Both algorithms are run several times, using different meshes to represent the system. Using a very coarse mesh leads to an inaccurate parameter estimate, due to the fact that the dynamics, and thus the measurements, cannot be well reconstructed using this coarse mesh. Refining the mesh leads to greater accuracy in the parameter estimation procedure, however it also greatly increases the number of DOFs, and thus the dimension of the dynamic state vector x^{dyn}. This leads to a tremendous increase in running time for the generic

[2] In this augmented state approach, a couple of sigma points are added to the $2n_x + 1$ (UKF) or $2n_\theta + 1$ (MUKF) sigma points, sampled to match the known moments of the process noise, assumed uncorrelated with the state vector. This approach is used in all the numerical experiments presented therein, but these additional sigma points are omitted from the discussion, since they are added for both the UKF and MUKF and thus do not participate in the performance variations observed between these two algorithms.

UKF, much more so than for the marginalized UKF, for which N_{sig} does not scale with the size of the state vector but solely with the number of parameters to be learnt, which in this case remains the same. This example shows clearly a significant drawback of the generic UKF for use in conjunction with finite element models, for which mesh refinement is a crucial element. It is also important to notice that both algorithms perform very similarly in terms of accuracy of both the mean and covariance estimates, for a given mesh (Fig. 4.3c), i.e., the marginalization procedure presented therein does not affect accuracy, while it enables large gains in computational time. This was expected for this case since the system behaves linearly, and it was shown that in this case the propagation equation is linear in the dynamic states, thus marginalization does not introduce any additional approximation compared to a generic UKF.

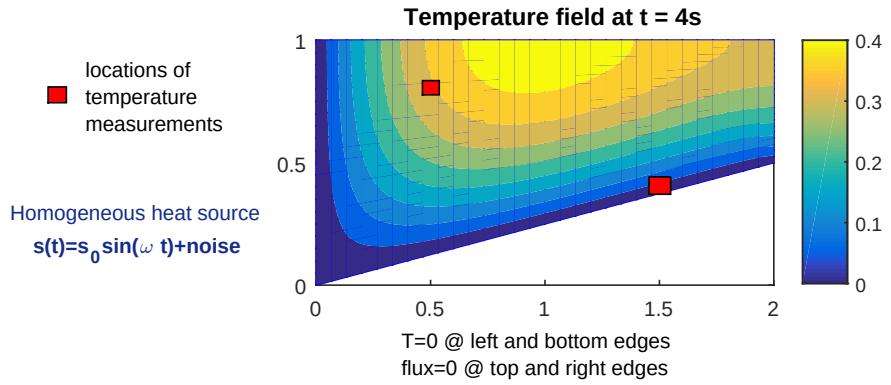
4.3.4 Application to linear hyperbolic systems

Formulation

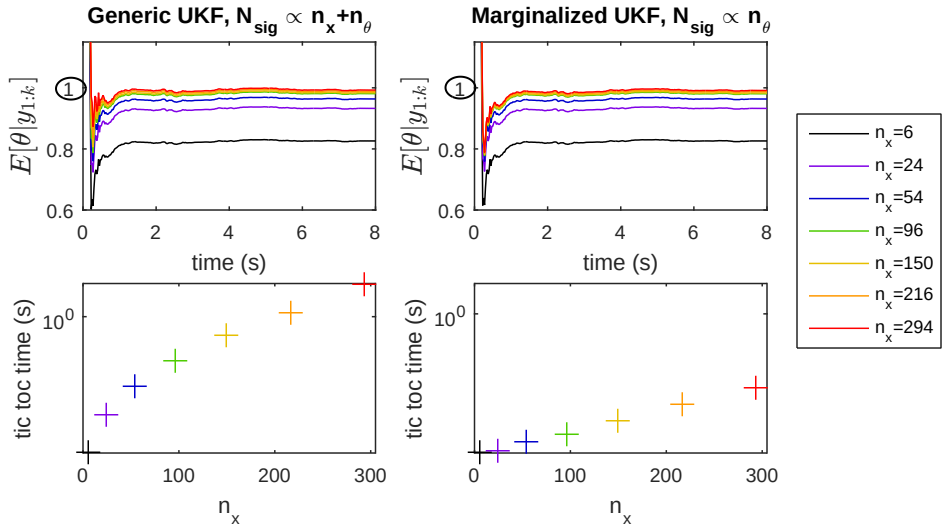
In this section, the method previously described is applied to the more common case of a linear hyperbolic system, whose dynamics can be represented by the following equation:

$$M\mathtt{a} + C\mathtt{v} + K\mathtt{u} = \mathrm{P} \tag{4.19}$$

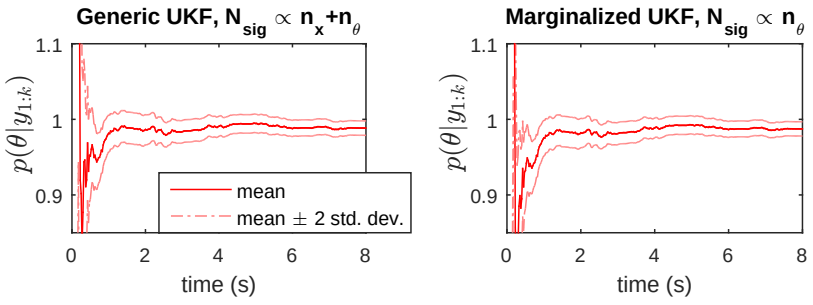
with initial conditions $\mathtt{u}_0, \mathtt{v}_0$. The system matrices M, C, K and external force vector P can be nonlinearly dependent on the parameters θ, but are not a function of the dynamic states. Following [Hughes, 2000], the initial value problem can be solved using one of the widely used Newmark



(a) Geometry of the 2D domain, measurement locations



(b) Parameter convergence (top) and running time per time step (bottom) for both algorithms when using various meshes for the inverse problem



(c) Convergence of both mean and std. dev. estimates for both algorithms, for a fine mesh

Figure 4.3: Performance of UKF vs. MUKF on a parabolic (heat conduction) problem.

methods, whose predictor/corrector implementation is written as follows:

$$\tilde{\mathtt{u}}_{k+1} = \mathtt{u}_k + dt\mathtt{v}_k + \frac{dt^2}{2}(1-2\beta)\mathtt{a}_k \quad \text{displacement predictor} \tag{4.20a}$$

$$\tilde{\mathtt{v}}_{k+1} = \mathtt{v}_k + (1-\gamma)dt\mathtt{a}_k \quad \text{velocity predictor} \tag{4.20b}$$

$$\left(M + \gamma dtC + \beta dt^2 K\right)\mathtt{a}_{k+1} = \mathrm{P}_{k+1} - C\tilde{\mathtt{v}}_{k+1} - K\tilde{\mathtt{u}}_{k+1} \quad \text{solve for acceleration} \tag{4.20c}$$

$$\mathtt{u}_{k+1} = \tilde{\mathtt{u}}_{k+1} + \beta dt^2 \mathtt{a}_{k+1} \quad \text{displacement corrector} \tag{4.20d}$$

$$\mathtt{v}_{k+1} = \tilde{\mathtt{v}}_{k+1} + \gamma dt\mathtt{a}_{k+1} \quad \text{velocity corrector} \tag{4.20e}$$

Using $\beta = \frac{1}{4}$ and $\gamma = \frac{1}{2}$ yields an implicit method with unconditional stability, often referred to as the average acceleration method or trapezoidal rule. This method will be used in the remainder of the paper to avoid keeping cumbersome notations.

With respect to the UKF, the state vector is now composed of both displacement and velocity terms at the free DOFs, i.e., $x_k^{dyn} = \begin{Bmatrix} \mathtt{u}_k \\ \mathtt{v}_k \end{Bmatrix}$. Similar to the parabolic problem, utilization of the marginalization scheme requires writing the propagation equation as a conditionally linear function of the dynamic states, i.e., $x_{k+1}^{dyn} = \varphi(\mathrm{P}_{k,k+1}, \theta) + G(\theta)x_k^{dyn}$. After some algebra, it is possible to show that the Newmark method can indeed be written in such a way, with

$$G = \begin{bmatrix} N^{-1}\left(M + \frac{dt}{2}C - \frac{dt^2}{4}K\right) & dtN^{-1}M \\ -dtN^{-1}K & N^{-1}\left(M - \frac{dt}{2}C - \frac{dt^2}{4}K\right) \end{bmatrix} \tag{4.21a}$$

$$N = \left(M + \frac{dt}{2}C + \frac{dt^2}{4}K\right) \tag{4.21b}$$

G is the Jacobian of the propagation equation with respect to the dynamic states, i.e., $G = J_f$; marginalization can be applied to the UKF in the exact same fashion as for a parabolic problem, using this new value of J_f in Alg. 4.

Regarding the measurement step, a conditionally linear measurement equation of the form

$$y_k = \begin{bmatrix} H_1 & H_2 \end{bmatrix} \begin{Bmatrix} \mathtt{u}_k \\ \mathtt{v}_k \end{Bmatrix} \tag{4.22}$$

encompasses strain, displacement, velocity as well as acceleration measurements (the latter is ob-

tained by setting $H_1 = -M^{-1}K, \quad H_2 = -M^{-1}C$). The Jacobian $J_h = \begin{bmatrix} H_1 & H_2 \end{bmatrix}$ can thus be used for marginalization in Alg. 5.

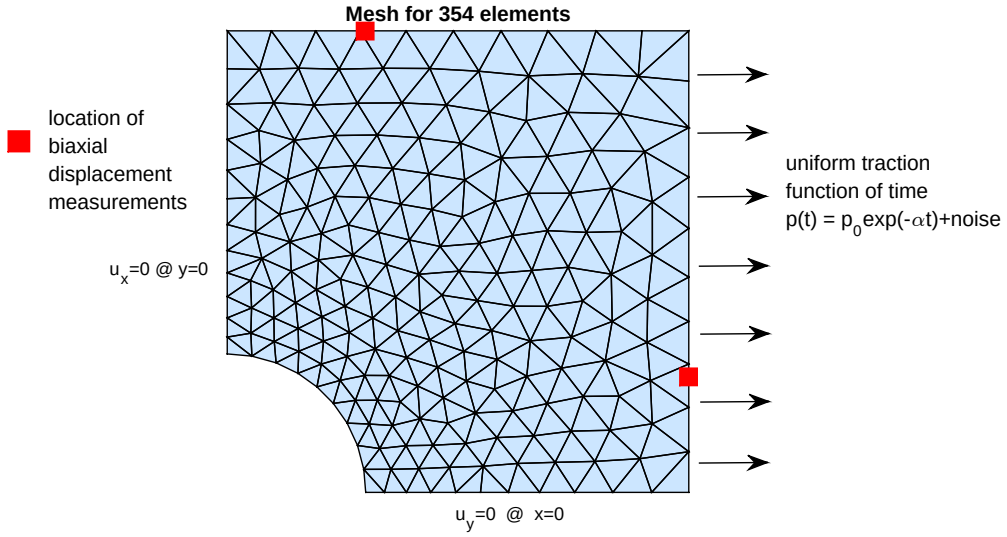
Numerical application: 2D linear elasticity

Performance of the marginalization procedure is first demonstrated on a 2D linear elasticity problem, as shown in Fig. 4.4a, where two material parameters must be identified (Young's modulus and Poisson ratio of the plate). The unstructured triangular mesh is generated using the DistMesh program ([Persson and Strang, 2004]). 10% RMS noise is added to both biaxial displacement measurements before using them for identification. As for the parabolic problem, the marginalization procedure does not negatively impact accuracy of the identification procedure (Fig. 4.4b), which was expected from the theory previously detailed. It however enables a large reduction in the number of sigma points compared to a generic UKF, yielding a significant reduction in computational time (Table 4.1).

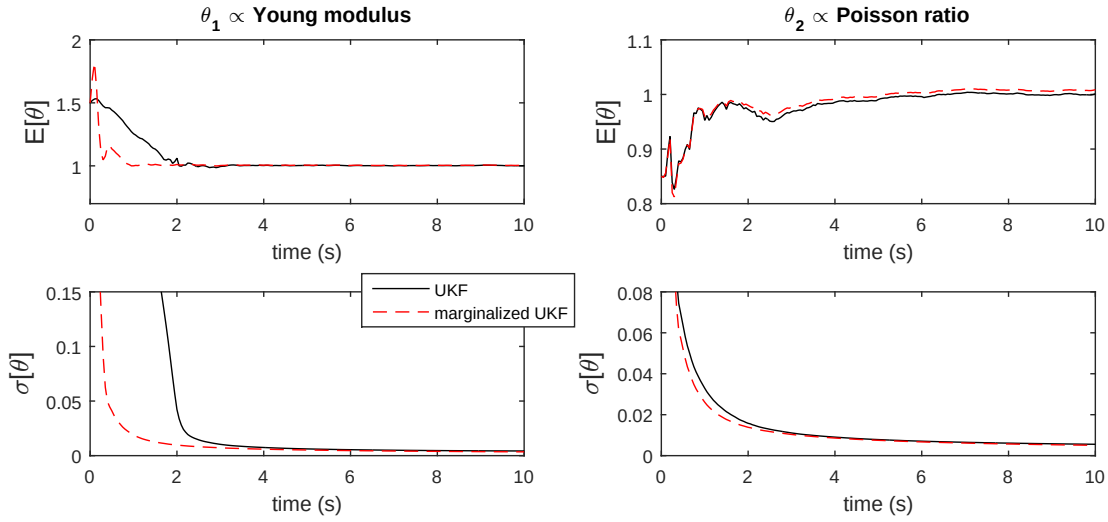
Numerical application: 2D truss subjected to seismic ground motion

This second example deals with a 2D linear truss, subjected to a vertical ground motion. Rayleigh damping is used as $C = \delta_1 M + \delta_2 K$, whose parameters $\delta_{1,2}$ are being learnt along with the Young's modulus of each of the 21 elements, appearing in the stiffness matrix K. This creates a nonlinearity *with respect to the parameters* in the propagation equation. A set of strains, displacements and accelerations are measured; at least one quantity is measured at each DOF to avoid unidentifiability issues.

Again, the UKF and MUKF perform similarly in terms of accuracy in their estimation of the mean of the posterior pdf $p(\theta|y_{1:N})$ (Figs. 4.6a and 4.6b). However, in this particular example the gain in computational time due to the marginalization procedure is not as substantial as in the previous cases, even though a gain of about 30% is still obtained (Fig. 4.2). The reason for this



(a) Mesh and excitation



(b) Convergence of moments of posterior pdf $p(\theta|y_{1:k})$ using UKF vs. MUKF

Figure 4.5: 2D linear elasticity example: performance of UKF vs. MUKF

	generic UKF	marginalized UKF
number of sigma points	$N_{sig} = 1383$	$N_{sig} = 7$
running time per time step	31 sec	0.8 sec

Table 4.1: Running time comparison for generic UKF vs. MUKF for 2D linear elasticity example

is that, in this specific problem, the number of parameters does increase with the dimension of the problem (number of elements), as does the number of sigma points in the MUT. Since marginalization also entails additional computations per sigma point compared to a generic UKF, the small reduction in sigma points is hardly capable of offsetting these additional computations. This again shows that the concept of marginalization will not necessarily yield large gains in computational time, unless the number of parameters is small compared to the number of DOFs, which is actually quite common in FE models.

4.3.5 A discussion on quasi-static vs. fully dynamic problems

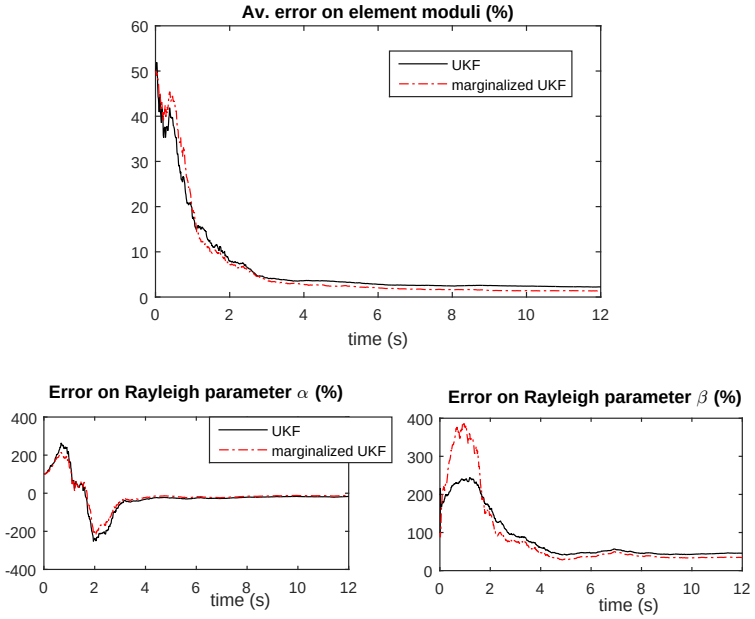
In this section we introduce a discussion on parameter estimation for static or quasi-static systems, i.e., systems for which the excitation is not time dependent, or varies slowly enough in time so that inertial effects can be neglected. These systems are then assumed to be in internal equilibrium at each time step k

$$K\mathrm{u}_k = \mathrm{P}_k \tag{4.23}$$

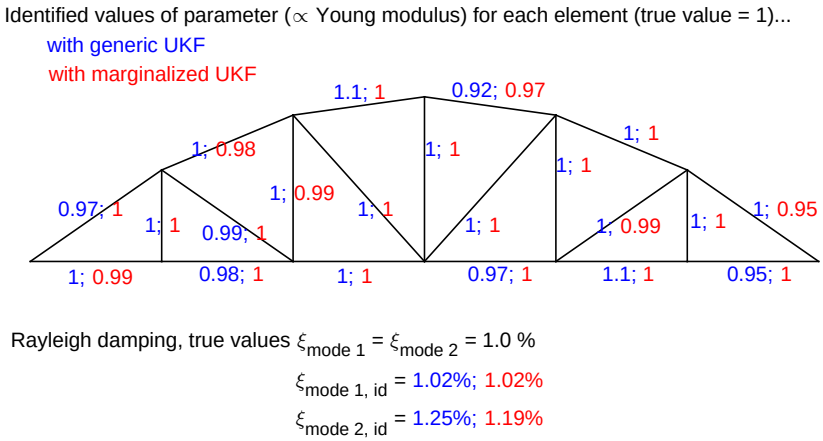
Intuitively, one can understand that for this type of systems, uncertainty around the unknown states does not govern the behavior of the UKF since this uncertainty does not propagate forward in time. Thus sigma points should only be needed to capture uncertainty around the unknown parameters, i.e., $N_{sig} = 2n_\theta + 1$. This can also be proved rigorously using the marginalization concept, since the propagation equation can be written as:

$$\mathrm{u}_k = f(\mathrm{u}_{k-1}) = K^{-1}\mathrm{P}_k + 0 \cdot \mathrm{u}_{k-1} \tag{4.24}$$

This equation is obviously linear in the dynamic vector u_{k-1}, with $G = 0$; the marginalization principle can be utilized and the number of sigma points need only to scale with the dimension of the parameter vector. This approach generalizes to nonlinear quasi-static systems, since again the propagation equation would be independent of the past dynamic state.



(a) Evolution in time of the error on identified parameters



(b) Final identified values of all parameters

Figure 4.7: Performance of UKF vs. MUKF for a 2D linear truss

	generic UKF	marginalized UKF
number of sigma points	$N_{sig} = 133$	$N_{sig} = 49$
running time per time step	0.30 s	0.20 s

Table 4.2: 2D linear truss example: running times of generic UKF vs. MUKF

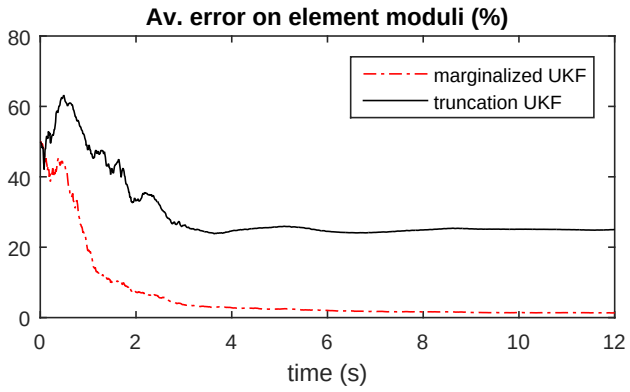


Figure 4.8: Error on estimating the Young's moduli parameters of the linear truss, MUKF vs. truncation UKF (fully neglecting uncertainty around the states).

As mentioned in the introduction of this chapter, this property of quasi-static systems is also discussed in [Xi et al., 2011], where a reduced-rank filter based on a SVD decomposition of the covariance matrix is utilized for identification. It is explained there that due to the quasi-staticity of the problem, the uncertainties primarily lie in the parameter space and only n_θ singular values need to be kept in the SVD decomposition of the covariance matrix. Here it is shown instead that no truncation, and no SVD decomposition, are actually necessary in this case since one can reduce the number of sigma points through the concept of marginalization.

A deeper study of the theory and possible applications of this method for quasi-static systems would deserve a more thorough investigation and is not presented in this current paper due to space limitations. However, it is important to notice at this stage that, if for quasi-static systems it is possible to somewhat reduce the uncertainty space to the parameter vector space, it is impossible to do the same for fully dynamic systems. Even though the main task of the joint filtering procedure may be to identify unknown parameters, and prior uncertainty around the parameters might be much larger than prior uncertainty around the states, the uncertainty around the dynamic states should not be neglected during the filtering procedure.

To show this, let us go back to our marginalization problem:

$$p\left(x_k|y_{1:k-1}\right) = \int_{x_{k-1}} p\left(x_k|x_{k-1}\right) p\left(x_{k-1}|y_{1:k-1}\right) dx_{k-1} \tag{4.25a}$$

$$= \int_{\theta} \underbrace{\int_{x_{k-1}^{dyn}} p\left(x_k|x_{k-1}^{dyn},\theta\right) p\left(x_{k-1}^{dyn}|\theta,y_{1:k-1}\right) dx_{k-1}^{dyn}}_{I_x\text{: solved using a point approximation}} p\left(\theta_{k-1}|y_{1:k-1}\right) d\theta_{k-1} \tag{4.25b}$$

Neglecting the uncertainty around the dynamic states could be simply performed by approximating integral I_x as a single point approximation, by evaluating it at the mean of the conditional probability $p\left(x_{k-1}^{dyn}|\theta,y_{1:k-1}\right)$. Then a sigma point approximation is only needed to evaluate an integral over $p\left(\theta|y_{1:k-1}\right)$. This leads to a similar algorithm as the MUKF, with the difference that the additional covariance terms (related to the Jacobians) are set to 0. Running this truncation algorithm on the previous truss system yields large discrepancies in the identified parameters (Fig. 4.8). This is due to the fact that in a fully dynamic system, uncertainties around the states propagate forward in time, and can in turn affect identification of the static parameters. Those results were further confirmed by running a reduced-rank filter on different types of fully dynamic, rapidly varying linear and nonlinear systems. For these systems, keeping only n_θ singular values in the decomposition yielded inaccurate results, proof that uncertainty around the states need to be taken into account for such systems. In [Moireau and Chapelle, 2011], this problem is alleviated using a first-stage observer to reduce uncertainty around the states. In the marginalization framework presented therein, this uncertainty is instead fully taken into account through the additional terms in the covariance calculations.

4.4 An extended-unscented Kalman filter for parameter estimation in nonlinear finite element models

4.4.1 Development of an extended-marginalized UKF

The extended-marginalized unscented transform

In the previous section it was shown that the marginalized unscented transform can be applied to joint state/parameter estimation in linear dynamical systems, potentially yielding large reductions in computational cost for cases where the number of parameters does not scale with the number of DOFs of the finite element model. Utilization of the MUT requires a certain format for the system equations, specifically both f, h should be conditionally linear in the dynamic states vector x_k, which prevents its direct utilization for parameter estimation in nonlinear dynamical systems.

However, a straightforward generalization of this method can be derived through partial linearization of the system equations. Indeed, going back to the marginalization procedure for a conditionally linear equation g, integrations over $p(a)$ and $p(b|a)$ are uncoupled: sigma points $\mathcal{A}^{(i)}$ are sampled to compute moments over $p(a)$, and, for each sigma point $\cdot^{(i)}$, moments over $p(b|\mathcal{A}^{(i)})$ are computed analytically, using the linear structure of g. Let us now consider the partial first order Taylor series expansion about subvector b of a more generic, non conditionally linear, function g:

$$g(a,b) \simeq g(a,\tilde{b}) + J_g(a,\tilde{b})(b-\tilde{b}) \tag{4.26}$$

where $J_g = \frac{\partial g}{\partial b}$ is the Jacobian of the transformation with respect to b. Using this linear representation of g, the marginalization procedure previously described can be taken advantage of, using $\tilde{b} = E\left[b|\mathcal{A}^{(i)}\right]$, the mean of the conditional pdf upon which integration is performed. A short comparison and summary of equations for the UT vs. extended marginalized unscented transform (eMUT) is presented in Fig. 4.9.

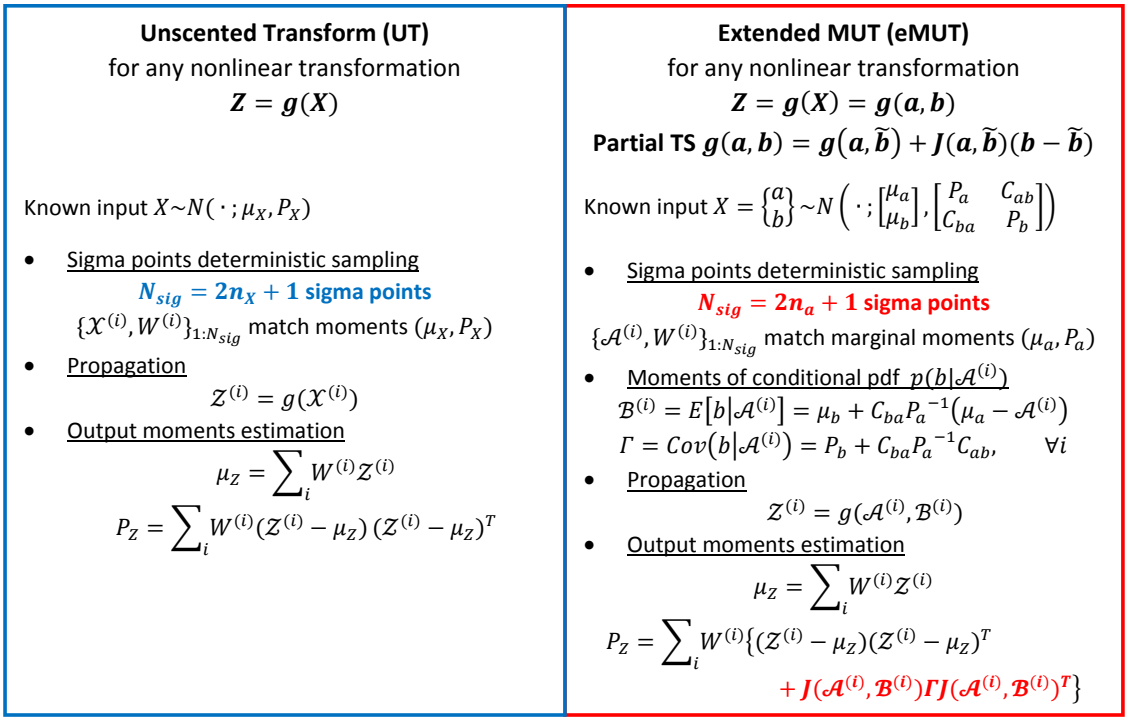


Figure 4.9: Principle of the unscented transform vs. extended marginalization unscented transform.

When integrated in a Gaussian filter, this approach can actually be interpreted as the combination of a UKF for treatment of uncertainties in the space of a and an EKF for filtering with respect to subvector b. As discussed in the previous chapter of this thesis, the EKF reaches a lower order of accuracy than the UKF with respect to moment propagation, however it may lead to a reduction in computational cost when used on systems *for which the Jacobian matrix can be computed efficiently*. Going back to finite element models, it is possible to show that, for certain discretization schemes, the Jacobian with respect to the dynamic states can be related to the system matrices computed during propagation. In particular, it is related to the tangent stiffness matrix of the problem, whose analytical form is usually coded in FEA software since nonlinear analysis often relies on Newton schemes, i.e., subsequent first order linearizations of the equilibrium or dynamics equations. Thus, by choosing a similar partitioning as for linear systems, i.e., $a \leftarrow \theta$ and

$b \leftarrow x^{dyn}$, the eMUT can be efficiently utilized to reduce computational costs, as will be demonstrated in subsequent examples. Before detailing computations of these Jacobians, it is useful to quickly describe the main components of a nonlinear dynamic analysis as usually performed in finite element analysis.

Solving for nonlinear problems in FEA software

To avoid repetitive developments, the detailed analysis is only laid out for the case of an hyperbolic problem. Performance of the method on a simpler parabolic problem will still be demonstrated in an example. In this section, the nonlinear systems considered exhibit the following dynamics:

$$M\mathtt{a} + C\mathtt{v} + \mathrm{F}(\mathtt{u}) = \mathrm{P}(\mathtt{u}) \tag{4.27}$$

where $\mathrm{F}(\mathtt{u})$ is the nonlinear restoring force (nonlinear stiffness term) and $\mathrm{P}(\mathtt{u})$ is the external excitation, which can also depend nonlinearly on the displacement vector. More general systems could encompass nonlinearities with respect to the velocity vector (nonlinear damping term $C\mathtt{v}$ for instance), which would be treated using a similar procedure as presented therein. However we restrict ourselves to the case defined above, knowing that it already encompasses a large variety of nonlinear systems.

We summarize in this section a well-known method to solve a nonlinear initial value problem, which combines a linear predictor through trapezoidal rule (or any chosen rule), and a nonlinear corrector using for instance the Newton-Raphson iteration method (see e.g. [Bathe and Cimento, 1980, Taylor, 2017]). Recall the Newmark formulas to solve for an hyperbolic initial value prob-

lem:

$$\mathtt{u}_{k+1} = \mathtt{u}_k + dt\mathtt{v}_k + \frac{dt^2}{4}(\mathtt{a}_k + \mathtt{a}_{k+1}) \tag{4.28a}$$

$$\mathtt{v}_{k+1} = \mathtt{v}_k + \frac{dt}{2}(\mathtt{a}_k + \mathtt{a}_{k+1}) \tag{4.28b}$$

$$M\mathtt{a}_{k+1} + C\mathtt{v}_{k+1} + \mathrm{F}(\mathtt{u}_{k+1}) = \mathrm{P}(\mathtt{u}_{k+1}) \tag{4.28c}$$

Since the momentum equation is nonlinear, the Newton-Raphson iterative procedure, which uses subsequent linearizations of the momentum equation, is used to solve for the previous system at time step $k+1$. It is recommended to start the iteration with $\mathtt{u}_{k+1}^{[0]} = \mathtt{u}_k$ (superscript refers to iteration number in the Newton-Raphson procedure), $\mathtt{v}_{k+1}^{[0]} = -\mathtt{v}_k$ and $\mathtt{a}_{k+1}^{[0]} = -\frac{4}{dt}\mathtt{v}_k - \mathtt{a}_k$. Then the update formulas, which satisfy the above Newmark formulas, are as follows:

$$\mathtt{u}_{k+1}^{[i+1]} = \mathtt{u}_{k+1}^{[i]} + \Delta\mathtt{u}_{k+1}^{[i+1]} \tag{4.29a}$$

$$\mathtt{v}_{k+1}^{[i+1]} = \mathtt{v}_{k+1}^{[i]} + \frac{2}{dt}\Delta\mathtt{u}_{k+1}^{[i+1]} \tag{4.29b}$$

$$\mathtt{a}_{k+1}^{[i+1]} = \mathtt{a}_{k+1}^{[i]} + \frac{4}{dt^2}\Delta\mathtt{u}_{k+1}^{[i+1]} \tag{4.29c}$$

The displacement increment $\Delta\mathtt{u}_{k+1}^{[i+1]}$ is obtained through linearization of the momentum equation:

$$K_t^{\star[i]}\Delta\mathtt{u}_{k+1}^{[i+1]} = \mathrm{P}(\mathtt{u}_{k+1}^{[i]}) - \mathrm{F}(\mathtt{u}_{k+1}^{[i]}) - C\mathtt{v}_{k+1}^{[i]} - M\mathtt{a}_{k+1}^{[i]} \tag{4.30}$$

where

$$K_t^{\star[i]} = K_t^{[i]} + \frac{2}{dt}C + \frac{4}{dt^2}M \tag{4.31}$$

In this last equation, the term $K_t^{[i]}$ represents the tangent stiffness matrix, as computed for a static problem, i.e., it is the Jacobian of the momentum equation with respect to the displacement vector $K_t^{[i]} = \frac{\partial(\mathrm{F}-\mathrm{P})}{\partial\mathtt{u}_{k+1}}$ evaluated at $\mathtt{u}_{k+1}^{[i]}$ (in the nonlinear cases the tangent stiffness is a function of the displacement).

Jacobian computations for the eMUT

Utilization of the MUKF requires evaluation of the Jacobian of both the propagation and measurement functions f, h. With respect to the propagation equation, derivation of an approximate Jacobian can be performed by differentiation of the Newmark formulas (Eqs. 4.28), which yields the following:

$$J_f = \frac{\partial f}{\partial x_k^{dyn}} = \begin{bmatrix} \left(K_t^{\star[i_f]}\right)^{-1}\left(\frac{4}{dt^2}M + \frac{2}{dt}C - K_t^{[0]}\right) & \left(K_t^{\star[i_f]}\right)^{-1}\left(\frac{4}{dt}M\right) \\ \left(K_t^{\star[i_f]}\right)^{-1}\left(-\frac{2}{dt}K_t^{[0]} - \frac{2}{dt}K_t^{[i_f]}\right) & \left(K_t^{\star[i_f]}\right)^{-1}\left(\frac{4}{dt^2}M - \frac{2}{dt}C - K_t^{[i_f]}\right) \end{bmatrix} \tag{4.32a}$$

$$K_t^{\star[i]} = K_t^{[i]} + \frac{2}{dt}C + \frac{4}{dt^2}M \tag{4.32b}$$

A more detailed derivation is provided in appendix D. This formula leads to an approximate Jacobian in the sense that the derivation does not explicitly take into account the different iterations of the Newton scheme. However, this Jacobian term appears solely in the dynamic state covariance computation and it is thus reasonable to believe that using an approximate Jacobian will not negatively affect learning of the parameters, as long as it is not completely ignored. If time steps are relatively large however, and the tangent stiffness K_t greatly varies within a time step, we would recommend to subdivide the time step and perform several subsequent propagations (similar to a bisection method, as used in nonlinear FE analysis when the Newton method does not converge).

With respect to the measurement update, many types of measurements lead to a measurement equation h that is truly linear in the states (as explained in the section that considered linear dynamical systems), thus marginalization can be applied exactly. If h happens to be nonlinear in the states, then its Jacobian J_h should be computed for the problem at hand. In the following examples, a small number of states were measured, leading to a simple measurement equation in that case.

Having computed the Jacobians of both transformations, the eMUKF can then be applied to nonlinear problems. The algorithms are identical to the ones presented for the linear systems (Algs. 4 and 5 in the previous section), using the Jacobians $J_{f,h}$ derived in this current section. This novel

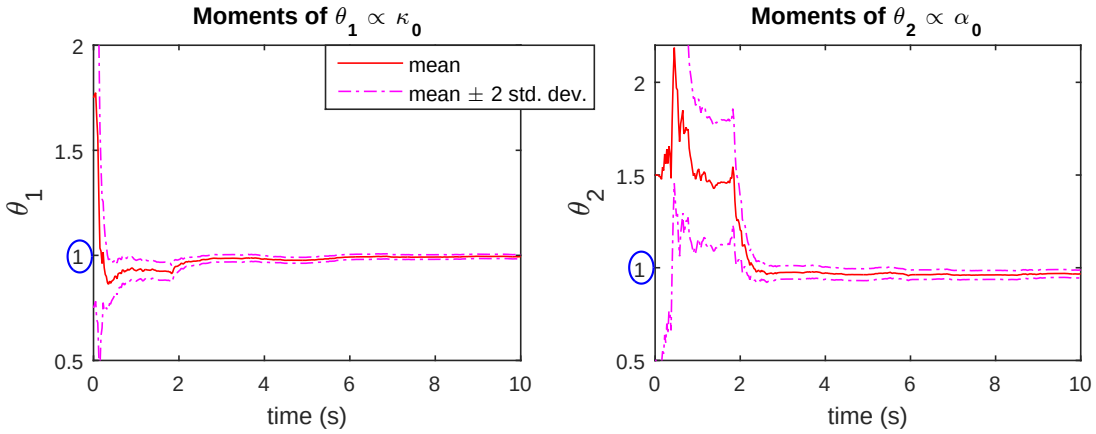
eMUKF was tested on two numerical examples, a parabolic and an hyperbolic problems.

4.4.2 Numerical example: heat conductivity on a plate with nonlinear conductivity

The linear heat conduction problem is revisited here, assuming that the material exhibits a nonlinear conductivity $\kappa(\mathtt{u}) = \kappa_0 e^{\alpha_0 \mathtt{u}}$, where both κ_0 and α_0 must be identified from data. For this specific example, the generic UKF fails after about 2s of data (numerical issues lead to a non-positive definite covariance matrix, preventing the cholesky decomposition required to sample sigma points). Artificial noise was added to both states and parameters to try and render the UKF more robust, which helped alleviate the problem but did not prevent numerical failure. On the contrary, the extended marginalized UKF performs very well here (Fig. 4.10a), in the sense that it is able to learn adequately both unknown parameters. This also draws attention to another crucial advantage of the eMUKF, namely that complex matrix operations such as taking an inverse or performing cholesky decomposition are performed only on small sized matrices (size $\propto n_\theta$), contrary to the generic UKF for which cholesky decomposition is performed on the full covariance matrix (size $\propto n_{x^{dyn}} + n_\theta$). In the eMUKF, operations involving the full covariance matrix are solely additions and matrix multiplications. This renders this algorithm more robust, and, as for the linear systems case, less computationally expensive (see Table in Fig. 4.3, where the running time of the UKF per time step was averaged over all the time steps happening before failure of the algorithm).

4.4.3 Hyperbolic problem: hyperelasticity

The hyperbolic system studied is an infinite thin cylinder made of a Mooney-Rivlin material. Due to symmetry properties, only a quarter cylinder, made of eight 3-dimensional elements is studied (Fig. 4.11a). As for all previous numerical experiments, the nonlinear FE analysis is performed



(a) Nonlinear heat conduction example: time evolution of posterior moments for unknown parameters when using a eMUKF for identification

	generic UKF	eMUKF
number of sigma points	$N_{sig} = 307$	$N_{sig} = 7$
running time per time step	60 s	1.4 s

Table 4.3: Nonlinear heat conduction problem: running times of generic UKF vs. eMUKF

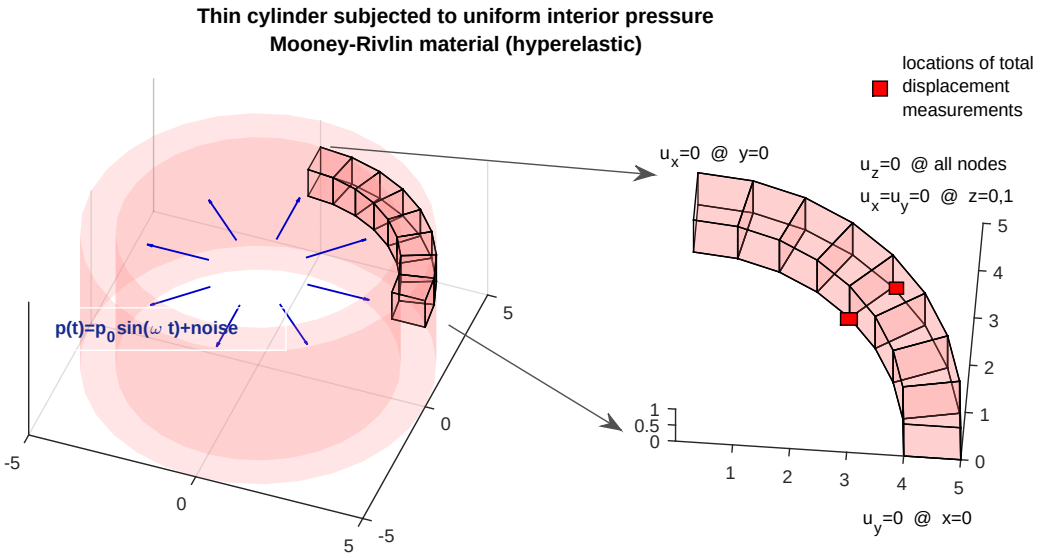
in MATLAB, the code is primarily based on the MATLAB codes given in [Kim, 2015] for static analysis, and are adapted for dynamic analysis as presented in section 4.4.1. The cylinder is subjected to a uniform internal pressure, which primarily varies as a sinusoidal function of time but also includes noise (8% RMS Gaussian noise). For this hyperelastic problem, the deformations are relatively large, which results in the pressure induced force $\mathrm{P} = p(t) \cdot Area$ being a function of the displacement. In the FEM community, such a force, whose magnitude or direction varies as a function of the system's displacement, is known as a *follower force*. It appears often in real life, a well-known example being aerodynamic forces on an aircraft wings, which will change orientation as the wings undergo large displacements or rotations. For such a case of follower force, the calculation of the tangent stiffness matrix should include the derivative of this external force, i.e., $K_t = \frac{\partial(\mathrm{F(u)}-\mathrm{P(u)})}{\partial \mathrm{u}}$. Total displacements at two locations is assumed to be measured and are later used to identify the Mooney-Rivlin material constants $\theta_1 \propto A_{10}$ and $\theta_2 \propto A_{01}$, as defined in [Kim,

2015].

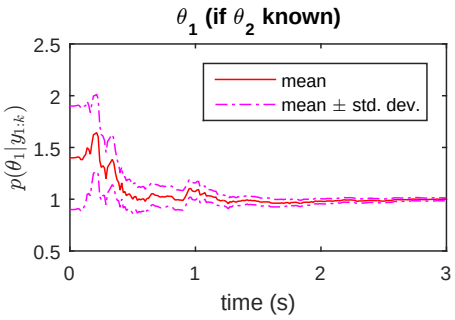
In this experiment, $\theta_2 = 1$ is assumed known and only θ_1 is being identified by the Kalman filtering algorithm. As before, the generic UKF fails after few iterations, due to numerical instability yielding a non-positive definite covariance matrix. The eMUKF on the contrary is able to accurately identify the unknown parameter (Fig. 4.11b).

In a second set of experiments, both parameters are being learnt simultaneously. In this interesting case, parameter θ_1 is still learnt quite accurately, while the true value of θ_2 cannot be accurately inferred. However, the eMUKF infers a large posterior variance for that parameter (middle plot on Fig. 4.11c), indicating that the algorithm is in fact unable to learn much about this parameter from the available data (measurements). Interestingly though, the algorithm also infers a strong posterior correlation between the two parameters, as can be seen on the right plot on Fig. 4.11c.

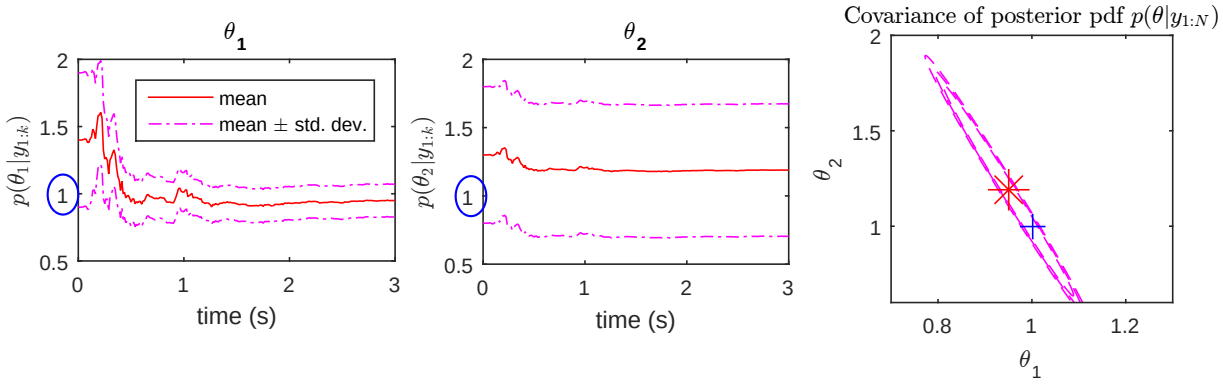
To understand these results, it was decided to investigate the relationship between the parameters and the measurements in an ad-hoc fashion, echoing a sensitivity-based approach. Briefly, sensitivity of the measurements with respect to varying parameters is studied by running the forward problem, and thus generating measurement time-series, using different values of the parameters. First, the sensitivity of the measurements with respect to varying only one parameter is studied, while the other parameter is fixed at its true value 1. Fig. 4.12a shows that the measurements are much less sensitive to variations of parameter θ_2 (i.e., varying θ_2 does not yield large variations in y), agreeing with our previous finding that this parameter is harder to identify than parameter θ_1 based on the available noisy measurements. In a second set of experiments, the forward problem is run, and measurements time-series generated, while varying both θ_1 and θ_2. A 15×15 size grid is generated in the $[\theta_1, \theta_2]$ space, and for each point (i, j), the error between the generated measurement time series using $\theta_1 = i, \theta_2 = j$ and the "true" measurement path ($\theta_1 = \theta_2 = 1$) is calculated, yielding the contour plot in Fig. 4.12b. This plot reveals that indeed, the measurements are not sensitive to variations of the parameters along a certain line that corresponds to the first



(a) Presentation of system studied



(b) Convergence of θ_1 when $\theta_2 = 1$ is known



(c) Convergence of $\theta_{1,2}$ when both are being identified and inferred posterior correlation

Figure 4.11: Performance of eMUKF on hyperelastic problem

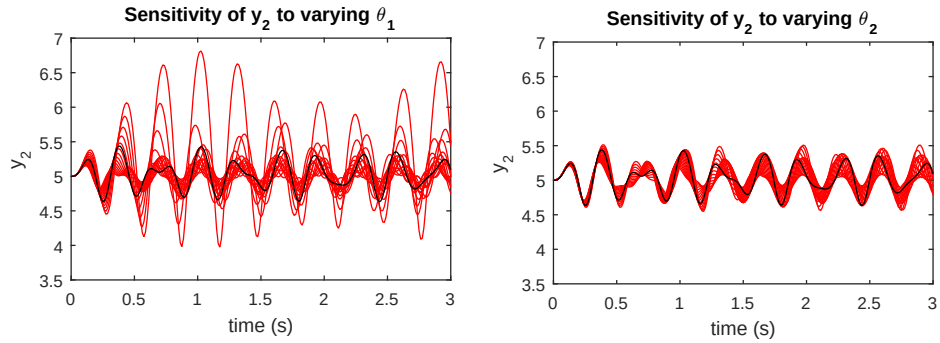
eigenvector of the posterior covariance plotted in Fig. 4.11c (red dotted line). This finding thus validates the posterior covariance previously inferred with the eMUKF.

This example highlights the problem of identifiability in inverse problems, i.e., are the parameters of the model identifiable based on available data? From this example it seems that filtering algorithms such as the UKF and its variants are able to provide the user with useful information with respect to the identifiability characteristics of the model parameters. This analysis is further developed in the next section of this dissertation, where the performance of filtering algorithms is studied for various simple systems whose identifiability properties can be comprehensively investigated using deterministic identifiability analysis.

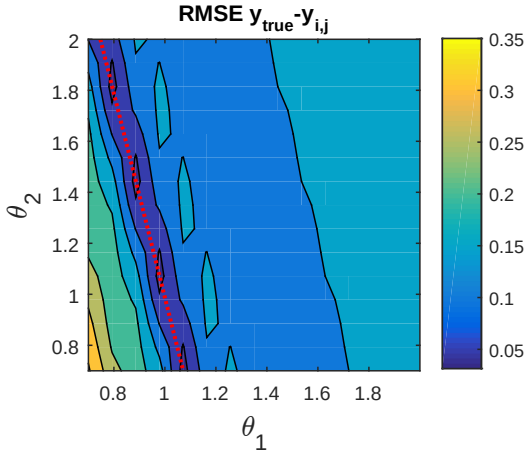
A final interesting remark about Fig. 4.12b is that measurements seem to be much more sensitive to variations in parameters for low values of the parameters (around 0.7 here), which may imply that the true posterior pdf based on these measurements is actually skewed, with low values of the parameters having very low probabilities. However, a Gaussian filter such as the ones studied in this present paper is not able to infer third order moments; a higher order Kalman filter or a particle filter should thus be utilized to study this phenomenon. The question then arises of the applicability of this marginalization method to higher order non Gaussian filters such as the ones presented in the previous chapter of this dissertation, a direction of our foreseen research.

4.5 Concluding remarks

In this chapter, a novel formulation was proposed to facilitate parameter estimation in dynamical systems represented by finite element models. The approach, based on the principle of marginalization, is shown to be more efficient and more robust than the generic UKF, on condition that the number of parameters to be learnt does not scale with the number of DOFs, which would be quite typical for a FE model. Very importantly, it is proven that the marginalization procedure would not



(a) In red, plots of measurement time-series generated using diffenret parameter values (left plot, θ_1 is varied while $\theta_2 = 1$ is fixed, and vice-versa for the rght plot), in black, "true" measurement path, i.e., generated using $\theta_1 = \theta_2 = 1$,



(b) RMSE between "true" measurement path and measurement generated using $\theta_1 = i, \theta_2 = j$ for different (i, j) values, the red dotted line shows the first eigenvector of the posterior covariance plotted in Fig. 4.11c

Figure 4.12: Sensitivity of measurements to varying parameter values for hyperelastic problem.

lead to any reduction in accuracy, compared to a generic UKF, when applied to linear dynamical systems.

For nonlinear systems, the approach makes use of an EKF to propagate and update uncertainties around the dynamic states. The EKF is known to be less accurate than the UKF with respect to moment propagation, especially when dealing with highly nonlinear systems. Furthermore, this approach relies on the fact that an approximate Jacobian of the propagation equation can be easily computed knowing the system matrices. For highly nonlinear hysteretic systems, involving

plasticity for instance, additional states must be added to the state vector in order to obtain a complete picture of the system at a given time step (hysteresis variable r in a Bouc-Wen model of hysteresis for instance). In this case, a Jacobian might not be easily accessible, and the EKF might lead to inaccurate results. Thus the particular case of plasticity and hysteresis should be further studied. Overall, the applicability of this approach and its validity for highly nonlinear systems should be investigated on larger, more complex models, which would require an implementation in an external FEA software such as FEAP for instance.

This page is intentionally left blank.

Part II

Accurate quantification of uncertainties in challenging systems

Chapter 5

Identifiability considerations in parameter estimation

5.1 The question of system identifiability: a challenge to parameter estimation

As shown in the motivating example in chapter 1, when one uses filtering algorithms to perform damage identification in structural systems, being able to accurately learn the posterior pdf of static parameters is crucial. Parameter identification in real-life applications is a very challenging task since 1) measurements from the structure will likely be sparse and noisy and 2) with respect to nonlinear systems, the inputs might not be of high enough amplitude to fully excite all the nonlinearities present in the system.

The fact that one has access to a limited number of measurements at specific locations on the structure raises the questions of state observability and parameter identifiability. Looking solely at identification of static parameters θ, the question of identifiability can be stated simply as whether running the inverse problem using the available measurements yields an infinite number of solu-

tions for the vector θ (unidentifiable), a finite number of solutions (local identifiability), or a unique solution (global identifiability). Knowing the system equations f, h, structural identifiability can be tested using various algorithms, see for instance [Chis et al., 2011] for a thorough comparison of methods applied to nonlinear biology models and [Chatzis et al., 2015b] for a discussion on the subject for nonlinear structural systems. Among all possible methods, we primarily used the software DAISY, [Bellu et al., 2007], since it is capable of distinguishing between local and global identifiabilities, which is known to be a challenging question for nonlinear systems. It has to be noted however that those tests assume that 1) the excitation is exactly known and 2) the system is noise-free. Since in real-life systems are always noisy, one will use probabilistic methods when identifying parameters, even though the noise-free system has a unique parameter solution. For nonlinear systems, introduction of noise in the system can actually render a parameter "unidentifiable", even though it is identifiable for the noise-free system (see example in section 5.2.1).

Identifiability of a system is a property of the system itself (equations of motion and available measurements) and does not depend on the method used to identify the parameters (frequentist vs. Bayesian, on-line vs. off-line). However, since identification methods are based on learning from the measurements, the identifiability properties of the system will have a direct impact on the posterior pdfs $p(\theta|y_{1:N})$, and thus on the choice of filtering algorithm used to perform the inverse problem, as will be explained through several examples in section 5.2. For systems that are globally identifiable (there exists a unique solution to the inverse problem), posterior pdfs would be expected to be unimodal, with relatively small variance, indicating that having access to measurements yields improved knowledge of the parameters. For unidentifiable parameters, one would expect the posterior and the prior pdfs to be relatively close to each other (no learning is possible using the measurements). For locally identifiable parameters (i.e., several parameters would lead to the same measured system response), one would expect multimodal posterior pdfs.

In [Vakilzadeh et al., 2017], an ABC technique (Approximate Bayesian Computation, an off-

line parameter estimation method that does not require explicit knowledge of the likelihood function, contrary to MCMC schemes) is used on several problems that exhibit various identifiability properties, and one can observe that this off-line technique is indeed able to capture non-Gaussian, possibly multi-modal posterior pdfs. In this chapter, we study the behavior of *online* algorithms to parameter identification in systems that also exhibit various identifiability characteristics, some of them adapted from [Vakilzadeh et al., 2017]. The idea is to provide an insight into the kind of behavior to expect from efficient online parameter learning algorithms, depending on the problem at hand, thus guiding the choice of algorithm for future users.

5.2 Performance of online filtering algorithms for systems exhibiting various identifiability properties

5.2.1 Globally identifiable Duffing oscillator

The equation of motion for a Duffing oscillator, with known mass $m = 1$, subjected to a known ground acceleration a_g, is:

$$\ddot{\xi}(t) + c\dot{\xi}(t) + k_{lin}\xi(t) + \beta\xi(t)^3 = -a_g(t) \tag{5.1}$$

where $\xi(t)$ represents the relative displacement of the mass, and c, k_{lin}, β the damping, linear and nonlinear stiffness parameters, respectively. Global identifiability of this system, when measuring either displacement or velocity, can be assessed using for instance the identifaibility software DAISY [Bellu et al., 2007]; meaning that the parameter vector $\theta = \begin{bmatrix} k_{lin} & c & \beta \end{bmatrix}^T$ can be theoretically identified in the absence of noise. If acceleration is measured, one can obtain an expression of the displacement through integration, with the condition that initial conditions are known; thus global identifiability results also apply when measuring acceleration.

In real life applications however, measurements will always incorporate some noise, which

renders identification of some parameters much harder to perform. In the case of a Duffing oscillator, even though theoretically the nonlinearity is excited as soon as $|\xi| > 0$, the presence of noise could render it impossible to detect unless it is excited at a high enough level. To incorporate noise in the simulated data, 10% root-mean-square (RMS) Gaussian noise is added to both acceleration time series ($\ddot{\xi}$ and a_g) before using them for parameter identification. Also, when performing real-time monitoring of a structure, it may not be reasonable to assume that initial conditions are known exactly. Thus, algorithms are initialized with a Gaussian prior on the states $\begin{bmatrix}\xi^T & \dot{\xi}^T\end{bmatrix}^T$ with a non-zero variance (standard deviation is chosen as 20% RMS noise of the corresponding state signal).

Performance of both a particle filter and a UKF are compared using this simple problem. Since the parameters appear linearly in the system equations, the concept of Rao-Blackwellisation can be used to increase efficiency of the PF; more specifically a MLN-RBPF, as presented in chapter 2, is used on this system. When running a UKF, only Gaussian prior pdfs can be considered, while more complex prior pdfs can be studied when running a particle filtering scheme. Thus, the influence of the prior is studied by running simulations with three different priors for the unknown parameters: priors 1 and 2 are Gaussian, with different mean and covariance. Prior 3 is similar to prior 1 (Gaussian) for parameters k_{lin}, c, but it is a mixture of 3 Gaussians for the nonlinear stiffness parameter β, which is harder to identify. In this way, this prior is more uniform over the admissible range of values for β, but it is still possible to run a MLN-RBPFsince the prior is a Gaussian mixture. Prior pdfs for the three parameters are shown in Fig. 5.1a (in order to simplify comparisons between parameters, the parameter vector θ is scaled so that $\theta_{true} = \begin{bmatrix}1 & 1 & 1\end{bmatrix}^T$).

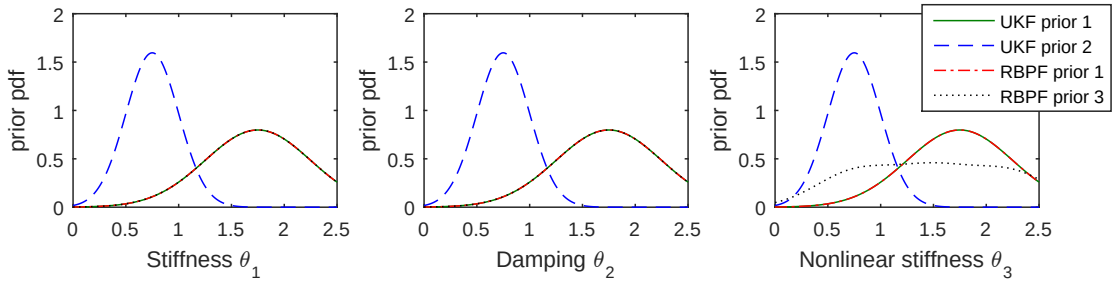
Fig. 5.1b shows results of the identification in the case where the excitation is of relatively high amplitude and the Duffing oscillator is fully excited. Recall that the posterior pdfs shown in this figure are Gaussian by construction for the UKFs, but they are a mixture of n_p Gaussians for the particle filter scheme (MLN-RBPF), with $n_p = 1000$ in this case. Clearly the two algorithms

perform well, and even the particle filters yield posterior pdfs that are very close to being Gaussian. These observations agree with results presented in the previous chapter, where it was concluded that the assumptions made in the UKF do not negatively impact learning of the static parameters in nonlinear systems in the case where parameters are identifiable and noise terms are Gaussian, in the sense that, after convergence of the parameters is achieved, the mean and variance estimates given by the UKF agree with the PF estimates. It also appears that in this case, the choice of prior pdf does not influence the posterior, which makes sense since the algorithm is able to learn from a long time series of measurements, thus somewhat forgetting prior information.

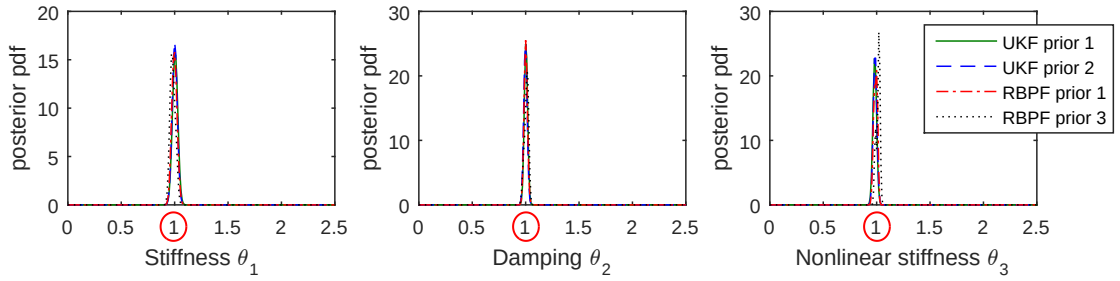
Fig. 5.1c shows results of the identification in the case where the excitation amplitude is reduced by 80%. In this case, the linear parameters k_{lin} and c are still well recovered, but the UKF is unable to learn the nonlinear parameter β. Its posterior pdf heavily depends on the choice of prior, which also makes sense since in this case the algorithm detects that measurements are not very informative, thus the prior knowledge has greater influence. The posterior mean obtained with a particle filter scheme is closer to the true value, implying that in this case, the assumptions made in the UKF (Gaussianity and order of accuracy of the UT) have a negative effect on its performance. Further simulations also showed that reducing the uncertainty on the initial conditions (variance of the prior for the states) seems to help the UKF, in the sense that posterior pdfs are closer to the ones estimated with a particle filter.

For both algorithms, and for any choice of prior variance for the states (level of knowledge of the initial conditions), the posterior pdf for parameter β exhibits a large variance, meaning that the algorithm is detecting that the measurements are not informative and that uncertainty on the identified value of β is high. Thus, even though the PF performs better in this case, running a UKF (much faster) already provides some very useful information about the posterior pdf. Table 5.1 displays the coefficient of variation (CoV) of the parameters identified with the UKF, defined as

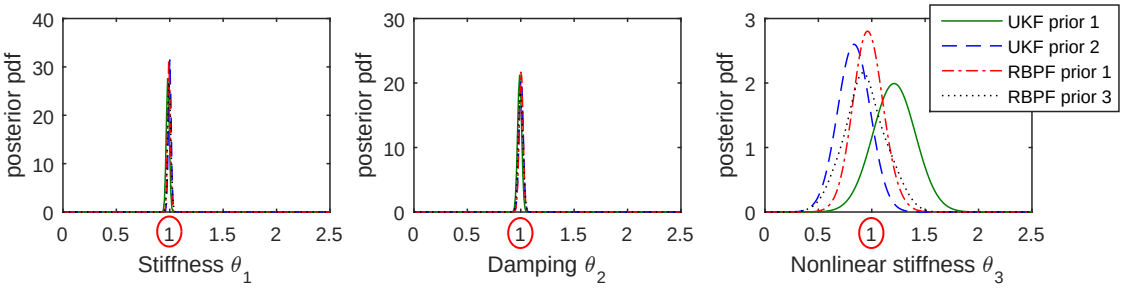
$$\text{CoV} = \frac{\sigma[\theta|y_{1:N}]}{E[\theta|y_{1:N}]} \tag{5.2}$$



(a) Prior pdfs of the three parameters



(b) Posterior pdfs, in the case where the excitation is of high amplitude and the nonlinear oscillator is fully excited



(c) Posterior pdfs, in the case where the excitation is of smaller amplitude and the system behaves almost linearly

Figure 5.1: Performance of UKF and particle filter (MLN-RBPF), with various priors, in estimating the parameters (globally identifiable) of a SDOF Duffing oscillator for different levels of excitation: algorithms differ in their estimation of the nonlinear stiffness β ($\propto \theta_3$) when the nonlinearity is not higly excited.

where $\sigma[\cdot]$ represents the standard deviation of a RV. One can observe that parameter β is not easily identified when the excitation is of small amplitude. This will be a sign for the user that the value identified with the UKF for this parameter should not be trusted, and that possibly running a particle filter on this data would lead to more accurate results.

This issue of unidentifiability due to presence of noise and low amplitude excitations will be

	CoV of identified parameters		
	stiffness θ_1	damping θ_2	nonlinear stiffness θ_3
High amplitude excitation	0.0251	0.0157	0.0177
Low amplitude excitation	0.0146	0.0189	0.1656 ✗

Table 5.1: Coefficient of variation (CoV) of the SDOF Duffing oscillator parameters identified with a UKF: when the nonlinear stiffness is not excited, the UKF is not able to learn it from the measurements, which results in a CoV for this parameter much larger than for other parameters.

inherent in many real-life nonlinear systems. In [Muto and Beck, 2008], the example is given of nonlinear hysteretic structures subjected to seismic event. If some parts of the structure exhibit only linear behavior during the event, no information on their yielding behavior (and associated parameters) will be available from the measurements. It is then crucial to make use of a learning algorithm that tells the user that it is not able to learn these parameters from data, as do the PF and UKF on the Duffing example described in this section. It is also important that the learning algorithm accurately estimates the posterior variance of these parameters, i.e., the posterior uncertainty on these parameters, since these estimates could be further used to estimate future behavior of the structure and quantify associated uncertainties (prognosis step).

5.2.2 Unidentifiable pendulum

In the previous section we have seen that the UKF gives useful information when it comes to noisy nonlinear systems that are not highly excited. Here we confirm this result by looking at the behavior of the UKF on an unidentifiable problem: a unit-length pendulum with rotational spring (stiffness k_{rot}) in free vibration, whose equation of motion is:

$$\ddot{\alpha}(t) = -g \cdot sin(\alpha(t)) - \frac{k_{rot}}{m}\alpha(t) \tag{5.3}$$

where g is the known acceleration of gravity. The angle $\alpha(t)$ is assumed to be measured. To generate artificial noisy data, some zero-mean Gaussian noise is added to both the equation of motion

(modeling error) and the measurements (measurement error). Covariances of both noise terms are assumed known when running the UKF for identification. The true values of the parameters are $m = 1$, $k_{rot} = 10$; we further define the vector of unknown parameters as $\theta = \begin{bmatrix} \frac{m}{1} & \frac{k_{rot}}{10} \end{bmatrix}^T$ whose true value is then $\theta_{true} = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$.

This system is clearly unidentifiable, since there exists an infinite number of vectors θ that would lead the same time series of measurements $y_{1:N}$: namely any vector θ for which the ratio $\frac{\theta_2}{\theta_1}$ equates to 1.

Results of the identification procedure with the UKF are shown in Fig. 5.2. The convergence plots show that the parameters are not identified correctly, since $E[\theta|y_{1:N}] = \begin{bmatrix} 1.7 & 1.7 \end{bmatrix}^T \neq \theta_{true} = \begin{bmatrix} 1 & 1 \end{bmatrix}^T$. This was expected since the system is unidentifiable. However, the standard deviation (st. dev.) around each identified value remains very large, meaning that the algorithm is able to detect that the measurements provided are not informative enough to learn the parameters. Also, the correct ratio $\frac{\theta_2}{\theta_1} = 1$ is identified quickly, and the posterior covariance shows a clear correlation along the line $\theta_1 = \theta_2$. Thus the estimation procedure still provides the user with useful information about this system, i.e., $\theta_1 = \theta_2$, which was expected in this simple case.

5.2.3 Locally identifiable system

We now consider a linear 2-DOF system, where only the second floor acceleration is measured (and the ground excitation is known/measured). This case is studied in [Katafygiotis and Beck, 1998, Vakilzadeh et al., 2017], and the system is shown to be locally identifiable. More specifically, defining $k_{lin,1} = \theta_1 k_{lin,0}$ and $k_{lin,2} = \theta_2 k_{lin,0}$, with known stiffness $k_{lin,0}$, it is shown in [Katafygiotis and Beck, 1998] that both vectors $\begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ and $\begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix} = \begin{bmatrix} 2 \\ 0.5 \end{bmatrix}$ lead to identical measurement time histories (when only the ground and second floor excitations are measured), a result that can be confirmed by running the identifiability software DAISY. The posterior pdf of $\begin{bmatrix} \theta_1 \\ \theta_2 \end{bmatrix}$ is then expected to be multimodal, a characteristic that we know the UKF cannot handle, because of its

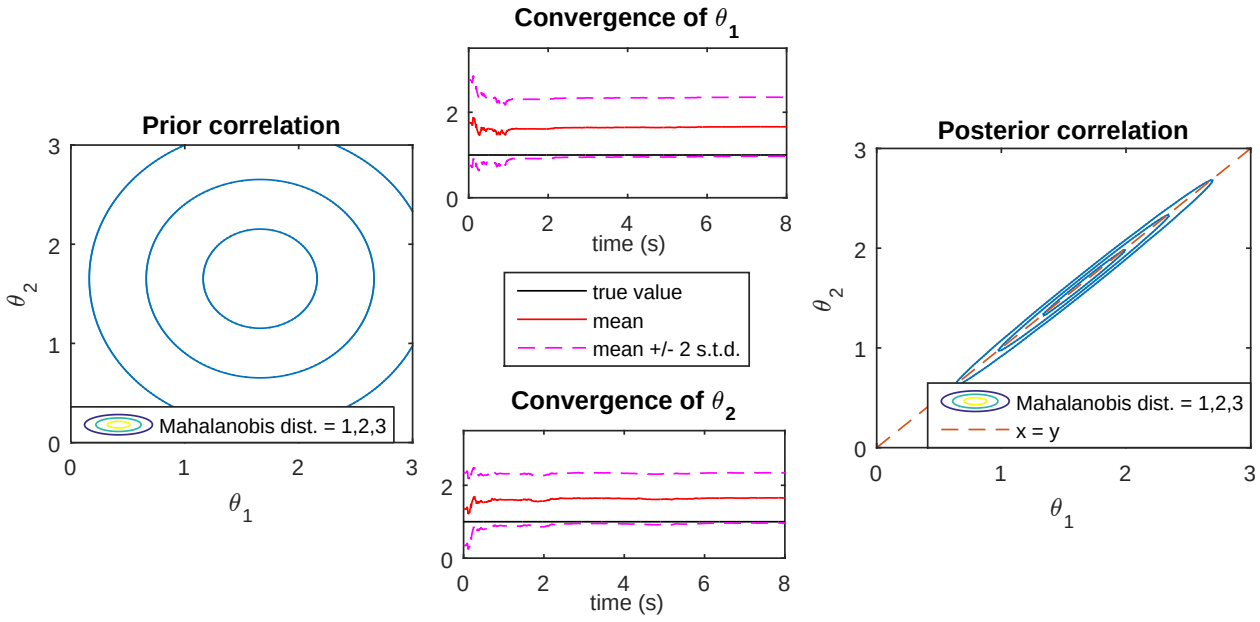


Figure 5.2: Pendulum with rotational spring (unidentifiable system): convergence of parameters (mean value and mean $\pm$ 2 standard deviations) and prior/posterior covariance obtained with the UKF

Gaussianity assumption. Indeed Fig. 5.3 shows two runs of the UKF (two different priors, on the same noisy data): depending on the prior chosen, the UKF will output one of the two acceptable $\theta_{1,2}$[1] vectors of this system, with very low posterior variance. The issue here is that if one runs only one UKF, only one outcome will be detected, with very low variance (i.e. low uncertainty on the outcome), the user will thus be given misleading information about the system.

For this type of system, a particle filter scheme clearly becomes the best option, since it is able to represent any type of distribution, even multi-modal ones. Again, the MLN-RBPFcan be used in this case. We choose a mixture of 9 Gaussians for the prior in the $[\theta_1, \theta_2]$ space, in order to have a prior that nicely explores the admissible space for these two parameters. Fig. 5.4 shows the evolution in time of the posterior pdf in the 2-dimensional space. Clearly the PF with $n_p = 4500$ particles is able to approximate the true posterior, which is in this case bi-modal. In Table 5.2,

[1] In this problem the damping parameters are also assumed unknown, so the size of the parameter vector is 4, however we focus our attention on the lcoally identifiable parameters $\theta_{1,2}$.

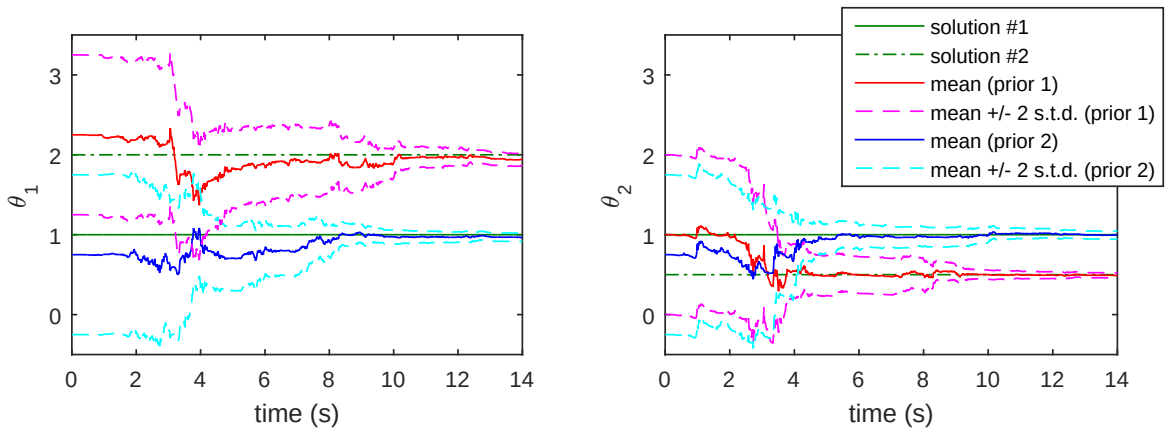


Figure 5.3: 2-DOF linear system: convergence of locally identifiable stiffness parameters when running two UKFs with different priors.

the performance of several PFs (MLN-RBPF), with various numbers of particles, is studied with regard to both accuracy of the posterior pdf estimates (location of both modes of the posterior pdf at $t_{final} = 14s$) and the algorithm's running time (measured by the tic-toc function of MATLAB, for a 14s long time series sampled at $dt = 0.01s$). With only 500 particles, the PF is usually unable to detect the two modes. When using 1000 particles or more (running time 7 min or more), the algorithm is able to detect and localize both modes, however, when using less than about 2000-2500 particles, estimates of the relative weights of each mode are very variable from run to run and quite inaccurate. A PF with 2500 particles outputs consistent estimates of the posterior pdf, and runs in about 18 min.

These results were obtained assuming that the initial conditions on the states were known (the prior was given the correct mean and a small variance). Further simulations were run while relaxing this assumption, i.e., the prior for the states is given the correct mean but a larger variance, 20% RMS of the state time-series, implying that there is uncertainty on the initial conditions of the system. This influences the behavior of the PF in the sense that more particles (about 3000) are needed in this case to consistently obtain an accurate estimate of the posterior pdf. On the contrary, the UKF_GM presented in the following section, which is an enhancement of the Gaussian UKF,

is more robust with regard to uncertainty on the initial conditions.

A limitation of the Rao-Blackwellised version of the particle filter lies in the fact that it requires both functions f and h to be linear in the parameters (recall presentation of the concept or Rao-Blackwellisation in chapter 2, section 2.2), which is quite restrictive. For instance it renders this algorithm impractical for systems in which the functions f, h consist of running an external program (FEM software for instance), for which it will not be possible to find such a conditionally linear partitioning. A generic bootstrap PF does not require that assumption, however it would need an enormous amount of particles to avoid degeneracy and collapse, rendering it quite inapplicable for medium to large dimensional problems. In the following section we use the framework described in the previous chapter of this dissertation to derive an unscented Gaussian mixture filter, thus combining the computational advantages of the UKF while allowing for multi-modal prior and posterior pdfs. This algorithm does not place limitations on the functions f, h, and we demonstrate that it yields very satisfactory estimates of the posterior pdfs in the numerical examples previously studied, while being computationally advantageous over a Rao-Blackwellised particle filter.

5.3 Performance of a non-Gaussian unscented Kalman filter

In the previous chapter of this dissertation, we presented a framework to derive higher order nonlinear filters that would expand the scope of UKF type algorithms to non-Gaussian, possibly multimodal, distributions and thus provide a good trade-off between the accuracy of the PF and the UKF low computational complexity. The main idea is to expand the capabilities of the UKF to non-Gaussian distributions - with non-fixed higher order moments, or even multi-modal distributions - while preserving the main advantages of the Kalman filter algorithms, i.e., working with an underlying distribution rather than a particle based distribution, which facilitates the measurement update step and avoids the degeneracy issue observed in the PF. To derive a nonlinear Kalman filter

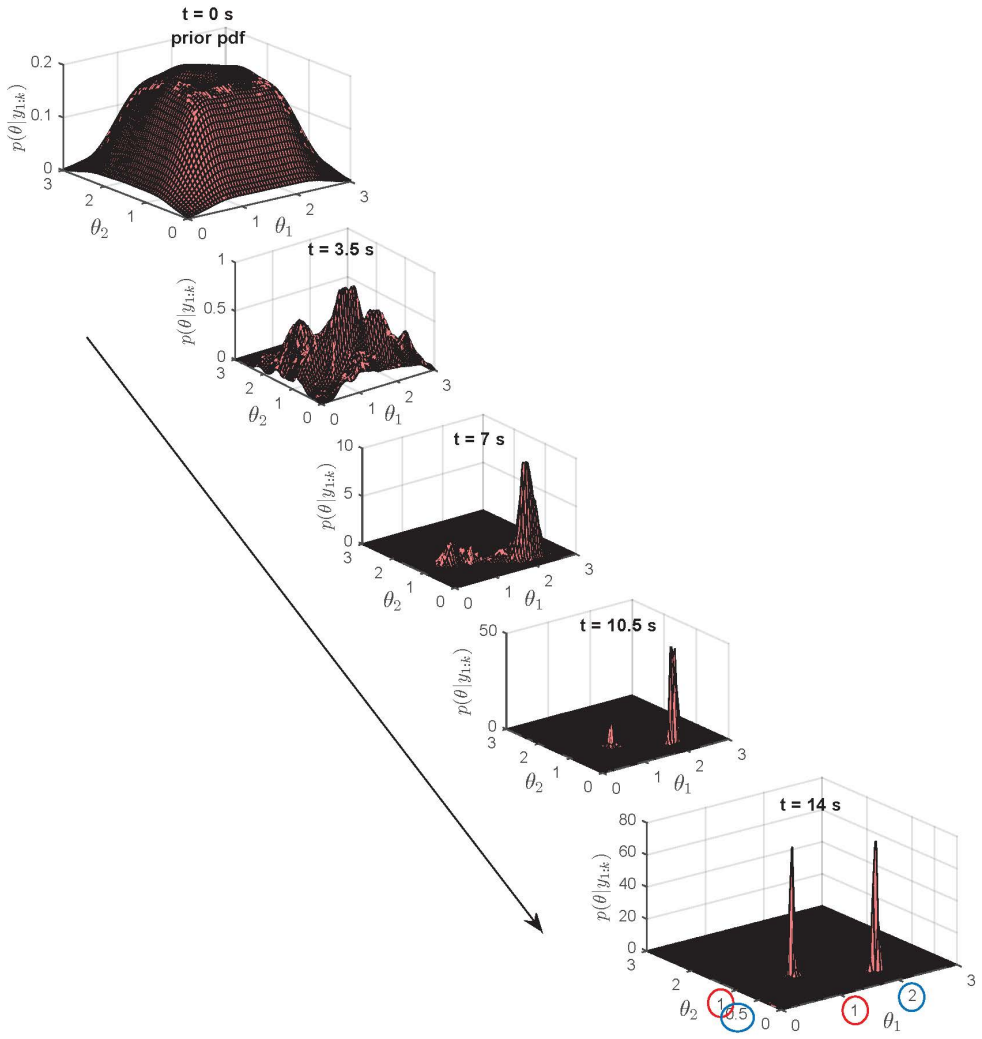


Figure 5.4: 2-DOF locally identifiable system: evolution in time of posterior pdfs $p(\theta|y_{1:N})$ when running a PF with $n_p = 4500$ particles.

number of particles	solution 1 $\theta_{true} = [1 \quad 1]^T$	solution 2 $\theta_{true} = [2 \quad 0.5]^T$	running time
$n_p = 500$	undetected	$[1.96 \quad 0.53]^T$	218 sec
$n_p = 2000$	$[1 \quad 0.97]^T$	$[1.97 \quad 0.54]^T$	856 sec
$n_p = 2500$	$[1.01 \quad 0.95]^T$	$[1.98 \quad 0.51]^T$	1074 sec

Table 5.2: 2-DOF locally identifiable system: performance comparison of PFs with increasing number of particles (accuracy in identification of the two modes of the posterior pdf, vs. computational time).

type algorithm in this framework, one needs two components:

- a baseline distribution that is closed under conditioning to enable derivation of the measurement update equations,
- a choice of approximation to propagate the distribution and its moments through a nonlinear function (Taylor series expansion, UT or Monte-Carlo simulation).

We have shown previously that for unidentifiable/locally identifiable systems, 1) the prior distribution may sometimes have a non-negligible effect on the posterior pdf, and 2) the posterior pdfs might sometimes be multi-modal. Using a mixture of Gaussians (GM) as a baseline distribution to a nonlinear Kalman filter algorithm could thus prove useful. In section 3.4.3, we proposed to use a MC approximation for propagation through nonlinear functions, since the parameters of a GM distribution are usually learnt through maximum likelihood (EM algorithm). However the EM algorithm becomes highly cumbersome in medium to large dimensional spaces. Here we propose a different approach, which uses the unscented transform to approximate the moments of the transformed random variables. The following procedure yields a unscented Gaussian mixture filter (referred to as UKF_GM from now on) which appears to have already been studied independently in other engineering fields such as tracking (see e.g. [Faubel et al., 2009, Luo et al., 2010]), for the purpose of dynamic state filtering in nonlinear systems featuring non-Gaussian noise, possibly causing non-Gaussianity of the posterior pdfs. We quickly present the derivation of this algorithm, afterwards we demonstrate its effectiveness for the purpose of parameter identification in cases where non-Gaussianity of the posterior pdfs arises from a lack of identifiability in the system.

5.3.1 Derivation of the UKF_GM algorithm

Propagation step

As previously mentioned, in this algorithm the posterior pdfs are assumed to be Gaussian mixtures. Let's assume that the posterior pdf at time step $k-1$ is a known mixture of L Gaussians:

$$x_{k-1}|y_{1:k-1} \sim \sum_{l=1}^{L} \pi_{k-1}^{(l)} \mathcal{N}\left(\cdot; x_{k-1|k-1}^{(l)}, P_{k-1|k-1}^{(l)}\right) \tag{5.4}$$

One obtains the prior at time step k by propagating this pdf through the propagation equation, which can also be written in probabilistic format as:

$$p\left(x_k|y_{1:k-1}\right) = \int \overbrace{p\left(x_k|x_{k-1}\right)}^{\text{relates to } f} p\left(x_{k-1}|y_{1:k-1}\right) dx_{k-1} \tag{5.5a}$$

$$= \sum_{l=1}^{L} \pi_{k-1}^{(l)} \underbrace{\int p\left(x_k|x_{k-1}\right) \mathcal{N}\left(x_{k-1}; x_{k-1|k-1}^{(l)}, P_{k-1|k-1}^{(l)}\right) dx_{k-1}}_{\text{propagation of a Gaussian RV through } f} \tag{5.5b}$$

The integral in Eq. (5.5b) represents the propagation of a Gaussian RV through the nonlinear function f, which can be approximated using the UT. Applying L independent unscented transforms yields Gaussian approximations $\mathcal{N}\left(\cdot; x_{k|k-1}^{(l)}, P_{k|k-1}^{(l)}\right)$ for the integrals, and a mixture of Gaussians approximation for the prior:

$$x_k|y_{1:k-1} = \sum_{l=1}^{L} \pi_{k-1}^{(l)} \mathcal{N}\left(\cdot; x_{k|k-1}^{(l)}, P_{k|k-1}^{(l)}\right) \tag{5.5c}$$

The UT is used for each Gaussian $\cdot^{(l)}$ in the mixture, meaning that the total number of sigma points required for approximation of the prior pdf is $N_{sig} = L \cdot (2n_x + 1)$, if the symmetric sigma point set is used for each Gaussian. Such a sigma point set captures up to the second order moment of the GM input RV $X = x_{k-1}|y_{1:k-1}$ (see proof in appendix C.2), and would achieves second order accuracy on mean estimate and first order accuracy on covariance estimate of the output RV $Z = f(X) = x_k|y_{1:k-1}$ (prior pdf, approximated as a GM). In this sense, this algorithm achieves the same order of accuracy as the generic UKF, however it is now able to 1) make use of a more

complex prior and 2) represent more complex pdfs, i.e. multi-modal, as will be demonstrated later with a numerical example.

Before finishing the derivation of this algorithm, it is important to notice that we are making the strong assumption that each transformed Gaussian can itself be approximated as a Gaussian (same assumption as in the UKF), yielding a GM for the output RV. In the cases of highly nonlinear non-Gaussian systems this approximation might lead to erroneous results, thus the UT should be replaced by a MC simulation, as proposed originally in the previous chapter of this dissertation. However, in the cases studied here, where multi-modality arises because of lack of identifiability in the system, making use of this assumption still yields acceptable results.

Measurement update step

Regarding the measurement update step, one must first estimate the joint probability pdf of the prior and the predicted measurement, i.e., $p\left(x_k, y_k | y_{1:k-1}\right)$, which is performed by transforming the prior through the measurement function h. This step can be performed in a similar fashion as for the propagation step, i.e., use independent UTs for each Gaussian constituting the prior, leading to a GM approximation of the pdf of interest as:

$$\begin{Bmatrix} x_k \\ y_k \end{Bmatrix} | y_{1:k-1} \sim \sum_{l=1}^{L} \pi_{k-1}^{(l)} \mathcal{N}\left(\cdot; \begin{bmatrix} x_{k|k-1}^{(l)} \\ y_{k|k-1}^{(l)} \end{bmatrix}, \begin{bmatrix} P_{k|k-1}^{(l)} & \Psi_k^{(l)} \\ \Psi_k^{(l),T} & S_k^{(l)} \end{bmatrix} \right) \tag{5.6}$$

using the same notations as in Eqs. (3.2).

The final step consists of computing the posterior pdf of the states, by conditioning the joint pdf $p\left(x_k, y_k | y_{1:k-1}\right)$ over the measured value y_k. Using the closure under conditioning property of

the GM distribution one can derive the following measurement update equations:

$$\pi_k^{(l)} = \frac{\pi_{k-1}^{(l)} \mathcal{N}\left(y_k; y_{k|k-1}^{(l)}, S_k^{(l)}\right)}{\sum_{j=1}^{L} \pi_{k-1}^{(j)} \mathcal{N}\left(y_k; y_{k|k-1}^{(j)}, S_k^{(j)}\right)} \tag{5.7a}$$

$$\mu_k^{(l)} = x_{k|k-1}^{(l)} + \Psi_k^{(l)} (S_k^{(l)})^{-1} (y_k - y_{k|k-1}^{(l)}) \tag{5.7b}$$

$$\Sigma_k^{(l)} = P_{k|k-1}^{(l)} - \Psi_k^{(l)} (S_k^{(l)})^{-1} \Psi_k^{(l),T} \tag{5.7c}$$

Observing these equations one sees that for each Gaussian $\cdot^{(l)}$ in the mixture, the update equations are exactly the same as for a Kalman filter, or a UKF. One equation is added to the set, and consists of updating the weights $\pi^{(l)}$ of each Gaussian, proportionally to their likelihood.

Treatment of noise terms.

In the present study noise terms are restricted to being additive Gaussian, thus they are introduced in the filtering equations in the exact same fashion as for a generic UKF, i.e., for each Gaussian $\cdot^{(l)}$ in the mixture, the process noise covariance Q is added to the prior covariance $P_{k|k-1}^{(l)}$ at the end of the propagation step, and the measurement noise covariance R is added to the covariance of the predicted measurement $S_k^{(l)}$ before performing the measurement update. This algorithm is also able to handle non-Gaussian noise, as long as it can be represented as a mixture of Gaussians. For more details on the subject, we refer the reader to e.g. [Faubel et al., 2009, Luo et al., 2010].

5.3.2 Comments on the algorithm and its computational time

The algorithm previously presented actually reduces to running several UKFs in parallel, and weighting each UKF at each time step according to their likelihood (Eq. (5.7a)). It is thus very easy to implement. This algorithm can be viewed as a trade-off between a UKF and a PF, in the sense that if the mixture is composed of one single Gaussian, this algorithm reduces to a generic UKF; if on the contrary the prior is composed of many Gaussians, each with an infinitely small

prior covariance, the algorithm becomes very similar to a particle filter, where all particles (which can be interpreted as degenerated Gaussians) are propagated and then weighted according to their likelihood.

The overall computational time of this algorithm grows proportionally with $n_x \cdot L$: it is thus proportional to both the dimension of the system and the number of Gaussians in the mixture, which governs the complexity of the pdfs of interest. If the algorithm requires highly complex Gaussian mixtures, this algorithm will then not be beneficial over a particle filter in terms of computational time. However, we have deduced from the previous numerical examples that if the parameters of the system, or at least some of them, are identifiable, then making use of a simple Gaussian prior for these parameters already yields accurate estimates. The idea in this new algorithm, when used for parameter identification purposes, is to start with a simple Gaussian prior in some of the dimensions, while using a more complex Gaussian mixture prior for the parameters that are harder to learn. Hence the number of Gaussians in the mixture will not grow excessively with the overall dimension of the system. A more detailed comparison of the running time for the PF vs. UKF_GM is provided for the following numerical example.

5.3.3 Numerical performance of the UKF_GM

Linear 2-DOF locally identifiable system

The UKF_GM is first tested on the 2-DOF locally identifiable system previously presented. The chosen prior is composed of a mixture of 49 Gaussians, leading to a quite uniform prior over the $[\theta_1, \theta_2]$ space (left plot in Fig. 5.5); however these 49 Gaussians collapse into a single Gaussian in the remaining dimensions. The idea here is that a much more complex prior (i.e., many more Gaussians) would be necessary to uniformly explore the n_x-dimensional space, which would lead to a very computationally expensive algorithm. However, by only exploring uniformly the $[\theta_1, \theta_2]$ space, one is already able to infer the multi-modal posterior pdf, as shown in Fig. 5.5 (right plot).

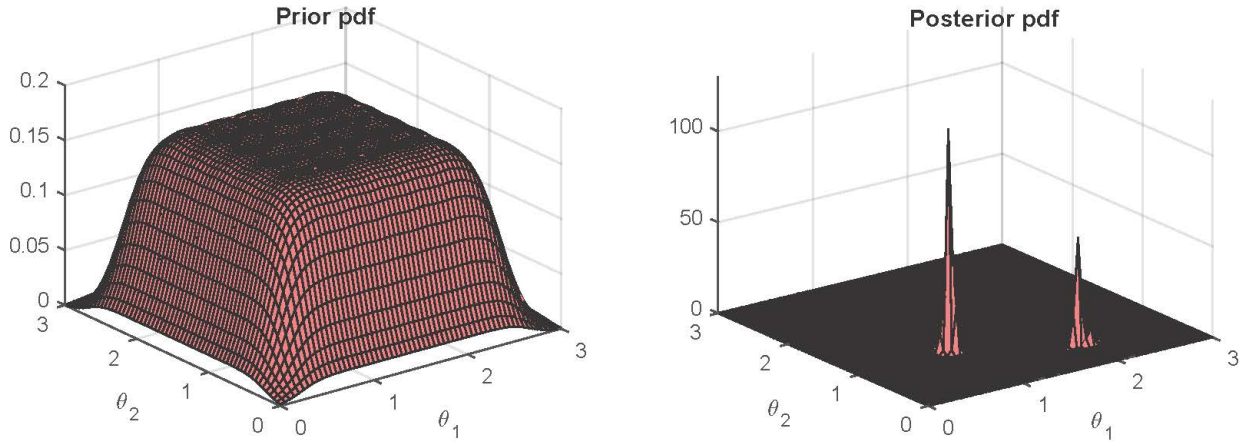


Figure 5.5: Performance of the UKF_GM (using a mixture of 49 Gaussians) on the 2-DOF locally identifiale system: prior and posterior pdfs of the stiffness parameters.

number of sigma points	solution 1 $\theta_{true} = [1 \quad 1]^T$	solution 2 $\theta_{true} = [2 \quad 0.5]^T$	running time
symmetric set $N_{sig} = L \cdot (2n_x + 1) = 833$	$[1 \quad 0.97]^T$	$[1.98 \quad 0.49]^T$	257 sec
simplex spherical set $N_{sig} = L \cdot (n_x + 2) = 490$	$[1 \quad 0.97]^T$	$[1.98 \quad 0.49]^T$	161 sec

Table 5.3: Performance comparison of the UKF_GM with two sigma point sets on the 2-DOF locally identifiable system: identification of the two modes of the posterior pdf, vs. computational time. For both algorithms, the function f consists of a RK2 discretization scheme, and the additive noise version of the UKF is used.

As previously mentioned, a tremendous advantage of the UKF_GM over a Rao-Blackwellised version of the PF is that it does not make any assumption on the functions f, h, while the MLN-RBPFrequires the functions to be linear in the parameters. In this specific case for instance, the measurements were generated by running the forward problem using a 4th order Runge-Kutta discretization scheme. However learning the parameters with the MLN-RBPFrequires writing the function f using an Euler discretization scheme. When running the UKF_GM on the contrary, we were able to use a Runge-Kutta discretization scheme for the function f, i.e., the model used to run the inverse problem is closer to the true system.

In Table 5.3, performance of this algorithm, using two different sigma point sets (the symmet-

ric set and the simplex spherical set, both plotted on Fig. 3.3), is analyzed in terms of accuracy of the posterior pdfs and running time, as done previously for the PF. One can see that the UKF_GM is able to quite accurately identify the two modes of the posterior pdf, and does so with a reduced computational time compared to a PF (MLN-RBPF). This can be explained by looking more precisely at the computational effort required for each algorithm. For the PF, the main computational load originates from evaluating $f(x_{k-1}^{(i)}, \theta^{(i)})$, $h(x_k^{(i)}, \theta^{(i)})$ as well as performing two linear measurement update equations (to update the conditional pdf of the conditionally linear parameters), and so for each particle. Regarding the UKF_GM, for each of the N_{sig} sigma points, with $N_{sig} = 833$ for the symmetric set and $N_{sig} = 490$ for the simplex set, the main computational load originates solely from evaluating $f(x_{k-1}^{(i)}, \theta^{(i)})$, $h(x_k^{(i)}, \theta^{(i)})$, then only L measurement updates are needed per time step. This algorithm thus reduces computational time in two ways compared to the Rao-Blackwellised version of the PF: reduce the number of particles/sigma points per time step and reduce the computational time per particle/sigma point (for the same function f).

The number of mixtures in the GM is chosen in advance by the user. This number should not be chosen too small for two reasons, 1) having a complex prior enables a good coverage of the prior parameter space, which is required in order to detect all possible parameter solutions and 2) we have observed in our experiments that if only few Gaussians are considered, the algorithm tends to collapse as in a PF, i.e., one Gaussian will be given all the weight and the posterior pdf will be Gaussian.

Duffing oscillator

The performance of the UKF_GM was also assessed on the Duffing oscillator problem previously considered, in the case where the nonlinearity is not fully excited. Recall that in this case, the UKF is not able to accurately learn the nonlinear stiffness parameter α. Very importantly, comparing behaviors of UKFs with different priors seems to indicate that the chosen prior has a non negligible

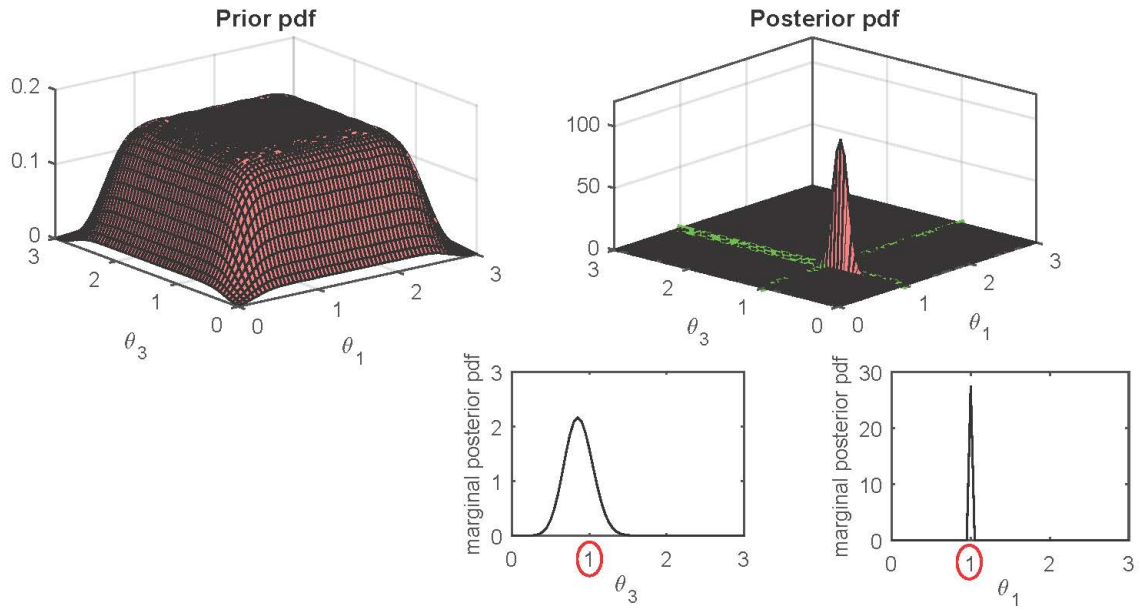


Figure 5.6: Performance of the UKF_GM (using a mixture of 49 Gaussians) on the SDOF Duffing oscillator (low amplitude excitation): prior and posterior pdfs of the linear and nonlinear stiffness parameters, $\theta_1 \propto k_{lin}$ and $\theta_3 \propto \beta$.

influence on the posterior. It is thus important in this case to be able to accurately represent the level of prior knowledge with a proper prior distribution. For this test we chose again as prior pdf a mixture of 49 Gaussians, which explores quite uniformly the 2-dimensional $[\theta_1, \theta_3] \propto [k, \alpha]$ space (since the two stiffness parameters are correlated, the algorithm yields much better results when this 2-dimensional space is explored uniformly, rather than just the θ_3 dimension). Both the prior and posterior distributions are shown in Fig. 5.6, demonstrating the ability of this new algorithm to learn the posterior value of α in a more accurate way than a single UKF, while still yielding a large variance for this parameter, indicating that measurements are not very informative in this case.

5.4 Concluding remarks

Overall this discussion demonstrates that on-line algorithms have excellent capabilities regarding joint state/parameter estimation, even when the lack of identifiability of the system causes the

posterior pdfs to be non-Gaussian, multi-modal. Furthermore, these algorithms make use at each time step of independent evaluations of the functions f, h at various points in the state space, they are thus quite easily parallelizable (see e.g. [Brun et al., 2002] for PF parallelization), making them very attractive for potential real-time monitoring of systems. This further enables fast decision making procedures, very important in the case of structural health monitoring. Also, these first posterior estimates could be used to speed up off-line methods such as MCMC schemes, by using them as starting point or as importance densities in the case of importance sampling, thus yielding more accurate estimates of the posterior pdfs.

The issue of parameter identifiability studied in this chapter will become of vital importance for real life, large dimensional, high fidelity models since the number of unknown parameters will surely increase, however the number of measurements might not be allowed to augment proportionally, for practical reasons. A challenge in high fidelity models (finite element models for instance), lies in the choice of the parameter vector to be learnt from data (engineering judgment vs. sensitivity-based methods [Jang and Smyth, 2017] for instance), and then to assess the identifiability of this parameter vector. For the small scale examples presented therein, whose system equations f and h were easy to write, one could assess the system's identifiability properties prior to learning, through the use of identifiability tests such as the DAISY software. Running identifiability tests on large dimensional problems, described by FE models, would be much more challenging. Thus, for large scale systems, identifiability characteristics of the parameter vector might not be known in advance, it is then of primordial importance to understand how on-line algorithms perform in order to accurately analyze and understand the results of the identification task.

Another issue that will surely arise when considering real life, possibly large dimensional systems, is a potential lack of knowledge of the excitation time series. Indeed, as mentioned in the introduction of this dissertation, we have assumed up to now that the excitation time series $e_{1:N}$

could be measured. This assumption would be reasonable for seismic excitation, where only the ground acceleration should be measured to know the force input at each DOF of the structure. However, in many cases, the input force might be impossible to measure at each DOF of the structure, i.e., the time series $e_{1:N}$ that appears in the filtering equations will not be available during monitoring. In the following chapter, this topic of unmeasured excitation is studied in the case where the excitation $e(t)$ is a stationary stochastic process, with a focus on wind excitation.

Chapter 6

Identification of systems subjected to unmeasured stochastic inputs

6.1 Filtering algorithms with unmeasured stochastic inputs: problem statement

6.1.1 Introduction of the excitation in the filtering equations

Using Bayesian filtering techniques on a physical model of the structure presents clear advantages for structural health monitoring purposes (better understanding of possible cause and extent of damage), however it shows a major drawback in the sense that the algorithm requires knowledge of the excitation e_k at each DOF, contrary to some statistical pattern recognition algorithms that can make use of black-box output-only models of the structure and thus do not require knowledge of the input, or at least not at each DOF. In order to use Bayesian filtering techniques, the excitation must be introduced in the equations as either:

- the term e_k in the process equation, i.e., the input excitation is known or measured at each

DOF, or

- the term v_k in the process equation, i.e., the input excitation is white noise.

The input white noise v_k could be either Gaussian, then a generic UKF could be used in a straightforward manner, or non-Gaussian, then more accurate results could be obtained using non-Gaussian Kalman filters previously described in this thesis, or particle filters. However the white-noise assumption is quite restrictive, and few inputs would actually fall in this category. On the other hand, Bayesian filtering techniques have been shown to perform very well on MDOF structures subjected to earthquake loading, for which only measurement of the ground acceleration $a_g(t)$ as well as reliable mass estimates are needed to obtain the input excitation at each DOF ($e(t) = -m \cdot a_g(t)$ is known). However, for many types of random loadings, the force will not be the same at each DOF, and thus many measurements of the input would be required, rendering the use of Bayesian filtering impossible. This is for instance the case with wind loading, as explained in the following sections.

One can also mention at this stage that Bayesian filtering algorithms can also be used in certain cases to perform joint state/input estimation, as presented for linear models in [Eftekhar Azam et al., 2015] for instance. These methods require first to identify the physical model (parameters) of the linear model, which can then be used in a second stage to perform joint estimation of the dynamic states and the input, knowing the physical model of the structure. In the following however, we are solely concerned with performing accurate online monitoring of static parameters, in possibly nonlinear systems, in the case where the excitations cannot be directly measured. As will be discussed later, because of observability/identifiability constraints, it is not reasonable to assume that one could simultaneously monitor/identify both the parameters of a system and the excitation to which it is subjected.

6.1.2 Introducing correlation in the noise terms of the filtering equations

At this stage it is useful to recall the system equations

$$x_k^{dyn} = f(x_{k-1}^{dyn}, \theta, e_{k-1}, v_{k-1}) \quad \text{(propagation equation)} \tag{6.1a}$$

$$y_k = h(x_k^{dyn}, \theta, \eta_k) \quad \text{(measurement equation)} \tag{6.1b}$$

where we now relax the assumption of noise additivity, i.e., the noise terms can appear nonlinearly in the equations. Both particle filtering and nonlinear Kalman filtering schemes are capable of handling such systems. For the UKF, which will be used in the remainder of this chapter, the noise terms are then treated similarly as the states, i.e., a set of sigma points is computed to capture their moments and propagate them through the nonlinear functions f, h. In its noise augmented form, the UKF is also capable of handling noise terms that exhibit correlation one time step apart, i.e., ($E[v_{k-1}\eta_k^T] \neq 0$). Lately, research in various engineering fields (control and automation, navigation, geodetic observation and monitoring) has grown towards ways to tackle accurately more complex noise terms:

- incorporating correlation at the same time step ($E[v_k\eta_k^T] \neq 0$), see e.g. [Chang, 2014b, Saha and Gustafsson, 2012] for an introduction in the particle filter and UKF respectively, and
- considering non-white noise, see e.g. [Masoumnezhad et al., 2014] for an introduction of colored noise with given structure (time correlation in the noise is described by a 1st order linear filter) and [Sui et al., 2015] for an introduction of noise described by an autoregressive moving average (ARMA) model.

As will be explained later, this chapter also tackles the introduction of non-white noise into the Bayesian framework, in order to take into account non-white and non measured excitations, more specifically wind excitation, in this filtering framework. We will show how linear filters and autoregressive (AR) models can both be used for this purpose. Also, in the literature the noise terms

are usually assumed additive, which will not be the case for our problems of interest, we then show how this issue can be bypassed by augmenting the state vector with a vector of intermediate states.

6.1.3 Wind turbulence stochastic process

In this section we introduce our assumptions regarding modeling of the wind loading exciting the structure whose states and parameters we are monitoring. We follow similar notations and definitions as in [Deodatis, 1996, Shinozuka and Deodatis, 1991]. Let $\upsilon(t)$ be a 1D-1V stationary stochastic process (its mean and variance do not vary over time) with zero mean value and autocorrelation function $R_\upsilon(\tau)$ and two-sided power spectral density function $S_\upsilon(\omega)$, with ω the frequency in rad/s. Then the following relations hold:

$$E\left[\upsilon(t)\right] = 0 \tag{6.2a}$$

$$E\left[\upsilon^2(t)\right] = R_\upsilon(0) \tag{6.2b}$$

$$E\left[\upsilon(t+\tau)\upsilon(t)\right] = R_\upsilon(\tau) \tag{6.2c}$$

$$S_\upsilon(\omega) = \frac{1}{2\pi}\int_{-\infty}^{\infty} R_\upsilon(\tau)e^{-i\omega\tau}d\tau \tag{6.2d}$$

$$R_\upsilon(\tau) = \int_{-\infty}^{\infty} S_\upsilon(\omega)e^{i\omega\tau}d\omega \tag{6.2e}$$

A white noise process has constant PSD and autocorrelation function $R(\tau) = \sigma^2\delta(\tau)$, i.e., two realizations at different time steps are uncorrelated.

Wind speed fluctuations are often modeled as a non-white stationary process (contrary to earthquake excitation for instance, which is clearly non stationary). Then one can adopt for instance the Kaimal spectrum to represent the time/frequency correlation of the wind speed fluctuations (around a mean value $U(z)$, in m/s):

$$S_\upsilon(\omega) = \frac{1}{2}\frac{200}{2\pi}u_*^2\frac{z}{U(z)}\frac{1}{\left(1+50\frac{\omega z}{2\pi U(z)}\right)^{\frac{5}{3}}} \tag{6.3}$$

with z the height in meters from the surface of the ground, ω frequency in rad/s, u_* the shear

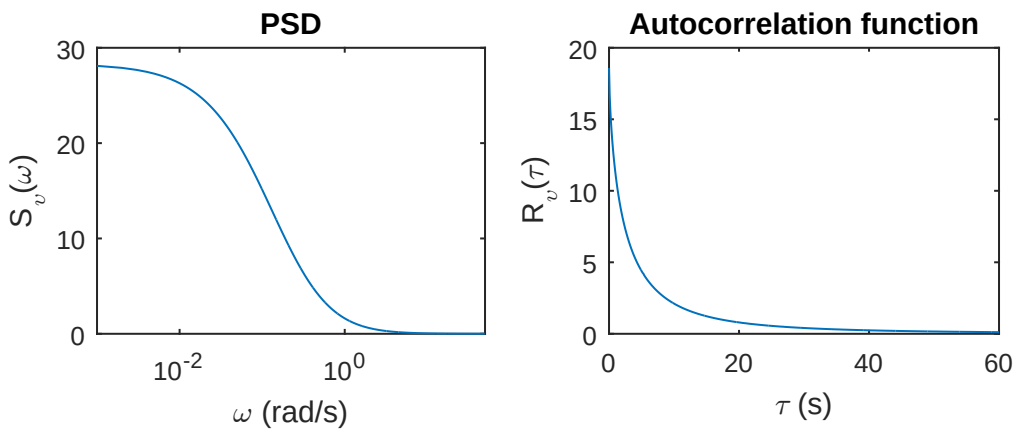


Figure 6.1: PSD $S_\upsilon(\omega)$ and autocovariance function $R_\upsilon(\tau)$ for wind turbulence following the Kaimal spectrum with $z = 25$ m, $U(z) = 43.52$ m/s and $z_0 = 0.001266$.

velocity of the flow, in m/s, which can be computed using the logarithmic law:

$$U(z) = \frac{1}{K} u_* ln\left(\frac{z}{z_0}\right) \tag{6.4}$$

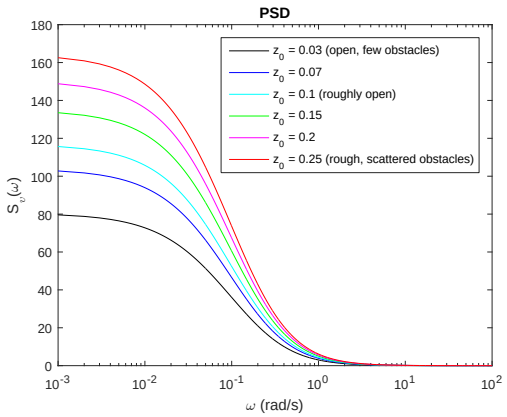
where z_0 is the roughness length, which depends on the surrounding area, and von Karman's constant $K \simeq 0.4$. Plots of the Kaimal PSD and associated autocorrelation function are shown in Fig. 6.1. The variance of the process can be computed as [Shinozuka and Deodatis, 1991]:

$$\sigma_\upsilon^2 = R_\upsilon(0) = 6u_*^2 \tag{6.5}$$

Note that in the definition of the spectrum, we can assume that z, the height in meters, and $U(z)$, the mean wind speed at height z can be measured quite accurately. Also, von Karman's constant is usually set to $K \simeq 0.4$ and we will keep this assumption. However, the roughness length z_0, which depends on the surrounding area, can be hard to measure/identify. For example, the table in Fig. 6.2a, reproduced from [Barthelmie et al., 1993], shows approximate values of z_0 for each type of surrounding area. Also, Fig. 6.2b shows plots of the Kaimal spectrum for several values of the parameter z_0 (from $z_0 = 0.03$ (open area) to $z_0 = 0.25$ (scattered obstacles)). From knowledge of the surrounding area, one obtains only a crude estimate of the parameter z_0; since this parameter

Terrain description	z_0 (m)
Sea - open sea, fetch at least 5 km	0.0002
Smooth - mud flats, snow: little vegetation, no obstacles	0.005
Open - flat terrain; grass, few isolated obstacles	0.03
Roughly open - low crops; occasional large obstacles	0.10
Rough - high crops; scattered obstacles	0.25
Very rough - orchards, bushes; numerous obstacles	0.5
Closed - regular large obstacle coverage; (suburb, forest)	1.0
Chaotic - city center with high- and low-rise buildings	> 2

(a) Roughness lenghts from terrain classification, reproduced from [Barthelmie et al., 1993].



(b) PSDs for different values of z_0 between 0.03 and 0.25.

Figure 6.2: Variation of input parameter z_0 and Kaimal spectrum depending on characteristics of surrounding area, for fixed $z = 35$ m and $U(z) = 45$ m/s. Turbulence intensity $\frac{\sigma_v}{U(z)}$ increases with z_0.

greatly affects the excitation, one can wonder how this uncertainty will affect identification of the structural parameters. This question is tackled in detail later in this chapter, using uncertainty propagation tools.

Wind turbulence processes at different points in space are also correlated (random field), which has to be taken into account when modeling a structure with several DOFs. To take into account this spatial correlation, one defines a coherence function $\gamma(\xi, \omega)$ that depends on the spatial distance ξ between two points of interest [Deodatis, 1996]. One can then define a power spectral density matrix (PSDM) that takes into account both the coherence function and the univariate Kaimal spectrum. In this chapter, only 1V-1D processes are considered and applied to SDOFs structures, both linear and nonlinear. However, the methods presented therein could be extended to multivariate processes and MDOFs systems in the future.

The force applied to a structure due to wind speed is computed as:

$$\varphi(t) = \frac{1}{2}\rho_{air} \cdot Area \cdot C \cdot (U(z) + v)^2 \tag{6.6}$$

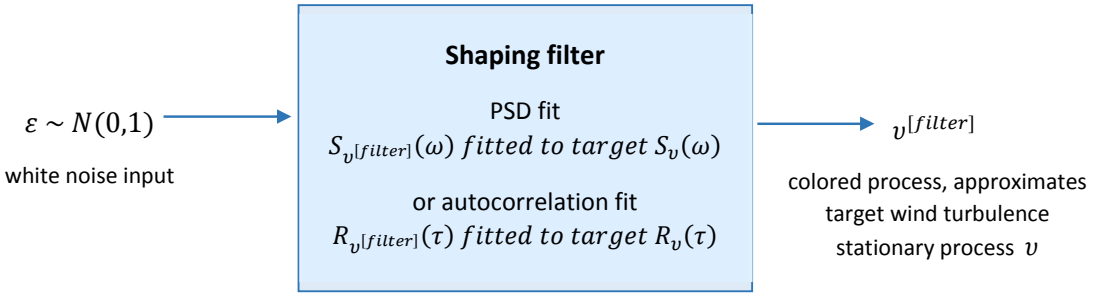


Figure 6.3: Principle of the shaping filter to generate univariate non-white processes from a white noise input.

Since the wind turbulence υ is non-white, one would need to obtain measurements of the wind turbulence or the excitation at each DOF of interest in order to perform Bayesian inference. This would become unfeasible for many systems. In the following section we present a way to create an approximation of the wind turbulence that can be directly incorporated in a UKF.

6.2 Shaping filters: introducing non-white input processes in the UKF

The idea here is to incorporate in the UKF a signal $\upsilon^{[filter]}$ that takes into account time/frequency correlation of the input excitation process. This can be achieved using a so-called shaping filter (see e.g. [Maybeck, 1979]), which takes as an input a white noise and outputs a colored noise (Fig. 6.3). To generate an appropriate wind turbulence, one also wants to match the PSD or autocorrelation function of the filter output to the target wind spectrum, here the Kaimal spectrum presented in the previous section.

6.2.1 Presentation of shaping filters

Continuous linear filter to fit the target PSD

We define a linear shaping filter in continuous time as:

$$\dot{q}(t) = Aq(t) + \sigma\varepsilon(t) \tag{6.7a}$$

$$\upsilon^{[filter]}(t) = Cq(t) \tag{6.7b}$$

where $\upsilon^{[filter]}$ is the shaping filter output, $q(t)$ an intermediate state, $\varepsilon(t) \sim \mathcal{N}(\cdot;0,1)$ the white noise input, thus the term $\sigma\varepsilon(t)$ has constant PSD $S_e(\omega) = \frac{\sigma^2}{2\pi}$. The transfer function $H_{\upsilon^{[filter]}}$ and PSD $S_{\upsilon^{[filter]}}$ of the filter output can be computed easily (linear filtering theory, see e.g. [Maybeck, 1979]):

$$H_{\upsilon^{[filter]}}(\omega) = \frac{\mathcal{L}(\upsilon^{[filter]})}{\mathcal{L}(\varepsilon)} = C(j\omega I - A)^{-1} \tag{6.8a}$$

$$S_{\upsilon^{[filter]}}(\omega) = |H_{\upsilon^{[filter]}}(\omega)|^2 S_e(\omega) \tag{6.8b}$$

One needs to create a filter so that $S_{\upsilon^{[filter]}}(\omega)$ matches the target Kaimal spectrum $S_\upsilon(\omega)$. A numerical example using a first order linear shaping filter is shown later in this chapter. The order of the filter can be increased in order to obtain a better fit to the target spectrum, however as will be explained later this will also lead to higher computational complexity of the UKF.

The filter must then be discretized in time to be integrated into the UKF. Since the filter is linear, one can discretize exactly (time step dt, see e.g. [Maybeck, 1979]) as:

$$q_k = A_d q_{k-1} + \sigma_e \varepsilon_{k-1} \tag{6.9a}$$

$$\upsilon_k^{[filter]} = C_d q_k \tag{6.9b}$$

with $\varepsilon_k \sim \mathcal{N}(\cdot;0,1)$ white noise (process noise term in the UKF equations) and

$$A_d = e^{Adt}, \quad C_d = C, \quad \sigma_e^2 = \int_{\tau=0}^{dt} e^{A\tau}\sigma^2 e^{A^T\tau} d\tau \tag{6.10}$$

AR model to fit the target autocorrelation function

In stochastic process theory, autoregressive models are also widely used to generate sample paths of stochastic processes, using as an input a white noise discrete process. AR models can be written as:

$$u_k = \sum_{l=1}^{p} \alpha_l u_{k-l} + \sigma_e \varepsilon_k \tag{6.11}$$

where $\varepsilon_k \sim \mathcal{N}(\cdot;0,1)$ and p is the order of the AR model. As done previously for the continuous filter, one could compute the transfer function of this AR model, then choose the parameters $\alpha_{1:p}$ and σ_e to fit the PSD of the target wind turbulence process.

Alternatively, one can also use the Yule-Walker equations to fit the AR model to the target spectrum. The Yule-Walker equations relate the autocorrelation function $R(\tau)$ of the AR process to its parameters $\alpha_{1:p}$ and σ_e:

$$\begin{Bmatrix} R(1) \\ R(2) \\ R(3) \\ \vdots \\ R(p) \end{Bmatrix} = \begin{bmatrix} R(0) & R(1) & R(2) & \cdots & R(p-1) \\ R(1) & R(0) & R(1) & \cdots & R(p-2) \\ R(2) & R(1) & R(0) & \cdots & R(p-3) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ R(p-1) & R(p-2) & R(p-3) & \cdots & R(0) \end{bmatrix} \begin{Bmatrix} \alpha_1 \\ \alpha_2 \\ \alpha_3 \\ \vdots \\ \alpha_p \end{Bmatrix} \tag{6.12a}$$

$$\sigma_e^2 = R(0) - \sum_{l=1}^{p} \alpha_l R(l) \tag{6.12b}$$

where $R(l)$ is the autocorrelation function computed at $\tau = l \cdot dt$, dt being the lag between two time steps k and $k+1$ (and $R(0)$ is the variance of the process). Using these equations, parameters $a_{1:p}$ and b_0 can then be chosen so that the autocorrelation function of the AR process matches the one of the target process, for $l \leq p$, by setting $R(l) = R_v(l), l \leq p$ and solving the Yule-Walker equations to find $\alpha_{1:p}$ and σ_e. Thus by increasing the order p of the AR model, one obtains a better fit to the target autocorrelation function, as can be seen in the numerical example presented in the next section.

In order to implement this method in the UKF, one defines the vector of intermediate states $q_k =$

$\begin{bmatrix} u_k & u_{k-1} & \dots & u_{k-p} \end{bmatrix}^T$ and the approximated excitation as $\upsilon_k^{[filter]} = u_k = \begin{bmatrix} 1 & 0 & \dots & 0 \end{bmatrix}^T q_k$. One can then directly write a discrete system similar to Eq. (6.9).

6.2.2 Numerical example

We here derive a linear 1st order shaping filter to match the Kaimal PSD ($K = 0.4$, $z_0 = 0.001266$, $z = 25m$, $U(35m) = 45m/s$). The PSD and autocorrelation function of a 1st order linear filter

$$\dot{q} = -\frac{1}{T_f} q + 2\pi \frac{Q_f}{T_f^2} \varepsilon \tag{6.13a}$$

$$\upsilon^{[filter]} = q$$

can be computed as ($\varepsilon \sim \mathcal{N}(\cdot; 0, 1)$):

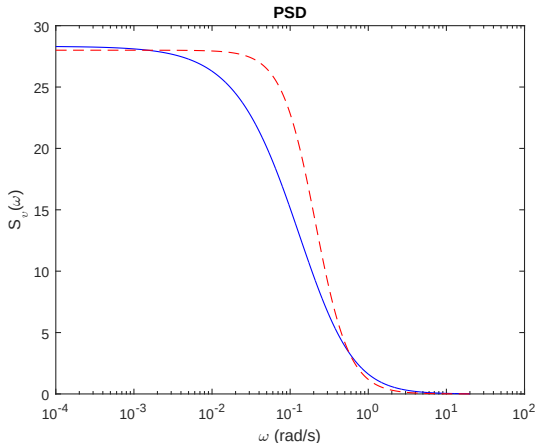
$$S_{\upsilon^{[filter]}}(\omega) = \frac{Q_f}{1 + T_f^2 \omega^2} \tag{6.14a}$$

$$R_{\upsilon^{[filter]}}(\tau) = \frac{\pi Q_f}{T_f} exp(-\frac{|\tau|}{T_f}) \tag{6.14b}$$

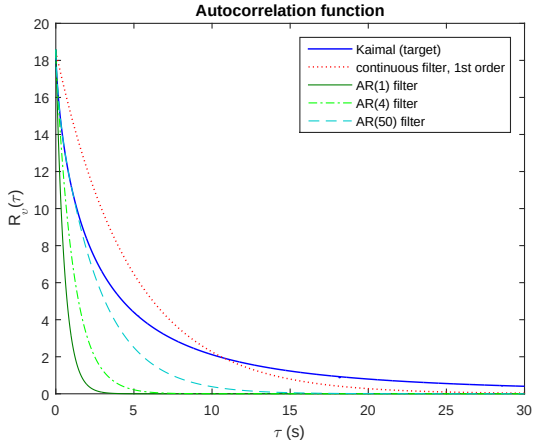
We decided here to choose Q_f and T_f so that the value of the PSD at $\omega \longrightarrow 0$ matches the Kaimal PSD, and the variance of the output process ($2\int_0^\infty S(\omega) d\omega = \frac{\pi Q_f}{T_f}$) matches the known true variance $6u_*^2$. The fit is shown in Fig. 6.4a.

Similarly, we used the Yule-Walker equations to derive several AR models, of order 1, 4 and 50. Autocorrelation function fits are shown on Fig. 6.4b.

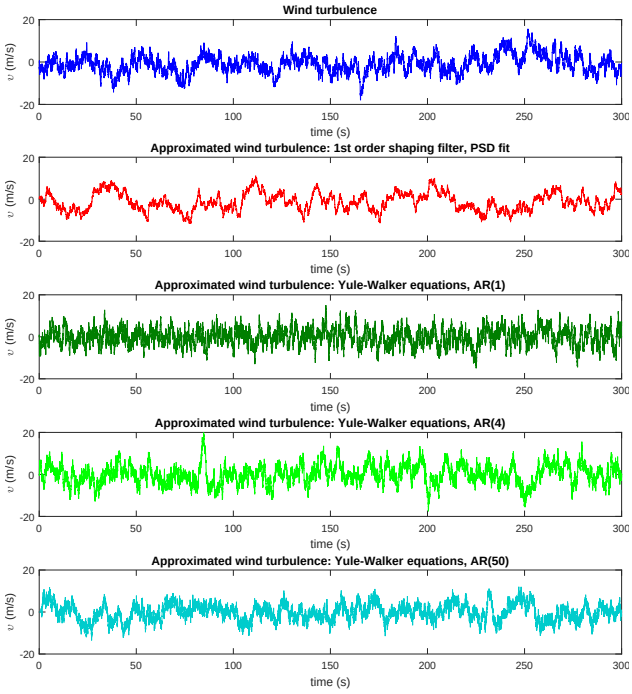
Also, in order to show the differences between time series generated using different methods, a sample path of the target excitation υ was generated using the Kaimal spectrum and the spectral representation method [Shinozuka and Deodatis, 1991], along with sample paths of the approximation processes $\upsilon^{[filter]}$, plotted on Fig. 6.4c. Those were generated by sampling independent normal RVs (ε_k) and applying the discretized version of the filter (Eq. (6.9)). One can see that the continuous filter (second plot from the top, red) overestimates the correlation between time



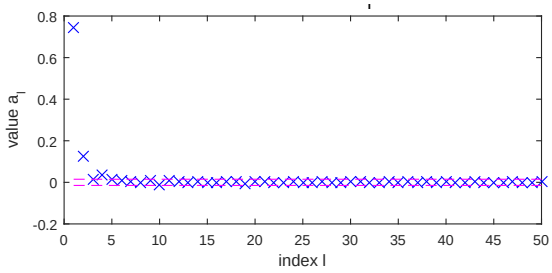
(a) PSD fit for the continuous 1st order filter



(b) Autocorrelation function fits for AR models



(c) Sample paths of v time series with different fits.



(d) AR coefficients α_l.

Figure 6.4: Presentation of several shaping filters: PSD/autocorrelation fits and sample paths.

steps, the excitation looks less ”random”, with less high frequency content. On the contrary, the AR(1) approximation is closer to a white noise, very random, which can also been seen in the autocorrelation plot since the autocorrelation for the AR(1) decays very quickly to 0. When the order of the AR model increases, the filter outputs resemble the target excitation more, and so does the autocorrelation function.

6.2.3 Implementation in the UKF

Recall that for both the continuous filter and the AR model, one ends up with a discrete filter that approximates the time correlation of the input wind excitation:

$$q_k = A_d q_{k-1} + \sigma_e \varepsilon_{k-1} \tag{6.15a}$$

$$v_k^{[filter]} = C_d q_k \tag{6.15b}$$

with $\varepsilon_k \sim \mathcal{N}(\cdot;0,1)$ a Gaussian white noise process. The excitation can then be incorporated in the UKF by defining the augmented state vector

$$z_k = \begin{Bmatrix} x_k^{dyn} \\ \theta \\ q_k \end{Bmatrix} \tag{6.16}$$

The propagation equation for this new system is:

$$x_k^{dyn} = f(x_{k-1}^{dyn}, \theta, v_{k-1}^{[filter]}) = f(x_{k-1}^{dyn}, \theta, C_d q_{k-1}) \tag{6.17a}$$

$$\theta_k = \theta_{k-1} \tag{6.17b}$$

$$q_k = A_d q_{k-1} + \sigma_e \varepsilon_{k-1} \tag{6.17c}$$

where $v_k^{[filter]}$ is the excitation term (wind speed) and $\sigma_e \varepsilon_{k-1}$ represents the process noise term, zero mean white noise, uncorrelated to the measurement noise. The measurement equation can also be written as function of the augmented state:

$$y_k = h(x_k^{dyn}, \theta_k, v_k^{[filter]}) + \eta_k = h(x_k^{dyn}, \theta_k, C_d q_k) + \eta_k \tag{6.18}$$

In this formulation, we consider additive measurement noise η_k; also, we consider that the process noise consists solely of noise due to the random excitation, i.e., there is no modeling error. These two assumptions could however be relaxed if needed. It is also important to highlight that in this formulation, no wind time series is ever simulated using the shaping filter (since it would never be able to match the actual unmeasured wind time series exciting the system); instead, we only

use the shaping filter parameters A_d, σ_e, C_d to take into account in the UKF the time correlation properties of the input excitation.

Since the state vector is augmented with the vector of intermediate states q_k, the computational complexity of the UKF ($\propto N_{sig} = 2n_z + 1$) will increase significantly if many intermediate states are added, i.e., if the shaping filter becomes very complex. For instance, an nth order continuous filter will use $n_q = n$ intermediate states, as will an AR(n) model. Thus, one should find a trade-off between a shaping filter complex enough to represent accurately the input excitation and keeping the size of the q vector small enough to keep the UKF as fast as necessary (especially for real-time monitoring). In the following sections, we show that with the use of a continuous first order filter, as computed in the numerical example previously detailed, the approach already performs well in identifying structural parameters in the case of wind excitation. Then we also briefly study the effect of increasing the accuracy of the shaping filter fit (increased AR order for instance), and we present a method to avoid large increase in computational time as an effect of the increase in the size of the augmented state vector z_k.

6.3 Results when the spectrum of the input random excitation is fully known

6.3.1 Structure of interest: linear SDOF system

A SDOF water tank is represented as a 25 m cantilever tower, with weight 445 kN, lateral stiffness 700 kN/m and damping ratio 5%. Its equation of motion is written as:

$$\ddot{x}(t) + 2\xi\omega_n\dot{x}(t) + \omega_n^2 x(t) = \frac{1}{m}\varphi(t) \tag{6.19a}$$

$$\ddot{x}(t) + 2\xi\omega_n\dot{x}(t) + \omega_n^2 x(t) = \frac{1}{m}\left(\frac{1}{2}\rho_{air}\cdot Area\cdot C\cdot(U(z)+\upsilon(t))^2\right) \tag{6.19b}$$

with $\rho_{air}, Area, C$ are the air density, area subjected to wind and drag coefficient respectively, $\omega_n = \sqrt{\frac{k}{m}}$ the fundamental frequency of the linear system and $\xi = 0.05$ its damping ratio.

A 300s ($dt = 0.02s$) wind time series is generated using the Kaimal spectrum and the spectral representation method and used as an input to generate response data from this SDOF structure (using a Runge-Kutta discretization scheme). 10% RMS noise is added to the displacement time series, which is used as measurement for the identification task, performed using the UKF. In the UKF, discretization for propagation of the dynamic states x_k^{dyn} is also performed using a Runge-Kutta scheme (function f in Eq. (6.17a)).

Two major issues should be tackled before undertaking structural identification with the UKF: 1) check identifiability of the parameter vector, as for any other system and 2) define a shaping filter that estimates the time correlation properties of the input wind excitation, since the actual excitation time series is not measured.

Identifiability, known excitation

For this linear system, identifiability of the parameter vector can be checked using the local identifiability test described in [Grewal and Glover, 1976]. Note that identifiability tests assume that the input is known, and there is no noise in the system.

Going back to the system of interest, the dynamics of the system can be written as

$$m\ddot{x}(t) + c\dot{x}(t) + kx(t) = \beta\varphi(t) \tag{6.20}$$

where, for a wind excitation, β is proportional to the drag coefficient C, a constant that is quite challenging to estimate for many structures. We then would like to estimate m, k, c and β, supposing we know the excitation $\varphi(t)$ (which is not exactly true since it is not measured, we only create an approximation of it) and a series of measurements from the structure $y_{1:N}$ (we suppose

displacement is measured). The system can be written in state-space format as:

$$\frac{d}{dt}\begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix}\begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} + \begin{bmatrix} 0 \\ \frac{\beta}{m} \end{bmatrix}\varphi(t) \qquad \left(A = \begin{bmatrix} 0 & 1 \\ -\frac{k}{m} & -\frac{c}{m} \end{bmatrix} \text{ and } B = \begin{bmatrix} 0 \\ \frac{\beta}{m} \end{bmatrix}\right) \tag{6.21a}$$

$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix}\begin{Bmatrix} x \\ \dot{x} \end{Bmatrix} \qquad \left(C = \begin{bmatrix} 1 & 0 \end{bmatrix} \text{ and } D = 0\right) \tag{6.21b}$$

The Markov parameter matrix of this system can be written as:

$$G(\theta) = \begin{bmatrix} D(\theta) \\ C(\theta)B(\theta) \\ C(\theta)A(\theta)B(\theta) \\ \vdots \\ C(\theta)A^{2n_A-1}(\theta)B(\theta) \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ \frac{b}{m} \\ -\frac{\beta c}{m^2} \\ -\frac{\beta}{m}(\frac{k}{m} - \frac{c^2}{m^2}) \end{bmatrix} \tag{6.21c}$$

and its Jacobian with respect to the parameter vector $\theta = \begin{bmatrix} m & k & c & \beta \end{bmatrix}^T$ is:

$$\frac{dG(\theta)}{d\theta} = \frac{1}{m}\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -\frac{\beta}{m} & 0 & 0 & 1 \\ \frac{2\beta c}{m^2} & 0 & -\frac{\beta}{m} & -\frac{c}{m} \\ \frac{\beta}{m}(\frac{2k}{m} - \frac{3c^2}{m^2}) & -\frac{\beta}{m} & \frac{2\beta c}{m^2} & -(\frac{k}{m} - \frac{c^2}{m^2}) \end{bmatrix} \tag{6.21d}$$

which has rank $3 < n_\theta = 4$, thus this parameter vector is non-identifiable. It is logical since writing the equations of motion using $\theta = \begin{bmatrix} m & k & c & \beta \end{bmatrix}^T$ or $\theta = \begin{bmatrix} \frac{m}{2} & \frac{k}{2} & \frac{c}{2} & \frac{\beta}{2} \end{bmatrix}^T$ would yield the same displacement time series, thus the 4 parameters of this system are non identifiable. However, if we re-parameterize the system using $\bar{k} = \frac{k}{m}$, $\bar{c} = \frac{c}{m}$ and $\phi = \frac{\beta}{m}$, then the matrices of the state-space formulation become:

$$A = \begin{bmatrix} 0 & 1 \\ -\bar{k} & -\bar{c} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \phi \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 \end{bmatrix}, \quad D = 0 \tag{6.22a}$$

Then the Markov parameter matrix and its Jacobian with respect to the parameter vector $\theta =$

$\begin{bmatrix} \phi & \bar{c} & \bar{k} \end{bmatrix}^T$ can be computed as:

$$G(\theta) = \begin{bmatrix} 0 \\ 0 \\ \phi \\ -\bar{c}\phi \\ -\bar{k}\phi + \bar{c}^2\phi \end{bmatrix}, \qquad \frac{dG(\theta)}{d\theta} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ -\bar{c} & -\phi & 0 \\ -\bar{k} + \bar{c}^2 & 2\bar{c}\phi & -\phi \end{bmatrix} \tag{6.22b}$$

The Jacobian has rank $3 = n_\theta$ if all parameters are non-zero, thus this system is identifiable. Thus for identifiabilty purposes of this system, either the mass or the drag coefficient C (and thus the constant β) should be known in advance (only the ratios $\bar{k} = \frac{k}{m}$, $\bar{c} = \frac{c}{m}$ and $\phi = \frac{\beta}{m}$ can be identified).

Performance comparison: assuming input excitation is white noise vs. shaping filter approximation

The second issue in our learning problem is that the excitation is not measured, thus one does not have access to the time series plotted in Fig. 6.5a. However, we assume that the characteristics of this excitation are known, i.e. we know that it can be represented by a Kaimal spectrum, with known coefficients $K, z_0, z, U(z)$. Thus one can follow the procedure previously described and integrate a shaping filter in the UKF to represent the time correlations properties of the excitation. The shaping filter used in this example consists of a first order continuous filter, as derived in the numerical example previously studied, and it is integrated in the UKF following the procedure explained in the preceding section.

For purposes of comparison with a generic UKF, we also tried to approximate the wind turbulence with a zero-mean Gaussian white noise $\upsilon^{[wn]}$:

$$\ddot{x}(t) + 2\xi\,\omega_n\dot{x}(t) + \omega_n^2 x(t) = \frac{1}{m}\left(\frac{1}{2}\rho_{air}\cdot Area\cdot C\cdot (U(z) + \upsilon^{[wn]})^2\right) \tag{6.23}$$

This equation can be easily discretized and the UKF can be used to recover the parameters, without making use of a shaping filter. The white noise excitation $\upsilon^{[wn]}$ (process noise in the UKF)

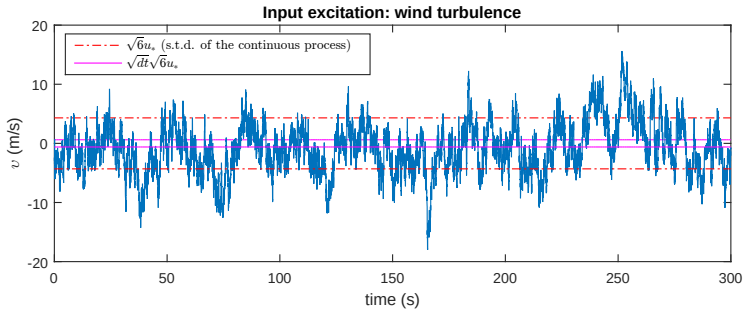
appears nonlinearly in the equations, thus one needs to use the noise augmented version of the UKF. However, this makes learning of the noise characteristics (mean, covariance) much more complex, more specifically they cannot be learnt through adaptive algorithms presented for instance in [Kontoroupi and Smyth, 2016] for the UKF and [Özkan et al., 2013] for the PF, since these algorithms assume additivity of the noise terms. One must then choose the variance Q of this process noise $\upsilon^{[wn]}$ to be used in the UKF. We know the variance of the input continuous stationary process from the Kaimal spectrum, used to generate the input data: $\sigma_\upsilon^2 = 6u_*^2$, thus one can choose to fit the variance, i.e. $Q = 6u_*^2$. For completeness we also ran another white noise filter where the value of Q is decreased ($Q = 6dt \cdot u_*^2$), which is more representative of the changes of amplitude observed in the true excitation between two consecutive time steps. Fig. 6.5a shows the actual excitation time series used to generate the data (assumed unknown when running the UKFs for structural identification) and the amplitude of the two white noises previously defined.

Convergence of the static parameters are plotted on Fig. 6.5b for the three algorithms (used on the same measurement time series $y_{1:N}$):

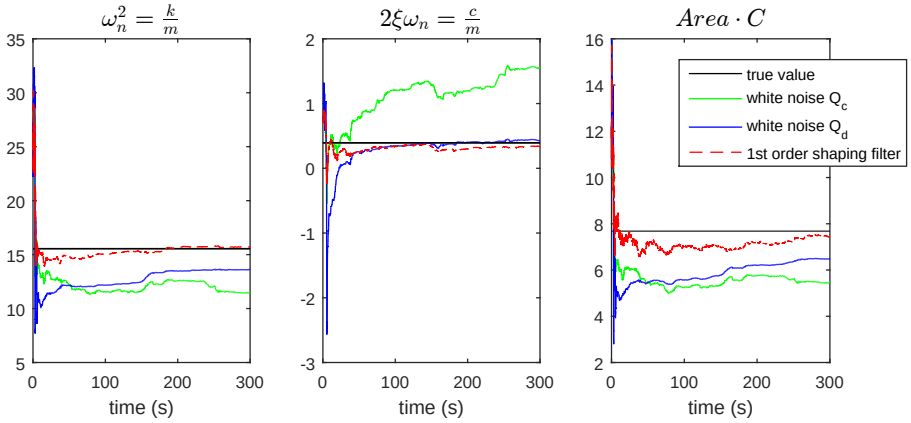
- UKF assuming that υ_k is a Gaussian white noise, of variance $6u_*^2$ (variance of the continuous process), method 1, green line on plots,
- UKF assuming that υ_k is a Gaussian white noise, of variance $dt \cdot 6u_*^2$, method 2, blue line on plots,
- UKF that uses a 1st order linear shaping filter to represent the excitation υ_k, method 3, red dashed line on plots[1].

Clearly, the UKF that uses the shaping filter performs the best in recovering the parameters. It is however slightly more computationally expensive since one state is added to the state vector (2 more sigma points). It might be possible to find a value of Q that would make the UKF with simple

[1]For this UKF, the prior mean for state $q = \upsilon^{[filter]}$ is set at 0, its prior variance is chosen as $6u_\star^2$, variance of the continuous process υ.



(a) Wind turbulence used to excite the system and simulate data from the system, generated using Kaimal's spectrum and the spectral representation method.



(b) Convergence of static parameters for linear SDOF

Figure 6.5: Performance of different UKF schemes (with or without shaping filter to take into account input correlation properties) on a SDOF linear system.

white noise perform well, however there is no easy way to learn that value, and learning it would probably increase the computational time of the UKF. The shaping filter is a way to better represent the excitation, as a non-white noise with certain characteristics fitted to the known spectrum of the actual excitation. Those results are confirmed in the following section, looking at a nonlinear SDOF.

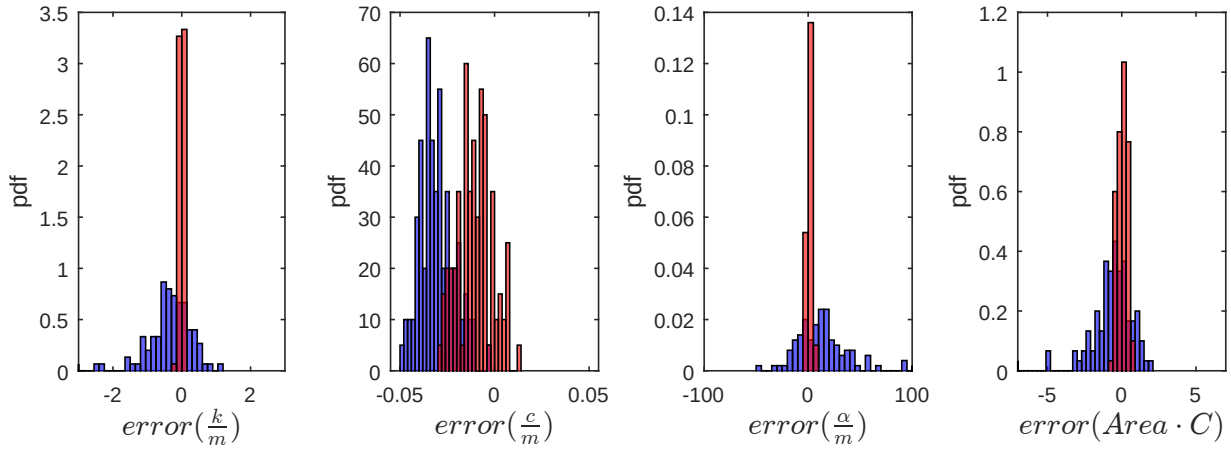


Figure 6.6: Pdf of error on identified parameters, i.e. $\theta_{true} - E[\theta|y_{1:N}]$, for a SDOF Duffing oscillator, when 1) a UKF is run assuming that the excitation is white noise (with variance $dt \cdot 6u_\star^2$, blue histogram) and 2) a shaping filter is used to aproximate the excitation (red histogram).

6.3.2 Extension to nonlinear systems: SDOF Duffing oscillator

Identifiability of parameters

A SDOF Duffing oscillator is considered here, with equation of motion:

$$\ddot{x}(t) + \frac{c}{m}\dot{x}(t) + \frac{k}{m}x(t) + \frac{\alpha}{m}x(t)^3 = \frac{\beta}{m}\varphi(t) \tag{6.24}$$

Since this equation is nonlinear, one cannot write this system as a linear state-space system and use the local identifiability method previously mentioned. Instead, we verify local identifiability of the parameters using the ORC method presented for instance in [Chatzis et al., 2015b]. In this method, the state vector is augmented with the static parameters and observability of this augmented state is studied. The method is explained for completeness in appendix E, and the Duffing oscillator example is detailed. The ORC test yields positive result on the observability of this system, thus the parameter vector $\theta = \begin{bmatrix} \frac{k}{m} & \frac{c}{m} & \frac{\alpha}{m} & \frac{\beta}{m} \end{bmatrix}^T$ is identifiable.

Performance of the shaping filter

We now compare methods 2 and 3, which performed the best on the linear system. 100 runs are performed: for each run a new excitation and response of the structure is generated, and the 2 solvers are run on this incoming noisy data (10% RMS noise added to the measurements, and the actual excitation time series is assumed unknown when running the UKF). The error between each identified parameter ($\theta = \left[\frac{k}{m} \quad \frac{c}{m} \quad \frac{\alpha}{m} \quad \frac{\beta}{m}\right]^T$) and its true value is computed for each run of the 2 solvers. Fig. 6.6 shows the histogram (approximated pdf) of this error. One can observe that our method performs much better in recovering the stiffness parameters $\frac{k}{m}$ and $\frac{\alpha}{m}$, as well as the drag coefficient term $Area \cdot C$, since the pdf of the errors is highly peaked around 0. Also, the UKF that uses white noise $\upsilon_k^{[wn]}$ shows a bias in recovering the damping parameter $\frac{c}{m}$, and is less pronounced for the method that uses the shaping filter.

6.3.3 Influence of the accuracy of the shaping filter fitting process

As previously mentioned, an important question that raises in this context is the question of the order of the shaping filter, which will govern how well this shaping filter is capable of representing the true input exciting the system. In this section we first study how the accuracy of the shaping filter design process influences learning of the structural parameters. This is easily done by fitting the autocorrelation function of the input using AR models of increasing order. First we go back to the linear SDOF system and compare performance of the UKF in recovering the identifiable parameter vector $\theta = \left[\frac{k}{m} \quad \frac{c}{m} \quad Area \cdot C\right]^T$ for several cases:

- UKF assuming that υ_k is a Gaussian white noise, of variance $6u_*^2$ (variance of the continuous process),
- UKF assuming that υ_k is a Gaussian white noise, of variance $dt6u_*^2$,
- UKF that uses a linear shaping filter to represent the excitation υ_k. We also vary the order of

the fitted AR process.

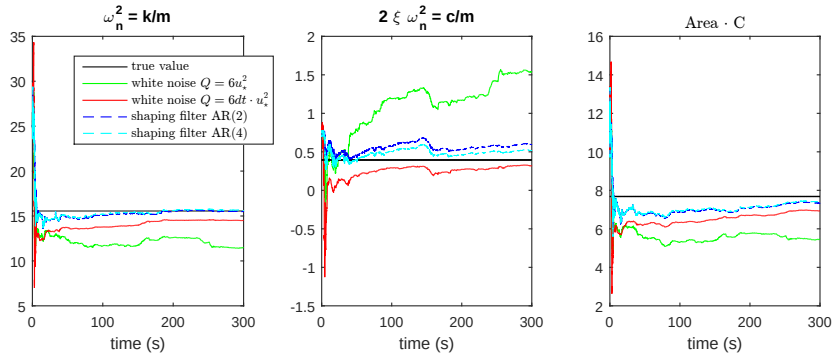
Convergence of the static parameters are plotted on Fig. 6.7a. Again, the UKF that uses the shaping filter performs the best in recovering the stiffness and drag coefficient parameters. The damping parameter is more challenging to identify, but it appears that increasing the order of the AR shaping filter increases accuracy of the identification procedure.

Those results are further confirmed by looking back at the SDOF Duffing oscillator, for which solvers 2 and 3 previously mentioned are compared. As before, 100 runs are performed and for each run, the error between each identified parameter ($\theta = \begin{bmatrix} \frac{k}{m} & \frac{c}{m} & \frac{\alpha}{m} & \frac{\beta}{m} \end{bmatrix}^T$) and its true value is computed. Fig. 6.7b shows the histogram of this error. Once again, using a shaping filter evidently helps the identification procedure: one can observe that the UKF with shaping filter performs much better in recovering the stiffness parameters $\frac{k}{m}$ and $\frac{\alpha}{m}$, as well as the drag coefficient term $Area \cdot C$, since the pdf of the errors is highly peaked around 0. Identification of the damping parameter is again more challenging, but one can observe on Fig. 6.7c that increasing the order of the AR shaping filter leads to much better approximations of the damping parameter.

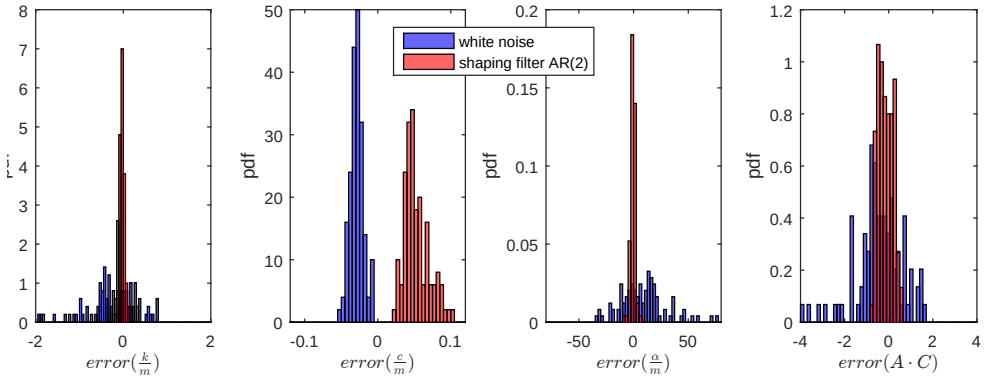
As previously mentioned, increasing the order p of the AR model fit also increases the size of the state vector, and thus the number of sigma points and the computational complexity of the overall filtering algorithm. However, it is possible to bypass this issue by making use of the marginalized UT, as presented in chapter 4, section 4.3.1. The augmented state vector is partitioned as:

$$a_{k-1} = \begin{Bmatrix} x_{k-1}^{dyn} \\ \theta_{k-1} \\ \upsilon_{k-1}^{[filter]} \end{Bmatrix} \qquad \text{and} \qquad b_{k-1} = \begin{Bmatrix} \upsilon_{k-2}^{[filter]} \\ \vdots \\ \upsilon_{k-p}^{[filter]} \end{Bmatrix} \tag{6.25}$$

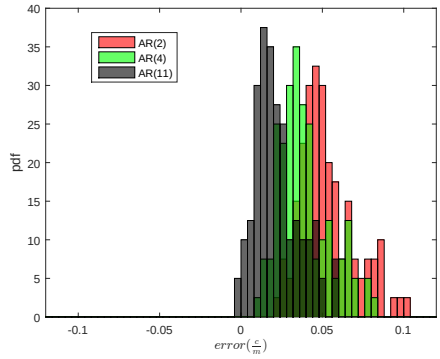
Then the propagation equation for the augmented system, which includes an AR shaping filter of



(a) Convergence of parameters for linear system.



(b) Pdf of error on identified parameters $E[\theta|y_{1:N}] - \theta_{true}$ for a sdof Duffing oscillator, comparison of assuming wind turbulence is white noise vs. use of an AR shaping filter.



(c) Pdf of error on identified damping parameter for a sdof Duffing oscillator, effect of increasing the order of the AR shaping filter.

Figure 6.7: Performance of AR shaping filters with increasing order on identification of structural parameters for SDOF linear and nonlinear systems.

order p with known coefficients $\{\alpha_{1:p}, \sigma_e\}$ can be written as:

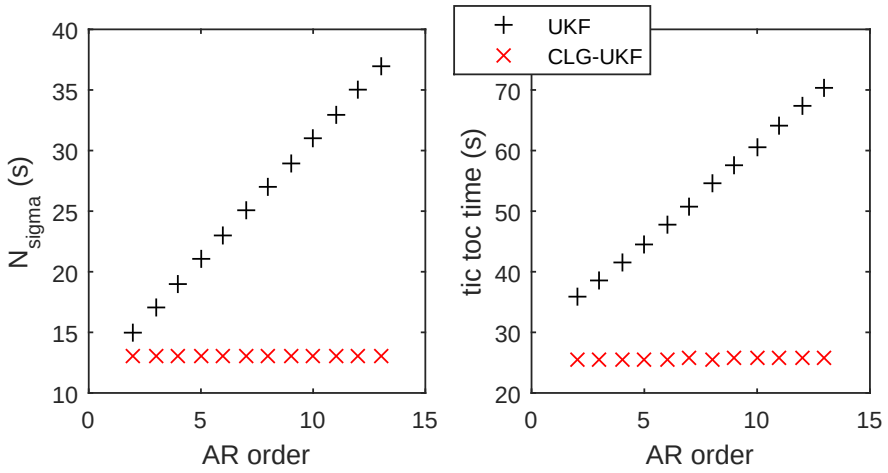
$$z_k = \begin{Bmatrix} x_k^{dyn} \\ \theta_k \\ v_k^{[filter]} \\ v_{k-1}^{[filter]} \\ v_{k-2}^{[filter]} \\ \vdots \\ v_{k-p+1}^{[filter]} \end{Bmatrix} = \underbrace{\begin{Bmatrix} f(x_{k-1}^{dyn}, \theta, v_{k-1}^{[filter]}) \\ \theta_{k-1} \\ \alpha_1 v_{k-1}^{[filter]} \\ v_{k-1}^{[filter]} \\ 0 \\ \vdots \\ 0 \end{Bmatrix}}_{\varphi(a_{k-1})} + \overbrace{\begin{bmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \alpha_2 & \alpha_3 & \cdots & \alpha_p \\ 0 & 0 & \cdots & 0 \\ 1 & 0 & \cdots & 0 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 1 & 0 \end{bmatrix}}^{G} \overbrace{\begin{Bmatrix} v_{k-2}^{[filter]} \\ v_{k-3}^{[filter]} \\ \vdots \\ v_{k-p}^{[filter]} \end{Bmatrix}}^{b_{k-1}} + \begin{Bmatrix} 0 \\ 0 \\ \sigma_e \varepsilon_{k-1} \\ 0 \\ 0 \\ \vdots \\ 0 \end{Bmatrix} \tag{6.26}$$

and the measurement equation as:

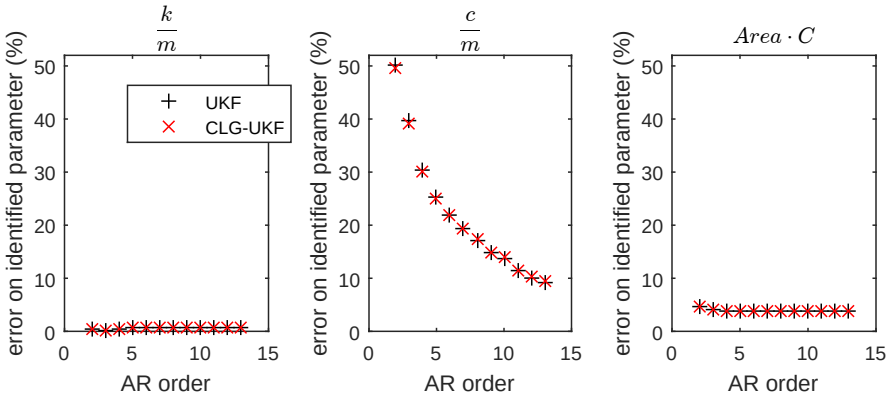
$$y_k = \underbrace{h(x_k^{dyn}, \theta_k, v_k^{[filter]})}_{\varphi(a_k)} + \overbrace{0}^{G} \cdot b_k + \eta_k \tag{6.27}$$

Both propagation and measurement equations are linear in subvector b, and the principle of marginalization can be used, i.e., integration over $p(a)$ is performed using the UT while integration over $p(b|a)$ is performed analytically, which reduces the number of sigma points and thus the number of function evaluations. Furthermore, one can see that in this specific case the Jacobians G are independent of the nonlinear subvector a, thus evaluation of G must be performed only once per time step (instead of once per sigma point). Thus in this case the marginalization procedure will quite surely lead to an overall reduction in computational time.

Fig. 6.8 shows a comparison of computational time and error on identified parameters when performing identification with a generic UKF vs. the UKF that uses this marginalization method (named CLG-UKF) on the SDOF linear system previously studied. The CLG-UKF performs almost identically to the generic UKF in terms of accuracy, i.e., increasing the order of the AR filter increases accuracy in identifying the damping parameters for both filters. The CLG-UKF however runs much faster, due to the fact that its number of sigma points does not increase as the order of



(a) Comparison of computational times for the generic UKF and the UKF that uses the UT for conditionally linear equations.



(b) Comparison of error on identified parameters for the generic UKF and the UKF that uses the UT for conditionally linear equations.

Figure 6.8: Performance of the UKF that uses the UT for conditionally linear equations.

the AR shaping filter increases.

Finally, it is good to mention here that since the equations of the UKF are nonlinear, it would be possible to use a nonlinear shaping filter to represent the excitation. This could afford access to more accurate representations of the excitations, without necessarily increasing the size of the augmented state vector q.

6.3.4 Partial conclusion.

Assuming that only some characteristics (spectrum) of the input exciting the system are known (the actual excitation time series is not measured), we are able to identify reasonably accurately structural parameters of a SDOF system, as long as the parameter vector is identifiable. This is performed using a shaping filter that takes as input a white noise and outputs a colored noise with given characteristics, which can be matched to the target spectrum. This shaping filter can be very easily integrated to the UKF formulation, but it increases its computational time. In the previous section, we have assumed that 1) the true excitation could be exactly represented by a Kaimal spectrum and 2) all the parameters of this spectrum are exactly known, which are two very strong assumptions. Assumption 1) can be relaxed by observing that when fitting the shaping filter to the spectrum/auto-correlation function, only some characteristics of the spectrum are used to obtain an approximation. When considering an AR model for instance, we only fit the auto-correlation function up to $\tau \leq p \cdot dt$, thus only knowledge of these values are required (and could be obtained through measurements), not the full spectrum/auto-correlation function. In the following section, we provide a method to relax assumption 2): i.e., we assume that the true excitation can be represented through the Kaimal spectrum, but one of its parameters is not known exactly (it is random). We then study how this uncertainty affects learning of the identifiable structural parameters.

6.4 What if the spectrum of the excitation is not fully known?

6.4.1 Can one learn both parameters representing the input excitation as well as structural parameters from the data?

We consider now the case where the spectrum of the wind turbulence excitation is not fully known. Would it be possible to use the displacement measurement time series to learn more about the excitation?

Going back to a Duffing oscillator, the continuous time augmented system used in the UKF with shaping filter can be written as:

$$\ddot{x}(t) + \frac{c}{m}\dot{x}(t) + \frac{k}{m}x(t) + \frac{\alpha}{m}x(t)^3 = \frac{\beta}{m}(U(z) + \upsilon(t))^2 \tag{6.28a}$$

$$\dot{\upsilon} = A\upsilon(t) + \sigma\varepsilon(t) \tag{6.28b}$$

where $e(t) \sim \mathcal{N}(\cdot; 0, 1)$. Parameters $U(z)$, A and σ represent (indirectly) the wind turbulence process. We now would like to estimate these parameters along with the structural parameters, i.e., the full parameter vector is $\theta = \begin{bmatrix} \frac{k}{m} & \frac{c}{m} & \frac{\alpha}{m} & \frac{\beta}{m} & U(z) & A & \sigma \end{bmatrix}^T$. To study observability/identifiability of this system, we consider $\varepsilon(t) \sim \mathcal{N}(\cdot; 0, 1)$ as a known input (which is not strictly correct since it is a random input). Using the ORC criterion previously studied, one finds out that this system is unobservable, even though we make the strong assumption that $\varepsilon(t)$ is known.

As previously stated, the mean speed value $U(z)$ is probably the easiest parameter to measure, thus we then consider it known and reduce the parameter vector to $\theta = \begin{bmatrix} \frac{k}{m} & \frac{c}{m} & \frac{\alpha}{m} & \frac{\beta}{m} & A & \sigma \end{bmatrix}^T$. In this case, the observability test yields positive results: the system is observable and parameters are identifiable, when we consider $\varepsilon(t)$ known/measured. Since $\varepsilon(t)$ is actually random, its mean and covariance are known, but its actual time series is unknown, we need to double-check this identifiabilty result, using for instance joint estimation with the UKF. We ran two slightly different algorithms: in one case we define the vector of static parameters as $\theta = \begin{bmatrix} \frac{k}{m} & \frac{c}{m} & \frac{\alpha}{m} & \frac{\beta}{m} & A & \sigma \end{bmatrix}^T$

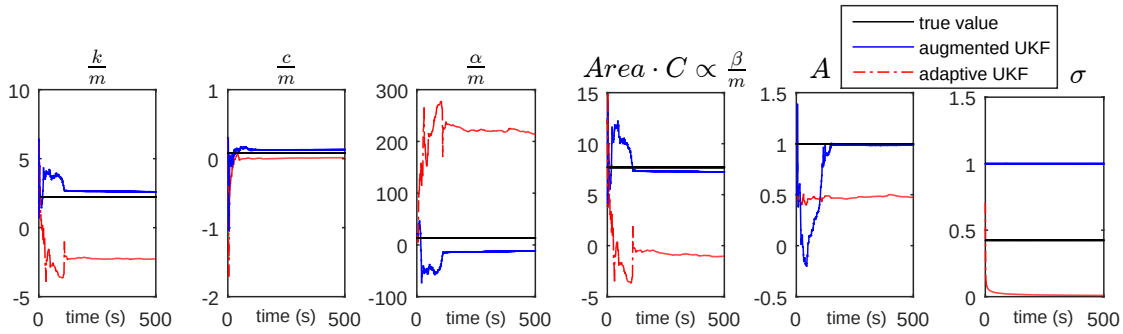


Figure 6.9: Parameter convergence when both structural and wind turbulence input parameters are unknown (blue line: the state vector is augmented with the unknown excitation parameters, red line: the adaptive UKF is used for learning of the covariance term).

(i.e., parameter σ is treated as any other parameter). In the second algorithm, we use the adaptive UKF derived in [Kontoroupi and Smyth, 2016] to learn the process noise covariance σ^2 instead. Results (Fig. 6.9) show poor results with both algorithms, which seems to imply that learning all these parameters is actually unfeasible in this case.

Note also that A and σ are only the parameters of a simple (linear) representation of the input, which is actually much more complex (Kaimal spectrum); depending on the method used to fit the shaping filter to the Kaimal spectrum, it might be difficult to learn specific parameters of the spectrum from values of A and σ.

6.4.2 Propagating uncertainties on input parameters to identified structural parameters

The idea here is that if one cannot learn exactly the parameters describing the input, then one can consider them as random and study how this uncertainty affects identification of the structural parameters, through uncertainty propagation schemes.

In uncertainty propagation (UP), one studies how uncertainty on an input parameter q, treated as a RV, affects a certain quantity of interest (QoI). One thus needs a mathematical model, which

takes as an input a sample of the RV $q^{(i)}$ and outputs a corresponding value of the QoI, $QoI^{(i)}$. Uncertainty propagation is usually performed to study the behavior of a structural system when the structural parameters/excitation parameters are random, i.e., $q \leftarrow$ structural/excitation parameters and $QoI \leftarrow$ response (displacement, acceleration) of the system.

Here instead we use UP to study how uncertainty in the input excitation parameters affect identification of the structural parameters, i.e., $q \leftarrow$ input excitation parameters and $QoI \leftarrow$ pdf or moments of $p(\theta|y_{1:N})$, obtained with the UKF or any other identification method. Our mathematical model consists then of running a UKF or any other identification method. One would need to use non-intrusive uncertainty quantification methods, such as Monte Carlo simulation (Fig. 6.10) or collocation methods, which both consist of running the mathematical model for several values of $q^{(i)}$, chosen either randomly or deterministically.

We also see that this approach shows an added complexity since the output QoI is not a single value, but a pdf $p(\theta|y_{1:N})$, or at least a few moments of this pdf. As will be explained later, we tackle this difficulty by marginalizing over the input RV to obtain the pdf or moments of the QoI (Fig. 6.11). We first explain our method using MC simulation, probably the most straightforward way to perform UP, then we show that similar results can be obtained with much less computational complexity by making use of the unscented transform.

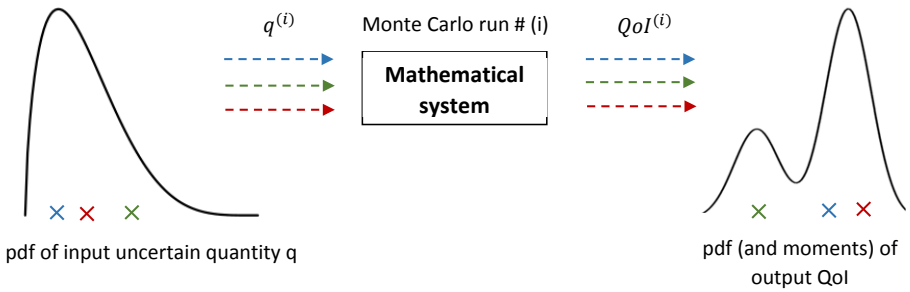


Figure 6.10: Uncertainty propagation through MC simulation.

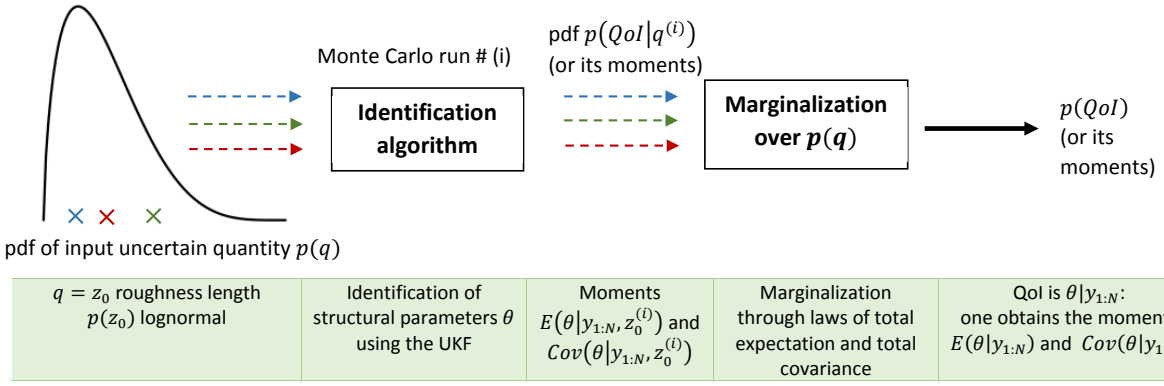


Figure 6.11: Identification of structural parameters θ, taking into account uncertainty in an input parameter q.

Uncertainty propagation through MC simulation.

The most straightforward way to perform UP is through MC simulation: sample a value $q^{(i)}$ of the random input parameter from its known pdf, run the mathematical model to obtain a value of the output quantity of interest $QoI^{(i)}$ (scheme in Fig. 6.10).

Going back to our system of interest, the mean wind speed $U(z)$ can be easily measured, so we focus our attention here on the input parameter z_0, the roughness length, which depends on the surroundings but is challenging to measure or identify exactly. We will consider this as the random input to our UP scheme, and assign it a lognormal pdf whose parameters were chosen using the table in Fig. 6.2a (the structure of interest is in an open terrain with some obstacles), as shown in Fig. 6.10. It is important to recall at this stage that we are not trying to learn this parameter z_0, but we want to study how uncertainty in this parameter affects identification of unknown structural parameters (thus the prior pdf $p(z_0)$ is known, here lognormal). One MC run then goes as follows:

1. sample $z_0^{(i)}$ from the known pdf $p(z_0)$,
2. fit a linear continuous shaping filter, as done in the previous section,
3. run the UKF to obtain the two QoIs $E\left[\theta|y_{1:N}, z_0^{(i)}\right]$ and $Cov\left(\theta|y_{1:N}, z_0^{(i)}\right)$.

Note here that the measurement time series $y_{1:N}$ is the same for each run, only the value of z_0 changes for each MC run.

Marginalization over $\mathbf{p(z_0)}$.

Using the procedure previously described, one can estimate the pdf of each of the two QoIs (see Fig. 6.12a for the pdf of QoI $E[\theta|y_{1:N}, z_0]$). However, our final goal is to estimate the posterior pdf $p(\theta|y_{1:N})$, which can be obtained by marginalizing out the random variable z_0:

$$p(\theta|y_{1:N}) = \int_{z_0} p(\theta|y_{1:N}, z_0)\, p(z_0) dz_0 \tag{6.29}$$

Looking only at the first two moments of the pdf, marginalization over $p(z_0)$ can be performed using the laws of total expectation and total variance. More explicitly, using results from the Monte Carlo simulation:

$$E[\theta|y_{1:N}] = E_{p(z_0)}[E[\theta|y_{1:N}, z_0]] \tag{6.30a}$$

$$= \underbrace{\frac{1}{N_{mc}} \sum_{i=1}^{N_{mc}} \underbrace{E\left[\theta|y_{1:N}, z_0^{(i)}\right]}_{\text{output of MC run (UKF) i}}}_{\text{averaging over all MC runs = marginalization over } p(z_0)} \tag{6.30b}$$

$$Cov(\theta|y_{1:N}) = E_{p(z_0)}[Cov(\theta|y_{1:N}, z_0)] + Cov_{p(z_0)}(E[\theta|y_{1:N}, z_0]) \tag{6.30c}$$

$$= \frac{1}{N_{mc}} \sum_{i=1}^{N_{mc}} Cov\left(\theta|y_{1:N}, z_0^{(i)}\right) + \frac{1}{N_{mc}-1} \sum_{i=1}^{N_{mc}} \left(E\left[\theta|y_{1:N}, z_0^{(i)}\right] - E[\theta|y_{1:N}]\right)\left(E\left[\theta|y_{1:N}, z_0^{(i)}\right] - E[\theta|y_{1:N}]\right)^T \tag{6.30d}$$

Results are shown in Fig. 6.12b: parameters $\frac{k}{m}$ and $Area \cdot C$ are identified with less than 5% error and variance is quite low. Damping parameters are harder to learn (about 16% error) but the mean $\pm$ standard deviation still contains the true value. Only for the nonlinear parameter $\frac{\alpha}{m}$ the mean $\pm$

	expected value (over $p(z_0)$) of the identified variances (%)	variance (over $p(z_0)$) of the identified means (%)
k/m	98.32	1.68
c/m	95.04	4.96
α/m	97.28	2.72
$Area \cdot C$	76.26	23.74

Table 6.1: Percentage of total variance $Var(\theta|y_{1:N})$ due to $E_{p(z_0)}[Var(\theta|y_{1:N}, z_0)]$ vs. $Var_{p(z_0)}(E[\theta|y_{1:N}, z_0)])$

one standard deviation does not contain the true value, and is approximated with about 8% error. This is due in part to the fact that the Duffing oscillator is not fully excited, and also because the shaping filter approximating the excitation is very simple (linear).

The formula for the variance is the sum of two terms: the first term represents the average over $p(z_0)$ of the identified variances (directly related to the process/measurement noise terms in the UKF), which is not taken into account if one looks only at the pdfs in Fig. 6.12a (thus these pdfs should be treated carefully). The second term represents the variance over $p(z_0)$ of the identified means. Table 6.1 shows the percentage of the total variance due to each of these two terms. For the first three parameters, the first term, related to the noise components of the UKF, clearly dominates. Note here that the process noise in the UKF consists of the shaping filter noise term, it is thus highly dependent on the choice of z_0; thus even though the first term dominates here, it does not mean that uncertainty on z_0 has no effect on the identification. For the fourth parameter $Area \cdot C \propto \frac{\beta}{m}$, which is directly related to the excitation (scaling factor), uncertainty around z_0 clearly affects identification through the second term of the total variance. Looking at the correlation plots (Fig. 6.13c, plot of $E[Area \cdot C|y_{1:N}, z_0]$ as a function of z_0), one can observe that these two parameters are negatively correlated, which makes sense since both have a similar effect on scaling the excitation.

As previously demonstrated in chapter 3, the Gaussianity assumption used in the UKF is actually acceptable, as long as the system parameters are identifiable and the noise terms are Gaussian. Thus one can assume in this case: $p\left(\theta|y_{1:N}, z_0^{(i)}\right) \simeq \mathcal{N}\left(\theta; \mu^{(i)}, C^{(i)}\right)$ with $\mu^{(i)}$ and $C^{(i)}$ the outputs

of UKF #i. Then the full posterior can be approximated as a mixture of Gaussians:

$$p(\theta|y_{1:N}) = \int_{z_0} p(\theta|y_{1:N}, z_0)\, p(z_0) dz_0 \tag{6.31a}$$

$$= \int_{z_0} \mathcal{N}\left(\theta; E\left[\theta|y_{1:N}, z_0^{(i)}\right], Cov\left(\theta|y_{1:N}, z_0^{(i)}\right)\right) \frac{1}{N_{mc}} \delta(z_0 - z_0^{(i)}) dz_0 \tag{6.31b}$$

$$= \frac{1}{N_{mc}} \sum_{i=1}^{N_{mc}} \mathcal{N}\left(\theta; E\left[\theta|y_{1:N}, z_0^{(i)}\right], Cov\left(\theta|y_{1:N}, z_0^{(i)}\right)\right) \tag{6.31c}$$

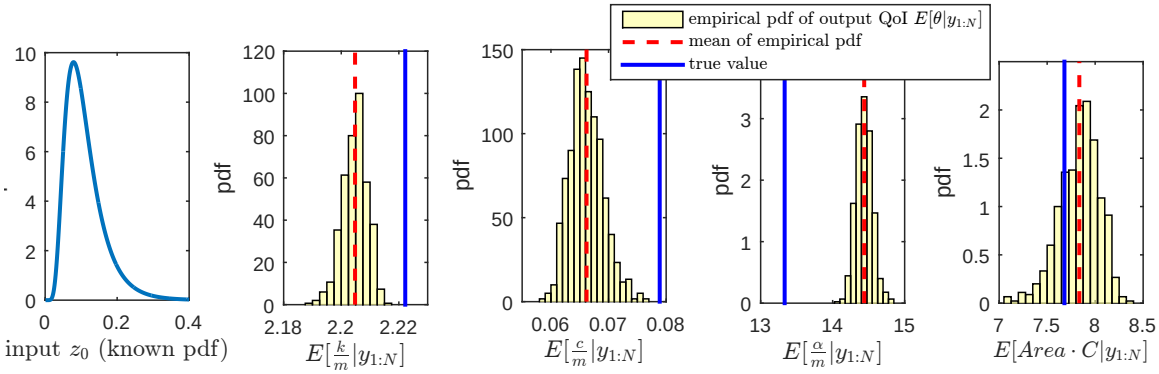
Plots of the posterior distributions (as a mixture of 600 Gaussians) are shown in Fig. 6.12c.

We thus have a way to compute the moments and approximate the distribution $p(\theta|y_{1:N})$, for one specific measurement time series $y_{1:N}$, even though we are not measuring the excitation time series and one parameter that characterizes the input excitation spectrum is uncertain. Clearly, uncertainty propagation is unavoidable for such cases where so many quantities are unknown. However, this requires many runs of the UKF (here we used 600) for the MC simulation. We show in the next section that almost similar results can be obtained in this case using only 3 runs, making use of the unscented transform.

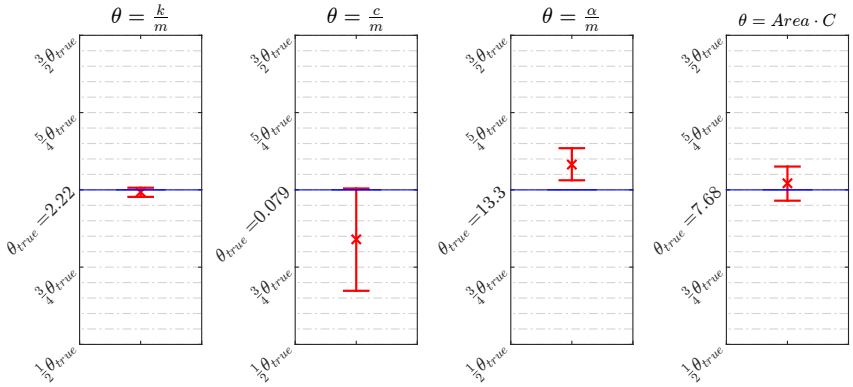
Moment propagation through the UT

The previous simulation enabled us to find the moments of the identified parameters by marginalizing over the random input parameter z_0. However it requires a large number of runs, each of them consisting of running a UKF over a relatively long time series, for the MC simulation. If one is mostly interested in the first two moments (Fig. 6.12b), then we propose here a much faster procedure, based on the unscented transform.

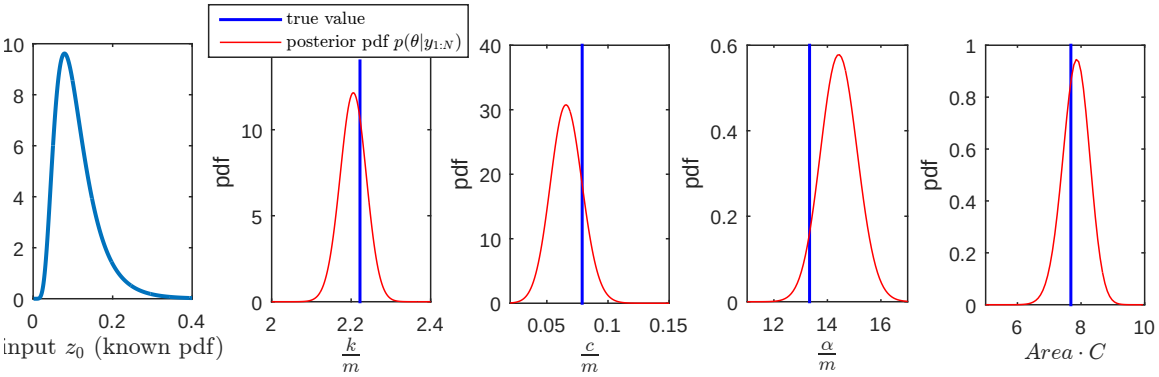
Recall that the unscented transform (UT) aims at computing the moments of an output RV $Z = g(X)$, when the moments of the input RV X are known. This concept is used in the laws of total expectation and covariance, setting $X = z_0$ and $Z = g(X) = E[\theta|y_{1:N}, z_0]$ or $Cov(\theta|y_{1:N}, z_0)$. Thus one can use the unscented transform to compute Eqs. (6.30a), (6.30c). Thus, to obtain the first and second order moments of $p(\theta|y_{1:N})$ (Eqs. (6.30a), (6.30c)) when using the UT instead of



(a) Pdfs of the input z_0 and the output QoIs $E[\theta|y_{1:N}]$. Note that for each MC run, the prior mean (starting point of the identification algorithm) was $\theta_0 = [3.27 \quad 0.30 \quad 21.4 \quad 12.0]^T$ and large diagonal covariance (not very informative prior).

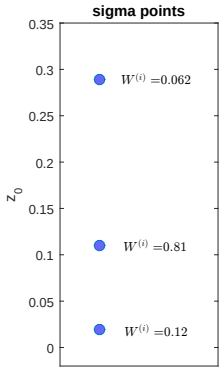


(b) Mean (x) $\pm$ one standard deviation (-) of identified parameters: $E[\theta|y_{1:N}]$ and $Std(\theta|y_{1:N})$, obtained using laws of total expectation/variance, marginalizing out the RV z_0.

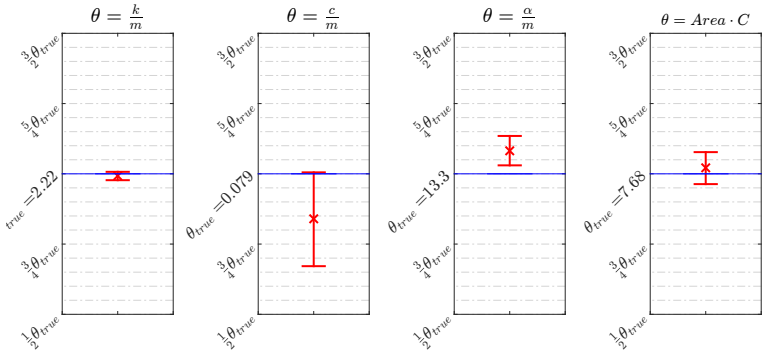


(c) Prior pdf of z_0 and posterior pdfs $p(\theta|y_{1:N})$ (as mixtures of $N_{mc} = 600$ Gaussians).

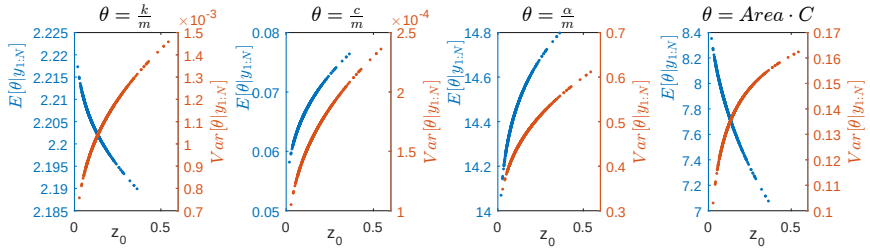
Figure 6.12: Results of Monte Carlo simulation to study propagation of uncertainty on z_0 to identification of structural parameters.



(a) Sigma points that match the first four order moments of the input RV z_0.



(b) Mean (x) $\pm$ one standard deviation (-) of identified parameters: $E\left[\theta|y_{1:N}\right]$ and $Std(\theta|y_{1:N})$, obtained using laws of total expectation/variance, marginalizing out the RV z_0 using the UT.



(c) Correlation between input z_0 and output QoIs $E\left[\theta|y_{1:N}\right]$ and $Cov\left(\theta|y_{1:N}\right)$.

Figure 6.13: Application of the unscented transform to propagate uncertainty on z_0 to identification of structural parameters.

a MC simulation, the following steps are proposed:

1. Generate the sigma points $\{\mathcal{Z}_0^{(i)}, W_{m,c}^{(i)}\}^1$ that capture the known moments of the input RV z_0,

2. Propagate them through the nonlinear function, i.e., for each sigma point, $i = 1 : N_{sig}$ do:

   - fit a shaping filter to the spectrum with parameter $\mathcal{Z}_0^{(i)}$, and
   - run a UKF to compute $E\left[\theta|y_{1:N}, \mathcal{Z}_0^{(i)}\right]$ and $Cov\left(\theta|y_{1:N}, \mathcal{Z}_0^{(i)}\right)$

3. Use the laws of total expectation and covariance to average over the sigma points as:

$$E\left[\theta|y_{1:N}\right] = E_{p(z_0)}\left[E\left[\theta|y_{1:N}, z_0\right]\right] \tag{6.32a}$$

$$= \sum_{i=1}^{N_{sig}} W_m^{(i)} E\left[\theta|y_{1:N}, \mathcal{Z}_0^{(i)}\right] \tag{6.32b}$$

$$Cov\left(\theta|y_{1:N}\right) = E_{p(z_0)}\left[Cov\left(\theta|y_{1:N}, z_0\right)\right] + Cov_{p(z_0)}\left(E\left[\theta|y_{1:N}, z_0\right]\right) \tag{6.32c}$$

$$= \sum_{i=1}^{N_{sig}} W_m^{(i)} Cov\left(\theta|y_{1:N}, \mathcal{Z}_0^{(i)}\right) + \sum_{i=1}^{N_{sig}} W_c^{(i)} \left(E\left[\theta|y_{1:N}, \mathcal{Z}_0^{(i)}\right] - E\left[\theta|y_{1:N}\right]\right)\left(E\left[\theta|y_{1:N}, \mathcal{Z}_0^{(i)}\right] - E\left[\theta|y_{1:N}\right]\right)^T \tag{6.32d}$$

Here note that the unscented transform used to average over $p(z_0)$ is completely independent from the UKF part, which also uses the unscented transform. Actually, any system identification tool could be used in place of the UKF, as long as it outputs the moments $E\left[\theta|y_{1:N}, \mathcal{Z}_0^{(i)}\right]$ and $Cov\left(\theta|y_{1:N}, \mathcal{Z}_0^{(i)}\right)$.

This method is applied on the same measurement data used previously for the MC simulation, using a fourth order sigma point set (comments on the order of accuracy are given later). Estimates of the moments of $\theta|y_{1:N}$ using this method are shown on Fig. 6.13b: they are actually very similar to the ones found with the MC simulation, however only 3 UKF runs were needed here, while 600 were used for the MC simulation.

A discussion on the order of accuracy of the UT.

At this stage, it is good to recall that the level of accuracy that can be achieved using the UT depends on two properties (see discussion in chapter 3):

[1] W_m is used for computations of means, W_c for computations of covariances, they sometimes differ, depending on the chosen sigma point set.

- the level of nonlinearity of the function g, and
- how many moments of the input RV X are matched by the sigma point set.

Going back to our simulation, the fourth order set used therein captures the skewness and kurtosis of the input RV $X = z_0$ (Fig. 6.13a shows this set of 3 sigma points, see e.g. [Julier, 1998] and appendix for details on the derivation of the set).

In our numerical experiment, the function g is very complex since it requires running a UKF. By running a UKF, for a given value of z_0, we actually make use indirectly of two functions: $g_1 : z_0 \longrightarrow E[\theta|y_{1:N}, z_0]$ and $g_2 : z_0 \longrightarrow Cov(\theta|y_{1:N}, z_0)$, i.e., one function evaluates the mean of the parameters knowing $z_0, y_{1:N}$, the other evaluates the covariance. Assessing the level of nonlinearity of these two functions is probably unfeasible, but one can get an idea by performing simulations for different values of z_0 and plot the results, for instance plot $E[\theta|y_{1:N}, z_0]$ as a function of z_0. We used our previous MC simulations to create these plots for both functions $g_{1,2}$, shown in Fig. 6.13c: it appears that $g_{1,2}$ could be approximated as a quadratic functions, for the range of z_0 of interest for our problem at least. Thus a UT that achieves second order accuracy seems reasonable in this case. These theoretical considerations explain why our UT scheme, with a fourth order set, achieves very accurate results compared to a MC simulation.

To show the importance of carefully designing the sigma point set used to perform UP, we also applied a symmetric sigma point set, which achieves 2nd order accuracy on the mean but only first order of accuracy on covariance estimation. Errors between the MC simulation and each of the two UT simulations are given in Table 6.2. First, the 4th order set consistently outputs better estimates (in better accordance with the MC simulation) than the less accurate symmetric set. Regarding parameters $\frac{k}{m}, \frac{c}{m}, \frac{\alpha}{m}$, results on the mean estimates are actually very good for both UT sets (always less than 1% error compared to MC simulation), likewise for the variance estimates. However, recall for these three parameters, the total variance is almost equal to the expected value (over $p(z_0)$) of the identified variances (outputted by the UKF), i.e. the first term in the law of total

	4th order (skewed) set	symmetric set
	error (%) with MC simulation for expected value of identified parameters	
k/m	0.0064	0.0255
c/m	-0.0856	-0.4237
α/m	-0.0399	-0.1394
$Area \cdot C$	0.0099	0.1465
	error (%) with MC simulation for standard deviation of identified parameters	
k/m	0.2965	0.4913
c/m	0.8406	1.8889
α/m	0.7731	1.7050
$Area \cdot C$	3.0058	7.8584

Table 6.2: Error between the MC simulation and each of the two UT simulations: 4th order sigma point set vs. symmetric set.

covariance. Thus in these cases, the UT is used to compute an expectation over $p(z_0)$, and thus second order accuracy is achieved with both sets (the 4th order set actually achieves 4th order accuracy on computation of expectations). However, for parameter $Area \cdot C$, the second term in the law of total covariance, (i.e., the variance (over $p(z_0)$) of the identified means) has a non negligible effect on the total covariance. For this term, only first order accuracy is achieved with a symmetric set, which explains the larger error (8% error on the standard deviation) observed for this set on parameter $Area \cdot C$. The more accurate 4th order set gives only 3% error on the standard deviation of this parameter.

The UT seems to be a good alternative to MC simulations for uncertainty propagation, since it requires fewer runs of the mathematical model. It was already used for instance in [Rocco Sanseverino and Ramirez-Marquez, 2014] for UP in reliability analysis. Here however we add a level of complexity since our QoI is now a pdf, which necessitates the use of a marginalization scheme (through laws of total expectation/covariance) to marginalize over the input parameter. Furthermore in this dissertation we have strongly insisted on the order of accuracy of the UT, which in many cases will strongly affect the results of an uncertainty propagation scheme, especially regarding estimation of variances. For our numerical simulations, we have given both theoretical

and numerical proof that the UT, with a 4th order set, will achieve very good accuracy. Note however that this level of accuracy might not be easily achievable for any scenario (i.e. effect of another input parameter than z_0 or even different input process): one might need a more accurate sigma point set, depending on the effect of the input random parameter q on the moments of $\theta|y_{1:N}, q$ (nonlinear effects). This can be studied as previously described: either comparison with MC results or plots of $Z = g(X)$ to understand the level of nonlinearity of the function and thus choose an accurate set of sigma points. One could also run the simulation with at least two different sigma point sets (as previously shown in the example) to check convergence of the results, especially the variance estimates.

6.5 Concluding remarks

In this final chapter we have presented a method to perform identification of structural parameters on nonlinear SDOF systems subjected to wind excitation, without having access to measurements from the excitation itself during monitoring. The method is based on the introduction in the UKF of a shaping filter, designed to capture the main correlation properties of the wind excitation, modeled as a stochastic process with *known* PSD/autocorrelation function. Since the shaping filter is only an approximation of the actual input, some modeling error is necessarily introduced in the system equations, and one can wonder how to account for this uncertainty in the posterior pdfs - maybe by performing several runs with various shaping filter fits. More importantly, we are now working on extending this method to multi-degree-of-freedom systems, which would require taking into consideration both spatial and time correlation of the input stationary process. Finally, it is important to note that this method could be applied to other types of stationary stochastic excitations (Pierson-Moskowitz spectrum for ocean waves for instance). Such developments would render filtering methods for damage detection applicable to a wide variety of systems.

Finally, we also acknowledge that it would not be reasonable to assume that the spectrum of the input excitation is exactly known. We thus present a method, based on uncertainty propagation tools, to study the effect of having uncertainties in the input parameters on identification of structural parameters. We also show that the unscented transform can be used to perform uncertainty propagation and marginalization over the random input, at a much lower computational cost than Monte Carlo simulations. Again, this method is very generic and could be applied to any input parameter that is uncertain (but whose pdf or moments are known) and cannot be learnt due to identifiability issues. When so many parameters/variables of a system are unknown (recall that we are not measuring the input excitation), making use of accurate probabilistic methods (Bayesian filtering) and uncertainty propagation tools is imperative.

This page is intentionally left blank.

Chapter 7

Conclusions

This dissertation has explored several topics related to the use of online Bayesian filtering algorithms for damage detection and characterization in structural systems.

As detailed in the second part of this thesis, Bayesian inference methods are very attractive for damage identification and characterization due to their ability to take into account uncertainties in the system and measurements. We have presented a method to take into account stochastic input excitations which cannot be measured during monitoring, by introducing in the filtering equations knowledge of the spectral properties characterizing the input process. We have also demonstrated that these online Bayesian filtering methods can also handle ill-conditioned problems, where not all parameters can be learnt from the available noisy data, a problem which will surely arise when considering large dimensional nonlinear systems.

A major challenge regarding these online Bayesian filtering algorithms lies in achieving good accuracy for large dimensional, nonlinear, potentially non-Gaussian systems. In the first part of this dissertation, both particle and nonlinear Kalman filtering techniques were reviewed in depth, focusing on their advantages and drawbacks with respect to damage detection in nonlinear systems. One possible way to increase accuracy and efficiency of online Bayesian algorithms for large dimensional systems is to build upon nonlinear Kalman filtering techniques, which are very

computationally efficient, and relax their Gaussianity assumption. We thus show how one can couple non-Gaussian filters and higher order moment propagation schemes to enhance the capabilities of nonlinear Kalman filters to different underlying probability distributions. On the other hand, particle filtering algorithms are sampling-weighting schemes which are theoretically accurate, i.e., lead to the true posterior pdf, for any nonlinear non-Gaussian system. However, they suffer from the so-called curse of dimensionality, and their computational complexity thus increases exponentially with the dimension of the system. An algorithmic enhancement based on the concept of Rao-Blackwellisation is used to mitigate this issue, yielding an algorithm which is particularly well-suited for damage detection purposes in structural systems with localized nonlinearities. It can be noted here that the concept of a curse of dimensionality is by no means restricted to particle filtering: it refers to several phenomena which arise when dealing with large dimensional spaces (optimization schemes, sampling...). In [Olivier and Smyth, 2016] for instance, a stochastic process simulation method based on the Karhunen-Loève decomposition is studied, which requires sampling from a multivariate Kernel density estimate. Again, it can be shown that using a dimension reduction approach in this context can enhance performance of the method.

Even though this dissertation has focused on applications to structural systems, these methods can potentially find a wider variety of applications including, but not restricted to, control and diagnostic of mechanical systems, state estimation of chemical processes, traffic or pollution spread, or improved modeling of biomechanical systems.

The work presented in this dissertation also raises questions which could provide a base for future work. Few points are already mentioned in the core of the thesis, such as studying applicability of the so-called *block particle filter* for parameter estimation tasks, or develop a multivariate version of the method presented in chapter 6 to take into account unmeasured inputs in the filtering procedure. Another topic of interest for SHM which has received limited attention up to now is the prognosis step of the SHM process. When using model-based methods such as the ones

presented in this dissertation, a model of the damaged structure can be constructed to study future behavior. The prediction task would most likely involve uncertainty propagation procedures, which are known to be quite cumbersome for large dimensional parameter spaces. Making use of the unscented transform, as mentioned in the last section of this dissertation, possibly associated with surrogate models, could potentially be beneficial for these tasks. Finally, another topic which is receiving a lot of attention lately, for instance in the finite element field, is the notion of *model class selection*, or *assessment*, see e.g. [Beck and Yuen, 2004, Muto and Beck, 2008], where one tries to select the ”best” representation of the system among a set of given models, where the term ”best” should incorporate both the notion of accurately fitting available data and allowing for prediction of future unseen data (i.e., avoid over-fitting). It has been shown that this model class selection principle can be implemented using the UKF [Kontoroupi and Smyth, 2017], thus it could be interesting to integrate this idea with the UKF for finite element models presented in chapter 4 of this dissertation, which would enable assessment of different finite element models (geometries, boundary conditions and so on) based on available measurements. This type of methods would certainly find applications in both monitoring and improved modeling in various fields of engineering and science.

This page is intentionally left blank.

Bibliography

Jaison Thomas Ambadan and Youmin Tang. Sigma-point Kalman filter data assimilation methods for strongly nonlinear systems. *Journal of the Atmospheric Sciences*, 66(2):261 – 285, 2009. doi:10.1175/2008JAS2681.1.

American Society of Civil Engineers. 2017 Infrastructure Report Card. http://www.infrastructurereportcard.org, 2017. Accessed: 2017-03-27.

Alfredo H-S. Ang and Wilson H. Tang. *Probability Concepts in Engineering: Emphasis on Applications to Civil and Environmental Engineering*. John Wiley & Sons, Inc, 2nd edition, 2007.

Rodrigo Astroza, Hamed Ebrahimian, and Joel P. Conte. Material parameter identification in distributed plasticity FE models of frame-type structures using nonlinear stochastic filtering. *Journal of Engineering Mechanics*, 141(5): 04014149, 2015. doi:10.1061/(ASCE)EM.1943-7889.0000851.

Luciana Balsamo, Raimondo Betti, and H. Beigi. A structural health monitoring strategy using cepstral features. *Journal of Sound and Vibration*, 333(19):4526 – 4542, 2014. doi:10.1016/j.jsv.2014.04.062.

R. J. Barthelmie, J. P. Palutikof, and T. D. Davies. Estimation of sector roughness lengths and the effect on prediction of the vertical wind speed profile. *Boundary-Layer Meteorology*, 66(1):19–47, 1993. doi:10.1007/BF00705458.

Klaus Jürgen Bathe and Arthur P. Cimento. Some practical procedures for the solution of nonlinear finite element equations. *Computer Methods in Applied Mechanics and Engineering*, 22(1):59 – 85, 1980. doi:10.1016/0045-7825(80)90051-1.

James L. Beck and Siu-Kiu Au. Bayesian updating of structural models and reliability using Markov chain Monte Carlo simulation. *Journal of Engineering Mechanics*, 128(4):380 – 391, 2002. doi:10.1061/(ASCE)0733-9399(2002)128:4(380).

James L. Beck and Ka-Veng Yuen. Model selection using response measurements: Bayesian probabilistic approach. *Journal of Engineering Mechanics*, 130(2):192 – 203, 2004. doi:10.1061/(ASCE)0733-9399(2004)130:2(192).

James L. Beck and Konstantin M. Zuev. Asymptotically Independent Markov Sampling: A new MCMC scheme for Bayesian inference. *International Journal for Uncertainty Quantification*, 3(5):445 – 474, 2013. doi:10.1615/Int.J.UncertaintyQuantification.2012004713.

Giuseppina Bellu, Maria Pia Saccomani, Stefania Audoly, and Leontina D'Angiò. DAISY: A new software tool to test global identifiability of biological and physiological systems. *Computer Methods and Programs in Biomedicine*, 88(1):52 – 61, 2007. doi:10.1016/j.cmpb.2007.07.002.

Thomas Bengtsson, Peter Bickel, and Bo Li. Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems. In *Probability and Statistics: Essays in Honor of David A. Freedman*, volume 2, pages 316 – 334. Institute of Mathematical Statistics, 2008. doi:10.1214/193940307000000518.

Luke Bornn, Charles R. Farrar, and Gyuhae Park. Damage detection in initially nonlinear systems. *International Journal of Engineering Science*, 48:909 – 920, 2010. doi:10.1016/j.ijengsci.2010.05.011.

Pierre Brasseur and Jacques Verron. The SEEK filter method for data assimilation in oceanography: a synthesis. *Ocean Dynamics*, 56(5):650 – 661, 2006. doi:10.1007/s10236-006-0080-3.

Rune Brincker, Lingmi Zhang, and Palle Andersen. Modal identification of output-only systems using frequency domain domain decomposition. *Smart Materials and Structures*, 10(3):441–445, 2001. doi:10.1088/0964-1726/10/3/303.

Olivier Brun, Vincent Teuliere, and Jean-Marie Garcia. Parallel particle filtering. *Journal of Parallel and Distributed Computing*, 62(7):1186 – 1202, 2002. doi:10.1006/jpdc.2002.1843.

P. Bui Quang, C. Musso, and F. Le Gland. An insight into the issue of dimensionality in particle filtering. In *Proceedings of the 13th Conference on Information Fusion (FUSION 2010)*, pages 1 – 8, 2010. doi:10.1109/ICIF.2010.5712050.

Olivier Cappé, Simon J. Godsill, and Eric Moulines. An overview of existing methods and recent advances in sequential Monte Carlo. *Proceedings of the IEEE*, 95:899 – 924, 2007. doi:10.1109/JPROC.2007.893250.

Carlos M. Carvalho, Michael S. Johannes, Hedibert F. Lopes, and Nicholas G. Polson. Particle learning and smoothing. *Statistical Science*, 25(1):88 – 106, 2010. doi:10.1214/10-STS325.

Guobin Chang. Loosely coupled ins/gps integration with constant lever arm using marginal unscented kalman filter. *Journal of Navigation*, 67(3):419 – 436, 2014a. doi:10.1017/S0373463313000775.

Guobin Chang. Marginal unscented Kalman filter for cross-correlated process and observation noise at the same epoch. *IET Radar, Sonar and Navigation*, 8(1):54 – 64, 2014b. doi:10.1049/iet-rsn.2013.0095.

Eleni N. Chatzi and Andrew W. Smyth. The unscented Kalman filter and particle filter methods for nonlinear structural system identification with non-collocated heterogeneous sensing. *Structural Control and Health Monitoring*, 16: 99 – 123, 2009. doi:10.1002/stc.290.

Eleni N. Chatzi and Andrew W. Smyth. Particle filter scheme with mutation for the estimation of time-invariant parameters in structural health monitoring applications. *Structural Control and Health Monitoring*, 20(7):1081 – 1095, 2013. doi:10.1002/stc.1520.

Manolis N. Chatzis, Eleni N. Chatzi, and Andrew W. Smyth. An experimental validation of time domain system identification methods with fusion of heterogeneous data. *Earthquake Engineering & Structural Dynamics*, 44(4): 523 – 547, 2015a. doi:10.1002/eqe.2528.

Manolis N. Chatzis, Eleni N. Chatzi, and Andrew W. Smyth. On the observability and identifiability of nonlinear structural and mechanical systems. *Structural Control and Health Monitoring*, 22(3):574–593, 2015b. doi:10.1002/stc.1690.

Jianye Ching and Yichu Chen. Transitional Markov chain Monte Carlo method for Bayesian model updating, model class selection, and model averaging. *Journal of Engineering Mechanics*, 133(7):816 – 832, 2007. doi:10.1061/(ASCE)0733-9399(2007)133:7(816).

Oana-Teodora Chis, Julio R. Banga, and Eva Balsa-Canto. Structural identifiability of systems biology models: A critical comparison of methods. *PLOS ONE*, 6(11):e27755, 2011. doi:10.1371/journal.pone.0027755.

Charles K. Chui and Guanrong Chen. *Kalman Filtering - with Real Time Applications*. Springer-Verlag Berlin Heidelberg, 4 edition, 2009. doi:10.1007/978-3-540-87849-0.

Dan Crisan and Arnaud Doucet. A survey of convergence results on particle filtering methods for practitioners. *IEEE Transactions on Signal Processing*, 50(3):736 – 746, 2002. doi:10.1109/78.984773.

George Deodatis. Simulation of ergodic multivariate stochastic processes. *Journal of Engineering Mechanics*, 122(8): 778 – 787, 1996. doi:10.1061/(ASCE)0733-9399(1996)122:8(778).

Arnaud Doucet and Adam M. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. In *The Oxford Handbook of Nonlinear Filtering*, pages 656 – 704. Oxford University Press, 2011.

Arnaud Doucet, Nando de Freitas, Kevin P. Murphy, and Stuart J. Russell. Rao-Blackwellised particle filtering for dynamic bayesian networks. In *Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI '00)*, pages 176 – 183. Morgan Kaufmann Publishers Inc., 2000.

Arnaud Doucet, Neil J. Gordon, and Vikram Krishnamurthy. Particle filters for state estimation of jump Markov linear systems. *IEEE Transactions on Signal Processing*, 49(3):613 – 624, 2001. doi:10.1109/78.905890.

Saeed Eftekhar Azam, Eleni Chatzi, Costas Papadimitriou, and Andrew Smyth. Experimental validation of the dual Kalman filter for online and real-time state and input estimation. In *Model Validation and Uncertainty Quantification, Volume 3: Proceedings of the 33rd IMAC, A Conference and Exposition on Structural Dynamics*, pages 1–13. Springer International Publishing, 2015. doi:10.1007/978-3-319-15224-0_1.

Charles R. Farrar and Keith Worden. An introduction to structural health monitoring. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 365(1851):303 – 315, 2007. doi:10.1098/rsta.2006.1928.

Brian F. Farrell and Petros J. Ioannou. State estimation using a reduced-order Kalman filter. *Journal of the Atmospheric Sciences*, 58(23):3666 – 3680, 2001.

Friedrich Faubel, John McDonough, and Dietrich Klakow. The split and merge unscented gaussian mixture filter. *IEEE Signal Processing Letters*, 16(9):786 – 789, 2009. doi:10.1109/LSP.2009.2024859.

Jacob Fish and Ted Belytschko. *A first course in finite elements*. John Wiley & Sons Ltd, 2007. ISBN 978-0-470-03580-1.

Mike Fisher. Assimilation techniques (5): Approximate Kalman filters and singular vectors, 2002.

S. Gillijns, O. Barrero Mendoza, J. Chandrasekar, B.L.R. De Moor, D.S. Bernstein, and A. Ridley. What is the ensemble Kalman filter and how well does it work? In *Proceedings of the 2006 IEEE American Control Conference*, pages 4448 – 4453, 2006. doi:10.1109/ACC.2006.1657419.

Graciela González-Farías, J. Armando Domínguez-Molina, and Arjun K. Gupta. The closed skew-normal distribution. In *Skew-elliptical distributions and their applications: a journey beyond normality*, chapter 2. Chapman & Hall/CRC, Boca Raton, 2004.

P. L. Green and K. Worden. Bayesian and Markov chain Monte Carlo methods for identifying nonlinear systems in the presence of uncertainty. *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 373:20140405, 2015. doi:10.1098/rsta.2014.0405.

Peter L. Green. Bayesian system identification of a nonlinear dynamical system using a novel variant of Simulated Annealing. *Mechanical Systems and Signal Processing*, 52-53(1):133 – 146, 2015. doi:10.1016/j.ymssp.2014.07.010.

M. Grewal and K. Glover. Identifiability of linear and nonlinear dynamical systems. *IEEE Transactions on Automatic Control*, 21(6):833–837, 1976. doi:10.1109/TAC.1976.1101375.

F. Gustafsson and P. Hriljac. Particle filters for system identification with application to chaos prediction. In *Proceedings of the 13th IFAC Symposium on System Identification*, 2003.

Fredrik Gustafsson. Particle filter theory and practice with positioning applications. *IEEE Aerospace and Electronic Systems Magazine*, 25(7):53 – 82, 2010. doi:10.1109/MAES.2010.5546308.

Fredrik Gustafsson and Gustaf Hendeby. Some relations between extended and unscented Kalman filters. *IEEE Transactions on Signal Processing*, 60(2):545 – 555, 2012. doi:10.1109/TSP.2011.2172431.

Thomas J. R. Hughes. *Finite element method - linear static and dynamic finite element analysis*. Dover Publications, 2000. ISBN 978-0-486-41181-1.

Jinwoo Jang and Andrew W. Smyth. Model updating of a full-scale FE model with nonlinear constraint equations and sensitivity-based cluster analysis for updating parameters. *Mechanical Systems and Signal Processing*, 83:337 – 355, 2017. doi:10.1016/j.ymssp.2016.06.018.

Simon Julier and Jeffrey K. Uhlmann. A general method for approximating nonlinear transformations of probability distributions. Technical report, Robotics Research Group, Dept. of Engineering Science, University of Oxford, 1996.

Simon J. Julier. A skewed approach to filtering. In *Proceedings of the SPIE Conference on Signal and Data Processing of Small Targets 1998*, volume 3373, pages 271 – 282, 1998. doi:10.1117/12.324626.

Simon J. Julier. The scaled unscented transformation. In *Proceedings of the 2002 IEEE American Control Conference*, volume 6, pages 4555 – 4559, 2002. doi:10.1109/ACC.2002.1025369.

Simon J. Julier. The spherical simplex unscented transformation. In *Proceedings of the 2003 IEEE American Control Conference*, volume 3, pages 2430 – 2434, 2003. doi:10.1109/ACC.2003.1243439.

Simon J. Julier and Jeffrey K. Uhlmann. A consistent, debiased method for converting between polar and cartesian coordinate systems. In *Proceedings of SPIE 3086, Acquisition, Tracking, and Pointing XI*, volume 3086, pages 110 – 121, 1997. doi:10.1117/12.277178.

Simon J. Julier and Jeffrey K. Uhlmann. Reduced sigma point filters for the propagation of means and covariances through nonlinear transformations. In *Proceedings of the 2002 IEEE American Control Conference*, volume 2, pages 887 – 892, 2002. doi:10.1109/ACC.2002.1023128.

L. S. Katafygiotis and J. L. Beck. Updating models and their uncertainties. II: model identifiability. *Journal of Engineering Mechanics*, 124(4):463–467, 1998. doi:10.1061/(ASCE)0733-9399(1998)124:4(463).

Nam-Ho Kim. *Introduction to nonlinear finite element analysis*. Springer, 2015. ISBN 978-1-4419-1745-4.

Thaleia Kontoroupi and Andrew W. Smyth. Online noise identification for joint state and parameter estimation of nonlinear systems. *ASCE-ASME Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*,

2(3):B4015006, 2016. doi:10.1061/AJRUA6.0000839.

Thaleia Kontoroupi and Andrew W. Smyth. Online Bayesian model assessment using nonlinear filters. *Structural Control and Health Monitoring*, 24(3):e1880, 2017. doi:10.1002/stc.1880.

Fredrik Lindsten. An efficient stochastic approximation EM algorithm using conditional particle filters. In *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6274 – 6278, 2013. doi:10.1109/ICASSP.2013.6638872.

Jiang Liu, Yujin Wang, and Ju Zhang. A linear extension of unscented kalman filter to higher-order moment-matching. In *Proceedings of the 53rd IEEE Conference on Decision and Control*, pages 5021 – 5026, 2014.

Ming Liu and Guobin Chang. Gravity matching aided inertial navigation technique based on marginal robust unscented Kalman filter. *Mathematical Problems in Engineering*, 2015, 2015. doi:10.1155/2015/592480.

Xiaodong Luo, Irene M. Moroz, and Ibrahim Hoteit. Scaled unscented transform gaussian sum filter: Theory and application. *Physica D: Nonlinear Phenomena*, 239(10):684–701, 2010. doi:10.1016/j.physd.2010.01.022.

K. V. Mardia. Measures of multivariate skewness and kurtosis with applications. *Biometrika*, 57(3):519 – 530, 1970.

M. Masoumnezhad, A. Jamali, and N. Nariman-zadeh. A robust unscented Kalman filter for nonlinear dynamical systems with colored noise. *Jounal of Automation and Control Engineering*, 2(3):310 – 315, September 2014. doi:10.12720/joace.2.3.310-315.

Peter S. Maybeck. *Stochastic processes and linear dynamic system models*, volume 1, chapter 4, pages 133 – 202. Academic Press, New York, 1979.

Philippe Moireau and Dominique Chapelle. Reduced-order unscented Kalman filtering with application to parameter identification in large-dimensional systems. *ESAIM: Control, Optimisation and Calculus of Variations*, 17(2):380 – 405, 2011. doi:10.1051/cocv/2010006.

Mark R. Morelande and Bill Moran. An unscented transformation for conditionally linear models. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP '07)*, volume 3, pages 1417 – 1420, 2007. doi:10.1109/ICASSP.2007.367112.

Virginia Mosquera, Andrew W. Smyth, and Raimondo Betti. Rapid evaluation and damage assessment of instrumented highway bridges. *Earthquake Engineering & Structural Dynamics*, 41(4):755–774, 2012. doi:10.1002/eqe.1155.

Suparno Mukhopadhyay, Hilmi Luş, and Raimondo Betti. Modal parameter based structural identification using input–output data: Minimal instrumentation and global identifiability issues. *Mechanical Systems and Signal Processing*, 45(2):283 – 301, 2014. doi:10.1016/j.ymssp.2013.11.005.

Matthew Muto and James L. Beck. Bayesian updating and model class selection for hysteretic structural models using stochastic simulation. *Journal of Vibration and Control*, 14(1-2):7 – 34, 2008. doi:10.1177/1077546307079400.

K. Krishnan Nair, Anne S. Kiremidjian, and Kincho H. Law. Time series-based damage detection and localization algorithm with application to the ASCE benchmark structure. *Journal of Sound and Vibration*, 291(1-2):349 – 368, 2006. doi:10.1016/j.jsv.2005.06.016.

H.A. Nasrellah and C.S. Manohar. A particle filtering approach for structural system identification in vehicle–structure interaction problems. *Journal of Sound and Vibration*, 329(9):1289 – 1309, 2010. doi:10.1016/j.jsv.2009.10.041.

Philippe Naveau, Marc G. Genton, and Xilin Shen. A skewed Kalman filter. *Journal of Multivariate Analysis*, 94(2): 382 – 400, 2005. doi:10.1016/j.jmva.2004.06.002.

Audrey Olivier and Andrew W. Smyth. Trade offs between statistical agreement and data reproduction in the generation of synthetic ground motions. *Probabilistic Engineering Mechanics*, 43:36 – 49, 2016. doi:10.1016/j.probengmech.2015.10.009.

Audrey Olivier and Andrew W. Smyth. On the performance of on-line parameter estimation algorithms in systems with various identifiability properties. *Frontiers in Built Environment*, 3(14), 2017a. doi:10.3389/fbuil.2017.00014.

Audrey Olivier and Andrew W. Smyth. Review of nonlinear filtering for SHM with an exploration of novel higher order kalman filtering algorithms for uncertainty quantification. *Journal of Engineering Mechanics*, 2017b. accepted for publication.

Audrey Olivier and Andrew W. Smyth. Particle filtering and marginalization for parameter identification in structural systems. *Structural Control and Health Monitoring*, 24(3):e1874, 2017c. doi:10.1002/stc.1874.

Audrey Olivier and Andrew W. Smyth. Damage detection through nonlinear filtering for systems subjected to unmeasured wind excitation whose spectral properties are known but uncertain. *Mechanical Systems and Signal*

Processing, under review.

Emre Özkan, Václav Šmídl, Saikat Saha, Christian Lundquist, and Fredrik Gustafsson. Marginalized adaptive particle filtering for nonlinear models with unknown time-varying noise parameters. *Automatica*, 49(6):1566 – 1575, 2013. doi:10.1016/j.automatica.2013.02.046.

Lauren E. Padilla and C.W. Rowley. An adaptive-covariance-rank algorithm for the unscented Kalman filter. In *Proceedings of the 49th IEEE Conference on Decision and Control (CDC)*, pages 1324 – 1329, 2010. doi:10.1109/CDC.2010.5717549.

Per-Olof Persson and Gilbert Strang. A simple mesh generator in MATLAB. *SIAM Review*, 46:329 – 345, 2004. doi:10.1137/S0036144503429121.

Ksenia Ponomareva, Paresh Date, and Zidong Wang. A new unscented Kalman filter with higher order moment-matching. In *Proceedings of the 19th International Symposium on Mathematical Theory and Systems*, pages 1609 – 1613, 2010. doi:10.13140/2.1.1472.6089.

G. Poyiadjis, S.S. Singh, and A. Doucet. Gradient-free maximum likelihood parameter estimation with particle filters. In *Proceedings of the 2006 IEEE American Control Conference*, pages 3062 – 3067, 2006. doi:10.1109/ACC.2006.1657187.

Patrick Rebeschini and Ramon van Handel. Can local particle filters beat the curse of dimensionality? *The Annals of Applied Probability*, 25(5):2809 – 2866, 2015. doi:10.1214/14-AAP1061.

Javad Rezaie and Jo Eidsvik. Kalman filter variants in the closed skew normal setting. *Computational Statistics & Data Analysis*, 75:1–14, 2014. doi:10.1016/j.csda.2014.01.014.

Claudio M. Rocco Sanseverino and José Emmanuel Ramirez-Marquez. Uncertainty propagation and sensitivity analysis in system reliability assessment via unscented transformation. *Reliability Engineering & System Safety*, 132:176 – 185, 2014. doi:10.1016/j.ress.2014.07.024.

Michael Roth, Emre Ozkan, and Fredrik Gustafsson. A Student's t filter for heavy tailed process and measurement noise. In *Proceedings of the 2013 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, pages 5770 – 5774, 2013. doi:10.1109/ICASSP.2013.6638770.

Michael Roth, Carsten Fritsche, Gustaf Hendeby, and Fredrik Gustafison. The ensemble Kalman filter and its relations to other nonlinear filters. In *Proceedings of the 23rd European Signal Processing Conference (EUSIPCO), 2015*, pages 1236 – 1240. IEEE, 2015. doi:10.1109/EUSIPCO.2015.7362581.

Yong Rui and Yunqiang Chen. Better proposal distributions: object tracking using unscented particle filter. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, volume 2, pages 786 – 793, 2001. doi:10.1109/CVPR.2001.991045.

Anders Rytter. *Vibrational Based Inspection of Civil Engineering Structures*. PhD thesis, Dept. of Building Technology and Structural Engineering, Aalborg University, Aalborg, Denmark, 1993.

Saikat Saha and Fredrik Gustafsson. Particle filtering with dependent noise processes. *IEEE Transactions on Signal Processing*, 60(9):4497 – 4508, 2012. doi:10.1109/TSP.2012.2202653.

R. Sajeeb, C. S. Manohar, and D. Roy. Rao-blackwellization with substructuring for identification of a class of noisy nonlinear dynamical systems. *International Journal of Engineering under Uncertainty: Hazards, Assessment and Mitigation*, 1(1-2):81 – 99, 2009a.

R. Sajeeb, C. S. Manohar, and D. Roy. A conditionally linearized monte carlo filter in non-linear structural dynamics. *International Journal of Non-Linear Mechanics*, 44(7):776 – 790, 2009b. doi:10.1016/j.ijnonlinmec.2009.04.001.

R. Sajeeb, C. S. Manohar, and D. Roy. A semi-analytical particle filter for identification of nonlinear oscillators. *Probabilistic Engineering Mechanics*, 25(1):35 – 48, 2010. doi:10.1016/j.probengmech.2009.05.004.

Simo Särkkä. *Bayesian Filtering and Smoothing*. Cambridge University Press, 2013.

Thomas Schön and Fredrik Gustafsson. Particle filters for system identification of state-space models linear in either parameters or states. In *Proceedings of the 13th IFAC Symposium on System Identification*, pages 1287 – 1292, 2003.

Thomas Schön, Fredrik Gustafsson, and Per-johan Nordlund. Marginalized particle filters for mixed linear / nonlinear state-space models. *IEEE Transactions on Signal Processing*, 53(7):2279 – 2289, 2005. doi:10.1109/TSP.2005.849151.

Z.Y. Shi, S.S. Law, and L.M. Zhang. Structural damage localization from modal strain energy change. *Journal of*

Sound and Vibration, 218(5):825–844, 1998. doi:10.1006/jsvi.1998.1878.

Masanobu Shinozuka and George Deodatis. Simulation of stochastic processes by spectral representation. *Applied Mechanics Reviews*, 44:191 – 204, 1991. doi:10.1115/1.3119501.

B.W. Silverman. *Density estimation for statistics and data analysis*. Chapman and Hall, 1986.

Ralph C. Smith. *Uncertainty quantification: theory, implementation, and applications*. Society for Industrial and Applied Mathematics, 2014.

Chris Snyder. Particle filters, the "optimal" proposal and high-dimensional systems. In *ECMWF Seminar on Data assimilation for atmosphere and ocean*, 2011.

Chris Snyder, Thomas Bengtsson, Peter Bickel, and Jeff Anderson. Obstacles to high-dimensional particle filtering. *Monthly Weather Review*, 136:4629 – 4640, 2008. doi:10.1175/2008MWR2529.1.

Chris Snyder, Thomas Bengtsson, and Mathias Morzfeld. Performance bounds for particle filters using the optimal proposal. *Monthly Weather Review*, 143(11):4750 – 4761, 2015. doi:10.1175/MWR-D-15-0144.1.

Antti Solonen, Tiangang Cui, Janne Hakkarainen, and Youssef Marzouk. On dimension reduction in Gaussian filters. *Inverse Problems*, 32(4):045003, 2016. doi:10.1088/0266-5611/32/4/045003.

Tommi Sottinen and Simo Särkkä. Application of Girsanov theorem to particle filtering of discretely observed continuous-time non-linear systems. *Bayesian Analysis*, 3(3):555 – 584, 2008. doi:10.1214/08-BA322.

M. S. Srivastava. A measure of skewness and kurtosis and a graphical method for assessing multivariate normality. *Statistics & Probability Letters*, 2(5):263 – 267, 1984. doi:10.1016/0167-7152(84)90062-2.

G. Storvik. Particle filters for state-space models with the presence of unknown static parameters. *IEEE Transactions on Signal Processing*, 50(2):281 – 289, 2002. doi:10.1109/78.978383.

Lifen Sui, Zhongkai Mou, Yu Gan, and Xianyuan Huang. Unscented Kalman filter algorithm with colored noise and its application in spacecraft attitude estimation. In *Proceedings of the 1st International Workshop on the Quality of Geodetic Observation and Monitoring Systems (QuGOMS 2011)*, pages 95 – 100. Springer International Publishing, 2015. doi:10.1007/978-3-319-10828-5_14.

Robert L. Taylor. *FEAP - A finite element analysis program, version 8.5 theory manual*. Department of Civil and Environmental Engineering, University of California at Berkeley, 2017.

Dirk Tenne and Tarunraj Singh. The higher order unscented filter. In *Proceedings of the 2003 IEEE American Control Conference*, volume 3, pages 2441 – 2446, 2003. doi:10.1109/ACC.2003.1243441.

Majid K. Vakilzadeh, Yong Huang, James L. Beck, and Thomas Abrahamsson. Approximate Bayesian Computation by Subset Simulation using hierarchical state-space models. *Mechanical Systems and Signal Processing*, 84(Part B):2 – 20, 2017. doi:10.1016/j.ymssp.2016.02.024.

R. van der Merwe, A. Doucet, N. De Freitas, and E. Wan. The unscented particle filter. In *Advances in Neural Information Processing Systems 13 (NIPS 2000)*, pages 584 – 590. MIT Press, 2001.

Rudolph van der Merwe. *Sigma-point Kalman Filters for Probabilistic Inference in Dynamic State-Space Models*. PhD thesis, OGI School of Science & Engineering, Oregon Health & Science University, Portland, OR, 2004.

Peter Jan van Leeuwen. Particle filtering in geophysical systems. *Monthly Weather Review*, 137(12):4089 – 4114, 2009. doi:10.1175/2009MWR2835.1.

Peter Van Overschee and Bart De Moor. *Subspace Identification for Linear Systems*. Springer US, copyright Kluwer Academic Publishing, 2006. doi:10.1007/978-1-4613-0465-4.

Eric A. Wan and Rudolph van der Merwe. The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium*, pages 153 – 158, 2000. doi:10.1109/ASSPCC.2000.882463.

Max Welling. Robust higher order statistics. In *Proceedings of the 10th International Workshop on Artificial Intelligence and Statistics (AISTATS 2005)*, pages 405 – 412, 2005.

K. Worden, Charles R. Farrar, J. Haywood, and M. Todd. A review of nonlinear dynamics applications to structural health monitoring. *Structural Control and Health Monitoring*, 15:540 – 567, 2008. doi:10.1002/stc.215.

Jiahe Xi, Pablo Lamata, Jack Lee, Philippe Moireau, Dominique Chapelle, and Nic Smith. Myocardial transversely isotropic material parameter estimation from in-silico measurements based on a reduced-order unscented Kalman filter. *Journal of the Mechanical Behavior of Biomedical Materials*, 4(7):1090 – 1102, 2011. doi:10.1016/j.jmbbm.2011.03.018.

Appendix A: Kalman filter and extended Kalman filter equations

A.1 Kalman filter equations

Let us consider the linear Gaussian system:

$$z_k = G_{k-1} z_{k-1} + g_{k-1} + v_{k-1} \tag{A.1a}$$

$$y_k = H_k z_k + h_k + \eta_k \tag{A.1b}$$

At each time step k the filtering posterior pdf of the state z_k knowing the measurements $y_{1:k}$ can be inferred exactly using the Kalman filter equations. As for any Bayesian filtering scheme, these equations consist of a propagation and an update step:

- Start with posterior at time step $k-1$: $\mathcal{N}\left(z_{k-1}; z_{k-1|k-1}, P_{k-1|k-1}\right)$
- Propagation equation

$$\left[z_{k|k-1}, P_{k|k-1}\right] = KF\,propagate_{(G_{k-1}, g_{k-1})}\left\{z_{k-1|k-1}, P_{k-1|k-1}\right\}$$

  which consists of:

$$z_{k|k-1} = G_{k-1} z_{k-1|k-1} + g_{k-1}$$

$$P_{k|k-1} = G_{k-1} P_{k-1|k-1} G_{k-1}^T + Q_{k-1}$$

- Update equation

$$\left[z_{k|k}, P_{k|k}\right] = KFupdate_{(y_k;H_k,h_k)}\left\{z_{k|k-1}, P_{k|k-1}\right\}$$

  which consists of:

$$z_{k|k} = z_{k|k-1} + K_k(y_k - H_k z_{k|k-1} - h_k)$$

$$P_{k|k} = (I - K_k H_k)P_{k|k-1}$$

$$K_k = P_{k|k-1}H_k^T L_k^{-1}$$

$$L_k = H_k P_{k|k-1}H_k^T + R_k$$

- End with posterior at time step k: $\mathcal{N}\left(z_k; z_{k|k}, P_{k|k}\right)$

A.2 Second order extended Kalman filter equations

Let's consider the nonlinear Gaussian system:

$$z_k = \varphi(z_{k-1}) + g_{k-1} + v_{k-1} \tag{A.2a}$$

$$y_k = \psi(z_k) + h_k + \eta_k \tag{A.2b}$$

where φ and ψ are nonlinear functions. In the extended Kalman filters, one linearizes those two functions using their Taylor series expansion around the mean. In the second order extended Kalman filter (EKF^2), terms up to the second order are kept in the expansion. Thus one can write for the propagation equation:

$$\varphi(z_{k-1}) = \varphi(z_{k-1|k-1} + \delta z) \approx \varphi(z_{k-1|k-1}) + G_z \delta z + \sum_i \frac{1}{2}\delta z G_{zz}^i \delta z \varepsilon_i$$

with G_z the jacobian of φ and G_{zz}^i the hessian of the i^{th} component of φ, both computed at point $z_{k-1|k-1}$. Also, ε_i is a vector of zeros with 1 at position i. For the measurement equation one has:

$$\psi(z_k) = \psi(z_{k|k-1} + \delta z) \approx \psi(z_{k|k-1}) + H_z \delta z + \sum_i \frac{1}{2}\delta z H_{zz}^i \delta z \varepsilon_i$$

with H_z the jacobian of ψ and H^i_{zz} the hessian of the i^{th} component of ψ, both computed at point $z_{k|k-1}$.

One can then show (see for example [Särkkä, 2013]) that the propagation and update equations of the second order extended Kalman filter can be written as follows:

- Propagation equation

$$\left[z_{k|k-1}, P_{k|k-1}\right] = EKF^2 propagate_{(\varphi)} \left\{z_{k-1|k-1}, P_{k-1|k-1}\right\}$$

  which consists of:

$$z_{k|k-1} = \varphi(z_{k-1|k-1}) + \frac{1}{2}\sum_i \varepsilon_i tr\{G^i_{zz} P_{k-1|k-1}\}$$

$$P_{k|k-1} = G_z P_{k-1|k-1} G_z^T + \frac{1}{2}\sum_{i,i'} \varepsilon_i \varepsilon_{i'}^T tr\{G^i_{zz} P_{k-1|k-1} G^{i'}_{zz} P_{k-1|k-1}\} + Q_{k-1}$$

- Update equation

$$\left[z_{k|k}, P_{k|k}\right] = EKF^2 update_{(y_k;\psi)} \left\{z_{k|k-1}, P_{k|k-1}\right\}$$

  which consists of:

$$z_{k|k} = z_{k|k-1} + K_k(y_k - \psi(z_{k|k-1}) - \frac{1}{2}\sum_i \varepsilon_i tr\{H^i_{zz} P_{k|k-1}\})$$

$$P_{k|k} = P_{k|k-1} - K_k L_k K_k^T$$

$$K_k = P_{k|k-1} H_z^T L_k^{-1}$$

$$L_k = H_z P_{k|k-1} H_z^T + \frac{1}{2}\sum_{i,i'} \varepsilon_i \varepsilon_{i'}^T tr\{H^i_{zz} P_{k|k-1} H^{i'}_{zz} P_{k|k-1}\} + R_k$$

Note: in the first order Kalman filter, only the first order terms are kept in the Taylor expansion. The equations for this filter can be easily obtained from the previous equations by setting all the Hessian terms to 0.

This page is intentionally left blank.

Appendix B: Properties of the CSN and GM distributions

B.1 Moments of the CSN distribution

In chapter 3, section 3.4.2, a nonlinear Kalman filter is derived that uses a closed skew normal distribution as its baseline pdf. Derivation of this filter involves moment matching, i.e., learn the parameters of the CSN distribution knowing its moments (propagated through the UT). In the case of a CSN with $q = 1$, one can analytically compute the moments of the CSN distribution since Φ_1 and ϕ_1 have closed form expressions. However those moments are related in a nonlinear way to the parameters of the distribution, thus rendering moment matching quite challenging. In the numerical example presented in section 3.4.2 we restricted ourselves to the case of $q = 1, \nu = 0$, for which one can derive:

$$E[X] = \mu + A^T \frac{2}{\sqrt{2\pi D}} \tag{B.1a}$$

$$cov(X,X) = \Sigma - \frac{2}{\pi D} A^T A \tag{B.1b}$$

$$E\left[(X - E[X])_j^3\right] = (16 - 4\pi)\left(\frac{a_j}{\sqrt{2\pi D}}\right)^3 \tag{B.1c}$$

where $A = \Gamma\Sigma$ and $D = \Delta + \Gamma\Sigma\Gamma^T$. Moment matching can be performed analytically in this way (no need for an optimization scheme), however this distribution might be unable to represent complex

high dimensional RVs.

If that is the case, one could think about extending to $q = 2$, for which we found that moments can also be derived analytically, using the following:

$$\frac{d\Phi_2(x,y)}{dx} = \int_y \phi_2(x,y)dy = \phi_1(x;\mu_1,\sigma_1)\Phi_1(y;\mu_2+\rho\sigma_2\frac{x-\mu_1}{\sigma_1},\sigma_2\sqrt{1-\rho^2}) \tag{B.2a}$$

$$\frac{d\Phi_2(x,y)}{dy} = \phi_1(y;\mu_2,\sigma_2)\Phi_1(x;\mu_1+\rho\sigma_1\frac{y-\mu_2}{\sigma_2},\sigma_1\sqrt{1-\rho^2}) \tag{B.2b}$$

where Φ_2 is the cdf of the 2-dimensional Gaussian distribution with mean $\begin{Bmatrix}\mu_1\\ \mu_2\end{Bmatrix}$, correlation coefficient ρ and standard deviation in each direction σ_1, σ_2. Performing moment matching will however require some sort of optimization scheme.

B.2 CSN distribution: closure under conditioning property

In the framework presented in chapter 3, the closure under conditioning property of the baseline pdf is used for the derivation of the measurement update equations. This property is given here for the CSN distribution, the reader is referred to [González-Farías et al., 2004] for proof of this property. If $x = \begin{Bmatrix}x_1\\ x_2\end{Bmatrix} \sim CSN_{n,q}(\mu,\Sigma,\Gamma,\nu,\Delta)$ with parameters partitioned as $\mu = \begin{Bmatrix}\mu_1\\ \mu_2\end{Bmatrix}$, $\Sigma = \begin{bmatrix}\Sigma_{11} & \Sigma_{12}\\ \Sigma_{21} & \Sigma_{22}\end{bmatrix}$ and $\Gamma = \begin{bmatrix}\Gamma_1 & \Gamma_2\end{bmatrix}$, then the conditional pdf of x_1 knowing $x_2 = \bar{x_2}$ is:

$$x_1|x_2 = \bar{x_2} \sim CSN_{n_1,q}(\mu_1,\Sigma_1,\Gamma_1,\nu_1,\Delta_1) \tag{B.3a}$$

with parameters

$$\mu_{1|2} = \mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(\bar{x_2} - \mu_2), \tag{B.3b}$$

$$\Sigma_{1|2} = \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}, \tag{B.3c}$$

$$\Gamma_{1|2} = \Gamma_1, \tag{B.3d}$$

$$\nu_{1|2} = \nu - (\Gamma_2 + \Gamma_1\Sigma_{12}\Sigma_{22}^{-1})(\bar{x_2} - \mu_2), \tag{B.3e}$$

$$\Delta_{1|2} = \Delta \tag{B.3f}$$

B.3 Gaussian mixture distribution: closure under conditioning property

An identical property needs to be derived for the Gaussian mixture distribution. If $x = \begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix}$ follows a multivariate Gaussian mixture (GM) distribution, i.e.,

$$\begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix} \sim \sum_{l=1}^{L} \pi^{(l)} \mathcal{N}\left(\begin{Bmatrix} x_1 \\ x_2 \end{Bmatrix}; \begin{Bmatrix} \mu_1 \\ \mu_2 \end{Bmatrix}^{(l)}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}^{(l)} \right) \tag{B.4}$$

then the conditional probability of x_1 knowing $x_2 = \bar{x_2}$ is also a GM distribution:

$$p\left(x_1 | x_2 = \bar{x_2}\right) = \frac{p(x_1, \bar{x_2})}{p(\bar{x_2})} = \sum_{l=1}^{L} \hat{\pi}^{(l)} \mathcal{N}\left(x_1 | \bar{x_2}; \mu_{1|2}^{(l)}, \Sigma_{1|2}^{(l)}\right) \tag{B.5}$$

with

$$\hat{\pi}^{(l)} = \frac{\pi^{(l)} \mathcal{N}\left(\bar{x_2}; \mu_2^{(l)}, \Sigma_{22}^{(l)}\right)}{\sum_{j=1}^{L} \pi^{(j)} \mathcal{N}\left(\bar{x_2}; \mu_2^{(j)}, \Sigma_{22}^{(j)}\right)} \tag{B.6a}$$

$$\mu_{1|2}^{(l)} = \mu_1^{(l)} + \Sigma_{12}^{(l)}(\Sigma_{22}^{(l)})^{-1}(\bar{x_2} - \mu_2^{(l)}) \tag{B.6b}$$

$$\Sigma_{1|2}^{(l)} = \Sigma_{11}^{(l)} - \Sigma_{12}^{(l)}(\Sigma_{22}^{(l)})^{-1}\Sigma_{21}^{(l)} \tag{B.6c}$$

This page is intentionally left blank.

Appendix C: Derivation and properties of some higher order sigma point sets for use in the UT

C.1 Sigma point set that captures marginal skewness in all directions

In chapter 3, section 3.4.2, a nonlinear Kalman filter is derived that uses a closed skew normal distribution as its baseline pdf, which enables tracking of third order moments of the posterior pdfs over time. This requires derivation of a higher order sigma point set, which captures skewness information of a RV X. A $N_{sig} = 2n_X + 1$ sigma point set that captures the known mean $E[X] = m$, covariance P_{XX} and marginal skew terms S^X of an input RV X is computed using the following steps:

1. compute the $2n_X$ points and weights of a set that captures the zero mean, identity covariance and marginal skew terms of a white RV Z as:

$$\begin{aligned} \mathcal{Z}^{(i)} &= -z_{1i}e_i & W^{(i)} &= w_{1i}, \quad i = 1 : n_X \\ \mathcal{Z}^{(n_X+i)} &= z_{2i}e_i & W^{(n_X+i)} &= w_{2i}, \quad i = 1 : n_X \end{aligned} \tag{C.1}$$

where e_i represents a column vector with 1 at position i and 0 elsewhere. As derived in [Julier, 1998], z_{1i} is the positive root of the quadratic equation

$$(w_{1i}+w_{2i})z^2+(w_{1i}+w_{2i})S_i^Z z-1=0 \tag{C.2a}$$

and

$$z_{2i}=z_{1i}\frac{w_{1i}}{w_{2i}} \tag{C.2b}$$

$$w_{1i}=\frac{(w_{1i}+w_{2i})}{(w_{1i}+w_{2i})z_{1i}^2+1} \tag{C.2c}$$

$$w_{2i}=(w_{1i}+w_{2i})-w_{1i} \tag{C.2d}$$

The sum of the weights $(w_{1i}+w_{2i})$ (positive) is prescribed in each direction, and the skew vector S^z is computed as:

$$S^Z=\left(\sqrt{P_{XX}}^{\wedge 3}\right)^{-1}S^X \tag{C.3}$$

where $\cdot^{\wedge 3}$ represents element-wise cubic power.

2. compute the $2n_X+1$ points and weights of the set that captures the known mean m, covariance P_{XX} and marginal skew terms S^X of X as:

$$\mathcal{X}^{(0)}=m \qquad W^{(0)}=w_0 \tag{C.4a}$$

$$\mathcal{X}^{(i)}=m+\sqrt{P_{XX}}\mathcal{Z}^{(i)} \qquad W^{(i)}=w_{1i}, \quad i=1:n_X \tag{C.4b}$$

$$\mathcal{X}^{(n_X+i)}=m+\sqrt{P_{XX}}\mathcal{Z}^{(n_X+i)} \qquad W^{(n_X+i)}=w_{2i}, \quad i=1:n_X \tag{C.4c}$$

where w_0 (positive) is also prescribed, constrained by $w_0+\sum_i(w_{1i}+w_{2i})=1$.

This procedure creates a set that captures up to the marginal skewness in each direction. However for complex distributions where the skewness is far from 0, the 4th order moments often have a considerable effect too, especially in the computation of the covariance of the transformed RV. The procedure above allows for some liberty in the choice of w_0 and $(w_{1i}+w_{2i})$ in each direction i,

thus one can choose these so that the error on the fourth order marginal moments is minimized. In our implementation, we choose w_0 to minimize this error on the fourth order moments, and $w_{1i} + w_{2i} = \frac{1-w_0}{n_X}$ in each direction.

An important note regarding Eq. (C.3): this equation is valid only because the set of points derived for the white RV Z sets all the cross-skew terms to 0. Only if those cross-skew terms are set to 0 one can write:

$$E\left[(X-m)_j^3\right] = E\left[\left(\sqrt{P_{XX}}_{(j,:)} Z\right)^3\right] = \sqrt{P_{XX}}_{(j,:)}^{\wedge 3} E\left[Z^{\wedge 3}\right] \tag{C.5a}$$

$$S_j^X = \sqrt{P_{XX}}_{(j,:)}^{\wedge 3} S^Z \tag{C.5b}$$

and thus obtain Eq. (C.3) in a matrix format.

C.2 Moments captured by the $L \cdot (2n_x + 1)$ Gaussian mixture sigma point set

In chapter 5, section 5.3, a unscented Kalman filter that uses a Gaussian mixture as its baseline distribution is derived. This requires usage of an appropriate sigma point set. In this section we prove that this set captures first and second order moments of a Gaussian mixture, thus achieving same order of accuracy than the generic symmetric sigma point set for unimodal distributions. We thus consider an input RV X, known to be distributed as a mixture of L Gaussians

$$X \sim \sum_{l=1}^{L} \pi^{(l)} \mathcal{N}\left(\cdot; \mu^{(l)}, \Sigma^{(l)}\right)$$

In this section we show that a sigma point set composed of L symmetric sets, one for each Gaussian $\cdot^{(l)}$ in the mixture, weighted according to the mixture weights $\pi^{(l)}$, captures the mean and covariance of the GM input RV X. This set will then achieve second order accuracy on estimation of the mean of a transformed RV $Z = g(X)$, and first order accuracy on the covariance estimation.

We define the set of $N_{sig} = L \cdot (2n_x + 1)$ sigma points $\{\mathcal{X}^{(i,l)}\}_{i=0:2n_x, l=1:L}$ and their associated weights $\{W^{(i,l)}\}_{i=0:2n_x, l=1:L}$ as:

$$
\begin{aligned}
&\forall l \in [1:L], \\
&\mathcal{X}^{(0,l)} = \mu^{(l)} \qquad W^{(0,l)} = \pi^{(l)} w^{(0)} \\
&\mathcal{X}^{(i,l)} = \mu^{(l)} + \sqrt{\kappa + n_x}(\sqrt{\Sigma^{(l)}})_i \qquad W^{(i,l)} = \pi^{(l)} w^{(i)}, \quad \text{for } i = 1:n_x \\
&\mathcal{X}^{(n_x+i,l)} = \mu^{(l)} - \sqrt{\kappa + n_x}(\sqrt{\Sigma^{(l)}})_i \qquad W^{(i,l)} = \pi^{(l)} w^{(n_x+i)}, \quad \text{for } i = 1:n_x
\end{aligned}
$$

where $\{w^{(i)}\}_{i=0:2n_x}$ are the weights associated to the symmetric set. For each Gaussian in the mixture, the symmetric set captures mean and covariance of the input Gaussian $\cdot^{(l)}$ by construction, i.e.,

$$
\begin{aligned}
&\sum_{i=0}^{2n_x} w^{(i)} \mathcal{X}^{(i,l)} = \mu^{(l)} \\
&\sum_{i=0}^{2n_x} w^{(i)} (\mathcal{X}^{(i,l)} - \mu^{(l)})(\mathcal{X}^{(i,l)} - \mu^{(l)})^T = \Sigma^{(l)}
\end{aligned}
$$

Then we show that the total set captures the global mean and covariance of the RV X (mixture of L Gaussians). Derivation for the mean is as follows:

$$
\begin{aligned}
\sum_{l=1}^{L} \sum_{i=0}^{2n_x} W^{(i,j)} \mathcal{X}^{(i,l)} &= \sum_{l=1}^{L} \pi^{(l)} \sum_{i=0}^{2n_x} w^{(i)} \mathcal{X}^{(i,l)} \\
&= \sum_{l=1}^{L} \pi^{(l)} \mu^{(l)} \\
&= E[X] \text{ (equation for mean of a Gaussian mixture)}
\end{aligned}
$$

and for the covariance:

$$\begin{aligned}
&\sum_{l=1}^{L}\sum_{i=0}^{2n_x} W^{(i,j)}(\mathcal{X}^{(i,l)} - E[X])(\mathcal{X}^{(i,l)} - E[X])^T \\
&= \sum_{l=1}^{L} \pi^{(l)} \sum_{i=0}^{2n_x} w^{(i)}(\mathcal{X}^{(i,l)} - \mu^{(l)} + \mu^{(l)} - E[X])(\mathcal{X}^{(i,l)} - \mu^{(l)} + \mu^{(l)} - E[X])^T \\
&= \sum_{l=1}^{L} \pi^{(l)} \sum_{i=0}^{2n_x} w^{(i)} \left((\mathcal{X}^{(i,l)} - \mu^{(l)})(\mathcal{X}^{(i,l)} - \mu^{(l)})^T + (\mu^{(l)} - E[X])(\mu^{(l)} - E[X])^T \right) \\
&+ \sum_{l=1}^{L} \pi^{(l)} \sum_{i=0}^{2n_x} w^{(i)} \left((\mathcal{X}^{(i,l)} - \mu^{(l)})(\mu^{(l)} - E[X])^T + (\mu^{(l)} - E[X])(\mathcal{X}^{(i,l)} - \mu^{(l)})^T \right)
\end{aligned}$$

The two last terms of this 4 terms sum can be shown to equate to 0, as:

$$\begin{aligned}
\sum_{i=0}^{2n_x} w^{(i)}(\mathcal{X}^{(i,l)} - \mu^{(l)})(\mu^{(l)} - E[X])^T &= \left(\left(\sum_{i=0}^{2n_x} w^{(i)} \mathcal{X}^{(i,l)} \right) - \mu^{(l)} \right) (\mu^{(l)} - E[X])^T \\
&= (\mu^{(l)} - \mu^{(l)})(\mu^{(l)} - E[X])^T \\
&= 0
\end{aligned}$$

which then gives for the covariance:

$$\begin{aligned}
&\sum_{l=1}^{L}\sum_{i=0}^{2n_x} W^{(i,j)}(\mathcal{X}^{(i,l)} - E[X])(\mathcal{X}^{(i,l)} - E[X])^T \\
&= \sum_{l=1}^{L} \pi^{(l)} \sum_{i=0}^{2n_x} w^{(i)} \left((\mathcal{X}^{(i,l)} - \mu^{(l)})(\mathcal{X}^{(i,l)} - \mu^{(l)})^T + (\mu^{(l)} - E[X])(\mu^{(l)} - E[X])^T \right) \\
&= \sum_{l=1}^{L} \pi^{(l)} \left(\Sigma^{(l)} + (\mu^{(l)} - E[X])(\mu^{(l)} - E[X])^T \right) \\
&= Cov(X) \text{ (equation for covariance of a Gaussian mixture)}
\end{aligned}$$

This page is intentionally left blank.

Appendix D: Derivation of Jacobians for use in the eMUKF for parameter estimation in FE models

D.1 Nonlinear hyperbolic problem

In this section, derivation of the approximate Jacobian $J_f = \begin{bmatrix} \frac{\partial \mathrm{u}_{k+1}}{\partial \mathrm{u}_k} & \frac{\partial \mathrm{u}_{k+1}}{\partial \mathrm{v}_k} \\ \frac{\partial \mathrm{v}_{k+1}}{\partial \mathrm{u}_k} & \frac{\partial \mathrm{v}_{k+1}}{\partial \mathrm{v}_k} \end{bmatrix}$ whose formula is given in chapter 4, section 4.4, Eq. (4.32) is explained in more details. First let us recall the Newmark formula for solving a nonlinear hyperbolic initial value problem:

$$\mathrm{u}_{k+1} = \mathrm{u}_k + dt\,\mathrm{v}_k + \frac{dt^2}{4}(\mathrm{a}_k + \mathrm{a}_{k+1}) \tag{D.1a}$$

$$\mathrm{v}_{k+1} = \mathrm{v}_k + \frac{dt}{2}(\mathrm{a}_k + \mathrm{a}_{k+1}) \tag{D.1b}$$

$$M\mathrm{a}_{k+1} + C\mathrm{v}_{k+1} + \mathrm{F}(\mathrm{u}_{k+1}) = \mathrm{P}(\mathrm{u}_{k+1}) \tag{D.1c}$$

Combining the two first equations to cancel the acceleration terms lead to:

$$\mathrm{v}_{k+1} + \mathrm{v}_k = \frac{2}{dt}(\mathrm{u}_{k+1} - \mathrm{u}_k) \tag{D.2}$$

From this equation it is easy to relate derivatives of velocity and displacement terms, which will

be further needed in the derivation. For instance, one can write:

$$\frac{\partial \mathrm{v}_{k+1}}{\partial \mathrm{u}_k} = \frac{2}{dt}\left(\frac{\partial \mathrm{u}_{k+1}}{\partial \mathrm{u}_k} - 1\right) \tag{D.3}$$

We focus on the derivation of the first term of the Jacobian, $\frac{\partial \mathrm{u}_{k+1}}{\partial \mathrm{u}_k}$. Derivation of the remaining terms follow the same logic. Re-arranging the first equation, and pre-multiplying by mass matrix M yields

$$M\mathrm{u}_{k+1} - \frac{dt^2}{4}M\mathrm{a}_{k+1} = M\mathrm{u}_k + dtM\mathrm{v}_k + \frac{dt^2}{4}M\mathrm{a}_k \tag{D.4}$$

From the equilibrium equation, one can write for both $j = k, k+1$

$$M\mathrm{a}_j = \mathrm{P}(\mathrm{u}_j) - C\mathrm{v}_j - \mathrm{F}(\mathrm{u}_j) \tag{D.5}$$

$$\frac{\partial M\mathrm{a}_j}{\partial \mathrm{u}_j} = -K_t|_{@j} \tag{D.6}$$

$$\frac{\partial M\mathrm{a}_j}{\partial \mathrm{v}_j} = -C \tag{D.7}$$

Thus, taking the partial derivative of Eq. D.4 with respect to u_k, being careful to apply the chain rule for the left hand size as $\frac{\partial}{\partial \mathrm{u}_k} = \frac{\partial}{\partial \mathrm{u}_{k+1}}\frac{\partial \mathrm{u}_{k+1}}{\partial \mathrm{u}_k} + \frac{\partial}{\partial \mathrm{v}_{k+1}}\frac{\partial \mathrm{v}_{k+1}}{\partial \mathrm{u}_k}$ gives:

$$M\frac{\partial \mathrm{u}_{k+1}}{\partial \mathrm{u}_k} - \frac{dt^2}{4}\left(-K_t|_{@k+1}\frac{\partial \mathrm{u}_{k+1}}{\partial \mathrm{u}_k} - C\frac{\partial \mathrm{v}_{k+1}}{\partial \mathrm{u}_k}\right) = M + \frac{dt^2}{4}\left(-K_t|_{@k}\right) \tag{D.8}$$

$$\left(M + \frac{dt^2}{4}K_t|_{@k+1}\right)\frac{\partial \mathrm{u}_{k+1}}{\partial \mathrm{u}_k} + \frac{dt^2}{4}C\frac{\partial \mathrm{v}_{k+1}}{\partial \mathrm{u}_k}\right) = M - \frac{dt^2}{4}K_t|_{@k} \tag{D.9}$$

Now making use of Eq. (D.3) and re-arranging gives

$$\left(M + \frac{dt^2}{4}K_t|_{@k+1} + \frac{dt}{2}C\right)\frac{\partial \mathrm{u}_{k+1}}{\partial \mathrm{u}_k} = M - \frac{dt^2}{4}K_t|_{@k} + \frac{dt}{2}C \tag{D.10}$$

$$\frac{\partial \mathrm{u}_{k+1}}{\partial \mathrm{u}_k} = \left(M + \frac{dt^2}{4}K_t|_{@k+1} + \frac{dt}{2}C\right)^{-1}\left(M - \frac{dt^2}{4}K_t|_{@k} + \frac{dt}{2}C\right) \tag{D.11}$$

which ends the proof for the first term of the Jacobian states in Eq. 4.32. Looking at this equation one can also see that it agrees with the linear Jacobian derived in Eq. (4.21), since for a linear system, $K_t|_{@j} = K, \forall j$. It should also be mentioned that this Jacobian is an approximation since

it does not take into account all intermediate steps of the Newton iteration scheme. The Jacobian derived therein only involves evaluations of the tangent stiffness matrix at both times k and $k+1$, while in reality this matrix is updated several times per time step, once per iteration of the Newton scheme. In the numerical experiments, we used $K_t|_{@k}$ as the first evaluation during time step $k+1$ and $K_t|_{@k+1}$ as the last evaluation at time step $k+1$. However, if the sampling time step dt is relatively large and the system highly nonlinear, the tangent stiffness matrix might vary significantly within a time step and this approximation might lead to erroneous results. We thus recommend in this case to perform several smaller propagation steps, similar to a bisection method used in FEA when the Newton does not converge.

D.2 Nonlinear parabolic problem

For a parabolic problem of the form $M\mathrm{v} + \mathrm{F}(\mathrm{u}) = \mathrm{P}(\mathrm{u})$, the trapezoidal rule with $\alpha = \frac{1}{2}$ can be used to solve for the initial value problem as:

$$M\mathrm{v}_{k+1} + \mathrm{F}(\mathrm{u}_{k+1}) = \mathrm{P}(\mathrm{u}_{k+1}) \tag{D.12a}$$

$$\mathrm{u}_{k+1} = \mathrm{u}_k + \frac{dt}{2}(\mathrm{v}_k + \mathrm{v}_{k+1}) \tag{D.12b}$$

Pre-multiplying the second equation by the mass matrix M and eliminating the velocity terms using the first equation yields:

$$M\mathrm{u}_{k+1} + \frac{dt}{2}\left(\mathrm{F}(\mathrm{u}_{k+1}) - \mathrm{P}(\mathrm{u}_{k+1})\right) = M\mathrm{u}_k - \frac{dt}{2}\left(\mathrm{F}(\mathrm{u}_k) - \mathrm{P}(\mathrm{u}_k)\right) \tag{D.13}$$

Taking the derivative with respect to u_k yields

$$\left(M + \frac{dt}{2}K_t|_{@k+1}\right)\frac{\mathrm{u}_{k+1}}{\mathrm{u}_k} = M - \frac{dt}{2}K_t|_{@k} \tag{D.14}$$

where the tangent stiffness matrix is defined as $K_t = \frac{\partial(\mathrm{F}-\mathrm{P})}{\partial \mathrm{u}}$. Again, this equation agrees with its linear counterpart given in Eq. (4.18).

This page is intentionally left blank.

Appendix E: ORC identifiability test for a SDOF Duffing oscillator

In this section we show an example of identifiability test, namely the ORC test, used to determine if a SDOF Duffing oscillator is or not identifiable (local identifiability). We consider here an affine-input nonlinear system (with only one input for simplicity), which can be written in the form:

$$\dot{X} = f(X) + g(X)u \tag{E.1a}$$

$$y_i = h_i(X) \tag{E.1b}$$

Then the ORC test has a simplified version, which is explained here using an example (see for instance [Chatzis et al., 2015b] for more detailed explanations on this test and identifiability in general). The problem of interest here is a Duffing oscillator:

$$\dot{x}_1 = x_2 \tag{E.1c}$$

$$\dot{x}_2 = -\theta_1 x_1 - \theta_2 x_2 - \theta_3 x_1^3 + \theta_4 u \tag{E.1d}$$

$$\dot{\theta} = 0 \tag{E.1e}$$

$$y_1 = x_1 \tag{E.1f}$$

where $\theta = \begin{bmatrix} \frac{k}{m} & \frac{c}{m} & \frac{\alpha}{m} & \frac{\beta}{m} \end{bmatrix}^T$. We perform the test with respect to the augmented vector $X = \begin{bmatrix} x_1 & x_2 & \theta^T \end{bmatrix}^T \in R^6$. The parameter vector is identifiable if the rank condition in the ORC test is

satisfied, i.e. if rank $d\Omega_{k+1} = 6$ when $k \leq 6-2 = 4$. The steps of the ORC test are as follows:

Starting point: $k = 0$, $\Omega_0 = \begin{bmatrix} h_1, h_2, \cdots \end{bmatrix} = x_1$, $\Delta\Omega_0 = \Omega_0 = x_1$, $d\Omega_0 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$ and rank $d\Omega_0 = 1$.

First iteration:

- $\Delta\Omega_1 = \begin{bmatrix} (d(\Delta\Omega_0) \cdot f)^T, (d(\Delta\Omega_0) \cdot g)^T \end{bmatrix}^T = [x_2, 0]^T$
- $\Omega_1 = \Omega_0 \cup \Delta\Omega_1 = \begin{bmatrix} x_1, x_2, 0 \end{bmatrix}^T$
- $d\Omega_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$
- rank $d\Omega_1 = 2 \neq$ rank $d\Omega_0 and \neq 6$ and $k = 0 < 4$ so go on to next iteration.
- (optional) Eliminate dependent rows from $d\Omega_1$ and corresponding rows from Ω_1, i.e. $d\Omega_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}$ and $\Omega_1 = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$

Second iteration ($k = 1$).

- $\Delta\Omega_2 = \begin{bmatrix} d(\Delta\Omega_1) \cdot f \\ d(\Delta\Omega_1) \cdot g \end{bmatrix} = \begin{bmatrix} -\theta_1 x_1 - \theta_2 x_2 - \theta_3 x_1^3 \\ \theta_4 \end{bmatrix}$
- $\Omega_2 = \Omega_1 \cup \Delta\Omega_2 = \begin{bmatrix} x_1 \\ x_2 \\ -\theta_1 x_1 - \theta_2 x_2 - \theta_3 x_1^3 \\ \theta_4 \end{bmatrix}$
- $d\Omega_2 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ -\theta_1 - 3\theta_3 x_1^2 & -\theta_2 & -x_1 & -x_2 & -x_1^3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$
- rank $d\Omega_2 = 4 \neq$ rank $d\Omega_1 and \neq 6$ and $k = 1 < 4$ so go on to next iteration.

Third iteration ($k = 2$).

- $\Delta\Omega_3 = \begin{bmatrix} d(\Delta\Omega_2)\cdot f \\ d(\Delta\Omega_2)\cdot g \end{bmatrix} = \begin{bmatrix} \theta_2(\theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^3) - x_2(\theta_1 + 3\theta_3 x_1^2) \\ 0 \\ -\theta_2\theta_4 \\ 0 \end{bmatrix}$

- $\Omega_3 = \Omega_2 \cup \Delta\Omega_3 = \begin{bmatrix} x_1 \\ x_2 \\ -\theta_1 x_1 - \theta_2 x_2 - \theta_3 x_1^3 \\ \theta_4 \\ \theta_2(\theta_1 x_1 + \theta_2 x_2 + \theta_3 x_1^3) - x_2(\theta_1 + 3\theta_3 x_1^2) \\ 0 \\ -\theta_2\theta_4 \\ 0 \end{bmatrix}$

- $d\Omega_3 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ -\theta_1 - 3\theta_3 x_1^2 & -\theta_2 & -x_1 & -x_2 & -x_1^3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -6\theta_3 x_1 x_2 + \theta_1\theta_2 + 3\theta_2\theta_3 x_1^2 & -\theta_1 - 3\theta_3 x_1^2 + \theta_2^2 & -x_2 + \theta_2 x_1 & \theta_1 x_1 + 2\theta_2 x_2 + \theta_3 x_1^3 & -3x_1^2 x_2 + \theta_2 x_1^3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\theta_4 & 0 & -\theta_2 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$

- rank $d\Omega_3 = 6$, the rank condition is satisfied and the parameter vector is identifiable.