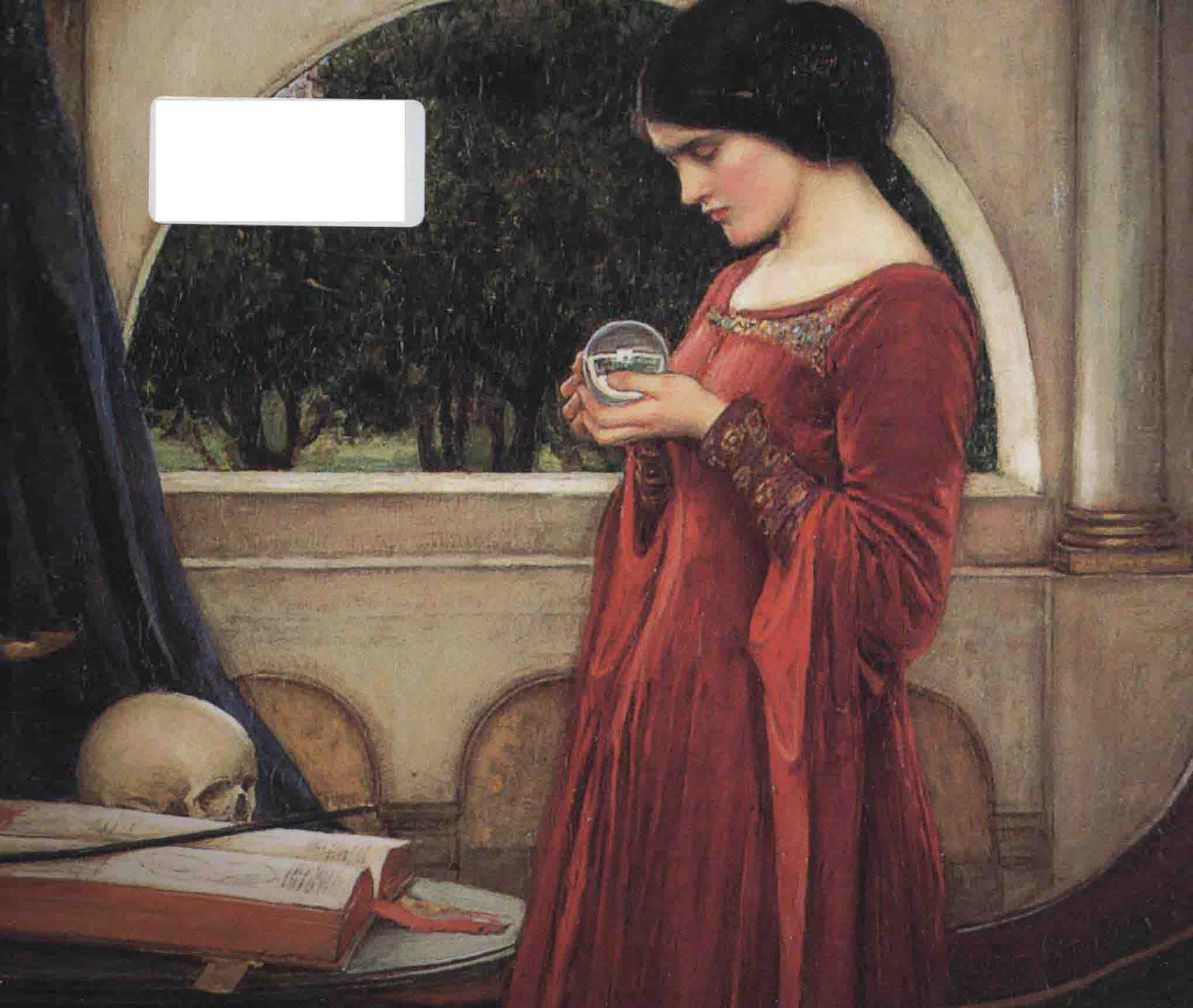

Probabilistic Forecasting and Bayesian Data Assimilation

Sebastian Reich and Colin Cotter

Probabilistic Forecasting and Bayesian Data Assimilation

SEBASTIAN REICH
University of Potsdam and University of Reading

COLIN COTTER
Imperial College London

CAMBRIDGE
UNIVERSITY PRESS

CAMBRIDGE
UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom

Cambridge University Press is part of the University of Cambridge.

It furthers the University's mission by disseminating knowledge in the pursuit of education, learning and research at the highest international levels of excellence.

www.cambridge.org
Information on this title: www.cambridge.org/9781107069398

First published 2015

Printed in the United Kingdom by Bell and Bain Ltd

A catalogue record for this publication is available from the British Library

ISBN 978-1-107-06939-8 Hardback
ISBN 978-1-107-66391-6 Paperback

Probabilistic Forecasting and Bayesian Data Assimilation

In this book the authors describe the principles and methods behind probabilistic forecasting and Bayesian data assimilation. Instead of focusing on particular application areas, the authors adopt a general dynamical systems approach, with a selection of low-dimensional, discrete time numerical examples designed to build intuition about the subject.

Part I explains the mathematical framework of ensemble-based probabilistic forecasting and uncertainty quantification. Part II is devoted to Bayesian filtering algorithms, from classical data assimilation algorithms such as the Kalman filter, variational techniques, and sequential Monte Carlo methods, through to more recent developments such as the ensemble Kalman filter and ensemble transform filters. The McKean approach to sequential filtering in combination with coupling of measures serves as a unifying mathematical framework throughout Part II.

The prerequisites are few. Some basic familiarity with probability is assumed, but concepts are explained when needed, making this an ideal introduction for graduate students in applied mathematics, computer science, engineering, geoscience and other emerging application areas of Bayesian data assimilation.

Preface

Classical mechanics is built upon the concept of *determinism*. Determinism means that knowledge of the current state of a mechanical system completely determines its future (as well as its past). During the nineteenth century, determinism became a guiding principle for advancing our understanding of natural phenomena, from empirical evidence to first principles and natural laws. In order to formalise the concept of determinism, the French mathematician Pierre Simon Laplace postulated an *intellect* now referred to as *Laplace's demon*:

> We may regard the present state of the universe as the effect of its past and the cause of its future. An intellect which at a certain moment would know all forces that set nature in motion, and all positions of all its items of which nature is composed, if this intellect were also vast enough to submit these data to analysis, it would embrace in a single formula the movements of the greatest bodies of the universe and those of the tiniest atoms; for such an intellect nothing would be uncertain and the future just like the past would be present before its eyes.[1]

Laplace's demon has three properties: (i) exact knowledge of the laws of nature; (ii) complete knowledge of the state of the universe at a particular point in time (of course, Laplace was writing in the days before knowledge of quantum mechanics and relativity); and (iii) the ability to solve any form of mathematical equation exactly. Except for extremely rare cases, none of these three conditions is met in practice. First, mathematical models generally provide a much simplified representation of nature. In the words of the statistician George Box: "All models are wrong, some are useful". Second, reality can only be assessed through measurements which are prone to measurement errors and which can only provide a very limited representation of the current state of nature. Third, most mathematical models cannot be solved analytically; we need to approximate them and then implement their solution on a computer, leading to further errors. At the end of the day, we might end up with a perfectly deterministic piece of computer code with relatively little correspondence to the evolution of the natural phenomena of interest to us.

[1] We have found this quote in the *Very Short Introduction to Chaos* by Smith (2007*b*), which has also stimulated a number of philosophical discussions on imperfect model forecasts, chaos, and data assimilation throughout this book. The original publication is *Essai philosophique sur les probabilités* (1814) by Pierre Simon Laplace.

Despite all these limitations, computational models have proved extremely useful, in producing ever more skilful *weather predictions*, for example. This has been made possible by an iterated process combining *forecasting*, using highly sophisticated computational models, with *analysis* of model outputs using observational data. In other words, we can think of a computational weather prediction code as an extremely complicated and sophisticated device for extrapolating our (limited) knowledge of the present state of the atmosphere into the future. This extrapolation procedure is guided by a constant comparison of computer generated forecasts with actual weather conditions as they arrive, leading to subsequent adjustments of the model state in the weather forecasting system. Since both the extrapolation process and the data driven model adjustments are prone to errors which can often be treated as *random*, one is forced to address the implied inherent *forecast uncertainties*. The two main computational tools developed within the meteorology community in order to deal with these uncertainties are *ensemble prediction* and *data assimilation*.

In ensemble prediction, forecast uncertainties are treated mathematically as random variables; instead of just producing a single forecast, ensemble prediction produces large sets of forecasts which are viewed as realisations of these random variables. This has become a major tool for quantifying uncertainty in forecasts, and is a major theme in this book. Meanwhile, the term *data assimilation* was coined in the computational geoscience community to describe methodologies for improving forecasting skill by combining measured data with computer generated forecasts. More specifically, data assimilation algorithms meld computational models with sets of observations in order to, for example, reduce uncertainties in the model forecasts or to adjust model parameters. Since all models are approximate and all data sets are partial snapshots of nature and are limited by measurement errors, the purpose of data assimilation is to provide estimates that are better than those obtained by using either computational models or observational data alone. While meteorology has served as a stimulus for many current data assimilation algorithms, the subject of uncertainty quantification and data assimilation has found widespread applications ranging from cognitive science to engineering.

This book focuses on the *Bayesian* approach to data assimilation and gives an overview of the subject by fleshing out key ideas and concepts, as well as explaining how to implement specific data assimilation algorithms. Instead of focusing on particular application areas, we adopt a general dynamical systems approach. More to the point, the book brings together two major strands of data assimilation: on the one hand, algorithms based on *Kalman's formulas* for Gaussian distributions together with their extension to nonlinear systems; and on the other, *sequential Monte Carlo methods* (also called *particle filters*). The common feature of all of these algorithms is that they use *ensemble prediction* to represent forecast uncertainties. Our discussion of ensemble-based data assimilation algorithms relies heavily on the *McKean approach* to filtering and the concept of *coupling of measures*, a well-established subject in probability which has not yet found widespread applications to Bayesian inference and data assimilation. Furthermore, while data assimilation can formally be treated as a special instance of the mathematical subject of filtering and smoothing, applications from the geosciences have highlighted that data assimilation algorithms are needed for very high-dimensional and highly nonlinear scientific models where the classical large sample size limits of statistics cannot be obtained in practice. Finally, in contrast with the assumptions of the *perfect model scenario* (which are central to most of mathematical filtering theory), applications from geoscience and other areas require data assimilation algorithms which can cope with systematic model errors. Hence robustness of data assimilation algorithms under finite ensemble/sample sizes and systematic model errors becomes of crucial importance. These aspects will also be discussed in this book.

It should have become clear by now that understanding data assimilation algorithms and quantification of uncertainty requires a broad array of mathematical tools. Therefore, the material in this book has to build upon a multidisciplinary approach synthesising topics from analysis, statistics, probability, and scientific computing. To cope with this demand we have divided the book into two parts. While most of the necessary mathematical background material on uncertainty quantification and probabilistic forecasting is summarised in Part I, Part II is entirely devoted to data assimilation algorithms. As well as classical data assimilation algorithms such as the Kalman filter, variational techniques, and sequential Monte Carlo methods, the book also covers newer developments such as the ensemble Kalman filter and ensemble transform filters. The *McKean approach* to sequential filtering in combination with coupling of measures serves as a unifying mathematical framework throughout Part II.

The book is written at an introductory level suitable for graduate students in applied mathematics, computer science, engineering, geoscience and other emerging application areas of Bayesian data assimilation. Although some familiarity with random variables and dynamical systems is helpful, necessary mathematical concepts are introduced when they are required. A large number of numerical experiments are provided to help to illustrate theoretical findings; these are mostly presented in a semi-rigorous manner. Matlab code for many of these is available via the book's webpage. Since we focus on ideas and concepts, we avoid proofs of technical mathematical aspects such as existence, convergence etc.; in particular, this is achieved by concentrating on finite-dimensional discrete time processes where results can be sketched out using finite difference techniques, avoiding discussion of Itô integrals, for example. Some more technical aspects are collected in appendices at the end of each chapter, together with descriptions of alternative algorithms that are useful but not key to the main story. At the end of each chapter we also provide exercises, together with a brief guide to related literature.

With probabilistic forecasting and data assimilation representing such rich and diverse fields, it is unavoidable that the authors had to make choices about the material to include in the book. In particular, it was necessary to omit many interesting recent developments in *uncertainty quantification* which are covered by Smith (2014). A very approachable introduction to data assimilation is provided by Tarantola (2005). In order to gain a broader mathematical perspective, the reader is referred to the monograph by Jazwinski (1970), which still provides an excellent introduction to the mathematical foundation of filtering and smoothing. A recent, in-depth mathematical account of filtering is given by Bain & Crisan (2009). The monograph by del Moral (2004) provides a very general mathematical framework for the filtering problem within the setting of Feynman–Kac formulas and their McKean models. Theoretical and practical aspects of sequential Monte Carlo methods and particle filters can, for example, be found in Doucet, de Freitas & Gordon (2001). The popular family of *ensemble Kalman filters* is covered by Evensen (2006). The monograph by Majda & Harlim (2012) develops further extensions of the classic Kalman filter to imperfect models in the context of turbulent flows. We also mention the excellent monograph on optimal transportation and coupling of measures by Villani (2003).

We would like to thank: our colleagues Uri Ascher, Gilles Blanchard, Jochen Bröcker, Dan Crisan, Georg Gottwald, Greg Pavliotis, Andrew Stuart and Peter Jan van Leeuwen for the many stimulating discussions centred around various subjects covered in this book; our students Yuan Cheng, Nawinda Chutsagulprom, Maurilio Gutzeit, Tobias Machewitz, Matthias Theves, James Tull, Richard Willis and Alexandra Wolff for their careful reading of earlier drafts of this book; Jason Frank, who provided us with detailed and very valuable feedback; Dan and Kate Daniels who provided childcare whilst much of Colin's work on the book was taking place; and David Tranah from Cambridge University Press who provided guidance throughout the whole process of writing this book.

Finally, we would like to thank our families: Winnie, Kasimir, Nepomuk, Rebecca, Matilda and Evan, for their patience and encouragement.

Contents

1 Prologue: how to produce forecasts

This chapter sets out a simplified mathematical framework that allows us to discuss the concept of forecasting and, more generally, prediction. Two key ingredients of prediction are: (i) we have a computational model which we use to simulate the future evolution of the physical process of interest given its current state;[1] and (ii) we have some measurement procedure providing partially observed data on the current and past states of the system. These two ingredients include three different types of error which we need to take into account when making predictions: (i) *precision errors* in our knowledge of the current state of the physical system; (ii) differences between the evolution of the computational model and the physical system, known as *model errors*; and (iii) *measurement errors* in the data that must occur since all measurement procedures are imperfect. Precision and model errors will both lead to a growing divergence between the predicted state and the system state over time, which we attempt to correct with data which have been polluted with measurement errors. This leads to the key question of data assimilation: *how can we best combine the data with the model to minimise the impact of these errors, and obtain predictions (and quantify errors in our predictions) of the past, present and future state of the system?*

1.1 Physical processes and observations

In this book we shall introduce *data assimilation algorithms*, and we shall want to discuss and evaluate their *accuracy* and *performance*. We shall illustrate this by choosing examples where the physical dynamical system can be represented mathematically. This places us in a somewhat artificial situation where we must generate data from some mathematical model and then pretend that we have only observed part of it. However, this will allow us to assess the performance of data assimilation algorithms by comparing our forecasts with the "true evolution" of the system. Once we have demonstrated the performance of such algorithms in this setting, we are ready to apply them to actual data assimilation problems

[1] It is often the case, in ocean modelling for example, that only partial observations are available and it is already challenging to predict the *current state* of the system (nowcasting). It is also often useful to reconstruct past events when more data become available (hindcasting).

where the true system state is unknown. This methodology is standard in the data assimilation community.

We shall use the term *surrogate physical process* to describe the model that we use to generate the true physical dynamical system trajectory for the purpose of these investigations. Since we are building the surrogate physical process purely to test out data assimilation algorithms, we are completely free to choose a model for this. To challenge these algorithms, the surrogate physical process should exhibit some complex dynamical phenomena. On the other hand, it should allow for numerically reproducible results so that we can make comparisons and compute errors. For example, we could consider a surrogate physical process described in terms of a finite-dimensional state variable $z \in \mathbb{R}^{N_z}$ of dimension $N_z \geq 1$, that has time dependence governed by an *ordinary differential equation* (ODE) of the form

$$\frac{\mathrm{d}z}{\mathrm{d}t} = f(z) + g(t), \quad z(0) = z_0, \tag{1.1}$$

with a chosen vector field $f : \mathbb{R}^{N_z} \to \mathbb{R}^{N_z}$ and a time-dependent function $g(t) \in \mathbb{R}^{N_z}$ for $t \geq 0$ such that solutions of (1.1) exist for all $t \geq 0$ and are unique. While such an ODE model can certainly lead to complex dynamic phenomena, such as *chaos*, the results are not easily reproducible since closed form analytic solutions rarely exist. Instead, we choose to replace (1.1) by a *numerical approximation* such as the *forward Euler scheme*

$$z^{n+1} = z^n + \delta t \left(f(z^n) + g(t_n) \right), \qquad t_n = n\,\delta t, \tag{1.2}$$

with iteration index $n \geq 0$, step-size $\delta t > 0$, and initial value $z^0 = z_0$.[2] Usually, (1.2) is used to approximate (1.1). However, here we will choose (1.2) to be our actual surrogate physical process with some specified value of δt (chosen sufficiently small for stability). This is then completely reproducible (assuming exact arithmetic, or a particular choice of rounding mode) since there is an explicit formula to obtain the sequence z^0, z^1, z^2, etc.

We shall often want to discuss time-continuous systems, and therefore we choose to use linear interpolation in between discrete time points t_n and t_{n+1},

$$z(t) = z^n + (t - t_n)\frac{z^{n+1} - z^n}{\delta t}, \quad t \in [t_n, t_{n+1}], \tag{1.3}$$

to obtain a completely reproducible time-continuous representation of a surrogate physical process. In other words, once the vector field f, together with the step-size δt, the initial condition z_0, and the forcing $\{g(t_n)\}_{n\geq 0}$, have been specified in (1.2), a unique function $z(t)$ can be obtained for $t \geq 0$, which we will denote by $z_{\mathrm{ref}}(t)$ for the rest of this chapter. It should be emphasised at this point that we need to pretend that $z_{\mathrm{ref}}(t)$ is not directly accessible to us during the data assimilation process. Our goal is to estimate $z_{\mathrm{ref}}(t)$ from partial

[2] Throughout this book we use superscript indices to denote a temporal iteration index, for example z^n in (1.2). Such an index should not be confused with the nth power of z. The interpretation of z^n should hopefully be clear from the circumstances of its use.

measurements of $z_{\text{ref}}(t)$, using imperfect mathematical models of the dynamical system. We will return to these issues later in the chapter.

To clarify the setting, we next discuss a specific example for producing surrogate physical processes in the form of a reference solution $z_{\text{ref}}(t)$.

Example 1.1 The *Lorenz-63 model* (Lorenz 1963) has a three-dimensional state variable $z := (\mathrm{x}, \mathrm{y}, \mathrm{z})^{\mathrm{T}} \in \mathbb{R}^{N_z}$, for scalar variables x, y, z, with $N_z = 3$. The variable z satisfies an equation that can be written in the form (1.2) with vector field f given by

$$f(z) := \begin{pmatrix} \sigma(\mathrm{y}-\mathrm{x}) \\ \mathrm{x}(\rho-\mathrm{z})-\mathrm{y} \\ \mathrm{xy}-\beta\mathrm{z} \end{pmatrix}, \tag{1.4}$$

and parameter values $\sigma = 10$, $\rho = 28$, and $\beta = 8/3$. We will use this vector field in the discrete system (1.2) to build a surrogate physical process with step-size $\delta t = 0.001$ and initial conditions

$$\mathrm{x}_0 = -0.587, \quad \mathrm{y}_0 = -0.563, \quad \mathrm{z}_0 = 16.870. \tag{1.5}$$

As we develop this example throughout this chapter, we will discuss model errors, defined as differences between the surrogate physical process and the imperfect model that we will use to make predictions. For that reason we include a non-autonomous forcing term g in (1.2), which will have different definitions in the two models. We shall define the forcing $g(t_n) = g^n = (g_1^n, g_2^n, g_3^n)^{\mathrm{T}} \in \mathbb{R}^3$ for the surrogate physical process as follows: set $a = 1/\sqrt{\delta t}$ and, for $n \geq 0$, define recursively

$$g_i^{n+1} = \begin{cases} 2g_i^n + a/2 & \text{if } g_i^n \in [-a/2, 0), \\ -2g_i^n + a/2 & \text{otherwise,} \end{cases} \tag{1.6}$$

for $i = 1, 2, 3$ with initial values

$$g_1^0 = a(2^{-1/2} - 1/2), \quad g_2^0 = a(3^{-1/2} - 1/2), \quad g_3^0 = a(5^{-1/2} - 1/2).$$

It should be noted that $g_i^n \in [-a/2, a/2]$ for all $n \geq 0$. In order to avoid an undesired accumulation of round-off errors in floating point arithmetic, we need to slightly modify the iteration defined by (1.6). A precise description of the necessary modification can be found in the appendix at the end of this chapter. A reader familiar with examples from the dynamical systems literature might have noticed that the iteration (1.6) reduces to the *tent map iteration* with $a = 1$ and the interval $[-1/2, 1/2]$ shifted to $[0, 1]$. The factor $a > 0$ controls the amplitude of the forcing and the interval has been shifted such that the forcing is centred about zero. We choose this for the surrogate physical process since it is completely reproducible in exact arithmetic, but has very complicated dynamics that can appear random.

The numerical solutions obtained from an application of (1.2) for $n = 0, \ldots, N-1$ with $N = 2 \times 10^5$ lead to a time-continuous reference solution $z_{\text{ref}}(t)$

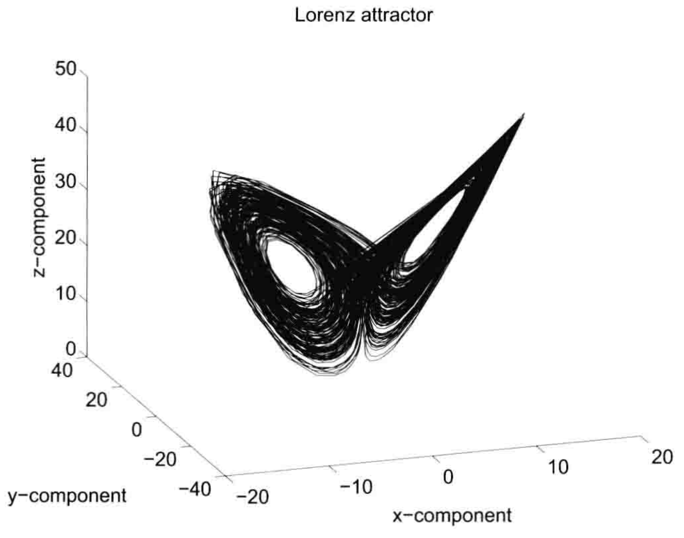

Figure 1.1 Trajectory of the modified Lorenz-63 model as described in Example 1.1. This trajectory provides us with the desired surrogate physical process. The cloud of solution points is part of what is called the model attractor.

according to the interpolation formula (1.3) for time $t \in [0, 200]$, which is used for all experiments conducted in this chapter. See Figure 1.1 for a phase portrait of the time series. Solutions asymptotically fill a subset of phase space $\mathbb{R}^3$ called the model *attractor*.

Next, we turn our attention to the second ingredient in the prediction problem, namely the measurement procedure. In this setting, neither $z_{\text{ref}}(t)$ nor (1.2) will be explicitly available to us. Instead, we will receive "observations" or "measurements" of $z_{\text{ref}}(t)$ at various times, in the form of measured data containing partial information about the underlying physical process, combined with measurement errors. Hence we need to introduce a mathematical framework for describing such partial observations of physical processes through measurements.

We first consider the case of an error-free measurement at a time t, which we describe by a *forward map* (or operator) $h : \mathbb{R}^{N_z} \to \mathbb{R}^{N_y}$

$$y_{\text{obs}}(t) = h(z_{\text{ref}}(t)), \tag{1.7}$$

where we typically have $N_y < N_z$ (corresponding to a partial observation of the system state z_{ref}). For simplicity, we shall only consider $N_y = 1$ in this chapter. Since h is non-invertible, we cannot deduce $z_{\text{ref}}(t)$ from simple inversion, even if the measurements are free from errors.

More realistically, a measurement device will lead to measurement errors, which may arise as the linear superposition of many individual errors $\eta_i \in \mathbb{R}$, $i = 1, \ldots, I$. Based on this assumption, we arrive at a mathematical model of

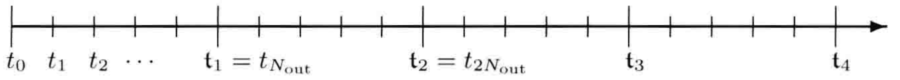

Figure 1.2 Diagram illustrating model timesteps t_0, t_1, etc. and observation times $\mathfrak{t}_1 = t_{N_{\text{out}}}$, $\mathfrak{t}_2$, etc. Here, $N_{\text{out}} = 5$.

type

$$y_{\text{obs}}(t) = h(z_{\text{ref}}(t)) + \sum_{i=1}^{I} \eta_i(t). \tag{1.8}$$

The quality of a measurement is now determined by the magnitude of the individual error terms η_i and the number I of contributing error sources. Measurements will only be taken at discrete points in time, separated by intervals of length $\Delta t_{\text{out}} > 0$. To distinguish the discrete model time $t_n = n\,\delta t$ from instances at which measurements are taken, we use Gothic script to denote measurement points, i.e.,

$$\mathfrak{t}_k = k\,\Delta t_{\text{out}}, \qquad k \geq 1,$$

and $\Delta t_{\text{out}} = \delta t N_{\text{out}}$ for given integer $N_{\text{out}} \geq 1$. This is illustrated in Figure 1.2.

We again consider a specific example to illustrate our "measurement procedure" (1.8).

Example 1.2 We consider the time series generated in Example 1.1 and assume that we can observe the x-component of

$$z_{\text{ref}}(t) = (\text{x}_{\text{ref}}(t), \text{y}_{\text{ref}}(t), \text{z}_{\text{ref}}(t))^{\text{T}} \in \mathbb{R}^3.$$

This leads to a linear forward operator of the form

$$h(z_{\text{ref}}(t)) = \text{x}_{\text{ref}}(t).$$

In this example, we shall use a modified tent map of type (1.6) to model measurement errors. More specifically, we use the iteration

$$\xi_{k+1} = \begin{cases} 2\xi_k + a/2 & \text{if } \xi_k \in [-a/2, 0), \\ -2\xi_k + a/2 & \text{otherwise,} \end{cases} \tag{1.9}$$

with $a = 4$ and starting value $\xi_0 = a(2^{-1/2} - 1/2)$ for $k \geq 0$. From this sequence we store every tenth iterate in an array $\{\Xi_i\}_{i\geq 1}$, i.e.,

$$\Xi_i = \xi_{k=10i}, \qquad i = 1, 2, \ldots\ . \tag{1.10}$$

An observation x_{obs} at time $\mathfrak{t}_1 = \Delta t_{\text{out}} = 0.05$ is now obtained as follows:

$$\text{x}_{\text{obs}}(\mathfrak{t}_1) := \text{x}_{\text{ref}}(\mathfrak{t}_1) + \frac{1}{20}\sum_{i=1}^{20} \Xi_i.$$

This procedure fits into the framework of (1.8) with $I = 20$ and $\eta_i(\mathfrak{t}_1) = \Xi_i/20$, $i = 1, \ldots, 20$.

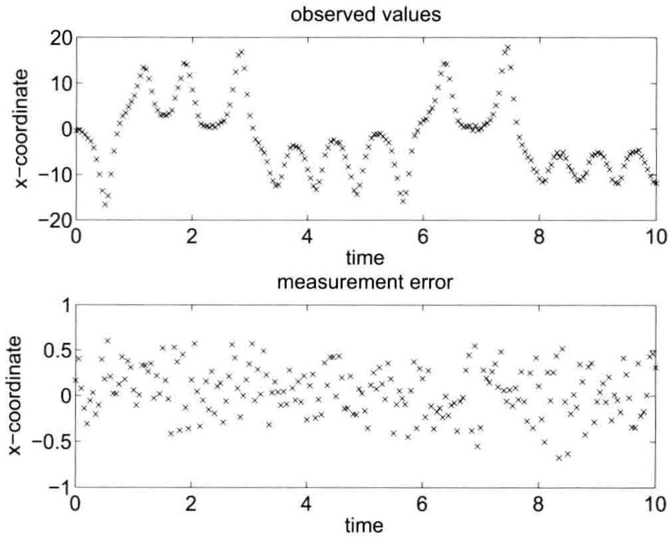

Figure 1.3 Observed values for the x-component and their measurement errors over the time interval $[0, 10]$ with observations taken every $\Delta t_{\text{out}} = 0.05$ time units.

For the next observation at $\mathrm{t}_2 = 2\Delta t_{\text{out}} = 0.1$ we use

$$\mathrm{x}_{\text{obs}}(\mathrm{t}_2) = \mathrm{x}_{\text{ref}}(\mathrm{t}_2) + \frac{1}{20}\sum_{i=21}^{40} \Xi_i,$$

and this process is repeated for all available data points from the reference trajectory generated in Example 1.1. Numerical results are displayed for the first 200 data points in Figure 1.3. Our procedure of defining the measurement errors might appear unduly complicated, but we will find later in Chapter 2 that it mimics important aspects of typical measurement errors. In particular, the measurement errors can be treated as *random* even though a perfectly deterministic procedure has defined them.

1.2 Data driven forecasting

We now assume that N_{obs} scalar observations $y_{\text{obs}}(\mathrm{t}_k) \in \mathbb{R}$ at $\mathrm{t}_k = k\,\Delta t_{\text{out}}$, $k = 1, 2, \ldots, N_{\text{obs}}$, have been made at time intervals of Δt_{out}. To define what we understand by a *forecast* or a *prediction*, we select a point in time t_{k_*} that we denote the *present*. Relative to t_{k_*}, we can define the *past* $t < \mathrm{t}_{k_*}$ and the *future* $t > \mathrm{t}_{k_*}$. A possible forecasting (or prediction) problem would be to produce an *estimate* for

$$y_{\text{ref}}(t) := h(z_{\text{ref}}(t))$$