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 $z_{\rm ref}(t)$, 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 $z_{\rm ref}(t)$.

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 $\mathrm{x}, \mathrm{y}, \mathrm{z}$, 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 $z_{\rm ref}(t)$

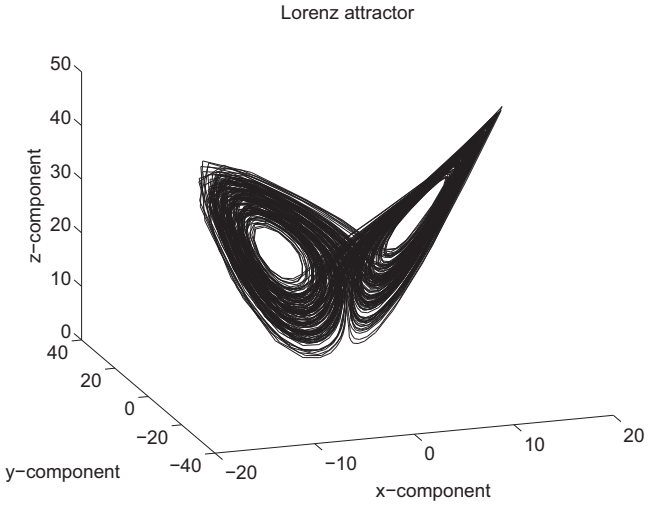

Figure 1.1 Trajectory of the modified Lorenz-63 model as described in Example 1.1. This trajectory provides us with the desired surrogate physical process. The cloud of solution points is part of what is called the model attractor.

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 $z_{\rm ref}(t)$ nor (1.2) will be explicitly available to us. Instead, we will receive "observations" or "measurements" of $z_{\rm ref}(t)$ 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) $h : \mathbb{R}^{N_z} \to \mathbb{R}^{N_y}$

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

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

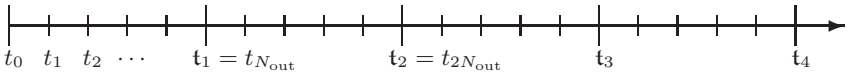

Figure 1.2 Diagram illustrating model timesteps t_0, t_1, etc. and observation times $\mathfrak{t}_1 = t_{N_{\rm out}}$, $\mathfrak{t}_2$, etc. Here, $N_{\rm out} = 5$.

type

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

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 $\Delta t_{\rm out} > 0$. 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.,

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

and $\Delta t_{\rm out} = \delta t N_{\rm out}$ for given integer $N_{\rm out} \geq 1$. 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

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

This leads to a linear forward operator of the form

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

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 $\mathrm{x}_{\rm obs}$ at time $\mathfrak{t}_1 = \Delta t_{\rm out} = 0.05$ is now obtained as follows:

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

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

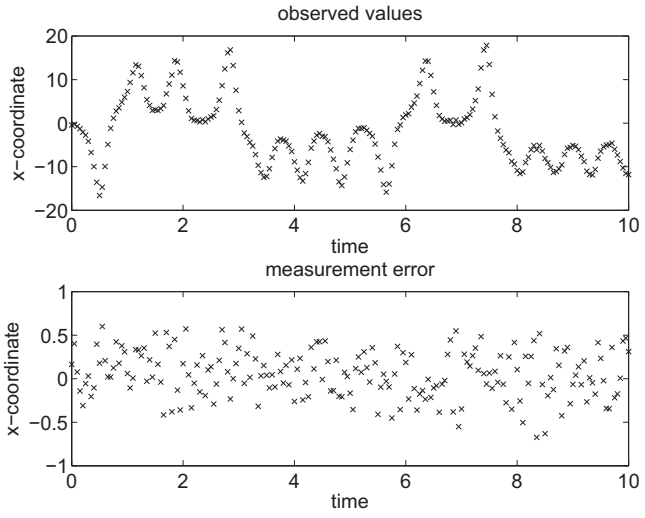

Figure 1.3 Observed values for the x-component and their measurement errors over the time interval $[0, 10]$ with observations taken every $\Delta t_{\rm out} = 0.05$ time units.

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

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

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 $N_{\rm obs}$ scalar observations $y_{\rm obs}(\mathrm{t}_k) \in \mathbb{R}$ at $\mathrm{t}_k = k\,\Delta t_{\rm out}$, $k = 1, 2, \ldots, N_{\rm obs}$, have been made at time intervals of $\Delta t_{\rm out}$. 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

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

with $t > \mathrm{t}_{k_*}$ and only observations from the past and present available. Such statements can be *verified* as soon as a future moment becomes the present and a new measurement becomes available. More generally, we would, of course, like to make predictions about the complete surrogate process $z_{\rm ref}(t)$ for $t > \mathrm{t}_{k_*}$ and not only about the quantity we can actually observe. We will come back to this more challenging task later in this chapter.

Returning to the problem of predicting future observations, we first utilise the concept of *polynomial interpolation*. Recall that there is a unique polynomial

$$q(t) = b_0 + b_1 t + b_2 t^2 + \cdots + b_p t^p \tag{1.11}$$

of order p with coefficients b_l through any $p+1$ data points. We would like to find a polynomial that interpolates observations at $p+1$ present and past observation times $\{\mathrm{t}_{k_*}, \mathrm{t}_{k_*-1}, \ldots, \mathrm{t}_{k_*-p}\}$ with the aim of using it to predict future observations. This leads to the interpolation conditions

$$q(\mathrm{t}_k) = y_{\rm obs}(\mathrm{t}_k), \qquad \mathrm{t}_k \in \{\mathrm{t}_{k_*}, \mathrm{t}_{k_*-1}, \ldots, \mathrm{t}_{k_*-p}\},$$

which determine the $p+1$ coefficients b_l in (1.11) uniquely. A predicted observation at $t > \mathrm{t}_{k_*}$ is then simply provided by $q(t)$. Since t is outside the interval of the observed data points, the prediction is an *extrapolation* from the data. For the linear case $p = 1$ we obtain

$$\begin{aligned} q(t) &= y_{\rm obs}(\mathrm{t}_{k_*}) + (t - \mathrm{t}_{k_*}) \frac{y_{\rm obs}(\mathrm{t}_{k_*}) - y_{\rm obs}(\mathrm{t}_{k_*-1})}{\mathrm{t}_{k_*} - \mathrm{t}_{k_*-1}} \\ &= y_{\rm obs}(\mathrm{t}_{k_*}) + (t - \mathrm{t}_{k_*}) \frac{y_{\rm obs}(\mathrm{t}_{k_*}) - y_{\rm obs}(\mathrm{t}_{k_*} - \Delta t_{\rm out})}{\Delta t_{\rm out}}. \end{aligned}$$

Upon setting $t = \mathrm{t}_{k_*+1}$ we obtain the extrapolation formula

$$y_{\rm predict}(\mathrm{t}_{k_*+1}) := q(\mathrm{t}_{k_*+1}) = 2 y_{\rm obs}(\mathrm{t}_{k_*}) - y_{\rm obs}(\mathrm{t}_{k_*-1}). \tag{1.12}$$

As soon as $y_{\rm obs}(\mathrm{t}_{k_*+1})$ becomes available, we can compare this prediction with the observed value. Furthermore, we can use this new observation point (and discard the oldest one from t_{k_*-1}) and a correspondingly updated linear extrapolation formula to obtain $y_{\rm predict}(\mathrm{t}_{k_*+2})$. This can be iterated over several time intervals, repeatedly using data to predict the new observation. To assess the accuracy of this procedure we introduce the following measure.

Definition 1.3 (Root mean square error) For a set of predictions and observations at times $\{\mathrm{t}_1, \mathrm{t}_2, \ldots, \mathrm{t}_N\}$ the root mean square error (RMSE) is given by

$$\text{time averaged RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^{N} |y_{\rm obs}(\mathrm{t}_k) - y_{\rm predict}(\mathrm{t}_k)|^2}. \tag{1.13}$$

In the case of linear interpolation, if there are $N_{\rm obs}$ observations then $N = N_{\rm obs} - 2$ since we cannot make predictions using linear interpolation for the first two observations.

We illustrate the linear interpolation prediction strategy by our next example.

Example 1.4 We utilise the observations generated in Example 1.2 for the first solution component of the Lorenz-63 system, i.e., $y_{\rm obs}(\mathrm{t}_k) = \mathrm{x}_{\rm obs}(\mathrm{t}_k)$. Recall that the observation interval is $\Delta t_{\rm out} = 0.05$. We set the first t_{k_*} equal to $\mathrm{t}_{k_*} = 100$, and make a total of 2000 verifiable predictions until we reach $t = 200$. The linear extrapolation formula (1.12) is used for making predictions of observations, and the quality of these predictions is assessed using the time averaged RMSE (1.13) with $N = 2000$. A snapshot of the computational results over a short time-window can be found in Figure 1.4. The time averaged RMSE over the whole interval is approximately 1.2951.

It is usually desirable to "extend the prediction window" by making predictions further into the future. In view of this, we modify the procedure so that at each time t_{k_*}, we attempt to predict the observation at time t_{k_*+2} instead of t_{k_*+1}. The associated linear extrapolation formula becomes

$$y_{\rm predict}(\mathrm{t}_{k_*+2}) := q(\mathrm{t}_{k_*+2}) = 3y_{\rm obs}(\mathrm{t}_{k_*}) - 2y_{\rm obs}(\mathrm{t}_{k_*-1}).$$

The results can also be found in Figure 1.4; the quality of the predictions is clearly worse over this larger window. This is confirmed by the time averaged RMSE which increases to approximately 3.3654.

The results of Example 1.4 show that linear interpolation does not provide good predictions over longer times. This suggests the accuracy of forecasts can be improved by extending the extrapolation formula (1.12) to use a linear combination of the present data point plus several previous data points of the form

$$y_{\rm predict}(\mathrm{t}_{k_*+1}) = \sum_{l=0}^{p} a_l \, y_{\rm obs}(\mathrm{t}_{k_*-l}). \tag{1.14}$$

We have already seen that linear extrapolation fits into this framework with $p = 1$ and coefficients $a_0 = 2$, $a_1 = -1$. We recall that the linear extrapolation formula (1.12) was based on first deriving the linear interpolation formula. Hence, as a first attempt at deriving coefficients a_l for (1.14) with $p > 1$, we shall use higher-order interpolation formulas. Interpolation formulas of order p can be conveniently based on the Lagrange polynomials (Süli & Mayers 2006) of order p

$$l_j(t) = \frac{\prod_{i\neq j}(t - \mathrm{t}_i)}{\prod_{i\neq j}(\mathrm{t}_j - \mathrm{t}_i)},$$

where the indices i and j run over the integers

$$\{k_*, k_* - 1, \ldots, k_* - p\}.$$

These polynomials have the useful property that

$$l_j(\mathrm{t}_i) = \begin{cases} 1 & \text{if } j = i, \\ 0 & \text{otherwise,} \end{cases}$$

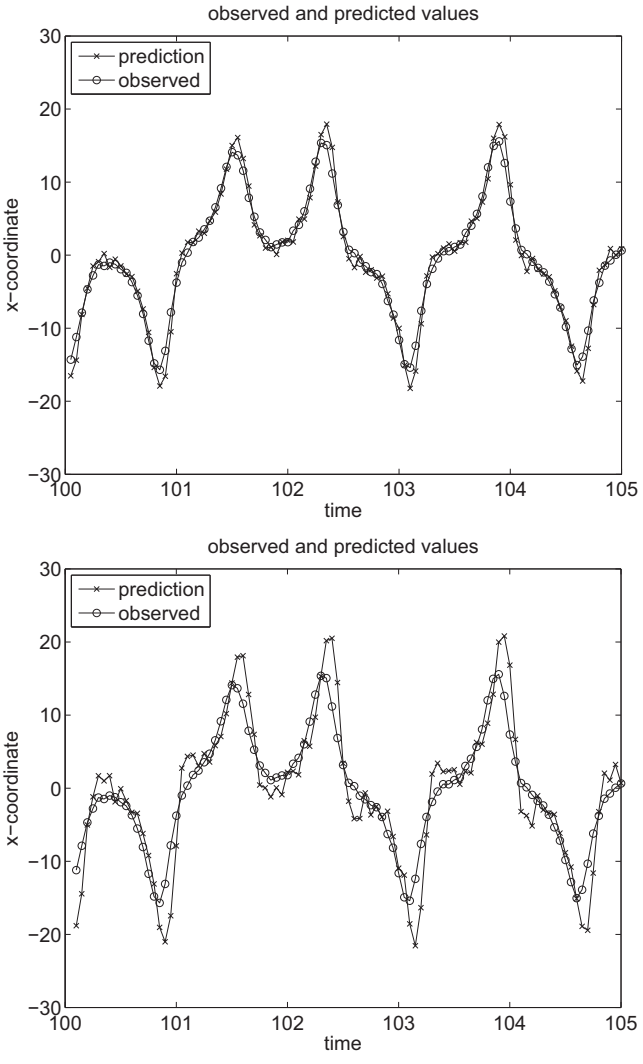

Figure 1.4 Observed values for the x-component and its predicted values using linear extrapolation. The figure at the top shows the results from linear extrapolation over a single observation interval $\Delta t_{\rm out} = 0.05$, while the figure beneath shows results when doubling the prediction interval to 0.1 time units.

which leads to the interpolation formula

$$q(t) = l_{k_*}(t)\, y_{\rm obs}(\mathrm{t}_{k_*}) + l_{k_*-1}(t)\, y_{\rm obs}(\mathrm{t}_{k_*-1}) + \cdots + l_{k_*-p}(t)\, y_{\rm obs}(\mathrm{t}_{k_*-p}). \quad (1.15)$$

The coefficients a_l in (1.14) are obtained by setting $t = \mathrm{t}_{k_*+1}$ in (1.15), i.e.

$$a_l = l_{k_*-l}(\mathrm{t}_{k_*+1}), \qquad l = 0, 1, \ldots, p.$$

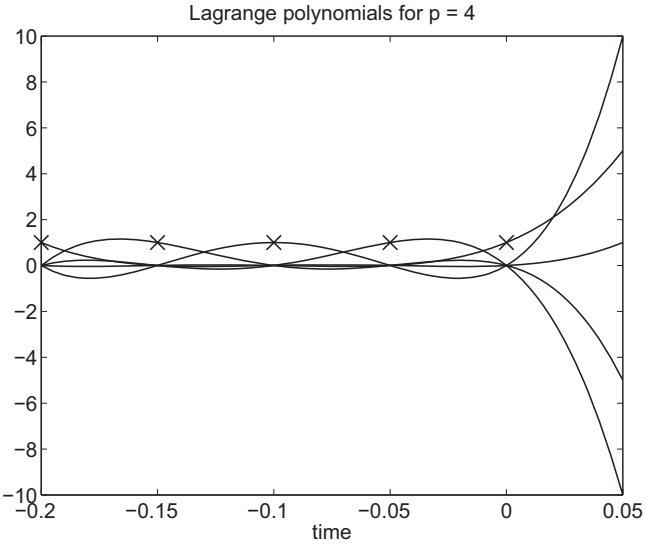

Figure 1.5 Lagrange polynomials $l_j(t)$ of order four corresponding to observations at $t_i = 0, -0.05, -0.1, -0.15, -0.2$. The coefficients a_l in (1.14) are equal to the values of the Lagrangian polynomials at $t = 0.05$. Crosses mark the points where each polynomial takes the value one. Note that the other polynomials are zero at those interpolation points, and note the steep increase in magnitude outside the interpolation interval $t \in [-0.2, 0]$.

Example 1.5 We consider extrapolation based on polynomial interpolation of order $p = 4$. The associated extrapolation coefficients in (1.14) are

$$a_0 = 5,\ a_1 = -10,\ a_2 = 10,\ a_3 = -5,\ a_4 = 1,$$

and the associated Lagrange polynomials are shown in Figure 1.5, taking $\mathfrak{t}_{k_*} = 0$ for simplicity. The values of the extrapolation coefficients can be obtained by inspecting the intersection of the Lagrange polynomials with the vertical line at $t = 0.05$.

The results of applying the fourth-order extrapolation formula to the data set from Example 1.2 are shown in Figure 1.6; the time averaged RMSE was 4.2707. This error is much larger than that observed for linear extrapolation (compare Example 1.4). The reason for this discrepancy can be found in the strong separation of the Lagrange polynomials outside the interpolation interval (compare Figure 1.5), which results in relatively large coefficients a_l in (1.14). Hence even relatively small measurement errors can be severely amplified and do not necessarily cancel out. This effect becomes even more pronounced when the prediction interval is doubled to $2\Delta t_{\text{out}}$. The associated extrapolation coefficients are now given by

$$a_0 = 15,\ a_1 = -40,\ a_2 = 45,\ a_3 = -24,\ a_4 = 5.$$