On shadowing methods for data assimilation

Printing: Ridderprint, www.ridderprint.nl
Cover: Ridderprint, www.ridderprint.nl

On shadowing methods for data assimilation

Over schaduwingsmethoden voor data-assimilatie

(met een samenvatting in het Nederlands)

Proefschrift

ter verkrijging van de graad van doctor
aan de Universiteit Utrecht
op gezag van de rector magnificus, prof. dr. H.R.B.M. Kummeling,
ingevolge het besluit van het college voor promoties
in het openbaar te verdedigen op
woensdag 22 december 2021
des middags te 12.15 uur

door

Bart Maarten de Leeuw

geboren 2 april 1990
te Haarlem

Promotor:	Prof. dr. ir. J. E. Frank
Copromotor:	dr. S. B. Dubinkina

Contents

Dankwoord

Het dankwoord is wellicht het meest gelezen onderdeel van een proefschrift en is zeker het leukste onderdeel om te schrijven. Hoewel het maken van een lijst met mensen die ik wil bedanken het gevaar van omissies inhoudt, ga ik deze lijst toch maken. Echter, degenenen die per abuis niet genoemd zijn, bedankt!

Ik wil mijn promotor Jason en copromotor Svetlana bedanken. Tijdens mijn werk aan de promotie heb ik veelvuldig met Jason en met Svetlana overlegd. Dit waren prettige overleggen. Hun belangstelling voor mij en mijn onderzoek en hun goede adviezen hebben sterk bijgedragen aan hoe dit proefschrift uiteindelijk geworden is en de hoofdstukken van dit proefschrift zijn ook grotendeels een gezamelijk werk met Svetlana. Ik wil zowel Jason als Svetlana hartelijk danken voor hun oprechte interesse en deskundige en persoonlijke begeleiding.

Verder wil ik Erik bedanken. Tijdens een bezoek aan Utrecht heeft hij me op schaduwing gewezen. Hoofdstuk 2 is in samenwerking met onder andere Erik geschreven, en andere hoofdstukken borduren hierop voort. Tijdens mijn werk bij het CWI mocht ik een aantal keer naar het buitenland reizen. Met name het bezoek aan Reading wil ik noemen en daarvoor wil ik Amos, Nancy en Peter Jan hartelijk danken. Ook ben ik uitgenodigd in Cottbus. Markus en Carsten, bedankt. De buitenlandse bezoeken omvatten ook een aantal conferenties. Hierbij wil ik met name twee Oberwolfach Seminars, twee edities van SciCADE, een EnKF workshop en SIAM-MPE noemen. Het is fijn dat deze conferenties georganiseerd zijn en het programma en de informele discussies waren nuttig. Binnen Nederland heb ik deelgenomen aan activiteiten van de Werkgemeenschap Scientific Computing en NDNS+. Dergelijke nationale activiteiten waren goed voor het opbouwen en onderhouden van een netwerk van wiskundigen.

Uiteraard gaat mijn dank ook uit naar mijn collega's bij het CWI. Met Sangeetika en Ki-Wai deelde ik een kantoortje. We hebben vele interessante

gesprekken over werk en andere zaken gehad. Ook de andere leden van de Scientific Computing groep wil ik bedanken. Ik denk hierbij onder andere aan Anne, Benjamin, Daan, Debarati, Fredrik, Jurriaan, Kees, Keith, Krzysztof, Laurent, Nada, Prashant, Yous en Willem-Jan. Verder wil ik de collega's bij het Mathematisch instituut bedanken, met name Rob, Tristan, Nick, Ties, Inan, Han, Ajinkya en uiteraard Felix en Anna. Hartelijk dank allemaal!

Ik heb jarenlang iedere week bij Schaakclub Woerden gespeeld. Daar ging ik iedere week met veel plezier heen, niet alleen voor het schaken, maar ook voor de gezelligheid. Daarvoor bedank ik onder andere Marcel, Marcel, Jan, Jan, Mark, Wick, Wim, Pia en Aldo. Verder wil ik mijn goede vriend Hans bedanken. We zijn al vanuit onze jeugd bevriend, en daar ben ik erg blij mee. Ook dank ik Leihua en zijn vrouw Jun voor hun gastvrijheid en vriendelijkheid. Uiteraard wil ik ook mijn familie voor hun belangstelling en morele steun bedanken. Dank aan al mijn neven en nichten, mijn ooms en tantes, mijn oma, mijn broer Simon en zus Jacqueline, mijn vader en Joan en mijn moeder en Karel. De aanmoedigingen van mijn familie om toch door te gaan met mijn promotietraject heb ik zeer gewaardeerd. Rest mij nog om mijn vrouw Xuewen te bedanken. Ik dank Xuewen voor alle fijne momenten samen, dat ze altijd achter me stond en dat ze me heeft geholpen tot de laatste loodjes het promotietraject door te zetten. Xuewen, ik ben heel gelukkig dat we elkaar hebben.

CHAPTER 1

Introduction

1.1 Data assimilation

Data assimilation methods combine orbits from a dynamical system model with observations to improve an estimate of the state of a physical system. Data assimilation is particularly relevant for time-dependent systems whose state cannot be directly and accurately observed. For example, the number of variables may be too large to observe a full state, or it may be possible to observe a full state, but the observations may be highly inaccurate. If the underlying time-dependent system is chaotic, an additional challenge is that even small errors in the state estimate increase over time as the model evolves. Systems to which the application of data assimilation is appropriate arise in many practical situations, such as weather forecasting, oceanography and subsurface modelling [59, 38, 77].

Such problems arise in the above-mentioned applications, because in said applications the physical systems that are being modelled exhibit chaotic behaviour. For example, fluid dynamics may be needed to be modelled. For the purpose of forecasting a chaotic system, it is particularly important to first obtain a high-quality nowcast, because errors in the nowcast tend to exponentially increase under the model dynamics used for forecasting. Another reason for the need for data assimilation is that the model dynamics may reside on a chaotic attractor. However, observational error tends to be random and may be inconsistent with the dynamic structure of the model.

Without data assimilation, the nowcast may be away from the attractor and the short-time dynamics would be for the system to return to the attractor. This then leads to an inaccurate forecast. So, it is important to perform data assimilation. This should lead to a nowcast that is accurate and consistent with model dynamics, in order to be able to perform an accurate forecast.

In this thesis we study problems for which both the dynamical system model and observations are discrete in time. We shall study deterministic models of the form

$$x_{n+1} = F_n(x_n), \quad x_n \in \mathcal{R}^d, \quad n = 0, \ldots, N-1, \tag{1.1}$$

where $F_n : \mathcal{R}^d \to \mathcal{R}^d$. We assume the model to be $\mathcal{C}^3$ for all n. In many applications the model is defined by the time-discretization of an ordinary differential equation (ODE)

$$\dot{x} = f(t, x), \quad x(t) \in \mathcal{R}^d, \tag{1.2}$$

which in turn may be defined as the space-discretization of a partial differential equation (or system of PDEs).

In practice, no models are perfect and all models have a model error. In this thesis, we study data assimilation algorithms under a few different assumptions on model error. In chapters 2 and 3 we assume there is no model error. In chapter 4 consider a model error that may be represented by adding stochastic terms to (1.1), as further explained below. In chapter 5 we consider a structural model error.

A model error that may be represented by adding stochastic terms to (1.1) gives rise to dynamic systems of the form

$$x_{n+1} = F_n(x_n) + \sqrt{\tau}\Sigma_m \eta_n, \quad x_n \in \mathcal{R}^d, \quad n = 0, \ldots, N-1. \tag{1.3}$$

In (1.3), we assume $\tau > 0$ and the symmetric positive definite $\Sigma_m \in \mathcal{R}^{d\times d}$ are given. We define the model covariance matrix $C_m := \tau\Sigma_m^2$. We assume the $\eta_n \in \mathcal{R}^d$ are independent and identically distributed standard normal random variables. If F_n is defined as the forward Euler discretisation of (1.2), then (1.3) is the Euler-Maruyama discretisation of the Îto stochastic differential equation

$$\dot{x} = f(t, x) + \Sigma_m \dot{W}, \quad x(t) \in \mathcal{R}^d, \tag{1.4}$$

where $W(t) \in \mathcal{R}^d$ denotes standard Brownian motion. Please note that, in the special case that $\Sigma = 0$, (1.4) reduces to (1.2) and (1.3) reduces to (1.1).

Let the sequence $\boldsymbol{X} := \{\mathcal{X}_0, \ldots \mathcal{X}_N\}$ be some orbit of (1.3), referred to as the *true solution* of the model, and presumed to be unknown. Suppose we are given a sequence of noisy observations $\boldsymbol{y} := \{y_0, \ldots y_N\}$ related to $\boldsymbol{X}$ via

$$y_n = H(\mathcal{X}_n) + \xi_n, \qquad y_n \in \mathbb{R}^b, \quad n = 0, \ldots, N, \tag{1.5}$$

where $H : \mathcal{R}^d \to \mathcal{R}^b$, $b \leq d$, is the observation operator, and the noise variables ξ_n are drawn from a normal distribution $\xi_n \sim \mathcal{N}(0, C_o)$ with zero mean and known observational error covariance matrix C_o. That is, the statistical distribution of errors is known.

The condition $b \leq d$ indicates that there are less observations than state variables. Throughout this thesis, we shall assume that $b \leq d$. It is of course possible that $b > d$, in which case it may be possible to de-noise the observations at each time step, before performing data assimilation, in particular if H is linear and of rank $\geq d$. In that way, a full, but noisy, state estimate could be obtained at each time step. So, the case where $b > d$ may be similar to the case where H is an identity operator. If $b > d$ and H is linear and of rank $\geq d$, the error of the de-noised observations is still Gaussian distributed with zero mean and known covariance matrix.

Define the residual functional

$$G(\boldsymbol{u}) = \begin{pmatrix} G_0(\boldsymbol{u}) \\ G_1(\boldsymbol{u}) \\ \vdots \\ G_{N-1}(\boldsymbol{u}) \end{pmatrix}, \qquad G_n(\boldsymbol{u}) = u_{n+1} - F_n(u_n), \quad n = 0, \ldots, N-1, \tag{1.6}$$

where $\boldsymbol{u} = \{u_0, u_1, \ldots, u_N\}, \quad u_n \in \mathcal{R}^d$. Then $G(\boldsymbol{u}) = 0$ if and only if $\boldsymbol{u}$ is an orbit of (1.1). If $\|G_n(\boldsymbol{u})\| < \varepsilon$, $n = 0, \ldots, N-1$, where $\|\cdot\|$ is a norm in $\mathcal{R}^d$, $\boldsymbol{u}$ is called an ε-orbit. Regarding notation, please note that we have used $\boldsymbol{x}$ to denote any orbit of (1.1), $\mathcal{X}$ to denote a particular orbit of (1.1) (the truth) and $\boldsymbol{u}$ to denote any vector in $\mathcal{R}^{d(N+1)}$, that may denote an orbit, pseudo-orbit or any other sequence of vectors in $\mathcal{R}^d$.

Data assimilation is the problem of finding a sequence $\boldsymbol{u}$, such that the observational mismatch $\|y_n - Hu_n\|$, $n = 0, \ldots, N$ and the residual $G(\boldsymbol{u})$ are small in an appropriately defined sense [91, 52, 63].

Data assimilation methods that impose the constraint $\|G(\boldsymbol{u})\| = 0$ are refered to as *strong constraint* data assimilation methods. Data assimilation methods for which the residual may be non-zero are called *weak constraint* data assimilation methods. In case that $\Sigma = 0$, it holds per definition that $G(\mathcal{X}) = 0$. Therefore, strong constraint data assimilation methods are most suitable when one does not consider model error in (1.1) (i.e. $\Sigma_m = 0$), while weak constraint data assimilation methods are more suitable for the imperfect model scenario [97, 63].

1.1.1 Linear models with Gaussian noise: Kalman filters

The classical Kalman filter is an example of a data assimilation algorithm [58]. In the special case that F_n and H are linear and the true initial condition $\mathcal{X}_0$

is distributed as $\mathcal{N}(x_0, C_b)$, the Kalman filter can be shown to yield an optimal solution to the data assimilation problem. The Kalman filter can be derived using a Bayesian approach.

In this case the initial condition, as well as all other uncertain terms, are distributed according to Gaussian distributions and the model is linear. Therefore, it can be derived that, conditioned on $n-1$, the unknown $\mathcal{X}_n$ is normally distributed for any n. The goal is then to find these distributions. Since the distributions are normal, it is sufficient to find the first two moments, or equivalently, the mean and covariance.

The mean and covariance of the Kalman filter can be expressed using a "forecasting" step and an "analysis" step [58, 38, 63]. Suppose that at discrete time n, the distribution for the state estimate is given by $\mathcal{N}(x_n, C_{b,n})$. For notational convenience, we assume that $F_n(u_n) = F_n u_n$ and $H(u_n) = Hu_n$, that is, we denote the linear operators as matrices with the same notation as the operators. This amounts to assuming there are no constant drift terms in F_n and H and does not result in an essential loss of generality. Then, the forecasting step is given by

$$x^f_{n+1} = F_n(x_n), \tag{1.7}$$

$$C^f_{b,n+1} = F_n C_{b,n} F_n^T + C_m. \tag{1.8}$$

The forecasting step is just the forward propagation of $\mathcal{N}(x_n, C_{b,n})$ by one time step. The observations are taken into account in the next step, which is referred to as the analysis step.

Let the Kalman gain $K_{n+1} = C^f_{b,n+1} H^T \left(H C^f_{b,n+1} H^T + C_o \right)^{-1}$. The analysis step is given by

$$x_{n+1} = x^f_{n+1} - K_{n+1}(H x^f_{n+1} - y_{n+1}), \tag{1.9}$$

$$C_{b,n+1} = (I - K_{n+1}H)\, C^f_{b,n+1}. \tag{1.10}$$

It is a property of the Kalman filter that the correct distribution of the state estimate at time $n+1$ is given by $\mathcal{N}(x_{n+1}, C_{b,n+1})$. In other words, the Kalman filter solves the Bayesian estimation problem exactly. It should be remarked that it is not always feasible in practice to solve the covariance equation. Also, for non-linear models the covariance equation cannot usually be solved exactly. Therefore, various algorithms for the approximation of the covariance have been developed [38].

Synchronization

As can be seen from (1.7) and (1.9), the state estimate of the Kalman filter is propagated through time as $x_{n+1} = (I - K_{n+1}H)F_n(x_n) + K_{n+1}y_{n+1}$. If we assume that the model F_n is perfect (but not necessarily linear) and that the

observations are not noisy (i.e. $C_o = 0$), then $y_{n+1} = H\mathcal{X}_{n+1} = HF_n\mathcal{X}_n$. The state estimate equation of the Kalman filter can be rewritten as

$$x_{n+1} = (I - K_{n+1}H)F_n(x_n) + K_{n+1}HF_n(\mathcal{X}_n). \tag{1.11}$$

In (1.11), a particularly simple example is if $K_{n+1}H$ is a projection matrix for each n. For example, if the model F_n is the Lorenz '63 model [69], given by

$$\dot{x}_1 = \sigma(x_2 - x_1), \quad \dot{x}_2 = x_1(\rho - x_3) - x_2, \quad \dot{x}_3 = x_1x_2 - \beta x_3 \tag{1.12}$$

where $\sigma = 10$, $\beta = \frac{8}{3}$ and $\rho = 28$. This is a chaotic ODE model. The Lyapunov exponents of the Lorenz attractor are $\lambda_1 \approx 0.906$, $\lambda_2 = 0$, $\lambda_3 \approx -14.572$. If it is discretized with sufficiently small time step, it can be shown that the discretized model shadows the truth [95]. We denote the respective variables at discrete time n by $x_{1,n}, x_{2,n}, x_{3,n}$. If $K_{n+1}H$ projects on $x_{1,n}$ or $x_{2,n}$ for all discrete times n, it was experimentally shown in [79] that $x_n \to \mathcal{X}_n$ at an exponential rate as $n \to \infty$. This algorithm is known as direct insertion synchronization, since one variable in the model is replaced by the same variable from the truth at each time instance, i.e. the truth is directly inserted into the model.

Synchronization of chaos has been studied extensively in the field of systems and control theory [79, 47, 42]. The use of synchronization for data assimilation was first applied in [34]. Because the model and observations are assumed to be perfect, it is possible to describe the truth and assimilation using a combined system of equations,

$$\mathcal{X}_{n+1} = F_n(\mathcal{X}_n), \tag{1.13a}$$

$$x_{n+1} = K_{n+1}H\mathcal{X}_{n+1} + (I - K_{n+1}H)F_n(x_n). \tag{1.13b}$$

The manifold $\mathcal{S} = \{(\mathcal{X}, x) \in \mathcal{R}^d \times \mathcal{R}^d : \mathcal{X} = x\}$ is invariant under these dynamics and is called the *synchronization manifold*. When $\mathcal{S}$ attracts a neighborhood of itself, then for x_0 within the basin of attraction, x_n synchronizes with $\mathcal{X}_n$. The dynamical system (1.13a)-(1.13b) has $2d$ Lyapunov exponents, of which d are the same as the original model Lyapunov exponents (because the d equations for the truth are the original model). The other d Lyapunov exponents are known as the conditional Lyapunov exponents. If x_n synchronizes with $\mathcal{X}_n$, the conditional Lyapunov exponents are all negative [79]. On the other hand, if the conditional Lyapunov exponents are all negative and x_0 is sufficiently close to $\mathcal{X}_0$, it holds under reasonable assumptions that $x_n \to \mathcal{X}_n$ exponentially and the rate is given by the largest conditional Lyapunov exponent [80]. However, whether synchronization is possible also depends on H. For example, if F_n is (a discretization of) the Lorenz '63 model and H projects on the last variable x_3, synchronization is impossible for most initial conditions [79]. It should also be noted that synchronization may be destabilized if there are errors in the model or data.

1.1.2 Variational methods for data assimilation

We now return to the data assimilation problem for which F_n and H are linear and the true initial condition $\mathcal{X}_0$ is distributed as $\mathcal{N}(x_0, C_b)$, and thus in which the Kalman filter is optimal. It can then be derived that the state estimate of the Kalman filter may be rewritten as the solution of a minimization problem,

$$x_{n+1} = \text{argmin}_x \frac{1}{2}\|x - F_n(x_n)\|^2_{C^f_{b,n+1}} + \frac{1}{2}\|Hx - y_{n+1}\|^2_{C_o}, \tag{1.14}$$

wherein the norm $\|x\|_C := \sqrt{x^T C^{-1} x}$, for any vector $x \in \mathcal{R}^d$ and any matrix $C \in \mathcal{R}^{d\times d}$. If the simplifying assumption is made that $C^f_{b,n+1}$ is constant, the resulting minimization problem is known as 3DVar [67, 68, 63]. This method may also be applied if F_n is nonlinear. Because $C^f_{b,n+1}$ is a constant matrix, there is no need for computing it at every step (which is a challenging problem in non-linear systems). Of course, the choice of matrix for $C^f_{b,n+1}$ is of importance for the performance of 3DVar. It is possible to generalize the 3DVar filter to a smoother, known as 4DVar, by accounting for observations at different points in time. This is further detailed below.

Strong constraint 4DVar

Four-dimensional variational data assimilation methods (see e.g.[85, 66, 92, 91] and references therein) are based on minimization of cost functions. These are smoother algorithms that take into account observations over longer time intervals. For example, one may define a minimization problem with a background cost function

$$J^{bg}(\boldsymbol{u}) = \frac{1}{2}\|u_0 - u_b\|^2_{C_b}, \tag{1.15}$$

and an observation cost function

$$J^{obs}(\boldsymbol{u}) = \frac{1}{2}\|H(\boldsymbol{u}) - \boldsymbol{y}\|^2_{C_o}. \tag{1.16}$$

As a convenient abuse of notation, C_o in (1.16) denotes a block diagonal matrix with $N+1$ identical blocks equal to the observational covariance matrix C_o. The vector u_{bg} is a first guess of the state at the beginning of the window and the matrix C_b the associated covariance, which is usually referred to a background matrix. The strong constraint 4DVar problem (s4DVar) is described by [85, 66, 92, 91]

$$\boldsymbol{u}^{s4d} = \text{argmin}_{\{\boldsymbol{u}|G(\boldsymbol{u})=0\}} \left(J^{bg}(\boldsymbol{u}) + J^{obs}(\boldsymbol{u})\right). \tag{1.17}$$

That is, in (1.17) we minimize the sum of background and observational cost functions, under the strong constraint that a model orbit should be satisfied (expressed here using (1.6), as $G(\boldsymbol{u}) = 0$). This means that the optimization

problem may be rewritten as an unconstrained problem depending only on the initial condition u_0. We remark that if N is big enough, the contribution of J^{bg} to (1.17) is negligible compared to the contribution of J^{obs}.

One drawback of s4DVar is that, for chaotic models, the number of local minima of the cost function increases dramatically with N [6, 74, 82]. This places a practical limit on the length of the assimilation window—the time period over which observations may be assimilated.

Weak constraint 4DVar

For the imperfect model scenario, one may use a weak constraint 4DVar algorithm (w4DVar) [85, 97, 39, 18, 36, 63]. In the weak constraint 4DVar problem, an extra term for model error is added to the cost function. Weak constraint 4DVar can be formulated using either the *forcing* or the *state space* formulation. The formulations differ in the choice of variables. Although those formulations are formally equivalent, in practice they lead to different optimization problems [39, 36].

The forcing formulation is defined as

$$J^{for}(\boldsymbol{e}) = \frac{1}{2}\|e\|^2_{C_m}, \tag{1.18}$$

$$\begin{pmatrix}\boldsymbol{u}^{w4f} & \boldsymbol{e}\end{pmatrix}^T = \operatorname{argmin}_{\{\boldsymbol{u},\boldsymbol{e}|G(\boldsymbol{u})=\boldsymbol{e}\}} \left(J^{bg}(\boldsymbol{u}) + J^{obs}(\boldsymbol{u}) + J^{for}(\boldsymbol{e})\right). \tag{1.19}$$

In the forcing formulation, $\boldsymbol{e}$ is a vector of forcings due to unmodelled terms. That is, $\boldsymbol{e}$ is an estimate of the effect of model error on the dynamics of the model. The forcing formulation of w4DVar is still a constrained optimization problem, with the constraint $G(\boldsymbol{u}) = \boldsymbol{e}$. Just like in the case for s4DWar, the forcing formulation of w4DVar can be rewritten as an unconstrained problem depending on u_0 and $\boldsymbol{e}$. This formulation of w4DVar is very similar to s4DVar and also for this problem the length of the assimilation window may be limited by the increase of the number of local minima as N increases. In practical situations, such as weather forecasting, it can be an advantage of the forcing formulation that the size of the problem may be reduced through a low-dimensional approximation of $\boldsymbol{e}$. For example, in some cases $\boldsymbol{e}$ may be assumed to be constant over time, thereby reducing the dimension of the optimization problem from $(N+1)d$ to $2d$ [39, 36].

The state space formulation is defined as

$$J^{mod}(\boldsymbol{u}) = \frac{1}{2}\|G(\boldsymbol{u})\|^2_{C_m}, \tag{1.20}$$

$$\boldsymbol{u}^{w4s} = \operatorname{argmin}_{\boldsymbol{u}} \left(J^{bg}(\boldsymbol{u}) + J^{obs}(\boldsymbol{u}) + J^{mod}(\boldsymbol{u})\right). \tag{1.21}$$

We remark that this formulation is somewhat similar to the shadowing methods discussed below in section 1.1.3. Due to this similarity the state

space formulation is expected to be suitable for w4DVar over long time windows [52, 10, 39, 36]. In the remainder of this thesis, we shall focus on the state space formulation of w4DVar and any mention of the w4DVar method will refer to this specific formulation.

Randomized maximum likelihood

The randomized maximimum likelihood (RML) method [77] is a data assimilation method for obtaining ensembles. This method is related to 4DVar. For the non-perfect model, or weak constraint, scenario, the RML method generates ensemble members $\boldsymbol{u}_{RML}$ as the minimizers of cost functions

$$J_{RML}(\boldsymbol{u}) := \frac{1}{2}\sum_{n=0}^{N} \|y_n - H(u_n) + \tilde{\xi}_n\|^2_{C_\mathrm{o}} + \frac{1}{2}\sum_{n=0}^{N-1} \|u_{n+1} - F_n(u_n) + \tilde{\eta}_n\|^2_{C_\mathrm{m}}, \quad (1.22)$$

where $N+1$ is the number of observations, $\tilde{\xi}_n$ and $\tilde{\eta}_n$ are random variables and C_o and C_m are given matrices. The matrices C_o and C_m may in principle be different depending on n, but the possible dependence has been omitted from the notation for brevity.

For example, if $\hat{\eta}_n$ is modeled as Gaussian noise with mean zero and covariance matrix C_m, the random variable $\tilde{\eta}_n$ should be chosen according to the same Gaussian distribution. Analogously, if ξ_n is Gaussian with mean zero and covariance matrix C_o, $\tilde{\xi}_n$ should be sampled from the same distribution. If F_n is furthermore linear, the minimizers of the RML cost functions sample from the Bayesian posterior. For completeness, we remark that if $\tilde{\eta}_n$ and $\tilde{\xi}_n$ are set to zero for all n, the RML cost function reduces to the weak 4DVar cost function (in this case in the state space formulation).

1.1.3 Shadowing methods for data assimilation

The data assimilation problem may also be solved using shadowing methods. A reason to look at shadowing methods is that there are various potential problems with variational methods. The variational methods define optimization problems to be solved. If the time window becomes longer, so if observations over longer time periods are taken into account, the optimization problems may become harder. This is especially the case for s4dvar [6, 74, 82]. Shadowing provides an alternative for the perfect model scenario with observations over longer times. For w4dvar, the forcing formulation suffers from the same challenges a s4dvar. The state space formulation has some superficial similarities to shadowing [52]. This is further expanded upon in chapter 4. We first discuss numerical shadowing in general, and then how it may be applied to strong constraint data assimilation and also how shadowing approaches have been used for weak constraint data assimilation. We then relate shadowing and variational methods. Our treatment of

numerical shadowing and shadowing for strong constraint data assimilation follows [20].

The review on shadowing methods is simplified by assuming full observations are available. Throughout this section, we assume that the observation operator H is the identity matrix on $\mathcal{R}^d$. We shall discuss how this assumption can be lifted in section 4.3.

Numerical shadowing

Suppose $\boldsymbol{u}$ is an ε-orbit of $\boldsymbol{F}$ in a neighborhood of a hyperbolic set with respect to $\boldsymbol{F}$. A hyperbolic set with respect to $\boldsymbol{F}$ is an $\boldsymbol{F}$-invariant and compact subset of $\mathcal{R}^d$ such that the tangent bundle of $\boldsymbol{F}$ may be split into a stable and an unstable subspace at each point of the hyperbolic set[9]. The shadowing lemma (e.g. Theorem 18.1.2 of [62]) states that , for every $\delta > 0$ there exists $\varepsilon > 0$ such that $\boldsymbol{u}$ is δ-shadowed by an orbit of $\boldsymbol{F}$, i.e. there exists an orbit $\boldsymbol{x}$ satisfying $G(\boldsymbol{x}) = 0$ such that $\|u_n - x_n\| < \delta$ for all $n = 0, \ldots, N$.

For example, suppose the components of $\boldsymbol{u}$ are the iterates of a numerical integrator of an autonomous ODE $\dot{x} = f(x)$ and assume that the local truncation error is bounded by ε. Then the components of $\boldsymbol{u}$ are an ε-orbit of an exact flow map (with the time step of the numerical integrator) of the autonomous ODE $\dot{x} = f(x)$. Shadowing refinement[43] uses shadowing to refine an ε-orbit to more closely approximate a true orbit. This may be done by applying a Newton's iteration for $G(\boldsymbol{u}) = 0$, with starting data given by the numerical iterates $\boldsymbol{u}$. If, instead of or in addition to the closer approximation a rigorous error bound on the global error is desired, this may be done by applying the Newton-Kantorovich theorem to the fore-mentioned Newton's iteration[7, 44, 45, 16, 17, 95]. In other words, *numerical orbits of chaotic processes represent true orbits* [45].

Shadowing for strong constraint data assimilation

For data assimilation, one can apply shadowing refinement methods to find model orbits close to observations. The observations are treated as an ε-orbit of a model orbit. Shadowing for strong constraint data assimilation may make use of Newton's method to find a root of the mismatch functional $G(\boldsymbol{u})$ close to observations [11, 10, 20]. The iteration is started by making the initial guess $\boldsymbol{u} = \boldsymbol{y}$. In the kth Newton's iteration an update $\boldsymbol{\delta}^{(k)}$ is sought by approximately solving

$$G(\boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)}) = 0. \tag{1.23}$$

The update is given by $\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)}$. The solution to (1.23) is approximated by iterating

$$G'(\boldsymbol{u}^{(k)})\boldsymbol{\delta}^{(k)} = -G(\boldsymbol{u}^{(k)}), \quad \boldsymbol{u}^{(k+1)} := \boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)} \tag{1.24}$$

to convergence. Each Newton's step is solved using the right pseudoinverse of G', i.e.

$$\boldsymbol{\delta}^{(k)} = -G'(\boldsymbol{u}^{(k)})^{\dagger} G(\boldsymbol{u}^{(k)}) = -G'^{T}(G'\,G'^{T})^{-1}G.$$

The function $G(\boldsymbol{u})$ by definition has a zero for every orbit of the model. The Jacobian of G has a $d(N-1) \times dN$ block structure:

$$G'(\boldsymbol{u}) = \begin{bmatrix} -F_0'(u_0) & I & & & \\ & -F_1'(u_1) & I & & \\ & & \ddots & \ddots & \\ & & & -F_{N-1}'(u_{N-1}) & I \end{bmatrix}.$$

We remark that through the use of the pseudo-inverse, for every Newton step, $\delta^{(k)}$ is the minimum 2-norm solution to (1.24). This minumum 2-norm solution may alternatively be obtained using a regularized quadratic cost function, that is,

$$\boldsymbol{\delta}^{(k)} = \lim_{\beta \downarrow 0} \operatorname{argmin}_{\boldsymbol{\delta}_\beta} \frac{1-\beta}{2} ||G'(\boldsymbol{u}^{(k)})\boldsymbol{\delta}_\beta + G(\boldsymbol{u}^{(k)})||_I^2 + \frac{\beta}{2}||\boldsymbol{\delta}_\beta||_I^2.$$

Another way of finding the minimum 2-norm solution of 1.24 is by using the QR decomposition of G'^{T}.

Another approach to shadowing for strong constraint data assimilation is known as pseudo-orbit data assimilation (PDA); see for instance [55, 90, 32] and references therein. While the Newton shadowing discussed above is concerned with finding a root of the mismatch functional $G(\boldsymbol{u})$, in PDA the equivalent problem of minimizing $\frac{1}{2}\|G(\boldsymbol{u})\|_I^2$ is solved. Just as in Newton shadowing, any model orbit solves the PDA problem. In order to stay close to observations, PDA is then initialized at observations and the minimization is approximately solved using a fixed number of gradient descent steps [32].

The gradient descent approach of PDA typically yields not an orbit but a pseudo-orbit. A pseudo-orbit is a sequence for which the mismatch is "small". In other words, a pseudo-orbit is an ε-orbit for which the precise norm used and the value of ε are left unspecified. The distance between the pseudo-orbit and the manifold of trajectories is then smaller than the distance between observations and the manifold of trajectories. The mid-point of this pseudo-orbit is then used as the initial condition for the analysis trajectory. PDA has for example been applied for parameter estimation [88] and as a method for finding reference trajectories for ensemble forecasting [32].

Shadowing for weak constraint data assimilation

The PDA methods have also been applied to the weak constraint problem [56, 52, 31, 33].

The first approach for applying PDA to the weak constraint problem is described in [56, 57]. Here, an extra set of variables $\boldsymbol{e} \in \mathbb{R}^{d(N-1)}$ is introduced. Then the goal is to find

$$\text{argmin}_{\boldsymbol{u},\boldsymbol{e}} \quad \frac{1}{2}\|G(\boldsymbol{u}) - \boldsymbol{e}\|_I^2.$$

In [31] this method is tested numerically. It is found that in those experiments, $\|\boldsymbol{u} - \boldsymbol{y}\|_I < \|\boldsymbol{\mathcal{X}} - \boldsymbol{y}\|_I$ and $\|G(\boldsymbol{u})\| > \|G(\boldsymbol{\mathcal{X}})\|$. This means that compared to the truth, the analysis is too close to the observations and too far from being a trajectory.

In [52] it is suggested to add an extra term penalizing the distance to observations to the shadowing cost function.

The most recent approach [31, 33] here is to still apply gradient descent to $\frac{1}{2}\|G(\boldsymbol{u})\|_I^2$, but only take a limited number of steps. This can be done by introducting a stopping criterion based on limiting the value of $\|\boldsymbol{u} - \boldsymbol{y}\|_{C_o}$.

Relation between shadowing and variational data assimilation methods

In [10] it is proposed to solve the strong constraint 4DVar problem as a limit of weak constraint 4DVar problems,

$$\boldsymbol{u}_{\text{strong}} = \lim_{\beta \downarrow 0} \text{argmin}_{\boldsymbol{u}_\beta} \frac{1-\beta}{2}||G(\boldsymbol{u}_\beta)||_{C_m}^2 + \frac{\beta}{2}||\boldsymbol{u}_\beta - \boldsymbol{y}||_{C_o}^2.$$

Taking the initial condition as the solution for $\beta = 1$ and then directly setting $\beta = 0$ constitutes a shadowing algorithm [10]. Solving a sequance of weak constraint 4DVar problems for decreasing β, where first guesses come from solutions with larger values for β is proposed as a strategy for finding global minima of the weak constraint 4DVar cost functions. It is argued that stopping the iteration at a small but finite value for β can regularize the strong constraint 4DVar problem [10].

In [52] it is suggested to solve the weak constraint shadowing problem by introducing an extra term penalizing the distance to observations. This leads to the cost function

$$J(\boldsymbol{u}) = \frac{1}{2}\|G(\boldsymbol{u})\|_{W_1}^2 + \frac{1}{2}\|\boldsymbol{u} - \boldsymbol{y}\|_{W_2}^2, \tag{1.25}$$

For some weighting matrices W_1, W_2. The weighting matrices W_1, W_2 can be chosen such that this cost function is identical to the cost function minimized in the state space formulation of w4DVar. It is however argued in [52] that that does not imply this shadowing method with extra term is the same as weak 4DVar. In particular, it is noted in [52] that W_1 should be interpreted as

a metric, not as a model covariance matrix, and $G(\boldsymbol{u})$ at optimum should be interpreted as a vector of model corrections. It is derived that at optimum

$$\mathbf{0} = J'(\boldsymbol{u}^{\text{opt}}) = G'^{T}(\boldsymbol{u}^{\text{opt}})W_1^{-1}G(\boldsymbol{u}^{\text{opt}}) + W_2^{-1}(\boldsymbol{u}^{\text{opt}} - \boldsymbol{y}).$$

This implies that at optimum, the difference between analysis and observations is nonlinearly related to the residual $\boldsymbol{G}(\boldsymbol{u}^{\text{opt}})$ by

$$\boldsymbol{u}^{\text{opt}} - \boldsymbol{y} = -W_2G'^{T}(\boldsymbol{u}^{\text{opt}})W_1^{-1}\boldsymbol{G}(\boldsymbol{u}^{\text{opt}}).$$

It immediately follows that for w4DVar with non-trivial observations and without background term, it holds that

$$H'^{T}(\boldsymbol{u}^{w4s})C_o^{-1}\left(H(\boldsymbol{u}^{w4s}) - \boldsymbol{y}\right) = -G'^{T}(\boldsymbol{u}^{w4s})C_m^{-1}\boldsymbol{G}(\boldsymbol{u}^{w4s}). \tag{1.26}$$

Because the difference between analysis and observations is not statistically independent from the residual at the solution of the w4DVar problem, it is argued in [52] that the solution of the w4DVar problem does not respect the assumptions made in the derivation of w4DVar. This then leads to an alternative explanation[52] of the terms of the cost function (1.25).

1.1.4 Iterative regularization methods

For shadowing data assimilation, the choice of optimization or root finding algorithm plays a big role. In chapter 4, section 4.3 we propose using an iterative regularization method. Here we briefly review some theory on a iteratively regularizing Levenberg-Marquardt scheme based on [46, 60, 50, 49]. In particular, in [50, 49] algorithms are suggested that use iterative methods to solve the strong constraint 4DVar problem (1.17). These methods use a regularizing Levenberg-Marquardt scheme [46, 60] to stably minimize (1.16) under a strong constraint.

Define the family of operators $\mathcal{M}_n : \mathbb{R}^d \to \mathbb{R}^{b(N+1)}$ by $\mathcal{M}_n(u_0) := H(F^n(u_0))$ for all $n = 0, \ldots, N$. Substituting the constraint into the strong constraint 4DVar cost function leads to the minimization problem

$$u_0^{s4reg} = \text{argmin}_{u_0}\|\boldsymbol{y} - \mathcal{M}(u_0)\|^2_{C_o}. \tag{1.27}$$

In (1.27), the background term is missing compared to the standard formulation of s4DVar (1.17). The main idea of the method is now to approximately minimize (1.27) with a cleverly chosen method, such that those minimizers are stable with respect to the noise [46, 60, 50, 49].

In particular, consider the iteration [46, 50]

$$\delta_0^{(k)} = -C_b\mathcal{M}'^{T}(u_0^{(k)})\left(\mathcal{M}'(u_0^{(k)})C_b\mathcal{M}'^{T}(u_0^{(k)}) + \alpha^{(k)}C_o\right)^{-1}\left(\mathcal{M}(u_0^{(k)}) - \boldsymbol{y}\right), \tag{1.28}$$

$$u_0^{(k+1)} = u_0^{(k)} + \delta_0^{(k)},$$

where $\alpha^{(k)}$ is a well-chosen regularization parameter. The choice of $\alpha^{(k)}$ and a stopping criterion for the iteration are discussed below. It can be derived (see e.g. [50] or section 4.3) that $\delta_0^{(k)}$ minimizes

$$J^{s4reg,(k)}(\delta) = \frac{1}{2}\left\|\mathcal{M}'(u_0^{(k)})\delta + \mathcal{M}(u_0^{(k)}) - \boldsymbol{y}\right\|_{C_0}^2 + \frac{\alpha^{(k)}}{2}\|\delta\|_{C_b}^2. \tag{1.29}$$

We remark that (1.29) does not stay fixed throughout the iteration. Indeed, at each iteration step k we get a different quadratic functional which is then minimized by the linear update (1.28).

Choice of the regularization parameter and stopping criterion

For the iterative regularization methods, the choice of regularization parameters and stopping criterion are important. A method based on Morozov's discrepancy principle is proposed in [46]. Assume $u_0^{(k)}$ is not a minimizer of (1.27). Define $u_0^\dagger$ as the minimizer closest to $u_0^{(k)}$ in 2-norm. Assume there exist $0 < \rho < 1 < \gamma$ such that

$$||\mathcal{M}'(u_0^{(k)})(u_0^\dagger - u_0^{(k)}) + \mathcal{M}(u_0^{(k)}) - \boldsymbol{y}||_{C_o} \leq \frac{\rho}{\gamma}||\mathcal{M}(u_0^{(k)})||_{C_o}. \tag{1.30}$$

Then $\alpha^{(k)}$ can be determined uniquely by imposing that

$$||\mathcal{M}'\left(u_0^{(k)}\right)\delta(\alpha^{(k)}) + \mathcal{M}(u_0^{(k)}) - \boldsymbol{y}||_{C_o} = \rho||\mathcal{M}\left(u_0^{(k)}\right)||_{C_o}, \tag{1.31}$$

where we made the dependence of the update step δ on the parameter $\alpha^{(k)}$ explicit. We remark that decreasing ρ will result in a decrease of the regularization strength α. The smaller α is, the larger the update step δ will be [46, 60]. For this choice of α, there is a convergence theory [46, 60]. This guarantees that in the noise-free case, if $||u_0^{(k)} - u_0^\dagger||$ is sufficiently small and $\mathcal{M}$ is smooth enough around $u_0^\dagger$, the algorithm will converge to a model orbit if $k \to \infty$.

If there is observational noise, the stopping criterion should terminate the iteration after a finite number of steps and can also be chosen according to a discrepancy principle. In particular, suppose the true minimum can be bounded by some known K, that is $\|\boldsymbol{y} - \mathcal{M}(u_0^\dagger)\|_{C_o} < K$. Let $r^{\text{stop}} > 1/\rho$ and assume $||u_0^{(k)} - u_0^\dagger||$ is sufficiently small. Then [46, 60] there exists a finite $k_* = O(1 + \ln K)$ such that for all $k < k_*$

$$\|\boldsymbol{y} - \mathcal{M}\left(u_0^{(k_*)}\right)\|_{C_o} \leq r^{\text{stop}}K < \|\boldsymbol{y} - \mathcal{M}\left(u_0^{(k)}\right)\|_{C_o}. \tag{1.32}$$

1.2 Structure of this thesis

This thesis comprises four more chapters and a conclusion. In chapter 2 we develop algorithms for data assimilation based upon a computational time

dependent stable/unstable splitting. Our particular method is based upon shadowing refinement and synchronization techniques and is motivated by work on Assimilation in the Unstable Subspace (AUS) [12, 94, 78] and Pseudo-orbit Data Assimilation (PDA) [55, 54, 32]. The algorithm utilizes time dependent projections onto the non-stable subspace determined by employing computational techniques for Lyapunov exponents/vectors. The method is extended to parameter estimation without changing the problem dynamics and we address techniques for adapting the method when (as is commonly the case) observations are not available in the full model state space. We use a combination of analysis and numerical experiments (with the Lorenz 63 and Lorenz 96 models) to illustrate the efficacy of the techniques and show that the results compare favorably with other variational techniques. This chapter intends to address data assimilation in the perfect model scenario and to provide a shadowing method that is more stable than s4dvar and computationally efficient due to the AUS aspects.

In chapter 3, we develop further an algorithm for data assimilation based upon the shadowing refinement technique of chapter 2 to take partial observations into account. The method is based on a regularized Gauss-Newton method. We prove local convergence to the solution manifold and provide a lower bound on the algorithmic time step. We use numerical experiments with the Lorenz 63 and Lorenz 96 models to illustrate convergence of the algorithm and show that the results compare favourably with a variational technique—weak-constraint four-dimensional variational method—and a shadowing technique–pseudo-orbit data assimilation. Numerical experiments show that a preconditioner chosen based on a cost function allows the algorithm to find an orbit of the dynamical system in the vicinity of the true solution. This chapter intends to further develop the method of chapter 2, in particular by allowing for partial observations. Partial observations are highly relevant in practice, because the full state space is not always observed.

In chapter 4, we consider a data assimilation problem for imperfect models. We propose a novel shadowing-based data assimilation method that takes model error into account following the Levenberg-Marquardt regularization approach. We illuminate how the proposed shadowing-based method is related to the weak constraint 4DVar method both analytically and numerically. We demonstrate that the shadowing-based method respects the distribution of the data mismatch, while the weak constraint 4DVar does not, which becomes even more pronounced with fewer observations. Moreover, sparse observations give weaker influence on unobserved variables for the shadowing-based method than for the weak constraint 4DVar. This chapter intends to provide a shadowing data assimilation method for the imperfect model scenario, point out some issues regarding statistical properties of the data and model mismatches of the analysis and to describe how shadowing and w4DVar are related.

In chapter 5, the algorithm of the previous chapters is developed yet further. We extend the algorithm to generate an ensemble of states, for estimating the uncertainties of the data assimilation algorithm using the concept of indistinguishable states. This chapter also includes some proofs regarding well-posedness, accuracy and consistency of the algorithm. The algorithm is applied to a non-perfect model to show how the unmodelled components of the model can be estimated using the data assimilation algorithm. This chapter intends to provide for an error estimate of the data assimilation result, through the use of an ensemble. Furthermore, this chapter intends to estimate unmodelled components of the system. In practice, such an estimate may be useful for improving the model.

CHAPTER 2

Projected Shadowing-based Data Assimilation

2.1 Introduction

Data assimilation methods combine orbits from a dynamical system model with measurement data to obtain an improved estimate for the state of a physical system. A version of this chapter may also be found in [20]. In this chapter we develop a data assimilation method in the context of a discrete deterministic model

$$x_{n+1} = F_n(x_n), \quad x_n \in \mathcal{R}^d, \quad n = 0, \ldots, N-1, \tag{2.1}$$

where $F_n : \mathcal{R}^d \to \mathcal{R}^d$. In many applications the model is defined by the time-discretization of an ordinary differential equation (ODE)

$$\dot{x} = f(t, x), \quad x(t) \in \mathcal{R}^d, \tag{2.2}$$

which in turn may be defined as the space-discretization of a partial differential equation (or system of PDEs).

Let the sequence[1] $\{\mathcal{X}_0, \mathcal{X}_1, \ldots, \mathcal{X}_N\}$ be a distinguished orbit of (1.1), referred to as the *true solution* of the model, and presumed to be unknown. Suppose we are given a sequence of noisy observations y_n related to $\mathcal{X}_n$ via

$$y_n = H\mathcal{X}_n + \xi_n, \qquad y_n \in \mathcal{R}^b, \quad n = 0, \ldots, N, \tag{2.3}$$

[1] In the sequel we will adopt the notation $\{\mathcal{X}_n; n = 0, \ldots, N\}$ or simply as $\{\mathcal{X}_n\}$ for a discrete orbit. The latter notation is also occasionally employed to denote an infinite sequence.

where $H : \mathcal{R}^d \to \mathcal{R}^b$, $b \leq d$, is the observation operator, and the noise variables ξ_n are drawn from a normal distribution $\xi_n \sim \mathcal{N}(0, E)$ with zero mean and known observational error covariance matrix E.

Data assimilation is the problem of finding an orbit (or pseudo-orbit, see 2.2.1) $\boldsymbol{u} = \{u_0, u_1, \ldots, u_N\}$, $u_n \in \mathcal{R}^d$, of the model (1.1), such that the differences $\|y_n - Hu_n\|$, $n = 0, \ldots, N$ are small in an appropriately defined sense. This is done with the aim of minimizing the unknown error $\|u_n - \mathcal{X}_n\|$; see for example [91, 63]. For instance, well known four-dimensional variational data assimilation (4DVar) aims at finding the optimal initial condition u_0 of (1.1) to minimize a cost function

$$C_{\text{var}}(u_0; \{y_n\}) = \sum_{n=1}^{N} (y_n - Hu_n)^T E^{-1} (y_n - Hu_n) + \lambda_n (u_n - F_n(u_{n-1})),$$

where the λ_n are Lagrange multipliers to ensure that the sequence $\{u_n\}$ defines an orbit of (1.1) (see e.g.[85, 66, 92, 91] and references therein). One drawback of variational data assimilation is that the number of local minima of the cost function increases dramatically with N [6, 74, 82]. This places a practical limit on the length of the assimilation window—the time period over which observations may be assimilated.

We propose a novel data assimilation method that overcomes this drawback: with the proposed method, increasing the length of the assimilation window may in fact lead to a better estimation. Instead of minimizing a cost function, we search for a zero of the *cost operator*

$$G(\boldsymbol{u}) = \begin{pmatrix} G_0(\boldsymbol{u}) \\ G_1(\boldsymbol{u}) \\ \vdots \\ G_{N-1}(\boldsymbol{u}) \end{pmatrix}, \qquad G_n(\boldsymbol{u}) = u_{n+1} - F_n(u_n), \quad n = 0, \ldots, N-1, \tag{2.4}$$

using a contractive iteration started from (a proxy of) complete, noisy observations. This approach is motivated by research on numerical shadowing methods. We stress that, as is the case with 4DVar, our approach attempts to find an exact orbit of (1.1) consistent with the observations. However, instead of solving directly for the initial condition, we solve for the whole orbit at once.

As stated, our approach assumes the availability of (noisy) observations of the complete state vectors $\mathcal{X}_n$. In other words, we assume that the observation operator H is the identity matrix on $\mathcal{R}^d$. When only partial observations are available, it is necessary to generate a proxy for complete observations. This can be done by some other cheap but inaccurate data assimilation method. For instance, in 2.6.1 we demonstrate this idea using direct insertion of noisy partial observations into the iteration (1.1).

Recent efforts [12, 94, 42, 78, 64, 84] to improve speed and reliability of data assimilation specifically address the partitioning of the tangent space into stable, neutral and unstable subspaces corresponding to Lyapunov vectors associated with negative, zero and positive Lyapunov exponents, respectively (see 2.2.2). In particular, Trevisan, d'Isidoro & Talagrand propose a modification of 4DVar, so-called 4DVar-AUS, in which corrections are applied only in the unstable and neutral subspaces [94, 78]. Research by Pecora & Carroll [79] indicates that when partial observations are sufficient to constrain the unstable subspace, an orbit of the chaotic Lorenz 63 system can be made to converge exponentially in time to a different, driving orbit, hence refining in the stable subspace. Their work has triggered a substantial body of research on the idea of synchronization of chaos (see review articles by Pecora et al. [81] and Boccaletti et al. [8]).

Motivated by the above, in this chapter we propose a new method for data assimilation that utilizes distinct treatments of the dynamics in the stable and non-stable directions. We find a numerical orbit compatible with observations by using Newton's method with updates projected on the non-stable subspace to emphasize the need to stay close to current observations in non-stable directions. In the stable subspace, we ensure that the trajectory is determined by past observations using a forward integration to synchronize the stable components. Although our focus here is on splitting into non-stable and stable components and then applying shadowing refinement and synchronization techniques, respectively, the splitting framework allows for other possibilities. In particular, if the non-stable subspace is relatively low dimensional this makes applications of techniques such as particle filters appealing. In addition, 4DVar or Kalman filter techniques may be applied to the stable system with the advantage that these techniques are being applied to a system with contractive dynamics. This also allows the split system to be put in a Bayesian data assimilation context.

In the next section we provide relevant background results. In 2.3 we describe the sense in which Newton's method is an effective data assimilation algorithm. While effective, the full Newton's iteration can be made more efficient by restricting the updates to just the non-stable tangent directions, as described in 2.4. The updates can then be synchronized in the stable directions as shown by the analysis in 2.A. We provide details of our implementation in 2.5. Finally, in 2.6 we provide numerical results for the Lorenz 63 model and compare the method to 4DVar for the Lorenz 96 model. We draw conclusions in 2.7.

2.2 Background

In 2.2.1 we review concepts from numerical shadowing, in 2.2.2 we describe the computation of the tangent space splitting used in this chapter, and in 2.2.3

we review synchronization of chaos.

2.2.1 Numerical shadowing

An ε-pseudo-orbit is a sequence $\boldsymbol{u} = \{u_0, u_1, \ldots, u_N\}$ satisfying $\|G_n(\boldsymbol{u})\| < \varepsilon$, $n = 0, \ldots, N-1$ where $\|\cdot\|$ is a norm in $\mathcal{R}^d$. For instance, suppose $F \equiv F_n$ is the exact time-τ flow map of an autonomous ODE $\dot{x} = f(x)$. If the components of $\boldsymbol{u}$ are the iterates of a numerical integrator with local truncation error bounded by ε, then these define an ε-pseudo-orbit of F. The shadowing lemma (e.g. Theorem 18.1.2 of [62]) states that in a neighborhood of a hyperbolic set for F, for every $\delta > 0$ there exists $\varepsilon > 0$ such that every ε-pseudo-orbit is δ-shadowed by an orbit of F, i.e. there exists an orbit $\{x_n\}$ satisfying $x_{n+1} = F(x_n)$ such that $\|u_n - x_n\| < \delta$ for all $n = 0, \ldots, N$. We remark that this means that the initial condition $x_0 \neq u_0$ in general. Rigorous bounds on the global error of numerical integrations with respect to a shadowing orbit can be proved. This is done by applying the Newton-Kantorovich theorem to Newton's iteration for $G(\boldsymbol{x}) = 0$ with starting data given by the numerical iterates $\boldsymbol{u}$ on a time interval that is long relative to the characteristic Lyapunov time [7, 44, 45, 16, 17, 95]. Shadowing is an important analysis technique for obtaining global error bounds on the numerical approximation to the solution of differential equations exhibiting chaos.

Numerical shadowing refinement is a residual correction technique that seeks to correct the residual (departure from being a true solution) by solving the error equation without constraining the initial condition. This makes it applicable over long time intervals for problems with positive Lyapunov exponents. When refining using a Gauss-Newton method the linear equations being solved are underdetermined so by using the pseudo inverse the Newton update is the minimum two norm solution correction of the residual. We can view data assimilation in the same vein by interpreting the data as some approximation to the model solution and set it as our goal to find a particular model solution that shadows the data.

With respect to shadowing, the *inverse problem* is to determine an optimal initial condition u_0 for a numerical integration, such that the numerical iterates $\boldsymbol{u}$ δ-shadow a desired orbit of (1.2). Shadowing refinement (see, e.g., [43]), employs the pseudo-orbit as an initial guess for $G(\boldsymbol{u}) = 0$ and, as opposed to proving the existence of a nearby zero of G, iteratively refines the pseudo-orbit to obtain an improved approximation of a true solution. This is clearly akin to the data assimilation problem.

Shadowing theory has already motivated a practical data assimilation algorithm known as pseudo-orbit data assimilation (PDA); see for instance [55, 54, 32]

and references therein. For the PDA approach a cost function

$$C_{\mathrm{PDA}} = \frac{1}{2}\sum_{n=0}^{N-1} G_n^T G_n$$

is minimized and the minimization is also initialized from observations. Obviously, the (nonunique) global minimum of C_{PDA} is zero and this value is reached if and only if $G(\boldsymbol{u}) = 0$, that is, if $\boldsymbol{u}$ is any model trajectory. The approach in [32] approximatly minimizes C_{PDA} by taking a fixed number of gradient descent steps starting from observations. This typically yields not an orbit but a (discrete) pseudo-orbit, i.e. the minimizing sequence satisfies $\|u_n - \mathcal{X}_n\| < \varepsilon$, for all $n = 0, \ldots, N$, and some constant ε. The distance between the pseudo-orbit and the manifold of trajectories is then smaller than the distance between observations and the manifold of trajectories. The mid-point of this pseudo-orbit is then used as the initial condition for a trajectory that should be consistent with model and data. PDA has been applied in operational weather models [54], parameter estimation [88] and as a method for finding reference trajectories for ensemble forecasting [32].

2.2.2 Tangent subspace decomposition

In this section we review the decomposition of the tangent space into stable, neutral and strongly unstable subspaces. This decomposition is central to the method described in this chapter. Let $\{x_n; n = 0, \ldots, N\}$ denote an orbit of (1.1). The fundamental matrix equation associated with $\{x_n\}$ is a matrix valued difference equation

$$X_{n+1} = F_n'(x_n)X_n, \quad n = 0, \ldots, N-1, \tag{2.5}$$

where $X_n \in \mathcal{R}^{d\times d}$. The iterates of (2.5) become increasingly ill-conditioned as the columns align with the dominant growth direction. To stably estimate X_n, one may introduce a time-discrete QR factorization. Let $X_0 = Q_0 R_0$, and write

$$Q_{n+1}R_{n+1} = F_n'(x_n)Q_n \quad \text{for} \quad n = 0, ..., N-1, \tag{2.6}$$

where $F_n'(x_n)Q_n$ is a matrix product of known quantities, and $Q_{n+1}R_{n+1}$ is the QR factorization found using the modified Gram-Schmidt process. Then $X_1 = F_0'(x_0)Q_0R_0 = Q_1R_1R_0$, $X_2 = F_1'(x_1)X_1 = F_1'(x_1)Q_1R_1R_0 = Q_2R_2R_1R_0$, etc. Note that this procedure is well defined for $Q_n \in \mathcal{R}^{d\times p}$ for $p \leq d$ provided $F_n'(x_n)Q_n$ is full rank for all n. The Gram-Schmidt process yields the unique upper triangular $R_n \in \mathcal{R}^{p\times p}$ with positive diagonal elements and, importantly, preserves the ordering of the columns of the Q_n.

The (local) p $(1 \leq p \leq d)$ largest Lyapunov exponents of the orbit $\{x_n\}$ are extracted from the time average of the logarithm of the diagonal of R_n

[29]:

$$\lambda_i = \limsup_{N\to\infty} \frac{1}{N} \sum_{n=1}^{N} \ln R_n^{(i,i)}, \quad i = 1, \ldots, p.$$

The method of construction ensures $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p$. Associated with λ_i is a Lyapunov vector $V_n^{(i)}$. The columns of Q_n generally (for most initial conditions) form an orthonormal basis for the Lyapunov vectors at time n (see [22, 23, 24]). The iteration (2.5) is a generalized power iteration. For each $\ell = 1, \ldots, p$, one finds in the limit $n \to \infty$ that $\text{span}\{V_n^{(1)}, \ldots, V_n^{(\ell)}\} = \text{span}\{Q_n^{(1)}, \ldots, Q_n^{(\ell)}\}$, where $Q_n^{(i)}$ denotes the ith column of Q_n [29, 26, 1, 5]. Positive (negative) λ_i correspond to tangent directions $Q_n^{(i)}$ in which perturbations grow (decay) exponentially. Consequently, if $\lambda_p \geq 0 > \lambda_{p+1}$, then the matrix $Q_n^u = (Q_n^{(1)}, \ldots Q_n^{(p)})$ provides an orthonormal basis for the non-stable tangent space at X_n. This time dependent tangent space decomposition is motivated by splittings due to integral separation (related to possessing an exponential dichotomy relative to a constant shift in the stability spectrum) and are analogous to having a decomposition via the Oseledets filtration.

We note here that by approximating the non-stable subspace we obtain information (see [42]) that may be used to analyze the error in data assimilation schemes, namely in terms of the degree to which observations constrain the uncertainty within the non-stable subspace. We remark that the dimension of the unstable subspace may be much less then the total dimension. In Carrassi et al. [13] it is shown that the AUS-framework gives good results for a quasi-geostrophic model described in [83]. This model is of dimension 14784, while the unstable subspace has a dimension of 24 [89].

We will use the computed factors $Q_n^u \in \mathcal{R}^{d\times p}$ to construct projection operators onto the non-stable tangent space. The Q_n^u are quantities that can be computed robustly with good forward error analysis properties (under reasonable assumptions closely related to the continuity of Lyapunov exponents with respect to perturbations and the integral separation or integral separation structure, see[1]). In particular, the results in [27, 28, 96, 2] show that the Q_n^u are continuous with respect to errors in $F'(x_n)$ and quantify the error in the Q_n as a function of the separation in growth/decay rates. In our context this ensures the time dependent projection operators $P_n = Q_n^u {Q_n^u}^T$ are robust.

2.2.3 Synchronization

Pecora & Carroll [79] demonstrated that an orbit of a chaotic dynamical system (the observer) can sometimes be made to synchronize with a second orbit (the driver) of that system, given partial observations of the driver signal. There is

a sizeable body of literature on synchronization of chaos, particularly in the field of systems and control theory [79, 47, 42]. The use of synchronization for data assimilation was first applied in [34].

For our purposes, the following coupled driver-response process is appropriate:

$$x_{n+1} = F_n(x_n), \tag{2.7a}$$
$$z_{n+1} = P_n x_{n+1} + (I - P_n)F_n(z_n), \tag{2.7b}$$

where the $P_n \in \mathcal{R}^{d\times d}$ are a sequence of appropriately chosen projection matrices. The manifold $\mathcal{S} = \{(x, z) \in \mathcal{R}^d \times \mathcal{R}^d : x = z\}$ is invariant under these dynamics and is called the *synchronization manifold*. When $\mathcal{S}$ attracts a neighborhood of itself, then for z_0 within the basin of attraction, z_n synchronizes with x_n. Defining $w_n = z_n - x_n$, $n = 0, 1, \ldots$, the transverse dynamics with respect to $\mathcal{S}$ is given by

$$\begin{aligned} w_{n+1} &= P_n x_{n+1} + (I - P_n)F_n(z_n) - F_n(x_n) \\ &= P_n F_n(x_n) + (I - P_n)F_n(z_n) - F_n(x_n) \\ &= (I - P_n)\left[F_n(x_n + w_n) - F_n(x_n)\right] \\ &= (I - P_n)F_n'(x_n)w_n + r_n(w_n), \end{aligned}$$

where $r_n(w)$ is assumed to be of higher order in w. The projectors P_n need to be chosen to ensure asymptotic stability of the origin under the transverse dynamics. From the stability theory of Lyapunov, it is known that if the sequence $\|w_n\|$ converges exponentially to zero for generic initial conditions, then the Lyapunov exponents of the transverse dynamics must necessarily all be negative. Such a necessary condition is argued by Pecora & Carroll in [80]. On the other hand, negativity of the Lyapunov exponents is also sufficient for convergence in a neighborhood of the origin, if F_n' is regular and r_n is at least second order in w. In our application to data assimilation we will choose P_n to project (in an approximate sense) onto the locally non-stable tangent space Q_n^u. 2.1 illustrates synchronization of the Lorenz 96 model (see [70] and 2.6.2) using the driver-response system (2.7a)-(2.7b) with projection $P_n = Q_n^u Q_n^{uT}$ for increasing dimension of the projection space p. In particular we observe exponential convergence only when p is greater than or equal to the dimension of the nonstable space, with exponential rate of convergence increasing with p.

2.3 Data assimilation via Newton's method

In this section we discuss the use of Newton's method for data assimilation, a context in which it was first applied in [11]. An important property of Newton's method is its local nature: when the initial guess is sufficiently close to a zero,

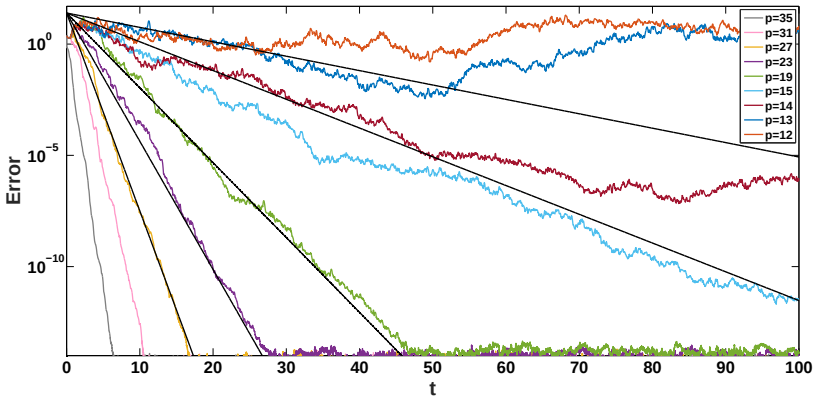

Figure 2.1: Synchronization in the L96 model (2.21) with 13 positive Lyapunov exponents in 36 dimensions. We plot the ℓ^∞-norm of the difference between the true solution and the synchronization approximation as a function of time. Forcing is done with projections of the true solution onto the non-stable space and the different graphs are for various values of p. In black we plot lines given by Error $= 25e^{\lambda_{p+1}\text{Time}}$, for $p+1 = 15, 16, 20, 24, 28$. It can be observed that after a transient time and for sufficiently large p (i.e. $p \geq 14$), convergence to the true solution is exponential.

the iterates converge to that zero at a quadratic rate. This statement is made formal in the Kantorovich Theorem [61].

Consequently, by analogy to the shadowing approach to global error estimation, we may construct a simple scheme for data assimilation by applying Newton's iterations to solve

$$G(\boldsymbol{u}) = 0,$$

where G is defined in (1.6) and starting data is provided by the noisy observations $\{y_n : n = 0, \ldots, N\}$ with observation operator the identity $\{H\mathcal{X}_n = \mathcal{X}_n : n = 0, \ldots, N\}$. (an assumption that can be relaxed, see 2.6).

In the kth Newton's iteration we seek an update $\boldsymbol{\delta}^{(k)}$ approximately solving

$$G(\boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)}) = 0. \tag{2.8}$$

We then update using $\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)}$. The solution to (2.8) is approximated by iterating

$$G'(\boldsymbol{u}^{(k)})\boldsymbol{\delta}^{(k)} = -G(\boldsymbol{u}^{(k)}), \quad \boldsymbol{u}^{(k+1)} := \boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)} \tag{2.9}$$

to convergence. We solve each Newton's step using the right pseudoinverse of G', i.e.

$$G'^{\dagger} = G'^{T}(G'\,G'^{T})^{-1},$$

where the linear system involving the block tridiagonal matrix $G'\,G'^T$ is solved using a block tridiagonal solver. The function $G(\boldsymbol{u})$ has a zero for every orbit of the model (1.1). The function $G : \mathcal{R}^{dN} \to \mathcal{R}^{d(N-1)}$ has a $d(N-1) \times dN$ Jacobian with block structure

$$G'(\boldsymbol{u}) = \begin{bmatrix} -F_0'(u_0) & I & & & \\ & -F_1'(u_1) & I & & \\ & & \ddots & \ddots & \\ & & & -F_{N-1}'(u_{N-1}) & I \end{bmatrix}.$$

To distinguish this method from the projected method to be described in 2.4, we shall refer to it as the *full Newton's method*.

The fact that Newton's method is a *local* root-finding method proves useful. The Kantorovich theorem can be generalized using the existence of a right inverse as opposed to an inverse. Then the solution $\boldsymbol{u}$ satisfies $G'(\boldsymbol{u})^T G(\boldsymbol{u}) = 0$ [4]. In [55] it is shown that $G'(\boldsymbol{u})^T G(\boldsymbol{u}) = 0$ implies $G(\boldsymbol{u}) = 0$, i.e. $\boldsymbol{u}$ is a trajectory.

Initializing the algorithm with observations, we can expect to find a trajectory close to observations, provided the initial observational error is not too large. The use of the pseudo-inverse minimizes the Newton step size in the 2-norm. By taking those shortest possible steps while starting from observations, the iterates should stay close to the observations, while converging to a trajectory. After convergence, it is possible to compute the distance between the assimilation and observations and use that to judge the quality of the assimilations [11, 10, 32].

The convergence of this approach with Newton's method can be demonstrated with numerical examples. Using the Lorenz 63 (L63) model [69] (see also 2.6.1) a set of true trajectories $\{X_n\}$ were integrated and perturbed. We used 1000 random initial conditions, which were all integrated for a spin-up time of length 5, after which we added noise as specified in (1.5) with noise covariance $E = I$. Experiments were performed for the L63 model integrated with both a forward Euler scheme and a classic 4th order Runge-Kutta (RK4) scheme. We used time steps of length $\Delta t = 0.005$ for both methods and observed the full state at each time step. For observations with sparser in time we refer to 2.6.2. The perturbed data was used as a starting guess for Newton's method. We used a single observational window of 2000 steps, corresponding to a time interval of length 10. Unless indicated otherwise, all errors are using the ℓ_2-norm. In 2.2, we show the median error over time as a dotted line, with the area between 25th and 75th percentiles shaded. The 2nd and 98th percentiles are shown as solid lines. The results for the forward Euler method are showin in blue and the results for the RK4 method in orange. The mean square error (MSE) is defined as

$$\text{MSE} = \frac{1}{N} \sum_{n=1}^{N} (u_n - \mathcal{X}_n)^T (u_n - \mathcal{X}_n). \tag{2.10}$$

Table 2.1: Application of Newton's method to L63. $C(\cdot)$ is defined in equation (2.11).

Property	**Value**
Observation error $C(\{\mathcal{X}_n\})$	2.9994 ± 0.0549
Obs. discrepancy of estimated trajectory $C(\boldsymbol{u})$ (Euler)	3.0062 ± 0.0639
Obs. discrepancy of estimated trajectory $C(\boldsymbol{u})$ (RK4)	2.9986 ± 0.0550

For the Euler method, the median of the MSE is equal to 0.027. Next we examine errors with respect to the observation operator

$$C(\{x_n\}) = \frac{1}{N}\sum_{n=1}^{N}(y_n - Hx_n)^T(y_n - Hx_n). \tag{2.11}$$

The mean square error in the observations (mean noise variance) is given by $\langle C(\{\mathcal{X}_n\})\rangle = 2.9994$, where $<$ ... $>$ stands for the average over the experiments. By comparison, the mean observation discrepancy of the Newton's solution is equal to $\langle C(\boldsymbol{u})\rangle = 3.0062$ for the Euler discretisation experiments and $\langle C(\boldsymbol{u})\rangle = 2.9986$ for the RK4 discretisation experiments. In 497 out of 1000 Euler experiments and in 860 out of 1000 RK4 experiments, we find that the discrepancy $C(\boldsymbol{u}) < C(\{\mathcal{X}_n\})$. That is, even though the trajectory found by Newton's method is not identical to the true trajectory, it is in fact a model orbit *closer* to the observations. This demonstrates that the method works well, even when observational noise prevents determining a unique viable trajectory.

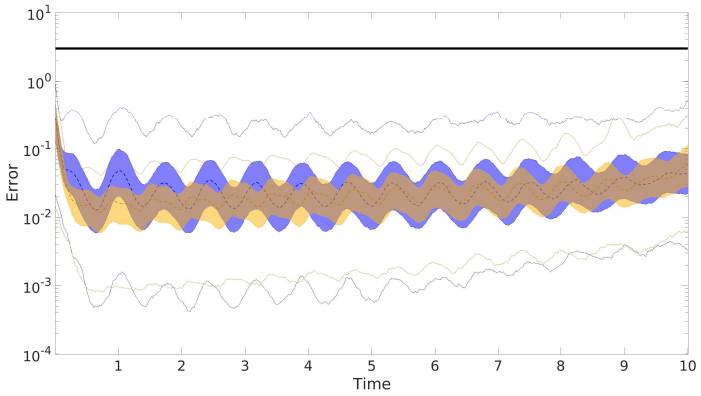

Figure 2.2: Application of Newton's method to the L63 model (1.12). The mean observational error is in black, the error over time of the estimation is in blue for forward Euler and orange for RK4.

Table 2.2: Application of Newton's method to L96. $C(\cdot)$ is defined in equation (2.11).

Property	**Value**
Observation error $C(\{\mathcal{X}_n\})$	35.9872 ± 0.3745
Obs. discrepancy of estimated trajectory $C(\boldsymbol{u})$ (Euler)	35.9227 ± 0.3751
Obs. discrepancy of estimated trajectory $C(\boldsymbol{u})$ (RK4)	35.9229 ± 0.3736

We have also repeated this numerical experiment for the Lorenz 96 model [70] (see also 2.6.2). For parameter values $\mathcal{F} = 8$ and $d = 36$, we integrated 1000 initial conditions over a run-up time of 5, after which we assimilated the data taken over a window of length 2.5. We used both forward Euler and RK4, with time step $\tau = 0.005$ and full observations. Results are in 2.3 and 2.2.

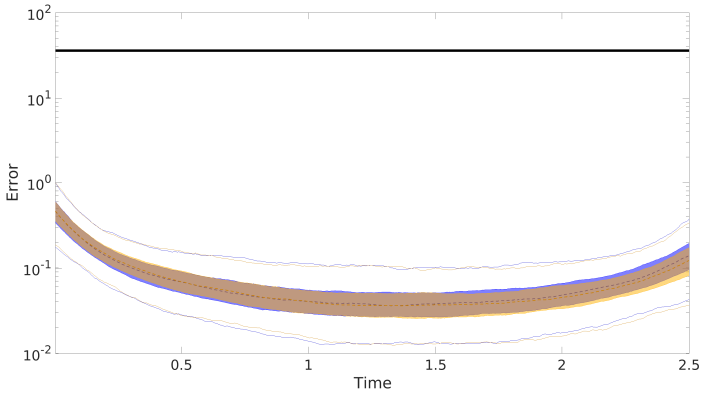

Figure 2.3: Application of Newton's method to the L96 model (2.21). The mean observational error is in black, the error over time of the estimation is in blue for forward Euler and orange for RK4.

For the Euler discretisation of L96 the median of the MSE of the median is 0.0558. In 994 out of 1000 Euler experiments and in 998 out of 1000 RK4 experiments, we find that the discrepancy $C(\boldsymbol{u}) < C(\{\mathcal{X}_n\})$. In the rest of the chapter we shall consider results for experiments with Euler discretisation. For comparisons, unless stated otherwise, truth and noise realisations are identical between experiments, except in 2.6.1, where we include results for an ensemble of noise realizations.

Remark. Our data assimilation method can be applied to parameter estimation as well. A standard approach to dealing with parameter estimation is to treat parameters as dependent variables with trivial dynamics. This approach adds neutral directions to the tangent space, which can hamper convergence of shadowing. Instead, Newton's method can be extended to

simultaneously estimate state space variables and parameters. Consider $G(\boldsymbol{u})$ defined in (4) and replace $G_n(\boldsymbol{u})$ with $G_n(\boldsymbol{u};\boldsymbol{\alpha}) = u_{n+1} - F_n(u_n;\boldsymbol{\alpha})$ where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, ..., \alpha_q)$. Linearization with respect to α_j takes the form

$$\frac{\partial F_n(u_n;\boldsymbol{\alpha})}{\partial \alpha_j} d\alpha_j, \;\; n = 0, ..., N-1.$$

In the presence of uncertain parameters, the linearization of G is modified. In particular, $G'(\boldsymbol{u})$ becomes $G'(\boldsymbol{u};\boldsymbol{\alpha})$. In the case of q parameters we have

$$G'(\boldsymbol{u};\boldsymbol{\alpha}) = [G'_{\boldsymbol{u}}|G'_{\boldsymbol{\alpha}}],$$

where $G_{\boldsymbol{\alpha}}$ is composed of q column vectors. Note that when forming the pseudoinverse of G', $G'^{\dagger} = G'^{T}(G'G'^{T})^{-1}$ we have

$$G'G'^{T} = G'_{\boldsymbol{u}}G'^{T}_{\boldsymbol{u}} + G'_{\boldsymbol{\alpha}}G'^{T}_{\boldsymbol{\alpha}}$$

so that $G'_{\boldsymbol{\alpha}}G'^{T}_{\boldsymbol{\alpha}}$ is a rank q perturbation of the block tridiagonal matrix $G'_{\boldsymbol{u}}G'^{T}_{\boldsymbol{u}}$. This allows for the use of Sherman-Morrision-Woodbury formulas and a solver for $G'_{\boldsymbol{u}}G'^{T}_{\boldsymbol{u}}$ to solve linear systems with matrix $G'G'^{T}$. In the case of time dependent parameters $\boldsymbol{\alpha}_n$, the diagonal blocks of $G'_{\boldsymbol{u}}G'^{T}_{\boldsymbol{u}}$ are modified and the overall block tridiagonal structure in maintained in $G'G'^{T}$. In 2.6.4 we illustrate this approach with numerical experiments.

2.4 Tangent space splitting of Newton's method

In the previous section we demonstrated that Newton's method applied to the residual (1.6) may converge from noisy observations to a model trajectory. On the other hand the computational and memory costs of the full Newton's method may be high. We will see that when the number of nonnegative Lyapunov exponents is moderate, substantial savings may be realized by computing Newton's updates only in the non-stable directions.

We start by decomposing the relation (2.8) into the equivalent system

$$\begin{aligned} \mathcal{P}G\left(\boldsymbol{u}^{(k)} + \widehat{\mathcal{P}}\boldsymbol{\delta}^{(k)} + (I - \widehat{\mathcal{P}})\boldsymbol{\delta}^{(k)}\right) &= 0, \\ (I - \mathcal{P})G\left(\boldsymbol{u}^{(k)} + \widehat{\mathcal{P}}\boldsymbol{\delta}^{(k)} + (I - \widehat{\mathcal{P}})\boldsymbol{\delta}^{(k)}\right) &= 0. \end{aligned}$$

Here, $\mathcal{P}$ and $\widehat{\mathcal{P}}$ are block diagonal projection matrices $\mathcal{P} = \text{blockdiag}\,(P_1, \cdots, P_N)$ and $\widehat{\mathcal{P}} = \text{blockdiag}\,(P_0, \ldots, P_N)$, where $P_0, P_1, ..., P_N \in \mathcal{R}^{d\times d}$ are projection matrices onto the non-stable subspace at time levels $n = 0, 1, \ldots, N$, respectively.

We propose to modify the Newton's iteration as follows. Instead of computing the update $\boldsymbol{\delta}^{(k)}$ by simultaneously solving the above system, we split the iterate into updates in the range and complement of $\widehat{\mathcal{P}}$. We also allow the projection operators $\mathcal{P}$ and $\widehat{\mathcal{P}}$ to be updated in each iteration. In the kth iteration, we first approximate the update in the range of $\widehat{\mathcal{P}}^{(k)}$, neglecting the term $(I-\widehat{\mathcal{P}}^{(k)})\boldsymbol{\delta}^{(k)}$ in the first equation above and solving

$$\mathcal{P}^{(k)} G\left(\boldsymbol{u}^{(k)} + \widehat{\mathcal{P}}^{(k)}\boldsymbol{\delta}^{(k)}\right) = 0 \tag{2.12}$$

for ${\boldsymbol{\delta}_{||}}^{(k)} = \widehat{\mathcal{P}}^{(k)}\boldsymbol{\delta}^{(k)}$. Next we approximate the update in the complement of $\widehat{\mathcal{P}}^{(k)}$ by solving

$$(I-\mathcal{P}^{(k)})G(\boldsymbol{u}^{(k)} + \widehat{\mathcal{P}}^{(k)}\boldsymbol{\delta}^{(k)} + (I-\widehat{\mathcal{P}}^{(k)})\boldsymbol{\delta}^{(k)}) = 0 \tag{2.13}$$

for ${\boldsymbol{\delta}_{\perp}}^{(k)} = (I-\widehat{\mathcal{P}}^{(k)})\boldsymbol{\delta}^{(k)}$. Then the update is computed as $\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + {\boldsymbol{\delta}_{||}}^{(k)} + {\boldsymbol{\delta}_{\perp}}^{(k)}$. Expressions (2.12) and (2.13) are solved approximately for the components ${\boldsymbol{\delta}_{||}}^{(k)}$ and ${\boldsymbol{\delta}_{\perp}}^{(k)}$ as described below.

2.4.1 Computation of projection matrices

The basis Q_n^u, $n = 0, \ldots, N$, for the non-stable tangent space along the true trajectory $\{\mathcal{X}_n\}$ is unknown. Instead, we approximate the Q_n^u along the most recent approximate trajectory $\{u_n^{(k)}\}$. In each iteration we update the projection matrices $P_n^{(k)}$ that project onto the non-stable tangent space. In the kth iteration we choose $P_n^{(k)} = Q_n^{u(k)}(Q_n^{u(k)})^T$, where $Q_n^{u(k)} \in \mathcal{R}^{d\times p}$ is a columnwise orthonormal matrix defined via the iteration (2.6) linearized along the most recently updated pseudo-orbit $\boldsymbol{u}^{(k)}$. That is, we take $x_n = u_n^{(k)}$, $n = 0, \ldots, N$, in (2.6). For the first iteration we use the observations: $u_n^{(0)} = y_n$, $n = 0, \ldots, N$.

The dimension p of the orthonormal basis $Q_n^{u(k)}$ should be equal to or greater than the number of non-negative Lyapunov exponents. In practice we take p to be a few more than the number of non-negative Lyapunov exponents to enhance the convergence rate of the synchronization step below [25].

2.4.2 Newton's step on the unstable space

Linearization of (2.12) yields a projected linear system for the update ${\boldsymbol{\delta}_{||}}^{(k)} = \widehat{\mathcal{P}}^{(k)}\boldsymbol{\delta}^{(k)}$:

$$\mathcal{P}^{(k)} G'(\boldsymbol{u}^{(k)})\widehat{\mathcal{P}}^{(k)}\boldsymbol{\delta}^{(k)} = -\mathcal{P}^{(k)} G(\boldsymbol{u}^{(k)}). \tag{2.14}$$

Supressing the iteration index k for the moment, define block matrices $\mathcal{Q} = \text{blockdiag}\,(Q_1^u, \ldots, Q_N^u)$ and $\widehat{\mathcal{Q}} = \text{blockdiag}\,(Q_0^u, \ldots, Q_N^u)$, and note the relations $\mathcal{Q}\mathcal{Q}^T = \mathcal{P}$, $\mathcal{Q}^T\mathcal{Q} = I$ with analogous expressions for $\widehat{\mathcal{Q}}$. Let

$\boldsymbol{\mu} = \widehat{\mathcal{Q}}^T\boldsymbol{\delta} = \widehat{\mathcal{Q}}^T\widehat{\mathcal{P}}\boldsymbol{\delta}$, $\widetilde{G}' = \mathcal{Q}^T G'(\boldsymbol{u})\widehat{\mathcal{Q}}$ and $\boldsymbol{b} = \mathcal{Q}^T G(\boldsymbol{u})$. Then the linear system for the update $\boldsymbol{\mu}$ may be written as

$$\widetilde{G}'\boldsymbol{\mu} = -\boldsymbol{b}, \tag{2.15}$$

where the matrix $\widetilde{G}'$ has the block structure

$$\widetilde{G}' = \begin{bmatrix} -R_0^u & I & & & \\ & -R_1^u & I & & \\ & & \ddots & \ddots & \\ & & & -R_{N-1}^u & I \end{bmatrix},$$

and consequently, $\widetilde{G}' \in \mathcal{R}^{Np\times(N+1)p}$. We solve (2.15) using the right pseudoinverse $\widetilde{G}'^{\dagger} = \widetilde{G}'^T(\widetilde{G}'\widetilde{G}'^T)^{-1}$ and define the intermediate update

$$\bar{\boldsymbol{u}}^{(k)} = \boldsymbol{u}^{(k)} + \widehat{\mathcal{P}}^{(k)}\boldsymbol{\delta}^{(k)} = \boldsymbol{u}^{(k)} + \widehat{\mathcal{Q}}^{(k)}\boldsymbol{\mu}^{(k)}. \tag{2.16}$$

2.4.3 Synchronization step in the stable space

We next turn to the treatment of (2.13). Inserting the definition (2.16) into (2.13) yields the relation

$$(I - \mathcal{P}^{(k)})G(\bar{\boldsymbol{u}}^{(k)} + (I - \widehat{\mathcal{P}}^{(k)})\boldsymbol{\delta}^{(k)}) = 0, \tag{2.17}$$

whose solution for ${\boldsymbol{\delta}_\perp}^{(k)} = (I - \widehat{\mathcal{P}}^{(k)})\boldsymbol{\delta}^{(k)}$ we wish to approximate. Again dropping the iteration index k for the moment, we expand (2.17) componentwise over the time index $n = 0, \ldots, N-1$.:

$$\begin{aligned} 0 &= \left[(I - \mathcal{P})G\left(\bar{\boldsymbol{u}} + (I - \widehat{\mathcal{P}})\boldsymbol{\delta}\right)\right]_n \\ &= (I - P_{n+1})\left[\bar{u}_{n+1} + (I - P_{n+1})\delta_{n+1} - F_n(\bar{u}_n + (I - P_n)\delta_n)\right]. \end{aligned}$$

The second equation is rewritten in the form

$$(I - P_{n+1})\delta_{n+1} = (I - P_{n+1})\left(F_n(\bar{u}_n + (I - P_n)\delta_n) - \bar{u}_{n+1}\right).$$

Adding $\bar{u}_{n+1}$ to both sides of this equation we get

$$\bar{u}_{n+1} + (I - P_{n+1})\delta_{n+1} = P_{n+1}\bar{u}_{n+1} + (I - P_{n+1})F_n(\bar{u}_n + (I - P_n)\delta_n),$$

or, defining $u_n^{(k+1)} = \bar{u}_n^{(k)} + (I - P_n)\delta_n^{(k)}$,

$$u_{n+1}^{(k+1)} = P_{n+1}\bar{u}_{n+1}^{(k)} + (I - P_{n+1})F_n(u_n^{(k+1)}). \tag{2.18}$$

The form of this iteration is identical to that of the receiver equation (2.7b) in the synchronization process. In other words, given the update $\bar{\boldsymbol{u}}^{(k)}$, corrected in the non-stable subspace (2.16), the correction to the stable subspace can be

implemented through a forward synchronization integration (2.18). In the Appendix, we prove that under suitable assumptions, if after k iterations, the error in $\bar{\boldsymbol{u}}^{(k)}$ exists entirely in the stable tangent space in the sense that $P_n \bar{u}_n^{(k)} = \bar{u}_n^{(k)}$ and $\|P_n \mathcal{X}_n - \bar{u}_n^{(k)}\| < \epsilon$, then the forward integration (2.18) converges exponentially to $\mathcal{X}_n$ as $n \to \infty$.

To summarize, the complete iteration step consists of:

1. Compute the approximate basis Q_n^u, $n = 0, \ldots, N$, for the tangent bundle along the pseudo-trajectory $\{u_n^{(k)}\}$.

2. Solve the linear system (2.14) for the update ${\boldsymbol{\delta}_\perp}^{(k)}$ in the non-stable subspace, and compute the intermediate update $\bar{\boldsymbol{u}}^{(k)}$ from (2.16).

3. Synchronize in the stable subspace using the forward iteration (2.18) to obtain $\boldsymbol{u}^{(k+1)}$.

The Newton's step in the unstable subspace is based upon residual ($G_n(\boldsymbol{u}) := u_{n+1} - F_n(u_n)$) correction with both the residual and the correction projected into the unstable subspace. If $P_n G_n(\boldsymbol{u}) = G_n(\boldsymbol{u})$ for all n, i.e, the residual is wholly within the unstable subspace, then the synchronization step in the stable subspace is trivial with $(I - P_n)\delta_n \equiv 0$ for all n. Thus, provided the Newton's iteration converges, all residual correction occurs within the unstable subspace. In the more general case in which the residual is contained, at least for some n, in both the stable and unstable subspaces, then the initialization of the synchronization step makes possible a reduction of the residuals in the stable subspace. This then generates an updated approximate trajectory to linearize about and obtain updated projections. In this case the Newton's step in the unstable subspace may again decrease the residual with respect to these new projections. The process then continues in the updated stable subspace and we continue until the desired tolerance is achieved or the method fails for lack of convergence of the projected Newton's iteration. In general the projected Newton's iteration will converge provided the residuals are small enough as compared to the strength of the hyperbolic structure (exponential dichotomies, etc.) in the projected system.

What we have observed is that better results are obtained by switching after each projected Newton's iterate to the synchronization step as opposed to switching to the synchronization after the projected Newton's has converged to tolerance. We attribute this to the variation in the projections that are produced. We note here that the basic splitting based upon projection into unstable and stable parts allows for different techniques to be employed for each subsystem. It provides a representation for the unstable subspace which we believe will prove useful in assessing the effectiveness and uncertainties in data assimilation techniques. We also emphasize that in contrast with traditional data assimilation techniques but similar to PDA, the only influence of the observations is via the initial guess for the projected

Newton's/synchronization scheme. Thus, in a perfect model scenario convergence to a solution depends on the initial guess being within its basin of attraction.

In the next section we demonstrate the algorithm and compare it to 4DVar for a number of test problems.

2.5 Implementation

In this section we provide details of the algorithm we implement and discuss some possible variations. The algorithm is "interval sequential" in the sense that the shadowing refinement is applied over an entire subinterval. This has the effect of simultaneously incorporating all observations over this subinterval into a single refinement step. In order to transition between subintervals we impose a continuity constraint in the stable subspace. Also discussed in this section are methods for obtaining an initial approximation of the solution trajectory. This is needed in order to determine the initial projections on each subinterval.

When observations are not available at every time step, we can redefine F to be the map corresponding to the composition of several time steps (examples are given in 2.6.2,2.6.3). When observations are not of the full model state, we can first apply a preprocessing step to the observations to infer an estimate of the full state at all observation times and then perform the main algorithm with the goal of substantial noise reduction. For the PDA method, where the same issue arises, this completion has been done using a variational analysis [54] or by just inserting climatological means for missing observations [31, 32]. In 2.6.1, we demonstrate an alternative preprocess motived by synchronization, whereby the observation data is directly inserted as a driving signal. The effectiveness of such an approach relies on the ability of the partial observational data to constrain the unstable tangent space. However, it is one of the main conclusions of this chapter that such a requirement on the data must hold anyway, if data assimilation is to be effective.

In our implementation, we decompose the time interval $t \in [0, T]$ of integration into $M+1$ non-overlapping time windows, and the data assimilation method is applied sequentially on each of these. We identify times τ_m, $m = 0, \ldots, M+1$, where $\tau_0 = 0$, $\tau_{M+1} = T$, $\tau_1 \in (0, T)$ is the length of the first time window, and $\tau_m = \tau_1 + (m-1)\Delta\tau$, $\Delta\tau = (T - \tau_1)/M$. The mth time window is the interval $t \in [\tau_{m-1}, \tau_m]$. In each window, an initial condition $(\delta_\perp)_0$ is needed for the synchronization step, and convergence of the stable directions requires this quantity to be small (see Appendix). In particular we implement (2.18) as

$$(I - P_{n+1})\delta_{n+1} = (I - P_{n+1})[F(\bar{u}_n^{(k)} + (I - P_n)\delta_n) - \bar{u}_{n+1}^{(k)}].$$

The initial condition $(\delta_\perp)_0 = (I - P_0)\delta_0$ on time window m is determined by

imposing

$$(I - P_0)\delta_0 = (I - P_0)[v_T - u_0^{(k)}]$$

where v_T is the converged iterate u at the terminal time on the time window $m-1$. Effectively, by imposing continuity in the stable directions during the full assimilation, also across window boundaries, when solving (2.18) we can define a unique solution (this analysis point of view is also taken up in [47] and [42]). To obtain a good initial condition for the algorithm we perform smoothing on an initialization window: i.e. we employ the full Newton's algorithm (see 2.3) on a short window and start the forced system (2.18) from there. This also improves the approximation of the unstable directions at the beginning of the window at which the projected method is started.

2.6 Numerical experiments

In the preceding sections we have outlined a data assimilation method based on a tangent space splitting into stable and non-stable subspaces. As described, the method assumes noisy observations of the full state of the system (i.e. observation operator H the identity map on $\mathcal{R}^d$) at each time step, and no restrictions are placed on the length of the time interval.

In this section we demonstrate the behavior of the method for low dimensional test problems: the Lorenz models L63 and L96. We study dependence on dimension of the projection operator and window lengths. We compare the method with 4DVar, and investigate the approaches for incomplete observations and parameter estimation.

In all experiments, the observations are generated from the truth by adding i.i.d. zero-mean Guassian noise as in equation (1.5) with diagonal covariance matrix $E = \nu^2 I$, where ν^2 denotes the variance of the noise process. As convergence criterion for the projected Newton's method we use that $\frac{\|\boldsymbol{b}\|_2}{\|\boldsymbol{u}\|_2} < 10^{-15}$, where $\boldsymbol{b}$ is the projected residual in (2.15).

2.6.1 Dependence on projector in the L63 model

The well-known Lorenz attractor [69] is a chaotic dynamical system commonly used as a test problem for data assimilation algorithms. The L63 model is

$$\dot{x}_1 = \sigma(x_2 - x_1), \quad \dot{x}_2 = x_1(\rho - x_3) - x_2, \quad \dot{x}_3 = x_1 x_2 - \beta x_3 \tag{2.19}$$

where $\sigma = 10$, $\beta = \frac{8}{3}$ and $\rho = 28$. The Lyapunov exponents of the Lorenz attractor are $\lambda_1 \approx 0.906$, $\lambda_2 = 0$, $\lambda_3 \approx -14.572$.

For the experiments in this section we generate a (single) set of observations computing a trajectory of L63 on $t \in [0, 20]$ with $T = 20$, using time step

Table 2.3: Application of the algorithm to L63 with $P = P_{x_1}$. Results are unsatisfactory. Please recall C and MSE are defined in equations (2.11) and (2.10) respectively.

Property	**Value**
Observation error $C(\{X_n\})$	11.9
Distance between Estimate and Observations $C(\boldsymbol{u})$	367
Error between Estimate and the Truth MSE	356

$\Delta t = 0.005$, and $\nu^2 = 4$. In all experiments in this section we use an assimilation window of length $\Delta\tau = 2.5$.

In 2.3 we observed that the full Newton's method successfully assimilates observations into L63. Now, we examine the proposed algorithm with projected Newton's and synchronization. Since the L63 model can be synchronized by coupling of the x_1-variables [79, 47], it is natural instead of computing Lyapunov vectors to try to take $P = P_{x_1}$, hence always projecting on the x_1-coordinate, and to iterate (2.16) and (2.18). Errors (2.10)–(2.11) are given in 2.3, where it is clear that for our algorithm the choice $P = P_{x_1}$ is insufficient to obtain an orbit that is close to observations. Since the projection operators P generally do not commute with the forward model solution operator F, the projected Newton's method does not yield a projection of the full model solution, which means in particular that there are important differences between our algorithm and synchronization in the sense of [79, 47].

Therefore we consider the projection operator on the subspace spanned by Lyapunov vectors. First, we choose the dimension of the projection operator to be $p = 1$. This means we use Newton's method in the (approximate) unstable direction, but not in the neutral or stable direction, because the L63 model has one positive, one zero and one negative Lyapunov exponent. This is not sufficient for Newton's method to always converge, since the method works well until $t = 20$ and after that Newton's method diverges. The results up to $t = 20$ are shown in 2.4, where in addition to errors we also display a measure of discontinuity at window boundaries defined as

$$D = \frac{1}{N}\sum_{n=1}^{N} \max |G_n(\boldsymbol{u})| \tag{2.20}$$

and the average number of iterations needed for Newton's method to converge is denoted as #. We remark that in principle we could restart the method using full Newton's at $t = 20$ and then continue with $p = 1$.

Next, we choose the dimension of the projection operator to be $p = 2$. This means we apply Newton's method to both the unstable and neutral direction. The results are shown in 2.5, 2.4, where it can be seen that the algorithm

Table 2.4: Application of the algorithm to L63 with P_1. Results for the time up to 20, since after that the algorithm diverges. Please recall C, MSE and D are defined in equations (2.11), (2.10) and (2.20) respectively.

Property	**Value**
Observation error $C(\{X_n\})$	11.9
Distance between Estimate and Observations $C(\boldsymbol{u})$	12.2
Error between Estimate and the Truth MSE	0.27
Discontinuity measure at window boundaries D	0.23
Average number of iterations #	8.7

Table 2.5: Application of the algorithm to L63 with P_2. Good results are obtained. Please recall C, MSE and D are defined in equations (2.11), (2.10) and (2.20) respectively. Numbers shown in the table are averages over 100 noise realizations, together with the corresponding standard deviations.

Property	**Value**
Observation error $C(\{X_n\})$	12.00 ± 0.17
Distance between Estimate and Observations $C(\boldsymbol{u})$	12.06 ± 0.18
Error between Estimate and the Truth MSE	0.09 ± 0.07
Discontinuity measure at window boundaries D	0.29 ± 0.08
Average number of iterations #	6.52 ± 0.15

becomes stable. Thus it is necessary to apply the projected Newton's method to both the unstable and neutral directions in this example. We repeated this numerical experiment for 100 different noise realizations. The case $p = d = 3$ is the full Newton's method, which gives the smallest MSE as was shown in 2.3. We remark, however, that in this section initialization (full Newton's) is performed only on the first assimilation window, which reduces computational costs. We conclude that the algorithm is capable of recovering a good approximation of the true trajectory and that this approximation *is* a trajectory of the L63 model.

As mentioned in the beginning of 2.6, observations of the full model state are not feasible and thus we need to relax this assumption. Therefore we now assume that the only available observations are of the x_1-coordinate. First, we perform a preprocessing procedure in order to complete the missing observations: we run (2.7b) with observations of x_1 as driving signal and $H\mathcal{X}_n = P_{x_1}\mathcal{X}_n$, $n = 0, \ldots, N$, as coupling. Subsequently, we apply the main algorithm with thus completed observations (which generally contain large errors due to the preprocessing). For the main algorithm we choose $p = 2$. Results are shown in 2.6, where we see that when only one coordinate is observed the error can be reduced and information on other coordinates can be obtained with synchronization as the preprocessing procedure.

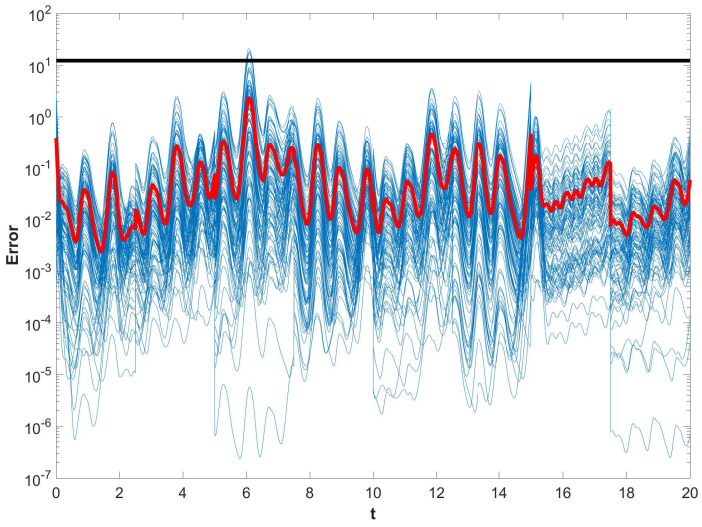

Figure 2.4: Application of the algorithm to L63 with P_2. We compare the time-averaged observational error (black) with the error of the estimations (blue) over time. We used 100 observational noise realizations. The average estimation error over time is shown in red.

Table 2.6: Application of the algorithm to L63 with P_2 and observations of the x_1-coordinate only. Synchronization is used as preprocessing step and errors are reduced by the projected Newton's method. Please recall C, MSE and D are defined in equations (2.11), (2.10) and (2.20) respectively.

Property	**Value**
Observation error $C(\{X_n\})$	3.97
Distance between Estimate and Observations $C(\boldsymbol{u})$	4.32
Error between Estimate and the Truth MSE	2.49
MSE for the observed coordinate	0.37
Discontinuity measure at window boundaries D	1.16
Average number of iterations #	7.0

2.6.2 Dependence on window length in the L96 model

Lorenz [70] proposed the following model as an example of a simple one-dimensional model with features of the atmosphere. The L96 model is

$$\dot{x_l} = -x_{l-2}x_{l-1} + x_{l-1}x_{l+1} - x_l + \mathcal{F}, \qquad (l = 1, ..., d), \qquad (2.21)$$

where the dimension d and forcing $\mathcal{F}$ are parameters. Cyclic boundary conditions are imposed. We implement the L96 model with the standard parameter choices $d = 36$ and $\mathcal{F} = 8$. The differential equations are discretized with a forward Euler scheme with time step $\tau = 0.005$ and the model initial conditions are chosen at random (standard Gaussian iid). Observations are obtained by perturbing a reference (true) trajectory with random Gaussian iid noise with zero mean and covariance with $\sigma = 0.3$. However, the observations are not drawn at every time step as for the L63 model but only every tenth time step, corresponding to observing a full model state every 6 hours. Then the map F_n (1.1) corresponds to ten forward Euler steps. This map is used to define G and the derivatives of this map are needed for the QR-decompositions and Newton's iteration. For the synchronization we observe that if $G(\boldsymbol{u}) = 0$, then we also have a trajectory under the forward Euler discretization with time step τ. This means that any model integration can just be done with the original discretization, with forcing only applied at points where we have observations.

For the projected Newton's method the dimension of the non-stable subspace p is chosen to be either 15 or 25. We carry out numerical experiments for various choices of window lengths: we use initialization windows with lengths between 0.75 and 15 time units and following windows with lengths between 0.75 and 5 time units. The total time length of assimilation is always 75 and identical observations are used for all choices of p and the respective window length. In 2.5,2.6 we show the error over time for a particular realization. In 2.7, 2.8 we show mean values and standard deviations for 20 realizations of true trajectory and noise. Initial conditions are generated at random and we use a spin-up time of 5, after which we have observations and assimilation over a total time of 75. It can be seen that the algorithm works well for both long and short windows, although when the windows are too long or too short the results deteriorate. In general, higher p decreases the estimation errors, although for the optimal choice of the window lengths—initialization window of 2.5 and following windows of 1.25—projection on $p = 15$ may result in better estimation, also see 2.5.

In 2.9 we investigate in more detail the dependence on the dimension of the projection p. We take total dimension $d = 40$ and illustrate how the distance of the refined orbit from observations undergoes a sharp transition around the number of positive Lyapunov exponents, which is equal to 13. This sharp transition has also been observed for 4DVar-AUS [94, 78] The values that are

Table 2.7: Application of the algorithm to L96 with P_{15}. Please recall C, MSE and D are defined in equations (2.11), (2.10) and (2.20) respectively. Averages and standard deviations over 20 trajectory and noise realizations. For window length 5, there where 4 experiments in which the method did not converge. Those results have been left out.

Property	**Value**			
Window length	5		2.5	
Initial window	15	5	15	5
Obs. err. $C(\{X_n\})$	3.23 ± 0.01			
Dist. Est. & Obs. $C(\boldsymbol{u})$	4.64±0.92	4.91±0.71	3.46±0.13	3.33±0.06
Err. Est. & Truth MSE	1.44±0.91	1.71±0.71	0.28±0.13	0.16±0.06
D	1.07±0.27	1.05±0.20	0.33±0.05	0.31±0.04
Avg. # its.	10.6±0.2	10.8±0.3	8.5±0.1	8.5±0.1

Property	**Value**			
Window length	2.5	1.25		0.75
Initial window	2.5		1.25	0.75
Obs. err. $C(\{X_n\})$	3.23 ± 0.01			
Dist. Est. & Obs. $C(\boldsymbol{u})$	3.33±0.04	3.23±0.03	3.25±0.03	3.50±1.09
Err. Est. & Truth MSE	0.16±0.03	0.11±0.03	0.13±0.03	0.46±1.12
D	0.31±0.03	0.24±0.03	0.25±0.02	0.31±0.11
Avg. # of its.	8.5±0.1	7.3±0.1	7.4±0.1	7.01±0.04

Table 2.8: Application of the algorithm to L96 with P_{25}. Please recall C, MSE and D are defined in equations (2.11), (2.10) and (2.20) respectively. Averages and standard deviations over 20 trajectory and noise realizations.

Property	**Value**			
Window length	5		2.5	
Initial window	15	5	15	5
Obs. err. $C(\{X_n\})$	3.23 ± 0.01			
Dist. Est. & Obs. $C(\boldsymbol{u})$	4.20±0.33	4.18±0.41	3.37±0.12	3.25±0.03
Err. Est. & Truth MSE	1.00±0.32	0.99±0.41	0.21±0.12	0.10±0.02
D	1.01±0.19	0.96±0.17	0.31±0.03	0.30±0.03
Avg. # of its.	9.2±0.2	9.2±0.2	7.9±0.1	7.9±0.1

Property	**Value**			
Window length	2.5	1.25		0.75
Initial window	2.5		1.25	0.75
$C(\{X_n\})$	3.23 ± 0.01			
$C(\boldsymbol{u})$	3.25±0.03	3.15±0.01	3.15±0.01	3.09±0.01
Est. & Tr. MSE	0.10±0.02	0.096±0.004	0.098±0.004	0.156±0.004
D	0.30±0.03	0.26±0.01	0.26±0.01	0.30±0.01
Avg. # of its.	7.93±0.04	7.01±0.02	7.01±0.02	6.87±0.03

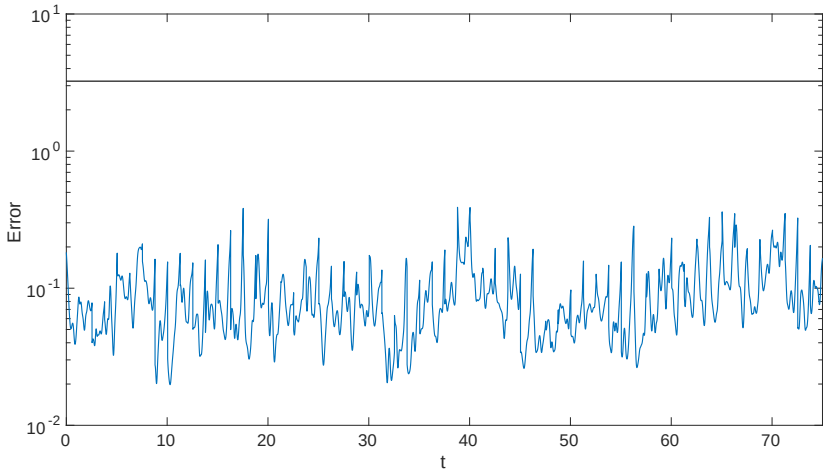

Figure 2.5: The best window length choice the algorithm, applied to L96 with $\sigma = 0.3$ and observations every 10 time steps. We use $p = 15$. The initialization window has length 2.5, following windows have length 1.25. In this realization, the mean square error MSE=0.09. The mean observational error is in black and the error over time of the estimation is in blue.

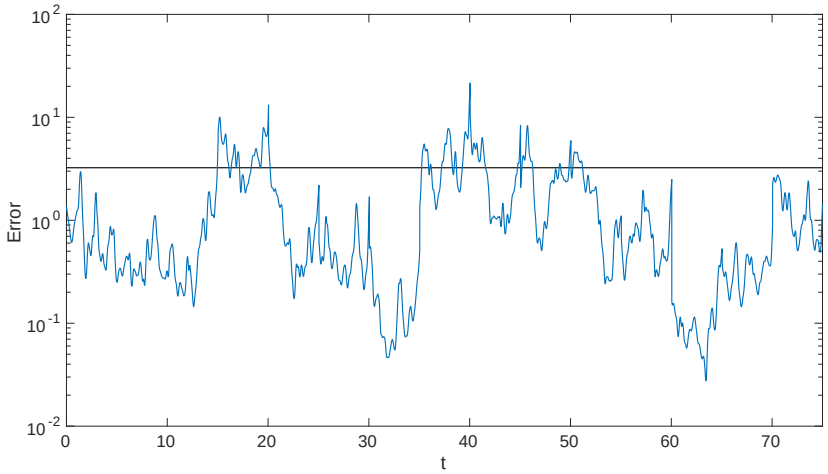

Figure 2.6: Here we use $p = 15$, with the longest window lengths: initialization window 15, following windows 5. Trajectories are found, but if error reduction is the main purpose longer windows do not necessarily result in smaller errors. In this realization, the mean square error MSE=1.47. The mean observational error is in black and the error over time of the estimation is in blue.

reported are based upon uniformly distributed noise in the interval $[-2, 2]$ and upon a total assimilation time of 4, subdivided into 4 windows of length 1. At the first window full Newton's is used (i.e. $p = 40$) and at the subsequent three windows p is as specified in crefLor95tab1; $C(\boldsymbol{u})$ is then computed over all 4 windows.

Table 2.9: Distance between Estimate and Observations $C(\boldsymbol{u})$ for L96.

p	5	10	13	15	20	30	40
$C(\boldsymbol{u})$	149.2	77.5	73.1	59.2	55.4	53.2	53.2

It is possible to carry out the same experiment using more projected windows of length 1, to achieve a total length of 10 or 20. However, for small p, the results get worse if the total length increases. For length 50 divergence of Newton's method is observed. So, the table can be reconstructed qualitatively, but not for unlimited times. At some point new spin-up windows with full-Newton's ($p = 40$) are needed. This eventual instability also occurs for L63 with $p = 1$. This problem does not occur when p is chosen large enough. Carrying out the numerical experiment with $p = 20$ results in the average distance to observations remaining compatible with the noise level for time lengths up to 2500 (using windows of length 1).

This illustrates that it is important to define the non-stable space to be large enough, ensuring the $(I - P)$-problem does not contain neutral or unstable directions. If p is chosen too small, the initialization at the spin-up window with full Newton's keeps the error somewhat in check over a few projected windows, but as we progress even further in time projected Newton's on an insufficiently large subspace is unable to keep the error in check. When errors get larger, this will eventually lead to divergence of Newton's method, but already before that the results from the data assimilation get progressively worse. However, if we choose p to be large enough, the method remains stable over long times.

2.6.3 Comparison to 4DVar

In the above sections we have argued that our algorithm aims at the same goals as the 4DVar algorithm and that for the L63 and L96 models we are able to reconstruct good trajectories based on observations. We now make a comparison with the standard 4DVar algorithm and demonstrate that our approach is a good alternative.

We perform a test using the L96 model with the same parameters as in the section above. Observations are drawn every fifth time step, which means we observe the full state every 0.025 time units, corresponding to 3 hours. We set $p = 25$. In our tests we use identical data, models and windows for both methods. We choose 25 windows of length 1, of which the first is used as initialization window for the shadowing method. On the initialization window

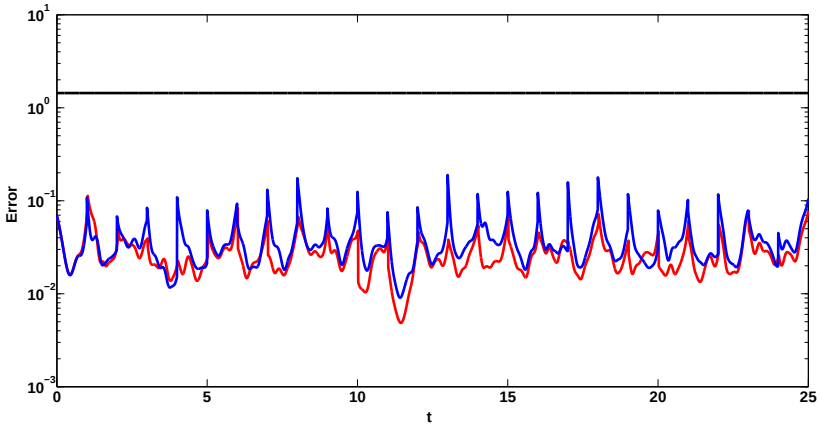

Figure 2.7: The error between the results of the approximation methods and the truth over time, measured using the ∞-norm. The error of 4DVar is shown in blue, the error for the shadowing method is shown in red and the mean observational error is in black.

Table 2.10: The results from the projected Newton's algorithm and 4DVar are of comparable quality, but convergence is much more quick for the projected Newton's algorithm.

Property	**Value**	
Method	Projected Newton's	4DVar
Obs. err. $C(\{X_n\})$	1.43	
Dist. Est. & Obs. $C(\boldsymbol{u})$	1.40	1.39
Err. Est. & Truth MSE	0.027	0.037
D	0.14	0.17
Avg. # of its.	6.3	418.3

5 iterations are needed for Newton's algorithm to converge. The initialization of 4DVar at the beginning of the first window is done with the first observation, since for neither of the two methods we have any prior knowledge of the system state. We do not use a background term for 4DVar. The gradient computation in the 4DVar method is done using the adjoint integration and the optimization is performed by a conjugate gradient method. Results are shown in 2.6.3, 2.10.

We can see in 2.10 that the 4DVar method returns a result that is slightly closer to the observations than the projected Newton's method, while the projected Newton's method is slightly closer to the truth. This minor difference might be related to the projection on the unstable space used for the projected Newton's method [94].

It can be seen that 4DVar and the projected Newton's method give comparable results in this test case, but that the projected Newton's method is faster and more suitable if we were to use longer windows. The choice of window length is determined by the requirement that the 4DVar method should still converge. Even though the results in the previous section suggest slightly longer windows would be better, this is not viable for this test. In fact, the number of iterations could be significantly reduced for 4DVar by shortening the window, but this would come at the cost of not taking enough data into account.

If the frequency of observations is significantly lowered, there may be benefits to using 4DVar compared to the projected Newton's method. Since the projected Newton's method makes use of linearizations of the model computed only at observation times, we expect the method to perform worse if observations are far enough apart in time for nonlinearities to be significant. In some cases it may be useful to combine 4DVar and the projected Newton's method. For instance, one could use the solution of the projected Newton's method to initialize the 4DVar algorithm.

We remark that one iteration of the projected Newton's method is more costly than one iteration of the 4DVar algorithm; if most directions are stable the difference in iteration cost would be less strong. In any case, the higher cost per iteration step of the projected Newton's method is more than compensated for since it requires far fewer iterations. For sufficiently long windows, the cost per iteration of 4DVar is dominated by the need to do one model integration and one adjoint integration, which scales as $\mathcal{O}(Nc)$, where c is defined as the typical cost of taking one time step in the non-linear model. An implementation of the full Newton's method in which the Jacobians are formed explicitly and are treated as dense matrices yields a cost per iteration of $\mathcal{O}(dNc)$ for the integrations needed and $\mathcal{O}(Nd^3)$ for solving the resulting linear system with a block tridiagonal method. The use of the projected Newton's method reduces this cost to $\mathcal{O}(pNc) + \mathcal{O}(Ndp^2)$. The main factors contributing to the cost per iteration of the projected Newton's method are two model integrations, p tangent linear model integrations and the application of the modified Gram-Schmidt method to a $p \times d$-matrix at each time step.

Reasons for the large difference in number of iterations needed for convergence could be that Newton's method has a quadratic convergence rate, while the optimization algorithm for 4DVar does not. We remark that it is not possible to choose an algorithm with quadratic convergence rate for 4DVar, since we do not have the Hessian of the 4DVar cost function available. A more important reason for the strong difference in number of iterations needed for convergence may be in the fact that 4DVar and projected Newton's really solve very different problems. An explanation for the large difference in needed iterations for this example and the robustness of the projected Newton's method can be found by analyzing what happens when window lengths are increased.

The (projected) Newton's method has to solve larger (but weakly coupled and

Table 2.11: Estimation of σ by shadowing-based data assimilation methodology from 2.3. The true value is 10.

Property	**Initial guess for σ**			
Initial σ	5	10	15	20
Obs. err. $C(\{X_n\})$	2.97			
Dist. Est. & Obs. $C(\boldsymbol{u})$	2.98	2.96	2.96	2.98
Err. Est. & Truth MSE	0.03	0.02	0.03	0.07
Estimated σ	10.08	10.03	10.05	10.06

not that strongly nonlinear) problems if interval length is increased, while for the 4DVar approach the size of the optimization problem stays constant, but the problem becomes more and more highly nonlinear as the interval length is increased. This problematic behavior of the 4DVar optimization problem is well known in the literature [6, 74, 82]. This can then lead to a large number of iterations needed for convergence, convergence to highly suboptimal local minima, or even to non-convergence of the 4DVar optimization. This difference between a large but weakly coupled and relatively easy root-finding problem that can be solved with an efficient method compared to a small but highly nontrivial optimization problem for which a slightly slower method has to be employed may give rise to the observed performance difference between the methods, both in terms of iterations needed to converge (and hence time needed to converge) and in the ability to still work for longer windows.

2.6.4 Parameter estimation

As described in 2.3, shadowing-based data assimilation methodology can be applied to the problem of parameter estimation. The results of σ estimation for the L63 model are shown in 2.11, where different values of initial σ were chosen—5, 10, 15, and 20—with the true σ being 10. Gaussian noise with identity covariance is added to the true solution and data assimilation is performed over one window of length 5 (when data assimilation is performed over multiple windows an estimate from the previous window can be taken as an initial parameter for the next window). It should be noted that similar results can be obtained for ρ or β estimation of the L63 model and for $\mathcal{F}$ estimation of the L96 model.

In 2.12, we show σ estimations obtained by 4DVar using a window length of 0.25. Instead of using the method of 2.3, we can also introduce trivial equations for the parameters to the shadowing-based data assimilation, which introduces extra zero Lyapunov exponents. As can be observed from 2.13 this method fails for σ estimations, though performs sufficiently well for ρ (see 2.14) or β estimations of the L63 model and for $\mathcal{F}$ estimation of the L96 model. Thus, adding trivial equations for the parameters to the shadowing-based data assimilation deteriorates its performance.

Table 2.12: Estimation of σ by 4DVar parameter estimation. The true value is 10.

Property	**Initial guess for σ**			
Initial σ	5	10	15	20
Obs. err. $C(\{X_n\})$	2.97			
Dist. Est. & Obs. $C(\boldsymbol{u})$	2.98	2.97	3.04	3.06
Err. Est. & Truth MSE	0.12	0.12	0.17	0.18
Estimated σ	9.92	9.95	9.91	9.94

Table 2.13: Estimation of σ by shadowing-based data assimilation methodology with trivial model for the parameters. The true value is 10 and the estimated σ is the mean estimate. Cases when Newton's method diverges are denoted by "∞" in the corresponding column.

Property	**Initial guess for σ**			
Initial σ	5	10	15	20
Observation error $C(\{X_n\})$	2.97			
Distance Estimate and Observations $C(\boldsymbol{u})$	360	5.50	∞	∞
Error Estimate and the Truth MSE	355	2.77	∞	∞
Estimated σ	8.85	9.83	∞	∞

Table 2.14: Estimation of ρ by shadowing-based data assimilation methodology with trivial model for the parameters. The true value is 28 and the estimated ρ is the mean estimate.

Property	**Initial guess for ρ**			
Initial ρ	14	28	42	56
Observation error $C(\{X_n\})$	2.97			
Distance Estimate and Observations $C(\boldsymbol{u})$	35.7	2.95	5.52	161
Error Estimate and the Truth MSE	32.7	0.02	2.74	159
Estimated ρ	26.7	28.0	28.4	37

2.7 Conclusions

We have introduced a new class of algorithms for data assimilation based upon shadowing refinement, synchronization, AUS, and PDA techniques. Projections are determined based upon techniques employed in the computation of Lyapunov exponents/vectors, in particular continuous QR techniques. This produces a splitting of the dynamics into non-stable and stable components, which allows for employing different techniques for the different components that are suited to their dynamics. Since the projections are a function of solutions of the state space model, these projection based techniques require at least an approximate solution to determine initial

projections. Assessing the uncertainty in obtaining an initial approximate solution and the impact of these uncertainties on the assimilation is a focus of our future work. These techniques are also amenable in a number of ways to a Bayesian framework and since we obtain an approximation of a time dependent orthonormal basis for the non-stable subspace one can assess the observation operator with respect to the unstable subspace. The stable component has contractive dynamics which is useful for error control and further assessing of uncertainties. In future work, we shall analyze the performance of the algorithm when observations are biased, correlated and/or sparse in time. The algorithm developed here is effective in parameter estimation without introducing a trivial ODE for the parameters as in traditional data assimilation methods. We used a combination of analysis and numerical experiments to show that the algorithm works effectively and we demonstrated that the results compare favorably to those of 4DVar. Rigourous convergence proofs of the method will be addressed in future work. The other avenues for future research include more efficient numerical linear algebra techniques (the shadowing refinement relies on a block tridiagonal linear system solve that we have performed with direct methods) and the use of parallel computing techniques.

2.A Convergence of the synchronization update in the stable subspace

2.A.1 Convergence in the linear, nonautonomous case

We study a synchronization process where there is some error made in the non-stable directions. If the model is linear but non-autonomous and at each step sufficiently close to the identity and the largest Lyapunov exponent of the stable subspace is negative, then the total error of the synchronized solution will not be much larger then the error in the non-stable directions. This holds in particular if the largest Lyapunov exponent of the stable subspace is small enough and if convergence to the Lyapunov exponents in the stable space is quick.

Let $\mathcal{X}_n$ be a solution to the nonautonomous linear model $\mathcal{X}_{n+1} = F'_n \mathcal{X}_n$, for $n \in \mathbb{N}$, and let $Q_{n+1}R_{n+1} = F'_n Q_n$. Let $\bar{u}_n \in \mathcal{R}^d$ be a sequence of vectors approximating the truth in the non-stable subspace as follows

$$P_n \bar{u}_n = \bar{u}_n, \qquad \|P_n \mathcal{X}_n - \bar{u}_n\| < \epsilon,$$

where $P_n := (Q_n^u)(Q_n^u)^T$, the orthogonal projector onto the first p columns of Q_n, i.e. p is the dimension of the non-stable subspace.

Define $\Delta_n := (F'_n - I)$ and $\Delta := \sup_n \|\Delta_n\|_F$. Let w_0 be arbitrary and

$$w_n = (I - P_n)F'_{n-1}\left(\bar{u}_{n-1} + w_{n-1}\right),$$

for $n \geq 1$. Define the error vector v_n as the difference between truth and approximation at time step n:

$$v_n := \mathcal{X}_n - \bar{u}_n - w_n, \tag{2.22}$$

and denote the projection on the κ-th column of Q_n by v_n^κ, for $\kappa = p+1, \ldots, d$. Let $\zeta := \left(\frac{\sqrt{2}\Delta}{1-\Delta} + \left[\frac{\sqrt{2}\Delta}{1-\Delta}\right]^2\right)(1+\Delta)\epsilon$ and $\delta^2 = \left(2\frac{\sqrt{2}\Delta}{1-\Delta}\Delta + \left[\frac{\sqrt{2}\Delta}{1-\Delta}\right]^2(1+\Delta)\right)$. Define modified Lyapunov exponents as $\hat{\lambda}_k := \lim_{n\to\infty} \frac{1}{n}\sum_{l=0}^{n} \log\left(R_\ell^{(k,k)} + \delta^2\right)$, for $k = p+1, \ldots, d$.

Assumption 0.1 *There exists a positive constant $\hat{\varepsilon} > 0$ and a positive integer $\mathcal{N}$ such that for all integers $\mathcal{N}' \geq \mathcal{N}$ and all $n > 0$,*

$$\left|\frac{1}{\mathcal{N}'}\sum_{\ell=0}^{\mathcal{N}'} \log\left(R_{(n+\ell)}^{(k,k)} + \delta^2\right) - \hat{\lambda}_k\right| < \hat{\varepsilon}, \quad k = p+1, \ldots, d.$$

Theorem 1 *Assume $\Delta < 1$, $\hat{\lambda}_{p+1} < 0$. Under (0.1), for any $p < \kappa \leq d$, $m_1 \in \mathbb{N}$ and $m_2 := m_1 + \mathcal{N} + 1$,*

$$|v_{m_2}^{(\kappa)}| \leq e^{(\hat{\lambda}_\kappa + \hat{\epsilon}_\kappa)\mathcal{N}}|v_{m_1}^{(\kappa)}| + \sum_{l=0}^{\mathcal{N}} \zeta \prod_{j=l}^{\mathcal{N}-1}\left(R_{j+m_1+2}^{(\kappa,\kappa)} + \delta^2\right) + \left\{\sum_{\iota=\kappa+1}^{d}\sum_{l=0}^{\mathcal{N}} |R_{l+m_1}^{(\iota,\kappa)} + \delta^2| \prod_{j=l+1}^{\mathcal{N}}\left(R_{j+m_1}^{(\kappa,\kappa)} + \delta^2\right)|v_{l+m_1}^{(\iota)}|\right\}.$$

Corollary 1.1 *Assume that when averaging over all $n \in \mathbb{N}$, for all $m \in \mathbb{N}$ and all $\iota_l, \kappa_l \in \{p+1, p+2, ..., d\}$, with $l \in \{n, n+1, ..., n+m\}$, the average of the product $\Pi_{l=n}^{n+m} R_l^{(\iota_l,\kappa_l)}$ can be expressed as the product of the averages: $\overline{\Pi_{l=n}^{n+m} R_l^{(\iota_l,\kappa_l)}} = \Pi_{l=n}^{n+m} \overline{R_l^{(\iota_l,\kappa_l)}}$. Then*

$$\overline{|v_{m_2}^{(\kappa)}|} < \frac{1}{1 - e^{(\hat{\lambda}_\kappa + \hat{\epsilon}_\kappa)\mathcal{N}}}\left(1 + \frac{1}{|\hat{\lambda}_\kappa|}\left(1 - e^{\hat{\lambda}_\kappa \mathcal{N}}\right)\right)\zeta + \mathcal{O}(\Delta^2),$$

where $\overline{|v_{m_2}^{(\kappa)}|}$ denotes taking the average of $|v_{m_2}^{(\kappa)}|$ over all m_2.

Proof 1 *To prove 1 we use, without loss of generality, a coordinate system such that at step n the orthonormal Lyapunov vectors coincide with the standard*

basis. Then we consider the equation for the projection of the error on the stable space:

$$(I - P_{n+1})v_{n+1} = (I - P_{n+1}) \left[Q_{n+1}R_{n+1}(\mathcal{X}_n - \bar{u}_n - w_n) \right]. \tag{2.23}$$

We now split the right side of (2.23) *between the range and the kernel of* P_n*:*

$$\begin{aligned}
&(I - P_{n+1})Q_{n+1}R_{n+1}(\mathcal{X}_n - \bar{u}_n - w_n) \\
&\quad =(I - P_{n+1})Q_{n+1}R_{n+1}P_n(\mathcal{X}_n - \bar{u}_n - w_n) \\
&\qquad + (I - P_{n+1})Q_{n+1}R_{n+1}(I - P_n)(\mathcal{X}_n - \bar{u}_n - w_n) \\
&\quad =(I - P_{n+1})Q_{n+1}R_{n+1}(P_n\mathcal{X}_n - \bar{u}_n) \\
&\qquad + (I - P_{n+1})Q_{n+1}R_{n+1} \left\{ (I - P_n)\mathcal{X}_n - w_n \right\}.
\end{aligned}$$

We first analyze the contribution from the error term $(I - P_{n+1})Q_{n+1}R_{n+1}(P_n\mathcal{X}_n - \bar{u}_n)$*, which is the contribution of the error parallel to the* p *leading Lyapunov vectors at time* n *to the error perpendicular to these vectors at time* $n + 1$*.*

By the definitions of Q_n*,* R_n *and* Δ_n*, we have that*

$$Q_{n+1}R_{n+1} = I + \Delta_n. \tag{2.24}$$

We recall that $P_n = Q_n^u(Q_n^u)^T$ *and that* Q_n *is the identity matrix in our coordinates. We can approximate* $||I - Q_{n+1}||_F \leq \frac{\sqrt{2}||\Delta_{n+1}||_F}{1-||\Delta_{n+1}||_2}$*, by Theorem 3.1 of [14]. It immediately follows that*

$$\begin{aligned}
&||(I - P_{n+1})Q_{n+1}R_{n+1}(P_n\mathcal{X}_n - \bar{u}_n)||_2 \\
&< \left(\frac{\sqrt{2}||\Delta_{n+1}||_F}{1 - ||\Delta_{n+1}||_2} + \left[\frac{\sqrt{2}||\Delta_{n+1}||_F}{1 - ||\Delta_{n+1}||_2} \right]^2 \right) (1 + ||\Delta_{n+1}||_2)\epsilon \\
&< \zeta.
\end{aligned}$$

For the convergence in the stable directions we proceed analogously to [95]. We remark that R_n *encodes the local approximation to the Lyapunov exponents [37]. We recall that the Gram-Schmidt algorithm ensures that all diagonal elements of* R_n *are positive for all* n*. Using the bound on the contribution of the unstable*

errors to the stable direction, we obtain

$$
\begin{aligned}
&||(I-P_{n+1})v_{n+1}||_2 \qquad (2.25)\\
&< ||(I-P_{n+1})F_n'\{(I-P_n)v_n\}||_2+\zeta\\
\leq &||(I-P_n)R_{n+1}\{(I-P_n)v_n\}||_2\\
&+||((I-P_{n+1})Q_{n+1}-(I-P_n))R_{n+1}\{(I-P_n)v_n\}||_2+\zeta\\
< &||(I-P_n)R_n\{(I-P_n)v_n\}||_2\\
+&\left(2\frac{\sqrt{2}||\Delta_{n+1}||_F}{1-||\Delta_{n+1}||_2}\Delta+\left[\frac{\sqrt{2}||\Delta_{n+1}||_F}{1-||\Delta_{n+1}||_2}\right]^2(1+\Delta)\right)||(I-P_n)v_n||_2+\zeta\\
\leq &||(I-P_n)R_n\{(I-P_n)v_n\}||_2+\delta^2||(I-P_n)v_n||_2+\zeta.
\end{aligned}
$$

We now compute a bound in the expected error by induction on the stable subspace dimension $d-p$. *If* $d-p=1$, *then* $(I-P_n)v_n=v_n^{(d)}$ *and for any* m_1, $m_2\in\mathbb{N}$, *with* $m_2>m_1$,

$$
|v_{m_2}^{(d)}|\leq\prod_{l=m_1}^{m_2-1}\left((R_{dd})_l+\delta^2\right)|v_{m_1}^{(d)}|+\sum_{l=m_1+1}^{m_2}\zeta\prod_{j=l+1}^{m_2}\left((R_{dd})_j+\delta^2\right). \qquad (2.26)
$$

Using (0.1) *and choosing* $m_2-m_1=\mathcal{N}+1$, *we get*

$$
\begin{aligned}
|v_{m_2}^{(d)}|&\leq\prod_{l=0}^{\mathcal{N}}\left((R_{dd})_{l+m_1}+\delta^2\right)|v_{m_1}^{(d)}|+\sum_{l=0}^{\mathcal{N}}\zeta\prod_{j=l}^{\mathcal{N}-1}\left((R_{dd})_{j+m_1+2}+\delta^2\right)\\
&\leq e^{(\hat{\lambda}_d+\hat{\epsilon}_d)\mathcal{N}}|v_{m_1}^{(d)}|+\sum_{l=0}^{\mathcal{N}}\zeta\prod_{j=l}^{\mathcal{N}-1}\left((R_{dd})_{j+m_1+2}+\delta^2\right)
\end{aligned}
$$

Now assume $d-p>1$ *and let* $p<\kappa\leq d$, *then*

$$
\begin{aligned}
|v_{m_2}^{(\kappa)}|\leq&\prod_{l=0}^{\mathcal{N}}\left((R_{\kappa\kappa})_{l+m_1}+\delta^2\right)|v_{m_1}^{(\kappa)}|+\sum_{l=0}^{\mathcal{N}}\zeta\prod_{j=l}^{\mathcal{N}-1}\left((R_{\kappa\kappa})_{j+m_1+2}+\delta^2\right)\\
&+\left\{\sum_{\iota=\kappa+1}^{d}\sum_{l=0}^{\mathcal{N}}|(R_{\iota\kappa})_{l+m_1}+\delta^2|\prod_{j=l+1}^{\mathcal{N}}\left((R_{\kappa\kappa})_{j+m_1}+\delta^2\right)|v_{l+m_1}^{(\iota)}|\right\}.
\end{aligned}
$$

We remark the first two terms of the above formula are the same as those in Eq.(2.26). *This finishes the proof of the theorem.*

1.1 can be proven by taking the average over all m_2. *We may now take averages*

over all $m_2 > \mathcal{N}+1$ *and use that* $m_1 = m_2 - \mathcal{N} - 1$.

$$\begin{aligned}
\overline{|v_{m_2}^{(d)}|} &\leq e^{(\hat{\lambda}_d+\hat{\epsilon}_d)\mathcal{N}}\overline{|v_{m_2-\mathcal{N}-1}^{(d)}|} + \sum_{l=0}^{\mathcal{N}} \zeta \prod_{j=l}^{\mathcal{N}-1} \left(\overline{(R_{dd})_{j+m_2-\mathcal{N}+1}} + \delta^2\right) \\
&\leq e^{(\hat{\lambda}_d+\hat{\epsilon})\mathcal{N}}\overline{|v_{m_2}^{(d)}|} + \sum_{l=0}^{\mathcal{N}} \zeta e^{\hat{\lambda}_d l} \\
&< e^{(\hat{\lambda}_d+\hat{\epsilon}_d)\mathcal{N}}\overline{|v_{m_2}^{(d)}|} + \zeta \left(\int_{t=0}^{\mathcal{N}} e^{\hat{\lambda}_d t}\,\mathrm{d}t + 1\right) \\
&\leq e^{(\hat{\lambda}_d+\hat{\epsilon}_d)\mathcal{N}}\overline{|v_{m_2}^{(d)}|} + \left(1 + \frac{1}{|\hat{\lambda}_d|}\left(1 - e^{\hat{\lambda}_d\mathcal{N}}\right)\right)\zeta.
\end{aligned}$$

From which it immediately follows that

$$\overline{|v_{m_2}^{(d)}|} < \frac{1}{1 - e^{(\hat{\lambda}_d+\hat{\epsilon}_d)\mathcal{N}}} \left(1 + \frac{1}{|\hat{\lambda}_d|}\left(1 - e^{\hat{\lambda}_d\mathcal{N}}\right)\right)\zeta. \tag{2.27}$$

For the last term in the expression with $d - p > 1$ *this yields*

$$\begin{aligned}
&\overline{\sum_{\iota=\kappa+1}^{d}\sum_{l=0}^{\mathcal{N}} |(R_{\iota\kappa})_{l+m_1} + \delta^2| \prod_{j=l+1}^{\mathcal{N}} \left((R_{\kappa\kappa})_{j+m_1} + \delta^2\right) |v_{l+m_1}^{(\iota)}|} \\
&= \sum_{\iota=\kappa+1}^{d}\sum_{l=0}^{\mathcal{N}}\prod_{j=l+1}^{\mathcal{N}} \left(\overline{(R_{\kappa\kappa})_{j+m_1}} + \delta^2\right) \overline{|(R_{\iota\kappa})_{l+m_1} + \delta^2|}\,\overline{|v_{l+m_1}^{(\iota)}|} \\
&< \sum_{\iota=\kappa+1}^{d} \left(1 + \frac{1}{|\hat{\lambda}_\kappa|}\left(1 - e^{\hat{\lambda}_\kappa\mathcal{N}}\right)\right)\left(\Delta + \delta^2\right)\overline{|v_{m_2}^{(\iota)}|}.
\end{aligned}$$

From which it immediately follows that

$$\overline{|v_{m_2}^{(\kappa)}|} < \frac{1}{1 - e^{(\hat{\lambda}_\kappa+\hat{\epsilon}_\kappa)\mathcal{N}}} \left(1 + \frac{1}{|\hat{\lambda}_\kappa|}\left(1 - e^{\hat{\lambda}_\kappa\mathcal{N}}\right)\right)\left(\zeta + \sum_{\iota=\kappa+1}^{d} \Delta\overline{|v_{m_2}^{(\iota)}|}\right). \tag{2.28}$$

Combining Eqs. (2.26) *and* (2.28) *and using that* $\zeta = \mathcal{O}(\Delta)$, *we conclude that*

$$\overline{|v_{m_2}^{(\kappa)}|} < \frac{1}{1 - e^{(\hat{\lambda}_\kappa+\hat{\epsilon}_\kappa)\mathcal{N}}} \left(1 + \frac{1}{|\hat{\lambda}_\kappa|}\left(1 - e^{\hat{\lambda}_\kappa\mathcal{N}}\right)\right)\zeta + \mathcal{O}(\Delta^2). \tag{2.29}$$

2.A.2 Bound for the nonlinear case

For the nonlinear case we do not have a convergence proof, but we can put a bound on the error. Let the truth $\mathcal{X}_n$ be a solution to the nonlinear model

$\mathcal{X}_{n+1} = F_n(\mathcal{X}_n)$, where F_n is a $\mathcal{C}^3$ function. Assume $\{\mathcal{X}_n\}$ lies on an attractor of F and on the attractor F admits an exponential splitting. Let $\epsilon_1 \geq \epsilon_2 > 0$, A_{ϵ_1} the neighborhood of size ϵ_1 around the attractor of F, $\alpha \geq 0$, $\delta > 0$ and $\tilde{\lambda} > \exp(\lambda_s)$, where $\lambda_s < 0$ is the largest Lyapunov exponent of the stable space. Let Π_n be projectors that project on the non-stable space at $\mathcal{X}_n$, let $K_2 = \frac{1}{2}\sup_{\chi \in A_{\epsilon_1}} |F_n''(\chi)|$ and $K_0 = \sup_{\chi \in A_{\epsilon_1}} |F_n(\chi)|$. Let $P_n \in \mathcal{R}^d \times \mathcal{R}^d$ be a sequence projectors and let $\bar{u}_n \in \mathcal{R}^d$ be a given sequence of vectors with $P_n \bar{u}_n = \bar{u}_n$ for all n. Let w_0 be some arbitrary vector and $w_n = (I - P_n)F(\bar{u}_{n-1} + w_{n-1})$, for $n > 1$. Define the error vector v_n as the difference between truth and approximation at time step n:

$$v_n := \mathcal{X}_n - \bar{u}_n - w_n.$$

Theorem 2 *Assume* $||v_n|| < \epsilon_1$, $||P_n v_n|| < \epsilon_2$ *and* $||P_{n+1}v_{n+1}|| = ||P_{n+1}\mathcal{X}_{n+1} - \bar{u}_{n+1}|| < \epsilon_2$. *Then there exists some* $\tilde{\alpha} > 0$ *such that if for all* $\tilde{v} \in \mathcal{R}^d$ *it holds that* $P_n\tilde{v} \in K^u_\alpha(\mathcal{X}_n)$, *where* $K^u_\alpha(\mathcal{X}_n)$ *is the non stable cone [62, 9] of size* α *at* $\mathcal{X}_n$, *and* $(I - P_n)\tilde{v} \in K^s_{\tilde{\alpha}}(\mathcal{X}_n)$, *where* $K^s_{\tilde{\alpha}}(\mathcal{X}_n)$ *is the stable cone [62, 9] of size* $\tilde{\alpha}$ *at* $\mathcal{X}_n$, *then*

$$||v_{n+1}|| < \epsilon_2 + K_2\epsilon_1^2 + 2\alpha(K_0\epsilon_2 + K_2\epsilon_2^2) + (\tilde{\lambda} + \delta)\epsilon_1. \tag{2.30}$$

Proof of 2.

Throughout this proof we use results of [62, 9, 42]. Since the error v_n is small and F_n is $\mathcal{C}^3$, we can approximate the nonlinear flow by a Taylor expansion around the truth

$$||v_{n+1} - F_n'(\mathcal{X}_n)v_n|| = \|F_n(\mathcal{X}_n) - F_n(\mathcal{X}_n - v_n) - F_n'(\mathcal{X}_n)v_n\| \leq K_2||v_n||^2. \tag{2.31}$$

By splitting $v_{n+1} = P_{n+1}v_{n+1} + (I - P_{n+1})v_{n+1}$, noting that $||I - P_{n+1}|| < 1$ and using (2.31) we obtain

$$||v_{n+1}|| < \epsilon_2 + K_2\epsilon_1^2 + ||(I - P_{n+1})F_n'(\mathcal{X}_n)v_n||. \tag{2.32}$$

Due to the exponential splitting of F_n, the non-stable cone becomes more narrow under the tangent dynamics, i.e. vectors in this non-stable cone tend to align more towards the non-stable directions under the dynamics. This means that $F_n'(\mathcal{X}_n)K^u_\alpha(\mathcal{X}_n) \subset \text{int}(K^u_\alpha(\mathcal{X}_{n+1})) \cup \{0\}$ (it follows for example from Proposition 5.4.1 of [9] or Lemma 6.2.10 of [62]). Hence we have that $F'(\mathcal{X}_n)P_n v_n \in \text{int}(K^u_\alpha(\mathcal{X}_{n+1})) \cup \{0\}$. Due to the dynamics on the non-stable cone, we expect the length of $P_n v_n$ to grow. A bound for the growth in any step is given by Taylor expansion of $F_n(\mathcal{X}_n - P_n v_n)$ around $F_n(\mathcal{X}_n)$ as $||F'(\mathcal{X}_n)P_n v_n|| \leq K_0||P_n v_n|| + K_2||P_n v_n||^2$, where $K_0 = \sup|F_n(\mathcal{X}_n)|$. However, the only part of $F'(\mathcal{X}_n)P_n v_n$ of interest is the component

$(I - P_{n+1})F'(\mathcal{X}_n)P_n v_n$. We have that

$$||(I-P_{n+1})F'(\mathcal{X}_n)P_n v_n|| < 2\alpha||F'(\mathcal{X}_n)P_n v_n|| < 2\alpha(K_0||P_n v_n|| + K_2||P_n v_n||^2). \tag{2.33}$$

For the part $(I-P_n)v_n$ in the stable cone we can use [9] Proposition 5.4.2 or [62] Lemma 6.2.11, which states that this vector shrinks under time evolution, where the amount depends on the width of our cone. To be precise: $\forall \delta > 0\ \exists \tilde{\alpha} > 0$ such that if $(I - P_n)v_n \in K^s_{\tilde{\alpha}}(\mathcal{X}_n)$, then

$$\|F'_n(\mathcal{X}_n)(I - P_n)v_n\| < (\tilde{\lambda} + \delta)\|(I - P_n)v_n\|. \tag{2.34}$$

Collecting the estimates (2.32)–(2.34), we find that

$$\begin{aligned}
||v_{n+1}|| &< \epsilon_2 + K_2\epsilon_1^2 + ||(I - P_{n+1})F'_n(\mathcal{X}_n)(P_n v_n + (I - P_n)v_n)|| \\
&\leq \epsilon_2 + K_2\epsilon_1^2 + ||(I - P_{n+1})F'_n(\mathcal{X}_n)P_n v_n|| + ||F'_n(\mathcal{X}_n)(I - P_n)v_n)|| \\
&< \epsilon_2 + K_2\epsilon_1^2 + 2\alpha(K_0||P_n v_n|| + K_2||P_n v_n||^2) + (\tilde{\lambda} + \delta)||(I - P_n)v_n|| \\
&< \epsilon_2 + K_2\epsilon_1^2 + 2\alpha(K_0\epsilon_2 + K_2\epsilon_2^2) + (\tilde{\lambda} + \delta)\epsilon_1.
\end{aligned}$$

□

CHAPTER 3

Shadowing-based data assimilation method for partially observed models

In this chapter we develop further an algorithm for data assimilation based upon a shadowing refinement technique [20] to take partial observations into account. A version of this chapter has been accepted for publication in the SIAM Journal on Applied Dynamical Systems. Our method is based on a regularized Gauss-Newton method. We prove local convergence to the solution manifold and provide a lower bound on the algorithmic time step. We use numerical experiments with the Lorenz 63 and Lorenz 96 models to illustrate convergence of the algorithm and show that the results compare favourably with a variational technique—weak-constraint four-dimensional variational method—and a shadowing technique–pseudo-orbit data assimilation. Numerical experiments show that a preconditioner chosen based on a cost function allows the algorithm to find an orbit of the dynamical system in the vicinity of the true solution.

3.1 Introduction

Data assimilation (DA) methods combine orbits from a dynamical model with measurement data to obtain an improved estimate for the state of a physical system [51]. Well known strong-constraint four-dimensional variational data assimilation (4DVar) aims at finding the optimal initial condition for the dynamical model such that the distance to observations is minimized under a

constraint of the estimate being an orbit of the dynamical model [91]. A drawback of strong-constraint 4DVar is that the number of local minima of the corresponding cost function increases dramatically with assimilation window—time window over which observations are assimilated into the dynamical model [6, 74, 82]. An existing remedy in 4DVar is introduction of a model error term in the cost function and is called weak-constraint 4DVar (WC4Var) [85, 93]. Then an estimate is a pseudo-orbit of the dynamical model rather than an orbit. An orbit satisfies PDE of a dynamical model exactly, while a pseudo-orbit up to a small ε. It has been shown in e.g. [93] that WC4Var allows longer assimilation windows compared to the strong-constraint 4DVar.

An alternative DA approach that allows long assimilation windows is based on a model having a shadowing property.

There exist several shadowing-type DA methods. A pseudo-orbit DA method (PDA) [32] and a noise reduction algorithm [11] seek a (pseudo-)trajectory of a dynamical model by minimizing a cost function. Local convergence to the solution manifold corresponding to $\dot{x} - f(x) = 0$ was proven for a class of iteration schemes assuming full observations [11]. In numerical experiments, the noise reduction algorithm uses the Laplace operator and PDA—an algorithmic time step to achieve convergence to the solution manifold, though without a robust answer whether these are the good choices for the convergence. Another shadowing-type DA method instead of minimizing a cost function, seeks zeros of a cost operator [20]. Obviously, the (nonunique) global minimum of the cost function is zero and this value is reached if and only if the corresponding cost operator is zero.

A shortcoming of existing shadowing-type DA methods is that for initialization they use *full* observations in space (or partial observations combined with an estimation obtained from another DA method). Up to now truly partial observations (without any preprocessing involving another DA method) have not been thoroughly considered in shadowing-type DA methods. Therefore, in this chapter we consider an initial guess for a shadowing-type DA method that consists of partial observations and a background trajectory, which was obtained from model propagation starting at an arbitrary initial condition and *without* DA. We develop further the shadowing-based DA method [20] to account for partial observations based on Levenberg-Marquardt regularization [65, 72], and prove local convergence following [11]. The Levenberg-Marquardt algorithm can be seen as a regularization of the Gauss-Newton method, which is used in the shadowing-based DA method of [20]. A regularization parameter controls algorithmic time step, making the Gauss-Newton method convergent to the solution manifold independently of the starting point. The Levenberg-Marquardt regularization is well used in nonlinear optimization and data assimilation in particular, e.g. for variational data assimilation [71],

and for an ensemble Kalman filter [15].

Despite being convergent to the solution manifold, a shadowing-type DA method might poorly approximate the true solution due to observations being used only as initial guess. Therefore, in this chapter we introduce a preconditioner for the corresponding gradient flow that modifies the direction of the search such that the estimate remains in the vicinity of observations. This is done in the spirit of trust region methods [76], which together with Gauss-Newton type methods have been an inspiration for new algorithms to solve nonlinear least-squares problems, see e.g. [21].

The rest of the chapter is organized as follows. In Section 3.2, we briefly recall the shadowing-based DA method for full observations. In Section 3.3, we introduce the shadowing-based DA method for partial observations and prove local convergence. In Section 3.4, we present results for the Lorenz 63 and the Lorenz 96 models. Finally, we draw the conclusions in Section 3.5.

3.2 Noise reduction

We consider a discrete deterministic model

$$x_{n+1} = F_n(x_n), \quad x_n \in \mathcal{R}^m, \quad n = 0, \dots, N-1, \tag{3.1}$$

where $F_n : \mathcal{R}^m \to \mathcal{R}^m$. We assume F_n to be $\mathcal{C}^3$ for all n. In many applications the model is defined by the time-discretization of an ordinary differential equation $\dot{x} = f(t, x)$, $x(t) \in \mathcal{R}^m$, which in turn may be defined as the space-discretization of a partial differential equation (or system of PDEs).

Let the sequence $\boldsymbol{\mathcal{X}} := \{\mathcal{X}_0, \dots, \mathcal{X}_N\}$ be a distinguished orbit of (3.1), referred to as the true solution of the model, and presumed to be unknown. Suppose we are given a sequence of partial noisy observations $\boldsymbol{y} := \{y_0, \dots y_N\}$ related to $\boldsymbol{X}$ via

$$y_n = H_n \mathcal{X}_n + \xi_n, \qquad y_n \in \mathbb{R}^d, \quad n = 0, \dots, N,$$

where $H_n : \mathcal{R}^m \to \mathcal{R}^d$, $d \leq m$, is the linear observation operator, and the noise variables ξ_n are drawn from a normal distribution $\mathcal{N}(0, R_n)$ with zero mean and known observational error covariance matrix R_n.

Data assimilation is the problem of finding a pseudo-orbit $\boldsymbol{u} = \{u_0, u_1, \dots, u_N\}$, $u_n \in \mathcal{R}^d$, of the model (3.1), such that the differences $\|y_n - Hu_n\|$ and $\|u_n - F_n(u_{n-1})\|$, $n = 1, \dots, N$ are small in an appropriately defined sense. This is done with the aim of minimizing the unknown error $\|u_n - \mathcal{X}_n\|$; see for example [91, 63]. Well known WC4DVar aims at minimizing a cost function

$$C_{\mathrm{v}}(\boldsymbol{u}) = \sum_{n=1}^{N} (y_n - Hu_n)^T R^{-1} (y_n - Hu_n) + (u_n - F_n(u_{n-1}))^T Q^{-1} (u_n - F_n(u_{n-1})),$$

where the Q is model error (see e.g.[85, 66, 92, 91] and references therein).

Instead of minimizing a cost function, the shadowing-based DA method [20] searches for a zero of the residual functional

$$G(\boldsymbol{u}) = \begin{pmatrix} G_0(\boldsymbol{u}) \\ G_1(\boldsymbol{u}) \\ \vdots \\ G_{N-1}(\boldsymbol{u}) \end{pmatrix}, \qquad G_n(\boldsymbol{u}) = u_{n+1} - F_n(u_n), \quad n = 0, \ldots, N-1, \tag{3.2}$$

using a contractive iteration started from (a proxy of) complete, noisy observations. Therefore we call this method noise reduction DA method. This approach is motivated by research on numerical shadowing methods. We stress that, just as with strong-constraint 4DVar, noise reduction DA attempts to find an exact orbit of (3.1) consistent with the observations. However, instead of solving directly for the initial condition, we solve for the whole orbit at once.

Noise reduction DA seeks an update $\boldsymbol{P}^{(k)}$ by approximately solving

$$G\left(\boldsymbol{u}^{(k)} + \boldsymbol{P}^{(k)}\right) = 0. \tag{3.3}$$

Here k denotes the index of the Newton's iteration and the solution to (3.3) is approximated using the right pseudo-inverse of G'

$$\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \boldsymbol{P}^{(k)}, \quad \boldsymbol{P}^{(k)} = -G'(\boldsymbol{u}^{(k)})^{\dagger} G(\boldsymbol{u}^{(k)}) = -G'^{T}(G'\,G'^{T})^{-1} G$$

with $\boldsymbol{u}^{(0)} = \boldsymbol{\mathcal{X}} + \boldsymbol{\xi}$, i.e. the iteration is started at (a proxy of) complete observations. Without loss of generality, we can assume that observation operator H is the identity matrix for a proxy of complete observations. The function $G(\boldsymbol{u})$ has a zero for every orbit of the model. The Jacobian of G has an $m(N-1) \times mN$ block structure:

$$G'(\boldsymbol{u}) = \begin{bmatrix} -F_0'(u_0) & I & & & \\ & -F_1'(u_1) & I & & \\ & & \ddots & \ddots & \\ & & & -F_{N-1}'(u_{N-1}) & I \end{bmatrix}. \tag{3.4}$$

The Jacobian appears only when acting on a given vector (unit vector for example), and therefore it could be efficiently approximated by finite differences. Thus we use an approximation $F'(u)v \approx 1/\varepsilon(F(u+\varepsilon v) - F(u))$.

3.3 Shadowing-based DA method

In this section, we assume that the observation operator H is not the identity matrix, but instead, that we are in a situation of partial observations.

Because not all variables are observed, the observation operator is taken to be a projection from the full model space to the observed variables. To obtain a proxy of complete observations, a background value is filled-in for the non-observed variables. The background values can then be treated as highly inaccurate observations. The noise reduction algorithm may then be applied, whereby an "observation covariance matrix" reflects that the observed variables are observed with a certain precision, while the background values are highly uncertain data points. We shall show some convergence results for this method and demonstrate the method can be used in various numerical experiments with incomplete observations.

We take as an initial guess for a shadowing-type DA method:

$$\boldsymbol{u}^{(0)} = H^T \boldsymbol{y} + (I - H^T H)\boldsymbol{x}^{\mathrm{b}}, \tag{3.5}$$

where $\boldsymbol{x}^{\mathrm{b}}$ is a so-called background trajectory—a solution of (3.1) with an arbitrary initial condition. It holds that $H\boldsymbol{u}^{(0)} = \boldsymbol{y}^{(0)}$, so for the observed variables the initial guess is at the observations. We seek an update $\boldsymbol{\Pi}^{(k)}$ by approximately solving

$$G\left(\boldsymbol{u}^{(k)} + \boldsymbol{\Pi}^{(k)}\right) = 0.$$

using the Levenberg-Marquardt regularization

$$\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \boldsymbol{\Pi}^{(k)}, \quad \boldsymbol{\Pi}^{(k)} = -\Sigma G'^T \left(G'\Sigma G'^T + \alpha^{(k)} Q\right)^{-1} G, \tag{3.6}$$

where $G = G(\boldsymbol{u}^{(k)})$ defined in (3.2), $G' = G'(\boldsymbol{u}^{(k)})$ defined in (3.4), Q is a given positive definite matrix, and $\alpha^{(k)} > 0$. Here

$$\Sigma := H^T R H + (I - H^T H) W (I - H^T H), \tag{3.7}$$

where W is a positive definite matrix that has an $mN \times mN$ block diagonal structure $W = \mathrm{blockdiag}(W_1, \ldots, W_N)$. The matrix R has a $dN \times dN$ block diagonal structure $R = \mathrm{blockdiag}(R_1, \ldots, R_N)$, Q has an $m(N-1) \times m(N-1)$ block diagonal structure $Q = \mathrm{blockdiag}(Q_1, \ldots, Q_N)$, and H has a $dN \times mN$ block diagonal structure $H = \mathrm{blockdiag}(H_1, \ldots, H_N)$,

To note a parallel to 4DVar, the solution $\boldsymbol{\Pi}^{(k)}$ to (3.6) is a minimizer of a cost function

$$\frac{1}{2}\left[G\left(\boldsymbol{u}^{(k+1)}\right)\right]^T Q^{-1}\left[G\left(\boldsymbol{u}^{(k+1)}\right)\right] + \frac{\alpha^{(k)}}{2}\left[\boldsymbol{\Pi}^{(k)}\right]^T \Sigma^{-1}\left[\boldsymbol{\Pi}^{(k)}\right].$$

In the above cost function, $\frac{1}{2}\left[G\left(\boldsymbol{u}^{(k+1)}\right)\right]^T Q^{-1}\left[G\left(\boldsymbol{u}^{(k+1)}\right)\right]$ is a cost attributed to model error. $\frac{\alpha^{(k)}}{2}\left[\boldsymbol{\Pi}^{(k)}\right]^T \Sigma^{-1}\left[\boldsymbol{\Pi}^{(k)}\right]$ is a cost attributed to the size of the update. The size of the update serves as an approximation for the observational mismatch, because the initial guess has zero observational mismatch and the regularization should ensure that the step size decreases during iterations.

3.3.1 Local convergence

The manifold of trajectories of F_n is defined by $\mathcal{M}$ by $\mathcal{M} = \{\boldsymbol{u} : \ G(\boldsymbol{u}) = 0\}$. Because F_n is $\mathcal{C}^3$, $\mathcal{M}$ is a manifold. We define $\phi(\boldsymbol{u})$ as

$$\phi(\boldsymbol{u}) = \boldsymbol{u} - \Sigma G'^T \left(G'\Sigma G'^T + \alpha Q\right)^{-1} G.$$

We note that

$$D\phi = I - \Sigma G'^T \left(G'\Sigma G'^T + \alpha Q\right)^{-1} G' \quad \text{for} \quad \boldsymbol{u} \in \mathcal{M}. \tag{3.8}$$

Since $\mathcal{M}$ is a manifold, we define the tangent and normal spaces of $\mathcal{M}$ at u as $\mathcal{T}_u\mathcal{M}$ and $\mathcal{N}_u\mathcal{M}$, respectively. We have $\mathcal{T}_u\mathcal{M} \perp \mathcal{N}_u\mathcal{M}$ and $\mathcal{T}_u\mathcal{M} = \ker\left(\Sigma G'^T \left(G'\Sigma G'^T + \alpha Q\right)^{-1} G'\right)$ for $\boldsymbol{u} \in \mathcal{M}$.

Lemma 1 *$\mathcal{M}$ is a set of fixed points for ϕ and there are no further fixed points in a neighbourhood of $\mathcal{M}$.*

Theorem 2 *Suppose $\mathcal{M}$ is compact and contained in an open set $\mathcal{U}$. Furthermore, suppose $D\phi$ is continuous in $\mathcal{U}$ and $\|D\phi|_{\mathcal{N}_u\mathcal{M}}\| < 1$ for all $\boldsymbol{u} \in \mathcal{M}$. Then the sequence $\boldsymbol{u}^{(k)} = \phi^k(\boldsymbol{u}^{(0)})$ converges for $k \to \infty$ to a point on $\mathcal{M}$ if $\boldsymbol{u}^{(0)}$ is sufficiently near to $\mathcal{M}$.*

For proof of both Lemma 1 and Theorem 2 we refer to [11], where local convergence for a class of general iterative schemes was proven.

Now we can prove a local convergence result for the shadowing-based DA method (3.6). First, we define $\Omega = G'^T Q^{-1} G'$.

Lemma 3 *Suppose Σ and Ω commute. Furthermore, suppose a positive α satisfies $\alpha > \lambda_{\max}(\Sigma\Omega|_{\mathcal{N}_u\mathcal{M}})/2 - \lambda_{\min}(\Sigma\Omega|_{\mathcal{N}_u\mathcal{M}})$. Then $\|D\phi|_{\mathcal{N}_u\mathcal{M}}\| < 1$ for all $\boldsymbol{u} \in \mathcal{M}$.*

Proof 1 *Using the Sherman-Morrison-Woodbury matrix inversion formula [41] and assuming that $\alpha \neq 0$, we can rewrite* (3.8) *as*

$$D\phi = I - \Sigma\Omega[\alpha I + \Sigma\Omega]^{-1}, \tag{3.9}$$

where we drop the iteration notation. It is noted that Σ and Ω are covariance matrices that commute. So, $[\alpha I + \Sigma\Omega]^{-1}$ is invertible, $\sigma\alpha[\alpha I + \Sigma\Omega]^{-1}$ is a covariance matrix and $D\phi$ is symmetric. For symmetric matrices the 2-norm is equal to the spectral radius, which is the largest absolute value of the eigenvalues of the matrix. Thus, the norm of $D\phi$ is

$$\|D\phi\| = \max\{|1 - \lambda_{\max}(\Sigma\Omega[\alpha I + \Sigma\Omega]^{-1})|, |1 - \lambda_{\min}(\Sigma\Omega[\alpha I + \Sigma\Omega]^{-1})|\}.$$

Moreover,

$$\begin{aligned} 0 &\le \lambda_{\max}(\Sigma\Omega[\alpha I + \Sigma\Omega]^{-1}) \\ &\le \lambda_{\max}(\Sigma\Omega)\lambda_{\max}([\alpha I + \Sigma\Omega]^{-1}) = \frac{\lambda_{\max}(\Sigma\Omega)}{\lambda_{\min}(\alpha I + \Sigma\Omega)} \\ &= \frac{\lambda_{\max}(\Sigma\Omega)}{\alpha + \lambda_{\min}(\Sigma\Omega)}. \end{aligned}$$

By choosing α *such that*

$$\frac{\lambda_{\max}(\Sigma\Omega)}{\alpha + \lambda_{\min}(\Sigma\Omega)} < 2,$$

we have $|1 - \lambda_{\max}(\Sigma\Omega[\alpha I + \Sigma\Omega]^{-1})| < 1$ *for* $\lambda_{\max}(\Sigma\Omega[\alpha I + \Sigma\Omega]^{-1}) > 0$.

Furthermore,

$$0 \le \lambda_{\min}(\Sigma\Omega[\alpha I + \Sigma\Omega]^{-1}) \le \lambda_{\max}(\Sigma\Omega[\alpha I + \Sigma\Omega]^{-1}) < 2.$$

Thus we have $|1 - \lambda_{\min}(\Sigma\Omega[\alpha I + \Sigma\Omega]^{-1})| < 1$ *for* $\lambda_{\min}(\Sigma\Omega[\alpha I + \Sigma\Omega]^{-1}) > 0$.

From (3.9) *it follows that* $\mathcal{T}_u\mathcal{M} = \ker\left(\Sigma\Omega[\alpha I + \Sigma\Omega]^{-1}\right)$ *for* $\boldsymbol{u} \in \mathcal{M}$. *Since* $\mathcal{T}_u\mathcal{M} \perp \mathcal{N}_u\mathcal{M}$, *we have*

$$\lambda\left(\Sigma\Omega[\alpha I + \Sigma\Omega]^{-1}|_{\mathcal{N}_u\mathcal{M}}\right) > 0.$$

Therefore by choosing $\alpha > \lambda_{\max}(\Sigma\Omega|_{\mathcal{N}_u\mathcal{M}})/2 - \lambda_{\min}(\Sigma\Omega|_{\mathcal{N}_u\mathcal{M}})$,
we have $\|D\phi|_{\mathcal{N}_u\mathcal{M}}\| < 1$ *for all* $\boldsymbol{u} \in \mathcal{M}$.

Corollary 2.1 *The sequence* $\boldsymbol{u}^{(k)} = \phi^k(\boldsymbol{u}^{(0)})$ *defined in* (3.6) *converges for* $k \to \infty$ *to a point on* $\mathcal{M}$ *if* $\boldsymbol{u}^{(0)}$ *is sufficiently near to* $\mathcal{M}$.

Proof 2 *The proof directly follows from Theorem 2 and Lemma 3.*

Corollary 2.2 *Suppose* G *has only one zero. Then for the sequence defined in* (3.6) *and a final iteration* K, $\boldsymbol{u}^{(K)} = \mathcal{X}$.

This rather trivial corollary shows that the shadowing-based DA method converges to the true solution for linear models or convex G^TG. Existence of several zeros of G is equivalent to the problem of several minima of G^TG.

We proved local convergence of the algorithm to the solution manifold. We are unable to provide any results on error bounds with respect to the true solution. However, we provide a necessary condition for an estimate to remain close to the observations, in the sense that $\|H\boldsymbol{u} - \boldsymbol{y}\|_E \le 1$. This result is useful since a background trajectory $\boldsymbol{x}^{\mathrm{b}}$ has larger error with respect to the truth than observations $\boldsymbol{y}$. We recall that an initial guess (3.5) for the algorithm consists of $\boldsymbol{x}^{\mathrm{b}}$ and $\boldsymbol{y}$. Then for a good estimate of the true solution, while

updating unobserved variables, the observed variables need to have Gauss-Newton updates that remain close to the observations $\boldsymbol{y}$.

Before we state the result, let us rewrite the shadowing-based DA method in the limit of continuous algorithmic time step, similarly to the approach used in [86, 87]. Assume we can set $\alpha^{(k)} = \alpha^{(0)}$ for all k. Then we introduce notation $h = 1/\alpha$ and rewrite (3.6) in terms of h

$$\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + h\boldsymbol{\Psi}^{(k)}, \quad \boldsymbol{\Psi}^{(k)} = -\Sigma G'^T \left(hG'\Sigma G'^T + Q\right)^{-1} G.$$

Then taking the limit of $h \to 0$, we get on $\tau \in [0\ 1]$

$$\frac{d\boldsymbol{u}}{d\tau} = \psi(\boldsymbol{u}), \quad \text{with} \quad \psi(\boldsymbol{u}) = -\Sigma G'^T(\boldsymbol{u}) Q^{-1} G(\boldsymbol{u}), \quad \text{and} \quad \boldsymbol{u}(0) = \boldsymbol{u}^0. \tag{3.10}$$

Defining $\Phi(\boldsymbol{u}) = \|Q^{-1/2}G(\boldsymbol{u})\|^2/2$, the ODE (3.10) becomes

$$\frac{d\boldsymbol{u}}{d\tau} = -\Sigma \nabla \Phi(\boldsymbol{u}). \tag{3.11}$$

This is a preconditioned gradient descent for $\Phi(\cdot)$ with a preconditioner Σ. We recall that Σ is defined in (3.7). We define the projection on the unobserved variables $H^\perp = (I - H^T H)$, which satisfies $H^\perp H = 0$ as well as $H \perp H^\perp$. We now have the following lemma in case of a convergent algortihm, i.e., in case $\|\nabla\Phi(\boldsymbol{u})\| < 1$.

Lemma 4 *Suppose* $\|\nabla\Phi(\boldsymbol{u})\| < 1$. *Furthermore, suppose* $\|H^\perp \boldsymbol{u}(1) - H^\perp \boldsymbol{x}^{\mathrm{b}}\|_W < \varepsilon$ *for a small positive* ε. *Then* $\|H\boldsymbol{u}(1) - \boldsymbol{y}\|_R < 1 - \varepsilon$.

Proof 3 *By multiplying* (3.11) *with either* H *or* $H^\perp$, *integrating from 0 to 1, and then taking the L2-norm, we have*

$$\|H\boldsymbol{u}(1) - \boldsymbol{y}\| = \|R \int_0^1 H\nabla\Phi(\boldsymbol{u})d\tau\|, \quad \textit{and}$$
$$\|H^\perp \boldsymbol{u}(1) - H^\perp \boldsymbol{x}^{\mathrm{b}}\| = \|W \int_0^1 H^\perp \nabla\Phi(\boldsymbol{u})d\tau\|.$$

Due to the assumption $\|H^\perp \boldsymbol{u}(1) - H^\perp \boldsymbol{x}^{\mathrm{b}}\|_W < \varepsilon$, *we have* $\| \int_0^1 H^\perp \nabla\Phi(\boldsymbol{u})d\tau\| < \varepsilon$. *This implies that* $\| \int_0^1 H\nabla\Phi(\boldsymbol{u})d\tau\| < 1 - \varepsilon$, $\|\nabla\Phi(\boldsymbol{u})\| < 1$ *and* $H \perp H^\perp$. *In turn, the inequality* $\| \int_0^1 H\nabla\Phi(\boldsymbol{u})d\tau\| < 1 - \varepsilon$ *implies* $\|H\boldsymbol{u}(1) - \boldsymbol{y}\|_R < 1 - \varepsilon$, *so the estimate* $H\boldsymbol{u}$ *consequently remains close to observations* $\boldsymbol{y}$, *that is,* $\|H\boldsymbol{u} - \boldsymbol{y}\|_E \leq 1$.

3.3.2 Existing shadowing-type DA methods

Now we point out differences between the shadowing-based DA method introduced in this chapter and the existing shadowing-type DA methods of

[11, 32], and of [20]. We write down the methods in terms of function ϕ:

$$\begin{aligned}\phi^{[11]} : &= u - G'^T \Lambda^{-1} G,\\ \phi^{[32]} : &= u - \gamma G'^T G,\\ \phi^{[20]} : &= u - G'^T (G' G'^T)^{-1} G,\\ \phi^{\mathrm{rSh}} : &= u - \Sigma G'^T (G' \Sigma G'^T + \alpha Q)^{-1} G.\end{aligned}$$

In $\phi^{[11]}$, Λ is chosen to be the Laplace operator. It is stated that the choice of Λ has great influence on the convergence, though without a rigorous statement whether the Laplace operator is a good choice. Local convergence is proven for the method as well as for a class of general iterative schemes. In $\phi^{[32]}$, an algorithmic time step γ is chosen by tuning. For sufficiently small γ convergence of the damped Gauss-Newton method is guaranteed but the convergence rate may be only linear [41]. In $\phi^{[20]}$, the convergence rate is quadratic due to the Gauss-Newton method, but the local nature of the Gauss-Newton method requires a good initial guess for convergence—thus (a proxy of) completed observations. In ϕ^{rSh}, on the one hand lower bound on α guarantees local convergence but on the other hand the preconditioner Σ might deteriorate the convergence rate. However, the preconditioner Σ, and in particular the component W, is required for a good estimation of the true solution.

3.4 Numerical experiments

We note that if $\Sigma = \epsilon I$, then Σ and Ω commute. For partially-observed models, however, Σ and Ω might not commute. Moreover, in practice $\mathcal{N}_u\mathcal{M}$ is not available. Therefore, we assume $\lambda_{\max}(\Delta t^2 \Sigma\Omega) > \lambda_{\max}(\Sigma\Omega|_{\mathcal{N}_u\mathcal{M}})$, where Δt is time step of a numerical discretization. Furthermore, we assume $\lambda_{\min}(\Sigma\Omega|_{\mathcal{N}_u\mathcal{M}}) > 0$. The latter assumption is fulfilled if projection onto $\mathcal{N}_u\mathcal{M}$ is defined in terms of G'. Then according to Lemma 3 we can choose α

$$\alpha = \Delta t^2 \lambda_{\max}(\Sigma\Omega)/2. \tag{3.12}$$

Numerical experiments show that choosing such an α provides convergence to the manifold $\mathcal{M}$. However, we do not have a rigorous answer whether the assumption $\lambda_{\max}(\Delta t^2 \Sigma\Omega) > \lambda_{\max}(\Sigma\Omega|_{\mathcal{N}_u\mathcal{M}})$ is fulfilled.

When computing α, we split the eigenvalue problem over one window length N in N eigenvalue problems over N windows length 1. Then in (3.12) we use maximum eigenvalue over N windows. Moreover, to save computational costs we compute α for an initial guess $\boldsymbol{u}^{(0)}$ only and fix the same α throughout the iteration. The maximum number of iteration is 100. Model error is assumed to be $Q = 10^{-3} I$. Other values such as 10^{-2} and 10^{-4} provide equivalent results to 10^{-3}. We define the weighting matrix $W = w^2 I$ in the preconditioner Σ and perform sensitivity analysis in terms of w.

We compare the shadowing-based DA method to WC4DVar and PDA. PDA is initialised at an initial guess $\boldsymbol{u}^{(0)}$ and an algorithmic time step is chosen as in [32], namely $\gamma = 0.1$. The maximum number of iterations is 100. We note that in [32] the maximum number of iterations is 1024. However, we keep the same number of iterations 100 for all DA methods.

Both the shadowing-based DA method and PDA provide an estimation at observation times only. Therefore we use an estimation at observation times as initial condition for forward model propagation to have an estimation at every time step of numerical discretization.

WC4DVar is initialised at a background trajectory $\boldsymbol{x}^{\mathrm{b}}$. The minimization of a cost function is done by a Matlab built-in Levenberg-Marquardt algorithm and stopping when the relative change in the cost function compared to the initial value is less then 10^{-6} unless 100 iterations are reached. Model error for WC4DVar is $10^{-2}I$, and the background covariance matrix is the identity.

In order to check robustness of the results, we perform 100 numerical experiments with different realizations of truth $\boldsymbol{\mathcal{X}}$, observations $\boldsymbol{y}$, and background trajectory $\boldsymbol{x}^{\mathrm{b}}$.

To analyze the shadowing-based DA method and compare it to other methods, we compute mean over time of G-error

$$\mathbb{E}^G = \frac{1}{N}\sum_{n=0}^{N-1} G_n^T G_n, \tag{3.13}$$

mean over time of estimation error with respect to the truth of observed variables

$$\mathbb{E}^{\mathrm{O}} = \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{E}_n^{\mathrm{O}}, \tag{3.14}$$

and of non-observed variables

$$\mathbb{E}^{\mathrm{N}} = \frac{1}{N}\sum_{n=0}^{N-1} \mathbb{E}_n^{\mathrm{N}}. \tag{3.15}$$

Here errors $\mathbb{E}_n^{\mathrm{O}}$ and $\mathbb{E}_n^{\mathrm{N}}$ defined as

$$\mathbb{E}_n^{\mathrm{O}} = \frac{1}{\mathrm{rank}(H)}(Hu_n - H\mathcal{X}_n)^T(Hu_n - H\mathcal{X}_n) \tag{3.16}$$

and

$$\mathbb{E}_n^{\mathrm{N}} = \frac{1}{\mathrm{rank}(I - H^T H)}[(I - H^T H)(u_n - \mathcal{X}_n)]^T[(I - H^T H)(u_n - \mathcal{X}_n)], \tag{3.17}$$

respectively, and n is the numerical time step index. We also compute a cost function with respect to observations

$$C = \frac{1}{(k_2 - k_1 + 1)\text{rank}(H)} \sum_{k=k_1}^{k_2} (Hu_k - y_k)^T (Hu_k - y_k), \tag{3.18}$$

where k is the index of the observation time step, $k_1 \geq 0$, and $k_2 \leq N-1$.

3.4.1 Application to the Lorenz 63 model

The well-known Lorenz attractor [69] is a chaotic dynamical system commonly used as a test problem for data assimilation algorithms. The L63 model is

$$\dot{x}^1 = \sigma(x^2 - x^1), \quad \dot{x}^2 = x^1(\rho - x^3) - x^2, \quad \dot{x}^3 = x^1 x^2 - \beta x^3, \tag{3.19}$$

where $\sigma = 10$, $\beta = \frac{8}{3}$ and $\rho = 28$. The differential equations are discretized with a forward Euler scheme with time step $\Delta t = 0.005$. (We have also considered Runge-Kutta 4th order, but since it gives similar results, it is omitted in the chapter.) We generate a set of observations by computing a trajectory of L63 on $t \in [0, 100]$, with a spin-up of $[-25, 0]$. Observations are obtained by perturbing a reference (true) trajectory with random Gaussian iid noise with zero mean and covariance $R = 8I$. The observations of x^1-variable only are drawn every $\Delta t_{\text{obs}} = 0.05$. Then the map F_n (3.1) corresponds to 10 forward Euler steps. This map is used to define G and the derivatives of this map are needed for the shadowing iteration. The assimilation windows is $\Delta t_{\text{ass}} = 5$.

In Figure 1 we display G-error (3.13) on the left and error with respect to the truth of non-observed variables (3.15) on the right as a function of iteration. We remark that small $w = 100$ gives quicker convergence to the manifold $\mathcal{M}$, while large $w = 1000$ requires more iterations to reach the same error on average. However, error with respect to the truth of non-observed variables is decreasing during iteration for large $w = 1000$, while increasing for small $w = 100$. In Figure 2 we plot error with respect to the truth of observed variables (3.14) on the left and cost function (3.18) on the right as a function of iteration number, where the solid black line is observation error. We see again that large $w = 1000$ gives better estimation of observed variables than small $w = 100$.

When analyzing the cost function, we see that for small $w = 100$ the cost function quickly underestimates the observation error. In inverse problems this phenomenon is often referred as observation overfitting, though in that context a cost function is decreasing not increasing and the observation error is overestimated not underestimated, see e.g. [46]. For the shadowing-based DA method the cost function (3.18) is zero at the first iteration, because the algorithm is initialized at $\boldsymbol{u}^{(0)}$ (3.5). The cost function increases over the course of iteration due to a search for a noise-free orbit. When the cost function is larger than the observation error R, an estimate is not in an

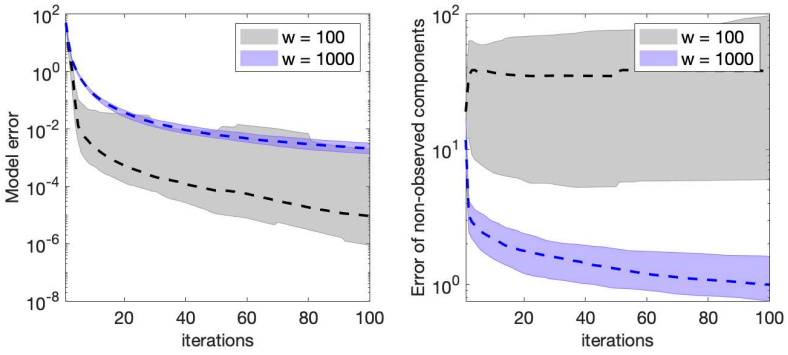

Figure 1: Application to L63. Error of the shadowing-based DA method as a function of iterations: median (dashed line) +/- one standard deviation (shadowed area) over 100 simulations. In grey error is shown for weighting matrix $w = 100$, in blue for $w = 1000$. On the left: mean over time of G-error. On the right: mean over time of error with respect to the truth of non-observed variables.

ellipsoid with principal axes defined by the eigenvectors of R and centred at the true trajectory, resulting in a larger error with respect to the truth. Therefore, we need to prevent the cost function becoming larger than R. A classical approach in inverse problems is to stop the iteration when this occurs. In the shadowing-based DA method this approach is questionable due to cost function increasing over the course of iteration. Instead, we propose to tune the preconditioner Σ (3.7), namely the weighting matrix W, to obtain the correct behaviour of the cost function. We see that the large value of $w = 1000$ results in the cost function approaching the observation error from below. This is an indication of correctly tuned w. Thus the role of preconditioner Σ is to keep descent steps in the direction of observed variables small compared to descent steps in the direction of non-observed variables. As the iteration proceeds, observed variables get denoised as well and the algorithm finds a (pseudo-)orbit compatible with observations. We would like to stress that the cost function (3.18) depends only on observations, not the truth.

In Figure 3 we compare the shadowing-based DA method with $w = 1000$ to WC4DVar and PDA, where we plot error with respect to the truth over time of observed variables (3.16) and of non-observed variables (3.17) on the left and right, respectively. We see that the correct choice of the preconditioner is essential for shadowing-type DA methods, since for fully observed L63 PDA and the shadowing-based DA method perform comparably (not shown) but for partially observed L63 PDA performs poorly. It is also remarkable that the shadowing-based DA method with tuned w outperforms WC4DVar.

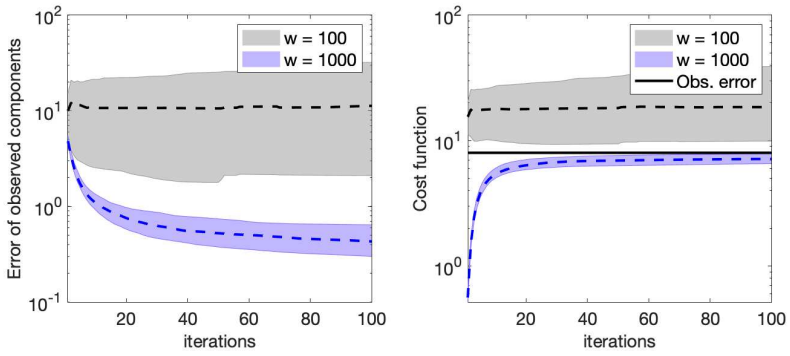

Figure 2: Application to L63. Error of the shadowing-based DA method as a function of iterations: median (dashed line) +/- one standard deviation (shadowed area) over 100 simulations. In grey error is shown for weighting matrix $w = 100$, in blue for $w = 1000$. On the left: mean over time of error with respect to the truth of observed variables. On the right: mean over time of cost function of observed variables.

3.4.2 Application to the Lorenz 96 model

Lorenz [70] proposed the following model as an example of a simple one-dimensional model with features of the atmosphere. The L96 model is

$$\dot{x^l} = -x^{l-2}x^{l-1} + x^{l-1}x^{l+1} - x^l + \mathcal{F}, \qquad (l = 1, ..., d), \tag{3.20}$$

where the dimension d and forcing $\mathcal{F}$ are parameters. Cyclic boundary conditions are imposed. We implement the L96 model with the standard parameter choices $d = 36$ and $\mathcal{F} = 8$. The differential equations are discretized with a forward Euler scheme with time step $\Delta t = 0.005$. (We have also considered Runge-Kutta 4th order but since it gives similar results, it is omitted in the chapter.) We generate a set of observations computing a trajectory of L96 on $t \in [0, 100]$, with a spin-up of $[-25, 0]$ for a true trajectory to reside on the attractor. Observations are obtained by perturbing a reference (true) trajectory with random Gaussian iid noise with zero mean and covariance $R = 8I$. The observations of every 2nd variable are drawn every $\Delta t_{\text{obs}} = 0.05$. Then the map F_n (3.1) corresponds to 10 forward Euler steps. This map is used to define G and the derivatives of this map are needed for the shadowing iteration. The assimilation windows is $\Delta t_{\text{ass}} = 5$.

In Figure 4 we display G-error (3.13) on the left and error with respect to the truth of non-observed variables (3.15) on the right as a function of iteration number. As for L63 displayed in Figure 1, large $w = 1000$ requires more iterations to reach the same G-error than small $w = 100$. Error with respect to the truth of non-observed variables decreases over iteration for large

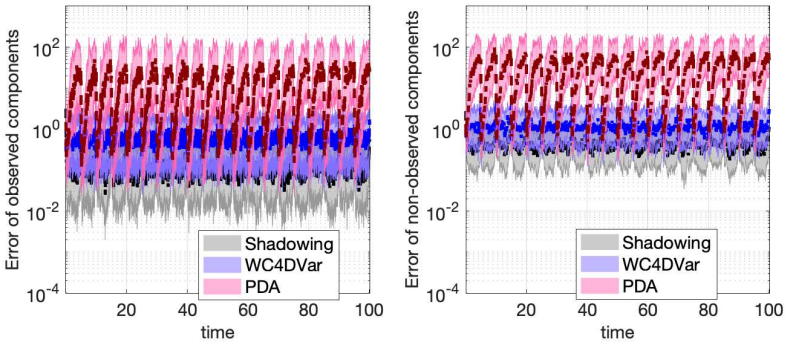

Figure 3: Application to L63. Error as a function of time: median (dashed line) +/- one standard deviation (shadowed area) over 100 simulations. On the left: error with respect to the truth of observed variables. On the right: error with respect to the truth of non-observed variables. The shadowing-based DA method with $w = 1000$ in grey, WC4DVar in blue, and PDA in pink.

$w = 1000$ while it increases for small $w = 100$. The preconditioner with $w = 1000$ yields better results.

In Figure 5, we plot error with respect to the truth of observed variables (3.14) and cost function (3.18) as a function of iteration on the left and on the right, respectively. A better estimation of observed variables is obtained with large $w = 1000$ than with small $w = 100$, as was the case for L63 displayed in Figure 2. Moreover, small $w = 100$ gives a considerable increase in the error. The cost function is underestimated with small $w = 100$ and well estimated with large $w = 1000$. Thus the preconditioner Σ with $w = 1000$ is optimal.

In Figure 6 we compare the shadowing-based DA method with $w = 1000$ to WC4DVar and PDA, where we plot error with respect to the truth over time of observed variables (3.16) and of non-observed variables (3.17) on the left and right, respectively. Here we see that the shadowing-based DA method with correctly chosen preconditioner Σ outperforms both WC4DVar and PDA.

3.5 Conclusions

We have introduced a shadowing-based DA method for partial observations based on the regularized Gauss-Newton method. We proved local convergence of the method and derived a lower bound for the algorithmic time step required for the method to converge to the manifold $G(u) = 0$. The shadowing-based DA method incorporates a preconditioner. The preconditioner scales the descent

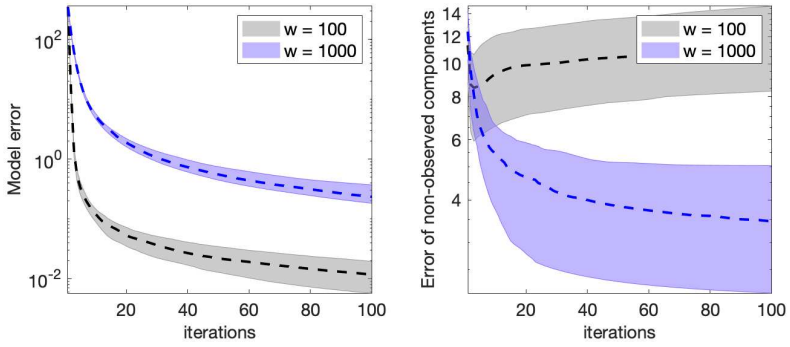

Figure 4: Application to L96. Error of the shadowing-based DA method as a function of iterations: median (dashed line) +/- one standard deviation (shadowed area) over 100 simulations. In grey error is shown for weighting matrix $w = 100$, in blue for $w = 1000$. On the left: mean over time of G-error. On the right: mean over time of error with respect to the truth of non-observed variables.

steps such that the descend step of non-observed variables is large compared to observed variables. This allows the algorithm to find a solution of $G(u) = 0$ in the vicinity of the truth. Numerical experiments with the Lorenz 63 and Lorenz 96 models show encouraging results: the shadowing-based DA method outperforms both WC4Var and PDA. The shadowing-based DA method is more expensive than PDA and WC4Var, since it requires finding eigenvalues at the first iteration, forming large matrices and inverting them. Therefore future directions include decreasing computational costs, a rigorous answer to the numerical choice of α, and error bounds with respect to the truth.

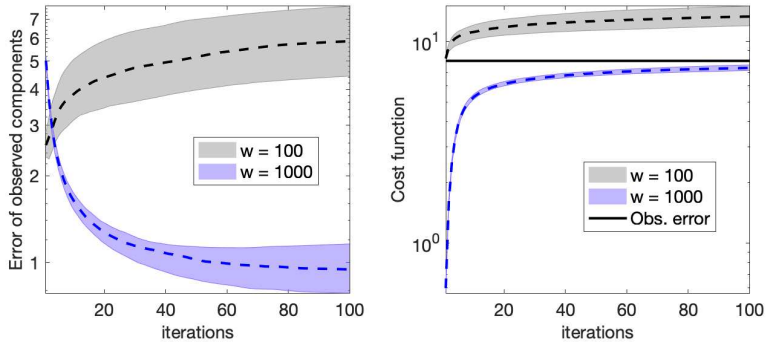

Figure 5: Application to L96. Error of the shadowing-based DA method as a function of iterations: median (dashed line) +/- one standard deviation (shadowed area) over 100 simulations. In grey error is shown for weighting matrix $w = 100$, in blue for $w = 1000$. On the left: mean over time of error with respect to the truth of observed variables. On the right: mean over time of cost function of observed variables.

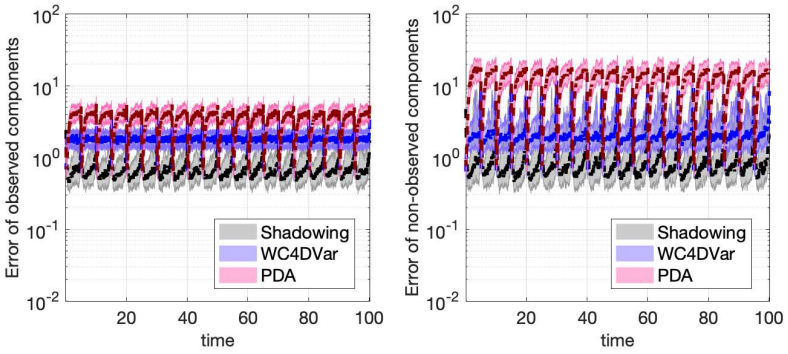

Figure 6: Application to L96. Error as a function of time: median (dashed line) +/- one standard deviation (shadowed area) over 100 simulations. On the left: error with respect to the truth of observed variables. On the right: error with respect to the truth of non-observed variables. The shadowing-based DA method with $w = 1000$ in grey, WC4DVar in blue, and PDA in pink.

CHAPTER 4

Regularized shadowing-based data assimilation method for imperfect models and its comparison to the weak constraint 4DVar method

4.1 Introduction

In this chapter we study problems for which the system model is discrete in both space and time and contaminated by model errors

$$x_{n+1} = F_n(x_n) + Q_n, \quad x_n \in \mathcal{R}^m, \quad n = 0, \ldots, N-1, \tag{4.1}$$

where $F_n : \mathcal{R}^m \to \mathcal{R}^m$ and Q_n is some random unknown model error drawn from a Gaussian distribution with zero mean and covariance matrix C_{m}. A version of this chapter may also be found in [19]. We assume F_n to be $\mathcal{C}^3$ for all n. Let the sequence $\boldsymbol{X} := \{\mathcal{X}_0, \ldots \mathcal{X}_N\}$ be a distinguished orbit of (4.1), referred to as the true solution of the model, and presumed to be unknown. Suppose we are given a sequence of noisy observations $\boldsymbol{y} := \{y_0, \ldots y_N\}$ related to $\boldsymbol{X}$ via

$$y_n = H(\mathcal{X}_n) + \xi_n, \qquad y_n \in \mathcal{R}^d, \quad n = 0, \ldots, N, \tag{4.2}$$

where $H : \mathcal{R}^m \to \mathcal{R}^d$, $d \leq m$, is the observation operator, and the noise variables ξ_n are drawn from a normal distribution with zero mean and known observational error covariance matrix C_{o}. The goal of data assimilation is to find $\boldsymbol{u} = \{u_0, u_1, \ldots, u_N\}$, $u_n \in \mathcal{R}^m$, such that the differences $\|y_n - H(u_n)\|$

and $\|u_{n+1} - F_n(u_n)\|$, $n = 0, \ldots, N$ are small in an appropriately defined sense.

The data assimilation problem may be solved through algorithms known as variational data assimilation methods, see [3] for a recent review of operational data assimilation. Those methods are based on minimization of cost functions. We assume that N is big enough so that the contribution of a so-called background cost function is negligible. Denote $\|\boldsymbol{v}\|_M := \sqrt{\boldsymbol{v}^T M^{-1} \boldsymbol{v}}$ and as a convenient abuse of notation denote C_o (C_m) as a block diagonal matrix with $N+1$ identical blocks equal to the covariance matrix C_o (C_m). Then we define the observation cost function

$$J_\mathrm{o} = \frac{1}{2} \|H(\boldsymbol{u}) - \boldsymbol{y}\|^2_{C_\mathrm{o}}, \tag{4.3}$$

and the model cost function

$$J_\mathrm{m} = \frac{1}{2} \|G(\boldsymbol{u})\|^2_{C_\mathrm{m}}, \tag{4.4}$$

where $G(\boldsymbol{u})$ is the residual functional defined as

$$G(\boldsymbol{u}) = \begin{pmatrix} G_0(\boldsymbol{u}) \\ G_1(\boldsymbol{u}) \\ \vdots \\ G_{N-1}(\boldsymbol{u}) \end{pmatrix}, \qquad G_n(\boldsymbol{u}) = u_{n+1} - F_n(u_n), \quad n = 0, \ldots, N-1. \tag{4.5}$$

Please note that the residual functional does not contain a term to account for the model error Q_n. So, unless $Q_n = 0$ for all n, $G(\boldsymbol{u}) = 0$ does not imply that $\boldsymbol{u}$ is an orbit of (4.1).

A strong constraint 4DVar problem [92] minimizes the observation cost functions (4.3) under the constraint that a perfect model orbit should be satisfied, $G(\boldsymbol{u}) = 0$. Thus the strong constraint 4DVar is only applicable for the models without model errors. When model error is present, one should consider a so-called weak constraint 4DVar problem [97], which in the state space formulation minimizes the sum of the observation (4.3) and model (4.4) cost functions.

The data assimilation problem may also be solved using shadowing-based methods. A shadowing data assimilation method known as pseudo-orbit data assimilation (PDA) [32] solves a problem of minimizing $\frac{1}{2}\|G(\boldsymbol{u})\|^2_I$. In order to stay close to observations, PDA is initialized at observations and the minimization is approximately solved using a fixed number of gradient descent steps. The PDA methods have also been applied to the weak constraint problem with the most recent approach [33] to still apply gradient descent to $\frac{1}{2}\|G(\boldsymbol{u})\|^2_I$, but only taking a limited number of steps using a stopping criterion.

Another shadowing-based approach to data assimilation is introduced in [20] and named here Newton shadowing. It is concerned with finding a root of the mismatch functional $G(\boldsymbol{u})$ rather than a minimum of a cost function, but with observations being an initial guess. It originates from numerical analyses to impose bounds on numerical error approximations of pseudo-orbits of dynamical systems [43]. Newton shadowing is, however, limited to models without model errors. In this chapter, we develop a shadowing-based data assimilation method for models with model errors. We adopt a regularization approach to find a solution of an imperfect model constrained by observations, namely an iterative regularizing Levenberg-Marquardt approach [46]. This method has been applied to data assimilation problems [50], though to stably minimize (4.3) under a strong constraint. The "classical" iterative regularizing Levenberg-Marquardt approach starts at a solution of a perfect model and aims at finding a solution of an optimization problem close to observations. This approach depends on a regularization parameter and a stopping criterion. According to Morozov's discrepancy principle, the regularization parameter can be determined uniquely. For noise-free observations and under some regularity conditions, the convergence theory guarantees that an iterative solution converges to a model orbit satisfying the observations exactly, when the number of iterations goes to infinity. For noisy observations, the stopping criterion chosen according to the discrepancy principle guarantees that at a finite iteration number the distance between an iterative solution and the minimum is smaller that the distance between the initial guess and the minimum. In the present chapter, we start at observations and aim at finding a solution of an imperfect model. Determining the regularization parameter uniquely and imposing an appropriate stopping criterion, we ensure the correctly distributed data mismatch. We call this approach weak constraint shadowing and compare it to the weak constraint 4DVar both analytically and numerically.

The chapter is organized as follows: we first recall the Newton shadowing data assimilation method in Section 4.2, then introduce the weak constraint shadowing data assimilation in Section 4.3. In Section 4.4 imperfect models are described and in Section 4.5 results of the numerical experiments with the imperfect models are discussed. Conclusions are drawn in the last section.

4.2 Newton shadowing data assimilation

In this section we describe Newton shadowing for strong constraint data assimilation developed in [20]. Newton shadowing finds model orbits close to observations by employing Newton's method for root searching of the mismatch functional $G(\boldsymbol{u})$ initialized at observations. Denoting by k the index of the Newton's iteration, we have at $k = 0$ $\boldsymbol{u} = \boldsymbol{y}$ and we seek an update $\boldsymbol{\delta}^{(k)}$

by approximately solving

$$G(\boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)}) = 0. \tag{4.6}$$

We then update using $\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)}$. The solution to (4.6) is approximated by iterating

$$G'(\boldsymbol{u}^{(k)})\boldsymbol{\delta}^{(k)} = -G(\boldsymbol{u}^{(k)}), \quad \boldsymbol{u}^{(k+1)} := \boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)} \tag{4.7}$$

to convergence. We solve each Newton's step using the right pseudo-inverse of G', i.e.

$$\boldsymbol{\delta}^{(k)} = -G'(\boldsymbol{u}^{(k)})^{\dagger} G(\boldsymbol{u}^{(k)}) = -G'^{T}(G'\,G'^{T})^{-1}G.$$

The function $G(\boldsymbol{u})$ has a zero for every orbit of the model. The Jacobian of G has a $m(N-1) \times mN$ block structure:

$$G'(\boldsymbol{u}) = \begin{bmatrix} -F_0'(u_0) & I & & & \\ & -F_1'(u_1) & I & & \\ & & \ddots & \ddots & \\ & & & -F_{N-1}'(u_{N-1}) & I \end{bmatrix}.$$

We remark that through the use of the pseudo-inverse, for every Newton step, $\boldsymbol{\delta}^{(k)}$ is the minimum 2-norm solution to (4.7).

4.3 Weak constraint shadowing data assimilation

Before we introduce weak constraint shadowing, we lift the Newton shadowing assumption of the observation operator being the identity. We introduce completed observations, where the existing observations are completed with long time "climatological" averages. Then the observation covariance matrix C_{o} is also completed by using the covariance of the completed observations.

4.3.1 Levenberg-Marquardt regularization

It is our aim to use an iterative regularization method to modify the Newton shadowing into a weak constraint shadowing data assimilation method. We propose using the Levenberg-Marquardt iteration

$$\boldsymbol{\delta}^{(k)} = -C_{\mathrm{o}}G'^{T}(\boldsymbol{u}^{(k)}) \left(G'(\boldsymbol{u}^{(k)})C_{\mathrm{o}}G'^{T}(\boldsymbol{u}^{(k)}) + \alpha^{(k)}C_{\mathrm{m}} \right)^{-1} G(\boldsymbol{u}^{(k)}), \tag{4.8}$$

$$\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)}, \tag{4.9}$$

for $\alpha^{(k)} > 0$. Under some regularity conditions and algorithms for choosing $\alpha^{(k)}$, convergence to a model orbit can be proven as $k \to \infty$ [46]. We remark

that if $C_\mathrm{o} = I$ and we choose $\alpha^{(k)} = 0$, for all k, then (4.8) reduces to the Newton shadowing (4.7). If $C_\mathrm{m} = C_\mathrm{o} = I$ and we choose $\alpha^{(k)} \to \infty$, for all k, then (4.8) reduces to the gradient descent algorithm of [55]

Dropping the trajectory dependency and iteration index for notational convenience, we rewrite (4.8) into a minimization problem. Assuming C_m and C_o are invertible, we have in exact arithmetic

$$\begin{aligned}
\boldsymbol{\delta} &= -C_\mathrm{o} G'^T (G' C_\mathrm{o} G'^T + \alpha C_\mathrm{m})^{-1} G \\
&\equiv -(G'^T C_\mathrm{m}^{-1} G' + \alpha C_\mathrm{o}^{-1})^{-1} (G'^T C_\mathrm{m}^{-1} G' + \alpha C_\mathrm{o}^{-1}) C_\mathrm{o} G'^T (G' C_\mathrm{o} G'^T + \alpha C_\mathrm{m})^{-1} G \\
&\equiv -(G'^T C_\mathrm{m}^{-1} G' + \alpha C_\mathrm{o}^{-1})^{-1} (G'^T C_\mathrm{m}^{-1} G' C_\mathrm{o} G'^T + \alpha G'^T) (G' C_\mathrm{o} G'^T + \alpha C_\mathrm{m})^{-1} G \\
&\equiv -(G'^T C_\mathrm{m}^{-1} G' + \alpha C_\mathrm{o}^{-1})^{-1} G'^T C_\mathrm{m}^{-1} (G' C_\mathrm{o} G'^T + \alpha C_\mathrm{m}) (G' C_\mathrm{o} G'^T + \alpha C_\mathrm{m})^{-1} G \\
&\equiv -(G'^T C_\mathrm{m}^{-1} G' + \alpha C_\mathrm{o}^{-1})^{-1} G'^T C_\mathrm{m}^{-1} G,
\end{aligned}$$

which implies that

$$G'^T C_\mathrm{m}^{-1} (G' \boldsymbol{\delta} + G) + \alpha C_\mathrm{o} \boldsymbol{\delta} = 0. \tag{4.10}$$

Introducing back the index notation and trajectory dependency, the solution $\boldsymbol{\delta}^{(k)}$ to (4.10) is the minimizer of

$$\frac{1}{2} \| G' \left(\boldsymbol{u}^{(k)} \right) \boldsymbol{\delta}^{(k)} + G \left(\boldsymbol{u}^{(k)} \right) \|^2_{C_\mathrm{m}} + \frac{\alpha^{(k)}}{2} \| \boldsymbol{\delta}^{(k)} \|^2_{C_\mathrm{o}} \quad \text{and} \quad \boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)}. \tag{4.11}$$

At the first iteration the weak constraint shadowing is identical to the weak constraint 4DVar, when initialized at the full observations. As the iteration proceeds, however, it becomes distinct since (4.11) does not stay fixed throughout the iteration.

4.3.2 Parameter choice

The regularization parameter $\alpha^{(k)}$ can be determined uniquely by imposing that for some $0 < \rho < 1$, $\alpha^{(k)}$ is the smallest non-negative scalar satisfying

$$\rho^{-1} \| \delta \left(\alpha^{(k)} \right) \|_{C_\mathrm{o}} \leq \sqrt{Nd} - \| H(\boldsymbol{u}^{(k)}) - \boldsymbol{y} \|_{C_\mathrm{o}}, \tag{4.12}$$

where we made the dependency of the update step δ on the parameter $\alpha^{(k)}$ explicit. In practice we fix ρ and search for $\alpha^{(k)}$ in a sequence $\{0, 2^\nu\}$ for $\nu \in \mathbb{N}$ by computing $\rho^{-1} \| \delta \left(\alpha^{(k)} \right) \|_{C_\mathrm{o}}$ and accepting the update if (4.12) is satisfied. At the next iteration $(k+1)$ we do not start from zero but rather from a previous $\alpha^{(k)}$.

For the stopping criterion we require that the distance between analysis and observations remains bounded. Denoting the principal square root of the observational precision by $C_\mathrm{o}^{-\frac{1}{2}}$, $C_\mathrm{o}^{-\frac{1}{2}} (H(\boldsymbol{X}) - \boldsymbol{y})$ is distributed according to a standard normal distribution. In particular, $\mathbb{E}(\| H(\boldsymbol{X}) - \boldsymbol{y} \|^2_{C_\mathrm{o}}) / Nd = 1$ and

when the number of observations is large enough we may assume $\|H(\boldsymbol{X}) - \boldsymbol{y}\|^2_{C_\mathrm{o}}/Nd \approx 1$ with high probability. Thus we stop the algorithm at the minimum k for which $\|H(\boldsymbol{u}^{(k)}) - \boldsymbol{y}\|^2_{C_\mathrm{o}}/Nd > r$ for a predefined parameter r close to 1.

4.3.3 Comparison to the weak constraint 4DVar

We compare weak shadowing to weak constraint 4DVar following [52]. The maximum likelihood principle assumes that both model mismatches $\boldsymbol{G}(\boldsymbol{u})$ and the observation mismatches $H(\boldsymbol{u}) - \boldsymbol{y}$ are independent Gaussian variables. The weak constraint 4DVar derived based on the maximum likelihood principle provides, however, a solution such that the model mismatches $\boldsymbol{G}(\boldsymbol{u})$ depend on the observation mismatches $H(\boldsymbol{u}) - \boldsymbol{y}$ as

$$H'^T(\boldsymbol{u})C_\mathrm{o}^{-1}\left(H(\boldsymbol{u}) - \boldsymbol{y}\right) = -G'^T(\boldsymbol{u})C_\mathrm{m}^{-1}\boldsymbol{G}(\boldsymbol{u}). \tag{4.13}$$

Therefore we compare the weak shadowing to the weak constraint 4DVar in terms of distributions of $C_\mathrm{o}^{-1/2}\left(H(\boldsymbol{u}) - \boldsymbol{y}\right)$ and of $C_\mathrm{m}^{-1/2}\boldsymbol{G}(\boldsymbol{u})$ for normally distributed $C_\mathrm{o}^{-1/2}\left(H(\boldsymbol{X}) - \boldsymbol{y}\right)$ and $C_\mathrm{m}^{-1/2}\boldsymbol{G}(\boldsymbol{X})$ at the true trajectory $\boldsymbol{X}$.

4.4 Imperfect Models

We assume that a perfect model is given by equation (4.1) with $Q_n = \sqrt{C_\mathrm{m}}\eta_n$, where η_n is drawn form the normal distribution. The imperfect model is given by equation (4.1) with $Q_n = 0$. The imperfect and perfect models are related to each other as the Euler and the Euler-Maruyama discretisation of an ODE or an SDE with Brownian motion, respectively.

4.4.1 The double-well model

The stochastic double-well model has been used to test variational data assimilation in [74]. It is described by

$$x_{n+1} = x_n + \tau\left(x_n(1 - x_n^2)\right) + \sqrt{\tau}\sigma_\mathrm{m}\eta_n, \tag{4.14}$$

where we choose the time step of the numerical discretization $\tau = 0.05$ and the model error $C_\mathrm{m} = \tau\sigma_\mathrm{m}^2$ with scalar σ_m. It has two stable equilibria at $x = \pm 1$ and one unstable equilibrium at $x = 0$. For sufficiently small stochastic noise, the model (4.14) stays near one of the stable equilibria for most of the time. Over long times, however, transitions between two stable equilibria occur. In the absence of the stochastic noise, transitions do not occur. This means that no orbit of the imperfect model is able to shadow the perfect model over long times.

4.4.2 The Lorenz 63 model

A stochastic version of the Lorenz 63 (L63) model [69] has been used in a data assimilation context in [73]. It is given by

$$\begin{aligned} x^1_{n+1} &= x^1_n + \tau 10(x^2_n - x^1_n) + \sqrt{\tau}\sigma_{\mathrm{m}}\eta^1_n, \\ x^2_{n+1} &= x^2_n + \tau(28x^1_n - x^2_n - x^1_n x^3_n) + \sqrt{\tau}\sigma_{\mathrm{m}}\eta^2_n, \\ x^3_{n+1} &= x^3_n + \tau(x^1_n x^2_n - 8/3x^3_n) + \sqrt{\tau}\sigma_{\mathrm{m}}\eta^3_n, \end{aligned}$$

where we use the standard parameter values, time step $\tau = 0.005$ and scalar σ_{m}.

4.4.3 The Lorenz 96 model

We also consider a stochastic version of the Lorenz 96 (L96) model [70] given by

$$x^l_{n+1} = x^l_n + \tau(-x^{l-2}_n x^{l-1}_n + x^{l-1}_n x^{l+1}_n - x^l_n + 8) + \sqrt{C_{\mathrm{m}}}\eta^l_n, \qquad l = 1, \ldots, 15, \tag{4.15}$$

where we use the standard parameter value for the forcing, time step $\tau = 0.005$, dimension size 15, and spatially correlated C_{m}

$$C_{\mathrm{m}} = \tau\sigma^2_{\mathrm{m}} \begin{pmatrix} 0.5 & 0.25 & 0 & \cdots & 0 & 0.25 \\ 0.25 & 0.5 & 0.25 & 0 & \cdots & 0 \\ 0 & 0.25 & 0.5 & 0.25 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & 0.25 & 0.5 & 0.25 \\ 0.25 & 0 & \cdots & 0 & 0.25 & 0.5 \end{pmatrix} \tag{4.16}$$

with scalar σ_{m}.

4.5 Numerical experiments

We use a spin-up of 5 time units for a true trajectory to reside on the attractor. We perform 100 numerical experiments with different truth and observation realizations in order to check the robustness of the results. We initialize the weak constraint 4DVar with (completed) observations, unless specified otherwise. The minimization of the cost function of the weak constraint 4DVar is done by a Matlab built-in Levenberg-Marquardt algorithm and stopping when the relative change in the cost function compared to the initial value is less than 10^{-6}. To find an adaptive $\alpha^{(k)}$ for the weak constraint shadowing we fix $\rho = 0.8$ and $r = 0.99$. We also check the performance of the weak constraint shadowing when $\alpha^{(k)} \equiv 1$, for all k, as this decreases the number of iterations.

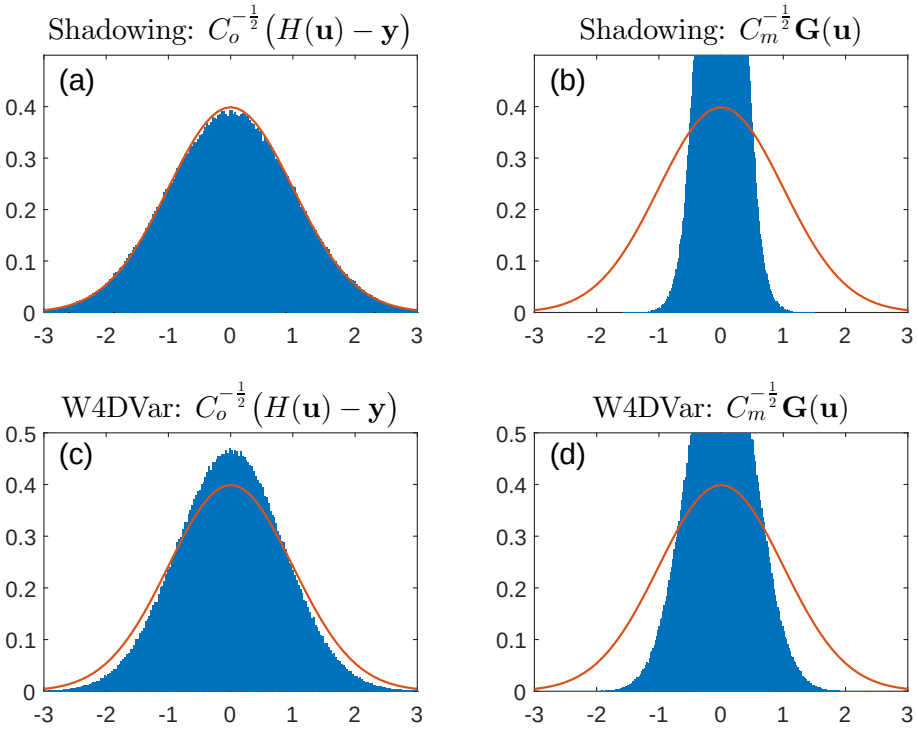

Figure 1: Histogram of the normalized data (left) and model (right) mismatches from the shadowing method (top) and the weak constraint 4DVar (bottom), for the stochastic double well model. In red we plot the standard normal distribution for reference.

4.5.1 Stochastic double well

For numerical experiments with the stochastic double well model we use $N = 4000$. We choose $\sigma_{\mathrm{m}} = 1$ and take observations at each time step with $C_{\mathrm{o}} = 0.16$. The histograms of the observation and model mismatches are shown in figure 1 for both the weak constraint shadowing and the weak constraint 4DVar. We see that the weak constraint 4DVar overfits both distance to the observations, though only slightly, and the model mismatch; while the weak constraint shadowing is overfitting only the model mismatch. In table 1, we show the mean and standard deviation over 100 experiments for the number of iterations, the observation cost function (4.3) and the model cost function (4.4). Results are given for the weak constraint 4DVar, the weak constraint shadowing with adaptive α and the weak constraint shadowing with fixed $\alpha \equiv 1$. We observe that adaptive shadowing is outperforming the weak constraint 4DVar but demands more iterations. By fixing $\alpha \equiv 1$ the shadowing results get only slightly worse but the number of

Method	Iterations	J_o/Nd	J_m/Nm
NA Shadowing	2 ± 0	0.516 ± 0.008	0.050 ± 0.002
Shadowing	6.8 ± 0.6	0.492 ± 0.001	0.062 ± 0.005
W4DVar	4.3 ± 0.5	0.365 ± 0.006	0.133 ± 0.003

Table 1: Results for the stochastic double well model averaged over 100 experiments and with standard deviations. The cost functions J_o and J_m are defined in equations (4.3) and (4.4) respectively. NA Shadowing stands for Non-Adaptive Shadowing.

iteration decreases considerably.

4.5.2 Stochastic Lorenz 63

For numerical experiments with the stochastic L63 model we use $N = 2000$ and observe only the x^1 coordinate every time step with observation error σ_o^2. The completed observations are obtained by averaging x^2 and x^3 of the deterministic L63 model over 2×10^7 time steps, $(\bar{x}^2\ \bar{x}^3)^T = (0.1015\ 24.3515)^T$. The observation covariance matrix is completed by the covariances of the long-time trajectories of x^2 and x^3 and is

$$C_\mathrm{o} := \begin{pmatrix} \sigma_\mathrm{o}^2 & 0 & 0 \\ 0 & 82.9135 & 0.3134 \\ 0 & 0.3134 & 67.2204 \end{pmatrix},$$

assuming there is no temporal correlation and no correlation between non-observed states and the observation. We choose $\sigma_\mathrm{o}^2 = 0.05$ and $\tau\sigma_\mathrm{m}^2 = 0.6$. In figure 2, we see that the shadowing method ensures the distances to observations are distributed approximately correctly, while the model mismatch is overfitted. For the weak constraint 4DVar both distance to observations and model mismatch are overfitted. Moreover, the model mismatch is not Gaussian distributed for either method.

Comparing the methods in table 2, we observe that the weak constraint 4DVar overfits the total mismatch as well. Shadowing takes only one iteration more on average compared to the weak constraint 4DVar. Fixing α to one does not decrease the number of iterations substantially, though increases the overfit, which could be improved by tuning r. We performed the same experiments but with observations every 10 time steps by completing the observations every unobservable time step. As we did not see qualitative differences compared to observing every time step, we omit the results here.

4.5.3 Stochastic Lorenz 96

For numerical experiments with the stochastic L96 model we use $N = 1000$ and observe x^1, x^6, and x^{11} coordinates every 10th time step with observation error

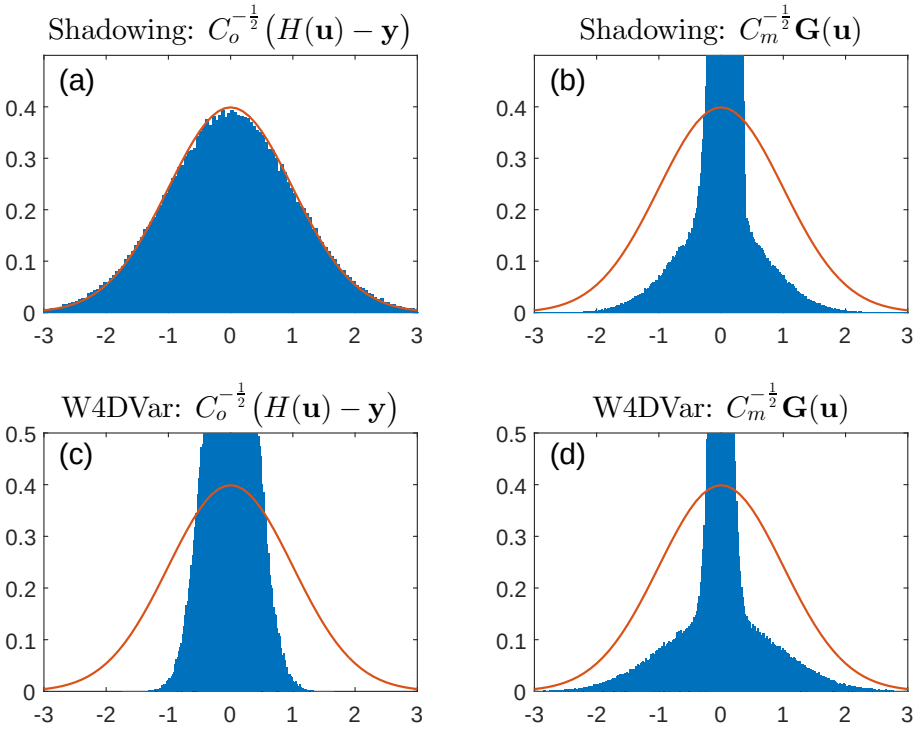

Figure 2: Histogram of the normalized data (left) and model (right) mismatches from the shadowing method (top) and the weak constraint 4DVar (bottom), for partially observed stochastic L63 model. In red we plot the standard normal distribution for reference.

Method	Iterations	J_o/Nd	J_m/Nm
NA Shadowing ($r = 0.9$)	3.7 ± 0.5	0.54 ± 0.08	0.10 ± 0.01
NA Shadowing ($r = 0.99$)	4 ± 0	0.6 ± 0.02	0.09 ± 0.003
Shadowing	6.2 ± 0.6	0.494 ± 0.002	0.101 ± 0.005
W4DVar	5.1 ± 0.3	0.064 ± 0.002	0.145 ± 0.005

Table 2: Results for the partially observed stochastic L63 model averaged over 100 experiments and with standard deviations. The cost functions J_o and J_m are defined in equations (4.3) and (4.4) respectively. NA Shadowing stands for Non-Adaptive Shadowing with different values of r in brackets.

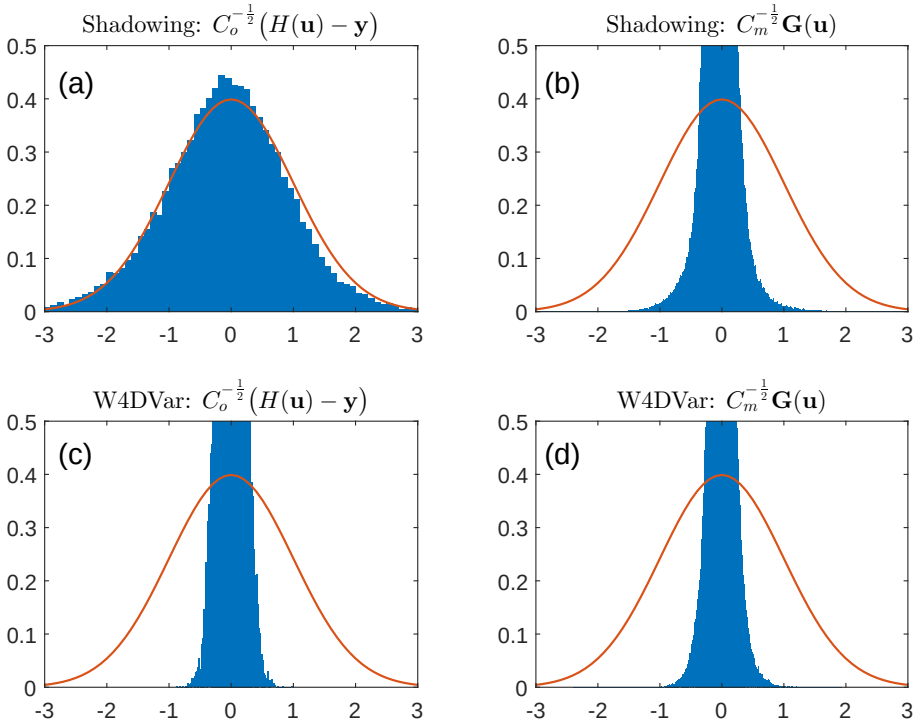

Figure 3: Histogram of the normalized data (left) and model (right) mismatches from the shadowing method (top) and the weak constraint 4DVar (bottom), for partially observed stochastic L96 model. In red we plot the standard normal distribution for reference.

$\sigma_o = 0.01$. The competed observations are averages of 2×10^6 time steps of the deterministic L96 model. For the model error we choose $\sigma_m = \sqrt{20}$. In figure 3, we see that the shadowing method provides the correct distribution for the data mismatch, while the weak constraint 4DVar drastically underestimates its variance.

In table 3, we observe that the weak constraint 4DVar takes an order of magnitude more iterations to converge than the shadowing method, independent of initialization. The weak constraint 4DVar overfits both the data and the model mismatch. Adaptive shadowing, on the other hand, provides the best data mismatch distribution and takes only 6 iterations on average. When fixing α to one and tuning r, an equivalent result can be achieved in even less iterations.

When so little data is given, data assimilation is not expected to give accurate results for unobserved variables. However, their behavior is of importance due

Method	Iterations	$10J_o/Nd$	J_m/Nm
NA Shadowing ($r = 0.9$)	3.2 ± 0.5	0.50 ± 0.06	
NA Shadowing ($r = 0.99$)	3.7 ± 0.6	0.59 ± 0.06	
Shadowing	6.5 ± 0.7	0.498 ± 0.002	0.03 ± 0.01
W4DVar (Bg)	55 ± 29	0.017 ± 0.002	0.014 ± 0.003
W4DVar (Obs)	49 ± 20	0.017 ± 0.002	0.011 ± 0.002

Table 3: Results for the partially observed stochastic L96 model with observations every 10 steps, averaged over 100 experiments and with standard deviations. The cost functions J_o and J_m are defined in equations (4.3) and (4.4) respectively. We adjust the cost function normalization, namely $Nd \mapsto 0.1Nd$ to take into account the sparsity in time of the observations. NA Shadowing stands for Non-Adaptive Shadowing with different values of r in brackets. W4DVar (Bg) stands for initialization at background and W4DVar (Obs) stands for initialization at observations.

to desirable dynamical consistency. Therefore we perform a long-time data assimilation experiment with $N = 10^5$. Computing a distance between the true solution $\boldsymbol{X}$ and a solution $\boldsymbol{u}$, namely $\|\boldsymbol{X} - \boldsymbol{u}\|_I^2/N(m-d)$, a long-time average for the deterministic L96 model gives 18.8, the weak constraint 4DVar 22.7, and the shadowing method 16.5. This mainly reflects that the distance to the original "true" state is not a good measure of the performance of data assimilation algorithms for the weak constraint problem. Instead, we study the distribution of unobserved variables. The width of this distribution signifies how far an analysis deviates from the initial guess, which is a long-time average for the deterministic L96 model and is 2.31 in this case. In particular, it can be seen in figure 4 that the weak constraint 4DVar produces some pseudo-trajectory for the unobserved components. This pseudo-trajectory, however, is not particularly close to the truth. For the shadowing method, the unobserved variables do not significantly deviate from the initial background guess, which is shown as a black line. This is related to the stability of the shadowing method with respect to perturbations in the initial condition. Therefore, with little meaningful information on the unobserved variables the shadowing method provides a solution that is both stable with respect to the noise realization and a good reflection of the lack of knowledge.

4.6 Conclusions

We proposed a data assimilation method for imperfect models. The method is based on combination of numerical shadowing, a weak constraint formulation, and regularization. Numerical shadowing ensures stability with respect to observational noise, the weak constraint formulation introduces model error in the method, and Levenberg-Marquardt regularization prevents overshooting in the estimation. The appropriately chosen regularization parameter together

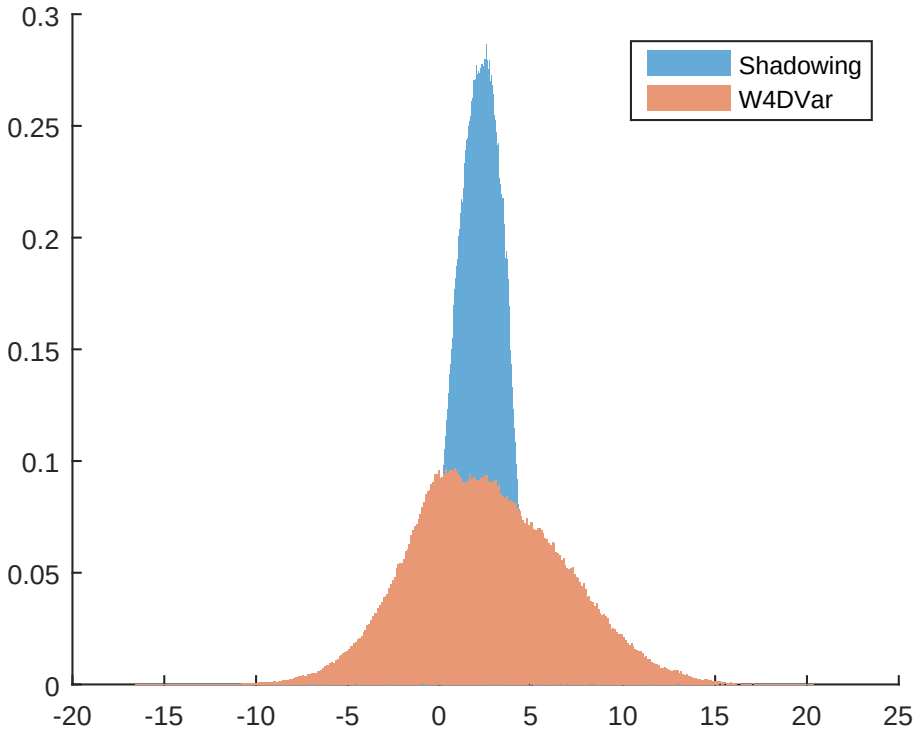

Figure 4: Histogram of all the unobserved variables for a partially observed long trajectory of stochastic L96. The weak constraint 4DVar is in red, and the shadowing method is in blue. The initial guess is in black.

with a data mismatch stopping criterion guarantees that the data mismatch is correctly distributed. We demonstrated that the shadowing method is successful for observations that are sparse both in space and in time.

We compared the proposed weak constraint shadowing-based method to the weak constraint 4DVar both analytically and numerically. We pointed out that they are identical only at the first iteration. Numerical experiments with stochastic models of double well, Lorenz 63, and Lorenz 96 confirmed analytic results that the shadowing method always estimates accurately distributions of the data mismatch. The weak constraint 4DVar, on the contrary, gives poor estimations of the data mismatch distributions, which become more deficient with fewer data. Moreover, unobserved variables only weakly deviate from an initial guess for the shadowing method, while strongly for the weak constraint 4DVar without being particularly close to the true trajectory. This is an advantage of the shadowing method, as sparse observations should not influence model states that are far away from the observation locations.

With respect to the model mismatch distributions, both the shadowing method and the weak constraint 4DVar perform poorly. This could be improved in the shadowing method by applying "classical" Levenberg-Marquardt regularization with a model mismatch stopping criterion. Then one gets the correct model mismatch distribution but at a price of misestimating the data mismatch distribution. As the goal of this study was to be close to accurate observations rather than to erroneous model estimations, this approach is unsuitable.

Future work will consist of further development of the shadowing method to use approximate adjoint models, generalizing the method to an ensemble approximation, and applying it to structurally incorrect models.

CHAPTER 5

Shadowing-based data assimilation method for imperfect models using an ensemble

5.1 Introduction

In this chapter we study models with structural errors. Assume the model is given by (3.1). When one assumes model error is present and is additive at each time step, the true model is

$$x_{n+1} = F_n(x_n) + \hat{\eta}_n, \quad x_n \in \mathcal{R}^m, \quad n = 0, \ldots, N-1. \tag{5.1}$$

In (5.1), $\hat{\eta}_n$ is unknown. We assume the dynamics F_n are known and we assume a statistical estimate of $\hat{\eta}_n$ is known. A simple approximation would involve that the model is unbiased and modelling $\hat{\eta}_n$ as a random variable drawn from a Gaussian distribution with covariance matrix C_{m}. One can consider (5.1) as a numerical discretization of an SDE $dx = f(x)dt + dW$. We remark that two cases can be distinguished in this example. Either the unknown $\hat{\eta}_n$ *is* indeed Gaussian noise, but the realization of the noise is not known; or the unknown $\hat{\eta}_n$ is not Gaussian noise, but could for example be a non-modeled deterministic term, possibly with non-zero mean (so, possibly with model bias).

We studied the case where the unknown $\hat{\eta}_n$ is Gaussian noise in [19]. Here, we focus on the case where $\hat{\eta}_n$ has known mean and covariance, but is not Gaussian distributed. Without essential loss of generality, we may assume that the known mean is zero. If the known mean would be non-zero, it is possible to

improve the model by adding this known mean forcing as a constant term. In our examples, the missing term will be described by deterministic differential equations.

The model we study in our examples is an idealized coupled atmosphere-ocean model [75, 48] and is given by

$$\dot{x_1} = \sigma(x_2 - x_1) + \alpha_c x_5 \tag{5.2}$$
$$\dot{x_2} = x_1(\rho - x_3) - x_2 + \alpha_c x_4 \tag{5.3}$$
$$\dot{x_3} = x_1 x_2 - \beta x_3 \tag{5.4}$$
$$\dot{x_4} = -\Omega x_5 - k(x_4 - w_*) - \alpha_c x_2 \tag{5.5}$$
$$\dot{x_5} = \Omega(x_4 - w_*) - k x_5 - \alpha_c x_1, \tag{5.6}$$

where $\sigma = 10$, $\beta = \frac{8}{3}$, $\rho = 28$, $\Omega = 1.5$, $k = 0.1$, $w_* = 2$, and $\alpha_c = 1$. This model can be interpreted as a highly simplified coupled atmosphere-ocean model [35]. In this interpretation, the variables x_1, x_2, x_3 model the atmospheric state and x_4, x_5 model the oceanic state. If the coupling parameter $\alpha_c = 0$, the equations for the atmosphere reduce to the Lorenz '63 model [69], while the equations for the ocean become a linear system. The Lyapunov exponents for the coupled model, based on a computation with 1 million time steps, are $0.89, 0.00, -0.08, -0.09, -14.72$.

Let the sequence $\boldsymbol{X} := \{\mathcal{X}_0, \dots \mathcal{X}_N\}$ be a distinguished orbit of (5.1), referred to as the true solution, and presumed to be unknown. If the model error $\hat{\eta}_n$ is stochastic, $\boldsymbol{X}$ is a realization of an SDE, if the model error $\hat{\eta}_n$ is deterministic but unknown, $\boldsymbol{X}$ is an orbit of an ODE. Suppose we are given a sequence of noisy observations $\boldsymbol{y} := \{y_0, \dots y_N\}$ related to $\boldsymbol{X}$ via

$$y_n = H(\mathcal{X}_n) + \xi_n, \qquad y_n \in \mathbb{R}^d, \quad n = 0, \dots, N,$$

where $H : \mathcal{R}^m \to \mathcal{R}^d$, $d \leq m$, is the observation operator, and the noise variables ξ_n are drawn from a normal distribution with zero mean and known observational error covariance matrix C_{o}. A goal of data assimilation is to find $\boldsymbol{u} = \{u_0, u_1, \dots, u_N\}$, $u_n \in \mathcal{R}^m$, such that the differences $\|y_n - H(u_n)\|$ and $\|u_{n+1} - F_n(u_n)\|$, $n = 0, \dots, N$ are small in an appropriately defined sense. A more ambitious goal of data assimilation is not only to find one such orbit $\boldsymbol{u}$, but rather to find a probability distribution of possible orbits. Most data assimilation algorithms attempt to find such a distribution using a Bayesian approach [63].

We remark that in this chapter we study data assimilation with imperfect models. That is, $F_n(u_n)$ is known, but the truth $\boldsymbol{X}$ is not an orbit of F_n, so $\|\mathcal{X}_{n+1} - F_n(\mathcal{X}_n)\| \neq 0$ in general. Indeed, in section 5.4, we shall show numerical examples where $\boldsymbol{X}$ is an orbit of (5.2)-(5.6), while F_n is either linear or the Lorenz '63 model. The data assimilation problem may, for example, be solved using a 4D variational data assimilation approach (4DVar), e.g. [92].

Variational data assimilation methods are based on minimization of a cost function to obtain the initial conditions. One reference trajectory is obtained. For the non-perfect model scenario (5.1), the so-called weak 4DVar method may be used [97]. However, some implementations of 4DVar are unsuitable for use over long time windows, that is, for large N and if the dynamics F_n is chaotic, the 4DVar optimization problem becomes untractable [6, 74, 82]. Strong-constraint variatonal data assimilation is suitable for the perfect model scenario [92].

A well-known method for obtaining data assimilation ensembles is the randomized maximimum likelihood (RML) method [77]. This method is related to 4DVar. For the non-perfect model, or weak constraint, scenario, the RML method generates ensemble members $\boldsymbol{u}_{RML}$ as the minimizers of cost functions

$$J_{RML}(\boldsymbol{u}) := \frac{1}{2}\sum_{n=0}^{N} \|y_n - H(u_n) + \tilde{\xi}_n\|^2_{C_\mathrm{o}} + \frac{1}{2}\sum_{n=0}^{N-1} \|u_{n+1} - F_n(u_n) + \tilde{\eta}_n\|^2_{C_\mathrm{m}}, \quad (5.7)$$

where $N+1$ is the number of observations, $\tilde{\xi}_n$ and $\tilde{\eta}_n$ are random variables and $\|.\|_{C_\mathrm{o}}$ and $\|.\|_{C_\mathrm{m}}$ are two-norms weighted by matrices C_o and C_m respectively. The matrices C_o and C_m may in principle depend on n, but we shall not study that case in this chapter.

For example, if $\hat{\eta}_n$ is modelled as Gaussian noise random variable with mean zero and covariance matrix C_m, the random variable $\tilde{\eta}_n$ should be sampled according to the same Gaussian distribution. Analogously, if ξ_n is a Gaussian random variable with mean zero and covariance matrix C_o, $\tilde{\xi}_n$ should be sampled from the same distribution. It is noted here that choosing a distribution for the model error and in particular choosing C_m is a non-trivial modelling assumption, on which we will comment in the numerical experiments section 5.4 below. If F_n is furthermore linear, the minimizers of the RML cost functions sample from the Bayesian posterior. For completeness, we remark that if $\tilde{\eta}_n$ and $\tilde{\xi}_n$ are set to zero for all n, the RML cost function reduces to the weak 4DVar cost function (in this case in the state space formulation).

Instead of minimizing a cost function, in [20] we proposed a strong constraint data assimilation method (when model error is absent)that searches for a zero of the residual functional $G(\boldsymbol{u})$ defined as

$$G(\boldsymbol{u}) = \begin{pmatrix} G_0(\boldsymbol{u}) \\ G_1(\boldsymbol{u}) \\ \vdots \\ G_{N-1}(\boldsymbol{u}) \end{pmatrix}, \qquad G_n(\boldsymbol{u}) = u_{n+1} - F_n(u_n), \quad n = 0, \ldots, N-1.$$

Instead of solving directly for the initial condition as in variational data

assimilation, we solve for the whole orbit at once. This approach is motivated by research on numerical shadowing methods [95].

5.2 Weak shadowing data assimilation

In [19], we formulated the shadowing-based method of [20] in a weak constraint form, when model error is present, and when observations are partial in space. Then the shadowing data assimilation is initialized at so-called completed observations $\boldsymbol{\mathcal{Y}}$, where the available observations $\boldsymbol{y}$ are combined with long-time averages of unobserved states of (5.1). The completed observation error covariance matrix $\mathcal{C}_{\mathrm{o}}$ is also a combination of the observation error covariance matrix C_{o} and a covariance matrix of unobserved states from long-time simulations of (5.1).

Initializing at completed observations, we take model error into account following the Levenberg-Marquardt regularization approach [46, 60, 50, 49] and impose a stopping criterion based on data mismatch. As also described below, Levenberg-Marquardt regularization is an algorithm combining properties of Newton's iteration and gradient descent. Denoting by k the index of Newton's iteration, we have at $k = 0$ $\boldsymbol{u} = \boldsymbol{\mathcal{Y}}$ and we seek an update $\boldsymbol{\delta}^{(k)}$ by approximately solving

$$G[\boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)}] = 0.$$

We then update using $\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)}$, where

$$\boldsymbol{\delta}^{(k)} = -\mathcal{C}_{\mathrm{o}} G'^{T}[\boldsymbol{u}^{(k)}] \left\{ G'[\boldsymbol{u}^{(k)}] \mathcal{C}_{\mathrm{o}} G'^{T}[\boldsymbol{u}^{(k)}] + \alpha^{(k)} C_{\mathrm{m}} \right\}^{-1} G[\boldsymbol{u}^{(k)}], \tag{5.8}$$

for $\alpha^{(k)} > 0$. As a convenient abuse of notation we denote $\mathcal{C}_{\mathrm{o}}$(respectively C_{m}) as a block diagonal matrix with $N+1$ identical blocks equal to the covariance matrix $\mathcal{C}_{\mathrm{o}}$(respectively C_{m}). The solution $\boldsymbol{\delta}^{(k)}$ to (5.8) is the minimizer of

$$\frac{1}{2} \left\| G'[\boldsymbol{u}^{(k)}] \boldsymbol{\delta}^{(k)} + G[\boldsymbol{u}^{(k)}] \right\|^2_{C_{\mathrm{m}}} + \frac{\alpha^{(k)}}{2} \left\| \boldsymbol{\delta}^{(k)} \right\|^2_{\mathcal{C}_{\mathrm{o}}}, \tag{5.9}$$

where $\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \boldsymbol{\delta}^{(k)}$ and $\|\boldsymbol{v}\|^2_C := \boldsymbol{v}^T C^{-1} \boldsymbol{v}$. We remark that if $\mathcal{C}_{\mathrm{o}} = C_{\mathrm{o}} = I$ and we choose $\alpha^{(k)} = 0$, for all k, then (5.8) reduces to the strong constraint shadowing of [20]. If $C_{\mathrm{m}} = I$, $\mathcal{C}_{\mathrm{o}} = C_{\mathrm{o}} = I$ and we choose $\alpha^{(k)} \to \infty$, for all k before minimizing, then (5.8) reduces to the gradient descent algorithm of PDA [32], for which the step size $\boldsymbol{\delta}^{(k)}$ is infinitesimally small. At the first iteration with $\alpha^{(1)} = 1$, the weak constraint shadowing is identical to the weak constraint 4DVar, when initialized at the full observations. As the iteration proceeds, however, it becomes distinct since (5.9) does not stay fixed throughout the iteration.

The regularization parameter $\alpha^{(k)}$ can be determined uniquely by imposing that for some $0 < \rho < 1$, $\alpha^{(k)}$ is the minimal non-negative scalar that satisfies

$$\rho^{-1} \left\| H(\delta(\alpha^{(k)})) \right\|_{C_o} \leq \sqrt{Nd} - \left\| H(\boldsymbol{u}^{(k)}) - \boldsymbol{y} \right\|_{C_o}, \tag{5.10}$$

where H is the observation operator and Nd is the total number of observations. This choice of $\alpha^{(k)}$ is optimal, in the sense that with this choice of $\alpha^{(k)}$ convergence to a minimum in a neighbourhood of the initial guess is achieved in the minimal number of iterations. However, it is not easy to find $\alpha^{(k)}$ according to (5.10) [46, 60]. As long as the regularization parameter is chosen at each step satisfies (5.10), convergence is still achieved, albeit more slowly than by choosing the smallest regularization parameter satisfying (5.10). So, in practice a regularization parameter can be chosen by checking the condition (5.10) for an increasing series of potential regularization parameters and selecting the first potential regularization parameter satisfying the condition (5.10).

During iterations, a zero of G is approximated. That is, at each iteration the pseudo-orbit $\boldsymbol{u}^{(k)}$ approaches an orbit. However, during iterations $\boldsymbol{u}^{(k)}$ will also tend to go further away from the initial guess. The initial guess is at observations. Therefore, a stopping criterion is imposed to prevent the iteration from straying too far from the observations. The stopping criterion requires the observational mismatch to remain bounded. Namely the algorithm stops at the first iteration k_* satisfying

$$\left\| H[\boldsymbol{u}^{(k_*)}] - \boldsymbol{y} \right\|_{C_o} \geq r\sqrt{Nd}, \tag{5.11}$$

where the predefined parameter $r \approx 1$ but $r < 1$. Since $\|H[\boldsymbol{u}^{(k)}] - \boldsymbol{y}\|_{C_o} \geq \|H[\boldsymbol{u}^{(0)}] - \boldsymbol{y}\|_{C_o} \equiv 0$ for any k, it is noted that, provided that $\alpha^{(k)}$ is selected according to (5.10), $\boldsymbol{u}^{(k)}$ either converges to an orbit $G(\boldsymbol{u}^{(k)}) = 0$ while staying within a ball $\left\| H[\boldsymbol{u}^{(k)}] - \boldsymbol{y} \right\|_{C_o} < r\sqrt{Nd}$ around observations, or the algorithm stops when $\left\| H[\boldsymbol{u}^{(k_*)}] - \boldsymbol{y} \right\|_{C_o} \approx r\sqrt{Nd}$, since k^* is the first iteration at which (5.11) is a satisfied. In other words, the algorithm starts with observations as initial guess and converges towards a trajectory. Since the model is imperfect, there may not be any trajectory in the vicinity of the observations. Therefore, the algorithm stops when the analysis drifts too far away from observations, where the parameter r governs what is "too far". When $r \approx 1$, the algorithm is stopped when the observational mismatch is comparable to the observational error. The parameter ρ governs the speed of convergence. When ρ is small, step length will be short and the algorithm will behave similarly to gradient descent. When ρ is large, step length will be longer, but there is a larger risk the solution overshoots the stopping criterion, leading to an observational mismatch larger than the observational error. For our numerical experiments, we set $r = 0.99$ and $\rho = 0.99$.

5.3 Ensemble shadowing data assimilation

We extend weak constraint shadowing to an ensemble approximation. Assume we have an ensemble of J samples of a random variable, $\{\boldsymbol{\eta}^j\}_{j=1}^J$, where $\boldsymbol{\eta}^j := \{\eta_0^j, \ldots, \eta_{N-1}^j\}$. We assume this ensemble samples the model error and it is drawn from a Gaussian distribution with zero mean and known model error covariance matrix C_{m}. Therefore we need to adjust the model (5.1) to account for model error. We seek a pseudo-orbit of

$$x_{n+1} = F_n(x_n) + \eta_n^j, \quad x_n, \eta_n^j \in \mathcal{R}^m, \quad n = 0, \ldots, N-1.$$

Then the model states are updated by $\boldsymbol{u}^{j(k+1)} = \boldsymbol{u}^{j(k)} + \boldsymbol{\delta}^{j(k)}$ with

$$\boldsymbol{\delta}^{j(k)} = -\mathcal{C}_{\mathrm{o}} G'^T[\boldsymbol{u}^{j(k)}] \left\{ G'[\boldsymbol{u}^{j(k)}] \mathcal{C}_{\mathrm{o}} G'^T[\boldsymbol{u}^{j(k)}] + \alpha^{j(k)} C_{\mathrm{m}} \right\}^{-1} \left\{ G[\boldsymbol{u}^{j(k)}] - \boldsymbol{\eta}^j \right\}, \tag{5.12}$$

where $\boldsymbol{u}^{j(0)} = \boldsymbol{\mathcal{Y}}$ for $j = 1, \ldots, J$. The regularization parameter $\alpha^{j(k)}$ is chosen according to (5.10) for $j = 1, \ldots, J$. By $\boldsymbol{\eta}^0$ we denote $\boldsymbol{\eta} \equiv \mathbf{0}$, which corresponds to a pseudo-orbit of (5.1) obtained by the LM regularization (5.8) and named here a reference solution.

This choice of algorithm intends to generate an ensemble of states that are indistinguishable from the true state [55, 56, 53]. This means that there is not enough information in the imperfect model and observations to distinguish the generated state from the truth based on what is known. Since we have assumed that the truth satisfies (5.1) and $\{\boldsymbol{\eta}^j\}_{j=1}^J$ samples the model error, an ensemble of orbits $\left\{\tilde{\boldsymbol{u}}^j\right\}_{j=1}^J$ satisfying $\left\{G(\tilde{\boldsymbol{u}}^j{}_{j=1}^J) = \boldsymbol{\eta}^j\right\} = 1^J$ samples the assumed distribution of the true orbit. After generating the sample, the shadowing algorithm is carried out in the same way as described previously. The jth ensemble member is then expected to have mismatches $\|G(Bu^{j(k_*)}) - \boldsymbol{\eta}^j\| \approx 0$ and $\left\|H[\boldsymbol{u}^{j(k_*)}] - \boldsymbol{y}\right\|_{C_{\mathrm{o}}} \approx r\sqrt{Nd}$.

In [55, 56], indistinguishable states $\tilde{\boldsymbol{u}}$ are defined as states for which both $\|G(\tilde{\boldsymbol{u}})\|$ and $\left\|H[\tilde{\boldsymbol{u}}^{j(k_*)}] - \boldsymbol{y}\right\|$ are bounded, for appropriately chosen norms and bounds. What norms and bounds are appropriate depends on the assumed (distributions of) model imperfections and observational errors, respectively.

In our case, we choose the model mismatch norm $\|...\|_{C_m}$ and bound $\|\boldsymbol{\eta}^j\|_{C_m}$, for each ensemble member. Other choices for the model mismatch and bound are also possible. In particular, it is not necessary to make a Gaussian estimate for the model mismatch. Another possible choice for an estimate of the model mismatch may be found in [40], in which the model mismatch is estimated to be time-independent over assimilation windows (for performance reasons).

For our model mismatch estimate, it holds that $\mathbb{E}(\|\boldsymbol{\eta}^j\|_{C_m}) = \sqrt{mN}$. However, in practice this model mismatch value may be not quite achieved,

since the algorithm is stopped according to a stopping criterion. This stopping criterion defines the norm and bound for the observational mismatch as $\left\|H[\boldsymbol{u}^{j(k_*)}] - \boldsymbol{y}\right\|_{C_\mathrm{o}} \leq r\sqrt{Nd}$. These criteria are considered to be appropriate for the assumptions regarding the model and observational mismatch. In this regard, it is remarked that, by the law of large numbers, $\mathbb{E}\|(G(\mathcal{X}))\|_{C_m} = \sqrt{mN}$ and $\mathbb{E}\left(\|H[\mathcal{X}] - \boldsymbol{y}\|_{C_\mathrm{o}}\right) = \sqrt{Nd}$. So, the ensemble members are chosen to approximate the ensemble of indistinghuishable states, as defined in detail in [55, 56, 53].

5.3.1 Well-posedness

We remark that the ensemble shadowing (5.12) can be described as

$$\boldsymbol{u}^{(k+1)} = \boldsymbol{u}^{(k)} + \frac{1}{\alpha^{(k)}}\Gamma(\boldsymbol{u}^{(k)}, \alpha^{(k)}), \tag{5.13}$$

where

$$\Gamma(\boldsymbol{u}^{(k)}, \alpha^{(k)}) = -\mathcal{C}_\mathrm{o}G'[\boldsymbol{u}^{(k)}]^T \left\{ \frac{1}{\alpha^{(k)}} G'[\boldsymbol{u}^{(k)}]\mathcal{C}_\mathrm{o}G'[\boldsymbol{u}^{(k)}]^T + C_\mathrm{m} \right\}^{-1} \left\{ G[\boldsymbol{u}^{(k)}] - \boldsymbol{\eta} \right\}.$$

We denote by $\mathcal{T} = \sum_{k=0}^{k_*-1} \frac{1}{\alpha^{(k)}}$ and by $h = \max[\frac{1}{\alpha^{(k)}}]$. Then taking the limit of $h \to 0$, we get on $\tau \in [0, \mathcal{T}]$

$$\frac{d\boldsymbol{u}}{d\tau} = g(\boldsymbol{u}), \quad \text{with} \quad g(\boldsymbol{u}) = -\mathcal{C}_\mathrm{o}G'(\boldsymbol{u})^T C_\mathrm{m}^{-1}[G(\boldsymbol{u}) - \boldsymbol{\eta}], \quad \text{and} \quad \boldsymbol{u}(0) = \boldsymbol{\mathcal{Y}}. \tag{5.14}$$

Stationary points of (5.14) only occur if $G(\boldsymbol{u}) = \boldsymbol{\eta}$ or if $G'(\boldsymbol{u})^T$ is not of full rank. Defining $\Phi(\boldsymbol{u}) = 0.5\|C_\mathrm{m}^{-1/2}[G(\boldsymbol{u}) - \boldsymbol{\eta}]\|^2$, the ODE (5.14) becomes

$$\frac{d\boldsymbol{u}}{d\tau} = -\mathcal{C}_\mathrm{o}\nabla\Phi(\boldsymbol{u}).$$

This is a preconditioned gradient descent for $\Phi(\cdot)$. We remark that the ensemble Kalman filter can be written as a gradient flow, see e.g. [86, 87] where the continuous time limit was studied to show well-posedness and convergence for a finite ensemble size. Assuming $\mathcal{C}_\mathrm{o}$ is invertible, it holds that

$$\frac{d}{d\tau}\frac{1}{2}\|C_\mathrm{m}^{-1/2}[G(\boldsymbol{u}) - \boldsymbol{\eta}]\|^2 = \frac{d}{d\tau}\Phi(\boldsymbol{u}(\tau)) = \left(\frac{d\Phi}{d\boldsymbol{u}}\right)^T \frac{d\boldsymbol{u}}{d\tau} = -\left\|\mathcal{C}_\mathrm{o}^{-1/2}\frac{d\boldsymbol{u}}{d\tau}\right\|^2 \leq 0,$$

which gives an a priori bound on $\|C_\mathrm{m}^{-1/2}[G(\boldsymbol{u}) - \boldsymbol{\eta}]\|$ but not on $\|\boldsymbol{u}\|$.

Theorem 3 *(Well-posedness for locally Lipschitz problems, e.g. [30]) Let the right-hand side g of* (5.14) *to be locally Lipschitz. Define $B = \{\boldsymbol{u} \in \mathcal{R}^{(N+1)m} : \|\boldsymbol{u} - \boldsymbol{\mathcal{Y}}\| \leq b\}$ and let $M = \max_{\boldsymbol{u}\in B}\|g(\boldsymbol{u})\|$. Then there exists a unique solution of* (5.14) *for $\tau \leq b/M$ which satisfies $u(\tau) \in B$.*

Corollary 3.1 *Under Theorem 3 the IVP* (5.14) *is well-posed on* $[0\ \mathcal{T}]$ *for any* ξ *and* $\hat{\eta}$ *such that* $\|\boldsymbol{u} - \mathcal{Y}(\xi, \hat{\eta})\| \leq b$ *if either*
a) $\boldsymbol{\mathcal{Y}} = \boldsymbol{y}$*, or*
b) $\boldsymbol{\mathcal{Y}} \neq \boldsymbol{y}$ *and there exist* $N^* \leq N$*,* $m^* \leq m$*, and* $M^* < M$ *for which* $\mathcal{T}(N^*, m^*) \leq b/M^*$*.*

Proof In order to show well-posedness on $[0\ \mathcal{T}]$, it is sufficient to show that $\mathcal{T} \leq b/M$. In case of full observations $\boldsymbol{\mathcal{Y}} = \boldsymbol{y}$, the final algorithmic time $\mathcal{T} = \sum_{k=0}^{k_*-1} \frac{1}{\alpha^{(k)}}$ does not depend on M but on the observation error due to the stopping criterion (5.11). It is noted here that the only case in which the stopping criterion is not reached is when the algorithm converges within the ball B and in which case the IVP is well posed for any $\mathcal{T}$. If the stopping criterion (5.11) is reached, choosing $\alpha^{(k)}$ according to (5.10) ensures there exists a unique solution to the IVP. In case of incomplete observations, the same argument holds for observed components of $\boldsymbol{u}$. However, unobserved components of $\boldsymbol{u}$ might blow up before $\mathcal{T}$ is reached due to large M. Thus in order for the IVP (5.14) to be well-posed on $[0\ \mathcal{T}]$, we need to find $M^* < M$ for which $\mathcal{T} \leq b/M^*$, or otherwise to find $N^* \leq N$ and $m^* \leq m$ such that $\mathcal{T}(N^*, m^*) \leq b/M^*$. Let us define $B^* = \{\boldsymbol{u} \in \mathcal{R}^{(N^*+1)m^*} : \|\boldsymbol{u} - \boldsymbol{\mathcal{Y}^*}\| \leq b\}$, where $\boldsymbol{\mathcal{Y}}^* = P\boldsymbol{\mathcal{Y}}$ with the projection operator $P : \mathcal{R}^{(N+1)m} \to \mathcal{R}^{(N^*+1)m^*}$ and $M^* = \max_{\boldsymbol{u} \in B^*} \|g(\boldsymbol{u})\|$. Then by Theorem 3 the IVP (5.14) is well-posed for $\tau \leq b/M^*$, and if $\mathcal{T}(N^*, m^*) \leq b/M^*$ it is well posed on $[0\ \mathcal{T}]$. This completes the proof. □

We remark that decreasing N leads to a shorter assimilation window. Decreasing m can only be performed in case of space dimension and thus by dividing the whole spatial domain into subdomains (akin to localization). The state, however, has to be *full*. Since well posedness is not always guaranteed, one has to find N^* and m^* by the trial-and-error approach. We note that in practice we search for $\alpha^{(k)}$ in a sequence $\{2^\nu\}$ for $\nu \in \mathbb{N}$, for which (5.10) is satisfied. Thus we have $\mathcal{T} < 2$. In Appendix 5.A, we show that $\mathcal{T}$ is, however, different for every ensemble member and the reference.

5.3.2 Accuracy

In order to show accuracy, we need to assume full observations. For linear models, the analysis is unbiased with respect to the true solution after any number of iterations. If $G(x) = Ax + b$ and α is chosen a priori, the covariance matrix of the error between the analysis and the true solution after one iteration is

$$C_{\text{err}} = K_\alpha(C_{\text{m}} + \tilde{C}_{\text{m}})K_\alpha^T + (I - K_\alpha A)C_{\text{o}}(I - K_\alpha A)^T,$$

where $\tilde{C}_{\text{m}}$ is the ensemble approximation of the model covariance matrix and

$$K_\alpha := C_{\text{o}}G'^T(\boldsymbol{y})\left[G'(\boldsymbol{y})C_{\text{o}}G'^T(\boldsymbol{y}) + \alpha C_{\text{m}}\right]^{-1}.$$

For the reference solution and $\alpha = 1$ the covariance matrix of the error becomes

$$C_{\text{err}} = (I - K_1 G) C_{\text{o}},$$

which is akin to that of the Kalman filter. The detailed derivation of accuracy can be found in Appendix 5.B.

5.3.3 Consistency

In order to show consistency, we need to assume full observations and that α is identical for all ensemble members and the reference. For nonlinear models, the ensemble members are unbiased with respect to the reference after one iteration. For linear models, the ensemble members are unbiased with respect to the reference after any number of iterations. This gives us first order consistency.

The ensemble shadowing is, however, second order inconsistent with respect to the Kalman filter. We denote by $\tilde{C}_{\text{m}}$ the ensemble approximation of the model error covariance matrix. Then for linear models and α chosen a priori, the covariance matrix of the ensemble after one iteration is

$$C_{\text{ens}} = K_\alpha \tilde{C}_{\text{m}} K_\alpha^T.$$

We note that if we initialize the ensemble shadowing with perturbed observations, then the covariance matrix of the ensemble after one iteration is consistent with respect to the Kalman filter. We denote by $\tilde{C}_{\text{o}}$ the ensemble approximation of the observation error covariance matrix, then

$$C_{\text{ens}} = K_\alpha \tilde{C}_{\text{m}} K_\alpha^T + (I - K_\alpha G)\, \tilde{C}_{\text{o}}\, (I - K_\alpha G)^T .$$

Numerical experiments with a *nonlinear* model showed, however, an extensive increase in ensemble spread throughout the iteration when perturbed observations were used to initialize the ensemble shadowing. The detailed derivation of consistency can be found in Appendix 5.C.

5.4 Numerical experiments

We perform some numerical experiments using shadowing for imperfect models. For the experimental window, we choose a length of 10 time units. To obtain a discrete model, we discretize using an Euler discretization with time step $\tau = 0.005$, which means the number of discrete steps $N = 2000$. In the perfect model scenario, this is a time length at which using strong constraint 4DVar may be problematic. For perfect model shadowing the length of the experimental window is not a limitation in and of itself.

In the numerical experiments, we shall compare the shadowing reference trajectory with a weak 4DVar solution. For ensemble comparisons, we shall

compare the above described shadowing ensembles with a randomized maximum likelihood (RML) method. The RML method consists of performing weak 4DVar optimizations on perturbed observations in order to obtain an ensemble. To stress the relation between weak 4DVar and RML, we shall call the RML method "ew4DV". We recall that, given covariance matrices C_{o} and C_{m}, ensemble w4DVar members minimize the cost function

$$J_{ew4DV}(\boldsymbol{u}) := \frac{1}{2}\|G(\boldsymbol{u}) - \eta^{ens}\|^2_{C_{\mathrm{m}}} + \frac{1}{2}\|H(\boldsymbol{u}) - \boldsymbol{y} - \xi^{ens}\|^2_{C_{\mathrm{o}}}, \tag{5.15}$$

where η^{ens} and ξ^{ens} are samples from Gaussian distributions with mean zero and covariance matrices C_{m} and C_{o} respectively.

We define the mismatch as

$$J_{\mathrm{o}}(\boldsymbol{u}) := \frac{1}{2}\|H(\boldsymbol{u}) - \boldsymbol{y}\|^2_{C_{\mathrm{o}}}. \tag{5.16}$$

An approach for studying structurally imperfect models consists of dropping out either the ocean or atmosphere part from the model. Then using only an imperfect model and observations of either the atmosphere or the ocean, one may attempt to reconstruct both. That is, given a solution from a data assimilation algorithm we may expect the observational discrepancy to be distributed according to the observational noise distribution and the model mismatch to reflect information about the excluded model variables.

An even simpler setting in which to study imperfect models is the case of a stochastically forced linear model. That is, we let the truth be of the form (5.1), where F_n is the linear ocean model and $\hat{\eta}_n$ is Gaussian noise sampled from a normal distribution with mean zero and covariance matrix

$$C_m = (\tau^2\alpha_c^2)\begin{pmatrix} 63.7917 & 64.7325 \\ 64.7325 & 84.3874 \end{pmatrix}, \tag{5.17}$$

so the dynamics is given by

$$\begin{pmatrix} x_4 \\ x_5 \end{pmatrix}_{n+1} = \begin{pmatrix} x_4 \\ x_5 \end{pmatrix}_{n} + \tau \begin{pmatrix} -k & -\Omega \\ \omega & -k \end{pmatrix} \begin{pmatrix} x_4 - w_* \\ x_5 \end{pmatrix}_{n} + \hat{\eta}_n. \tag{5.18}$$

We remark that the covariance matrix of (5.17) is obtained as the covariance matrix of the forcing due to the atmosphere model variables on the ocean model variables for a long-time integration of the coupled atmosphere-ocean model (5.2)-(5.6).

For this linear and Gaussian case, it is well known that the Bayesian data assimilation problem is exactly solved by a Kalman smoother and that the ew4DV method samples from the posterior distribution of the Kalman smoother. This means we can not expect the shadowing method to be a more

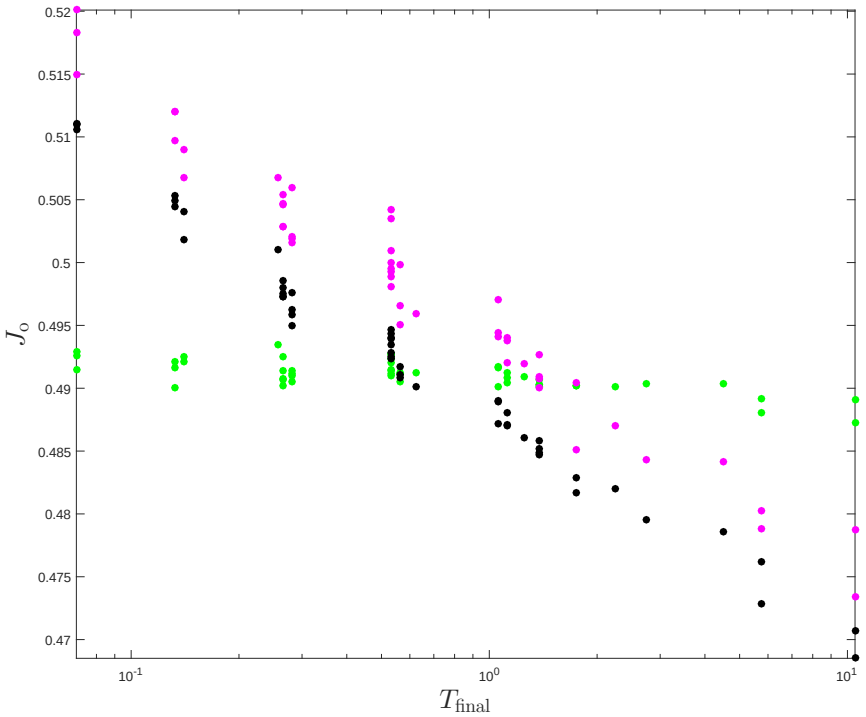

Figure 1: Observational cost function J_o values for 50 experiments of a linear and Gaussian model (5.18). Results for shadowing are in green, w4DVar is in black and the truth is in magenta.

accurate sample of the exact posterior distribution than the ew4DV method.

To gain some insight in the differences between the reference solution for shadowing and the w4DVar result, we ran 50 data assimilation experiments of the stochastically forced ocean model (5.18). In figure 5.4 we plot the value of the observational cost function J_o against the final shadowing time for the shadowing solutions. For reference, we also plot $J_o(\mathcal{X})$ and w4DVar solutions for identical observations at the corresponding shadowing times. We remark that for w4DVar solutions the corresponding final time is always 1 (reached in one step) and no algorithmic final time exists for the truth, so the plot is just for easy comparison between identical experiments.

It can be seen that, as expected, there is some statistical variation in the value of J_o for the truth. The value for J_o obtained through w4DVar is always slightly less then the true value. This is also known from literature [52, 19]. Because of the stopping criterion, the shadowing algorithm consistently obtains a value $J_o \approx 0.49$. This then leads to a wide range of values for the

Property	Shadowing	W4DVar	Truth
T_{final}	0.0087 ± 0.0007	1	N/A
J_{o}	0.493 ± 0.002	11 ± 1	0.50 ± 0.01
J_{t}	0.11 ± 0.01	10 ± 1	0

Table 1: Results for carrying out 50 numerical experiments when the imperfect model (5.18) is incorrectly modelling the atmosphere, where the atmosphere consists of the first three variables of (5.2)-(5.6)

.

Property	Shadowing	W4DVar	Truth
T_{final}	0.0011 ± 0.0005	1	N/A
J_{o}	0.492 ± 0.001	5.9 ± 0.9	0.50 ± 0.01
J_{t}	0.025 ± 0.005	5.4 ± 0.9	0

Table 2: Same as table 1, but with observational noise parameter $\epsilon = 1$.

final time T_{final}.

In this simple test case the optimal solution is known to be the w4DVar solution. Indeed, the distance between the w4DVar solution and the truth is consistently smaller than the distance between shadowing solution and truth. We have also tested integration up to $T = T_{\text{final}}$ or up to $T = 1$ using small shadowing steps. These methods perform comparably but slightly worse than shadowing and w4DVar respectively. The best shadowing results are found if $T_{\text{final}} \approx 1$. This is the case if the value of J_{o} for the truth is close to, but slightly smaller than, the theoretical expectation of 0.5. Then the first shadowing step would be up to time 1 in one step. This is identical to doing w4DVar. Minor further corrections to shadowing do not worsen the result considerably. As we shall see below, for more difficult test cases deriving the optimal final time is less trivial, but shadowing results for this final time are more consistent.

For our next test, we let the truth be the full model (5.2)-(5.6). As above, the data assimilation is done with the linear ocean model (5.18) and the covariance matrix is given by (5.17). This is an imperfect model, but the choice of stochastic forcing (5.17) is optimal. We take full observations and set the observational noise parameter $\epsilon = 0.1$. The experiments are repeated for $\epsilon = 1$ and $\epsilon = 0.01$, yielding similar results. We then compute the reference solution for shadowing and the w4DVar solution for 50 different realizations of truth and observations. In table 1, we show the results of those numerical experiments. We denote by

$$J_{\text{t}}(x) := 0.5||x - \mathcal{X}||^2_{C_{\text{o}}}, \tag{5.19}$$

a normalized distance to the truth.

It is clear from tables 1, 2 that for these numerical experiments, shadowing

Property	Shadowing	W4DVar	Truth
T_{final}	0.103 ± 0.008	1	N/A
J_{o}	0.494 ± 0.001	9 ± 1	0.50 ± 0.01
J_{t}	0.29 ± 0.02	9 ± 1	0

Table 3: Same as table 1, but then with observational noise parameter $\epsilon = 0.01$.

results are much more consistent with observations and closer to the truth. In fact, the data assimilation result from w4DVar tends to be much further from the truth then the observations are to begin with. We also observe that the final time from shadowing is much less than 1 and the variance in final times is small.

We shall now focus on ensemble results for this same ocean model (5.18). It is clear from table 1, that for the observed variables, assimilation with shadowing yields results closer to the truth than observations, while w4DVar yields solutions that are significantly further from the truth. For the ensemble results, we compare a true trajectory of the unobserved x_1 coordinate with a normalized mismatch.

In particular, it is expected to hold for some analysis trajectory u_a of the ocean model (5.18), that $G_n(u_a) \approx -\left(x_2\ x_1\right)_n / \tau$. In figures 2 and 3, it can be seen that the shadowing reference trajectory more closely approximates the truth. For w4DVar, the peaks in the true trajectory tend to be damped out. This may be a consequence of the mismatch term being part of the w4DVar cost function. The ew4DV ensemble members are, as expected, clustered around the w4DVar trajectory, but the error tends to be underestimated. For the shadowing ensemble, the spread is wide enough but the error is overestimated. When the observational error is larger, $\epsilon = 1$, there does not seem to be any relation anymore between the ew4DV mismatch result and the true unmodeled state, as can be seen in Figure 4. These results are different from the theoretical results with only one step.

In the figures 2 and 4, we show results for 20 shadowing ensemble members and the reference shadowing result. To further clarify the properties of using an ensemble, in table 4, for ensemble sizes 5, 10, 20 and 40 and for the different model variables, we show mean ensemble spreads and mean rms distances between ensemble mean and reference solutions. We remark that the shadowing method is a non-linear estimation method, and therefore we prefer to use the reference solution as the best estimate of the truth.

We may contrast the results of table 4 with the results that arise from applying shadowing initialized with perturbed observations. The spread and distance between reference and mean for shadowing ensembles with perturbed observations is shown in table 5. It can be seen that the spread is severely over-estimated, especially for the excluded model variables.

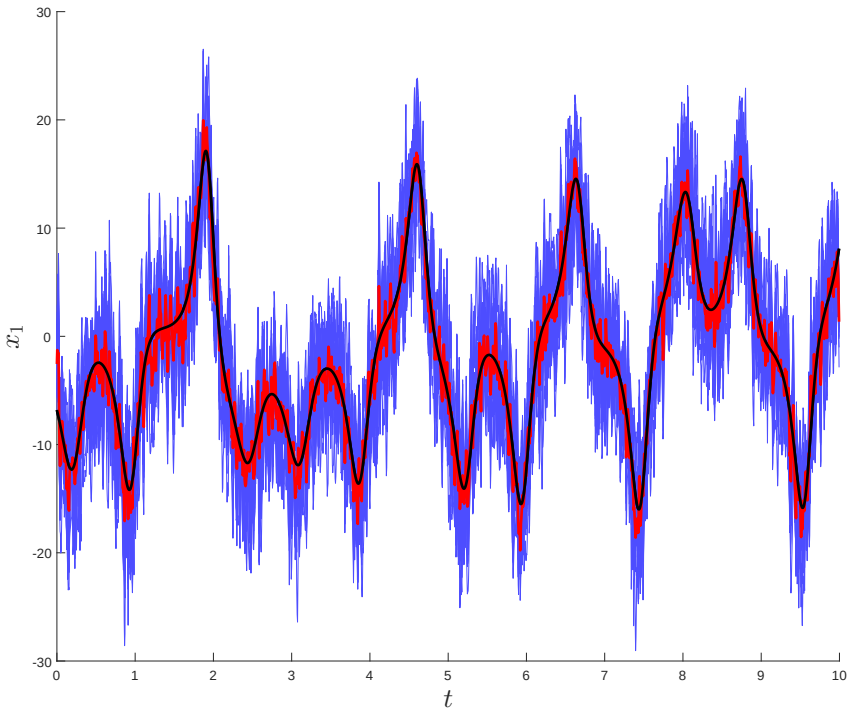

Figure 2: An estimate of an excluded model variable using shadowing. The truth is in black and the reference shadowing result is in red. Observational noise $\epsilon = 0.1$. The thin blue lines are 20 shadowing ensemble members. The true model is given by (5.2)-(5.6) and the imperfect model is given by (5.18).

Ens. size	x_1	x_2	x_4	x_5
5	0.029 (0.020)	0.032 (0.018)	6.4 (3.1)	7.4 (3.5)
10	0.031 (0.017)	0.033 (0.015)	6.7 (2.2)	7.6 (2.5)
20	0.031 (0.016)	0.034 (0.013)	6.7 (1.6)	7.7 (1.8)
40	0.031 (0.015)	0.034 (0.011)	6.8 (1.1)	7.8 (1.3)

Table 4: Spread and (RMS distance) between ensemble mean and reference, for different model variables and different shadowing ensemble sizes. The true model is given by (5.2)-(5.6) and the imperfect model is given by (5.18). The entries first show spread and RMS distance in brackets behind it.

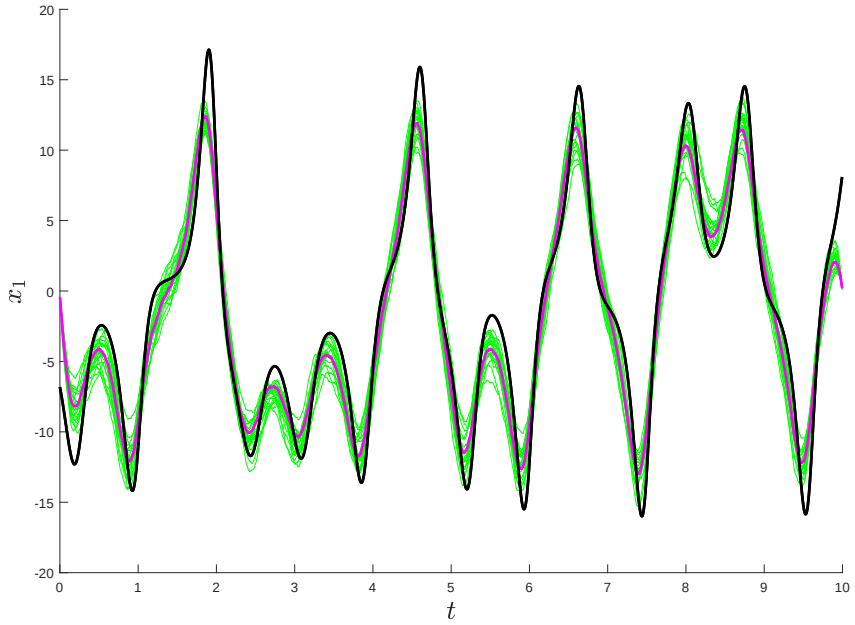

Figure 3: An estimate of an unmodelled trajectory component using ew4DV. The truth is in black and the w4DVar solution in magenta. Observational noise $\epsilon = 0.1$. The thin green lines are 20 ew4DV ensemble members. The true model is given by (5.2)-(5.6) and the imperfect model is given by (5.18).

Ens. size	x_1	x_2	x_4	x_5
5.	0.094 (0.11))	0.096 (0.11)	26 (30)	27 (30)
10	0.096 (0.10)	0.099 (0.099)	27 (28)	28 (28)
20	0.098 (0.10)	0.099 (0.097)	28 (26)	28 (27)
40	0.098 (0.097)	0.099 (0.094)	28 (25)	28 (26)

Table 5: Spread and RMS distance between ensemble mean and reference, for different solution components and different shadowing ensemble sizes. The true model is given by (5.2)-(5.6) and the imperfect model is given by (5.18). Ensembles are obtained using ensemble members with perturbed observations.

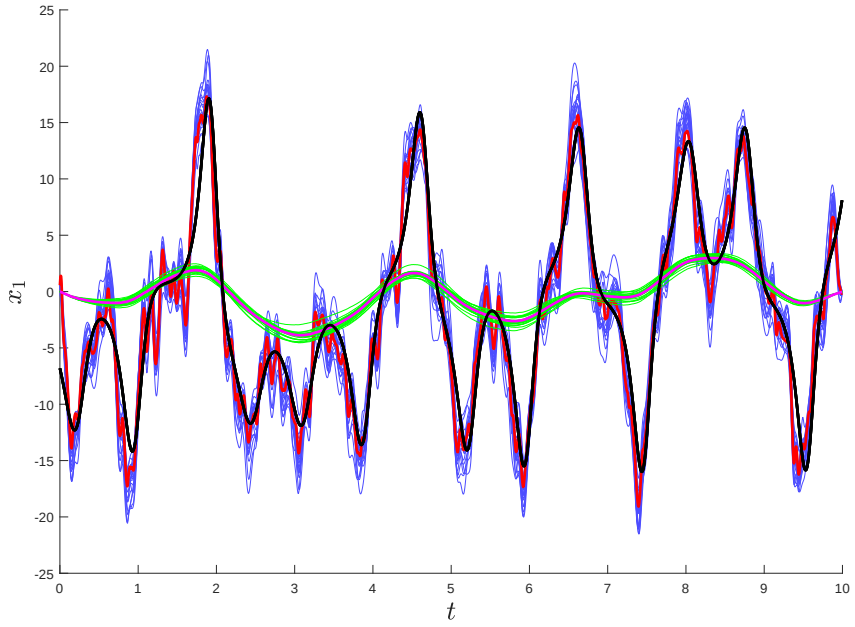

Figure 4: Same as figures 2 and 3 plotted together in one plot, but with observational noise $\epsilon = 1$.

An even harder test case arises when the truth follows the Molteni model (5.2)-(5.6), while the assimilation is done using the L63 model with stochastic forcing. This means that the atmosphere is modeled, but the ocean is not. The ocean can then be modeled with the covariance matrix

$$C_{\mathrm{m}} = \tau^2 \alpha_c^2 \begin{pmatrix} 70.7957 & -5.1121 & 0 \\ -5.1121 & 58.3686 & 0 \\ 0 & 0 & 0 \end{pmatrix}.$$

Or, to explicitly repeat the equations of this imperfect atmosphere model,

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}_{n+1} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix}_n + \tau \begin{pmatrix} -\sigma & \sigma & 0 \\ \rho - x_3 & -1 & 0 \\ x_2 & 0 & -\beta \end{pmatrix}_n \begin{pmatrix} \sigma(x_1 - x_2) \\ x_1(\rho - x_3) - x_2 \\ x_1 x_2 - \beta x_3 \end{pmatrix}_n + \hat{\eta}_n, \tag{5.20}$$

where $\hat{\eta}_n$ is normally distributed with mean zero and covariance matrix C_{m}. We conduct a numerical experiment where the first two components of the atmosphere are observed with observation standard deviation $\epsilon = 0.1$. This means that of the truth, the first two components are observed, the third is correctly modeled but not observed and the last two components are neither modeled nor observed, but instead described using Gaussian noise.

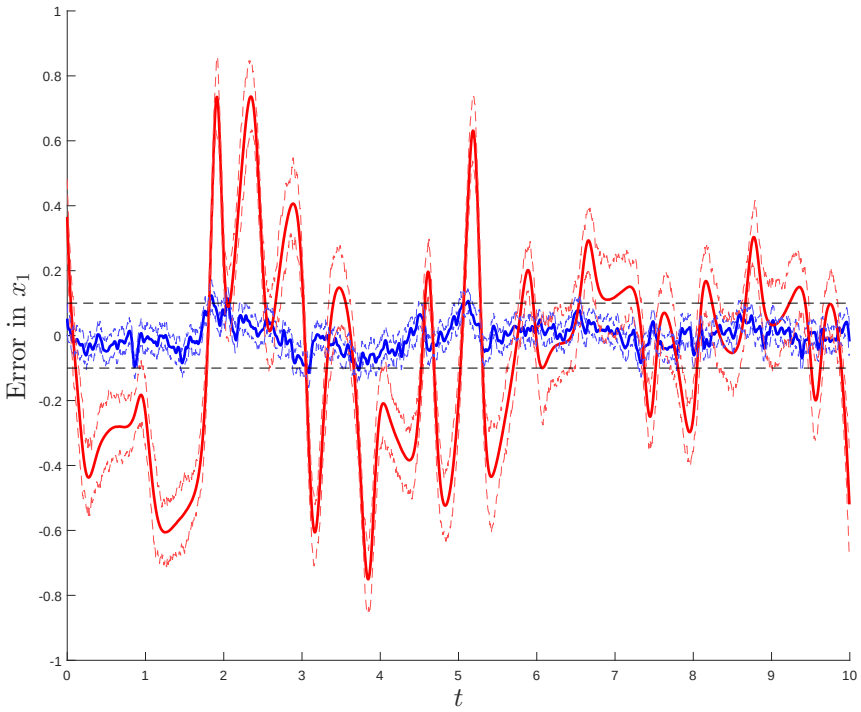

Figure 5: Error over time of an observed trajectory component using both shadowing and ew4DV. The reference shadowing result is in blue, and the w4DVar solution in red. Dotted lines are used to denote one standard deviation $\epsilon = 0.1$, the black dotted line denotes the observational standard deviation. The true model is given by (5.2)-(5.6) and the wrong model is given by (5.20)

For an example run, we compare ensemble shadowing with ew4DV. Results are shown in figures 5, 7, and 9. For the observed coordinate x_1 and for the unobserved, but modeled, coordinate x_3 the errors are small relative to the signal. Therefore, we plot the distance between truth and assimilation over time. The ensemble is indicated by plotting dotted lines at one standard deviation from the reference solution. We use blue for the shadowing results and red for the ew4DV results. In the figure for the observed x_1, the observational standard deviation is shown as a black-dotted line. For the non-modeled x_5-component, the errors are larger and we can show the trajectories (with standard deviations) as well as the truth. We again use blue for the shadowing results and red for the ew4DV results. The truth is in black.

It can be seen in figures 5, 6, 7, 8, 9 and 10, that shadowing results in solutions closer to the truth. For the estimation of the observed variable x_1,

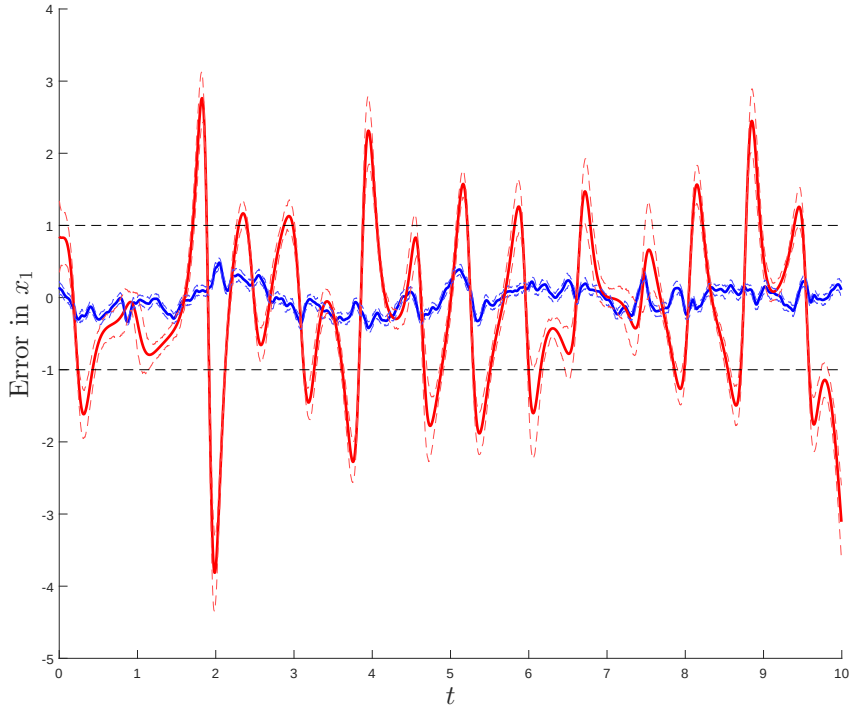

Figure 6: Same as figure 5, but then with observational noise $\epsilon = 1$. Please note the scale of the y-axis is different from figure 5.

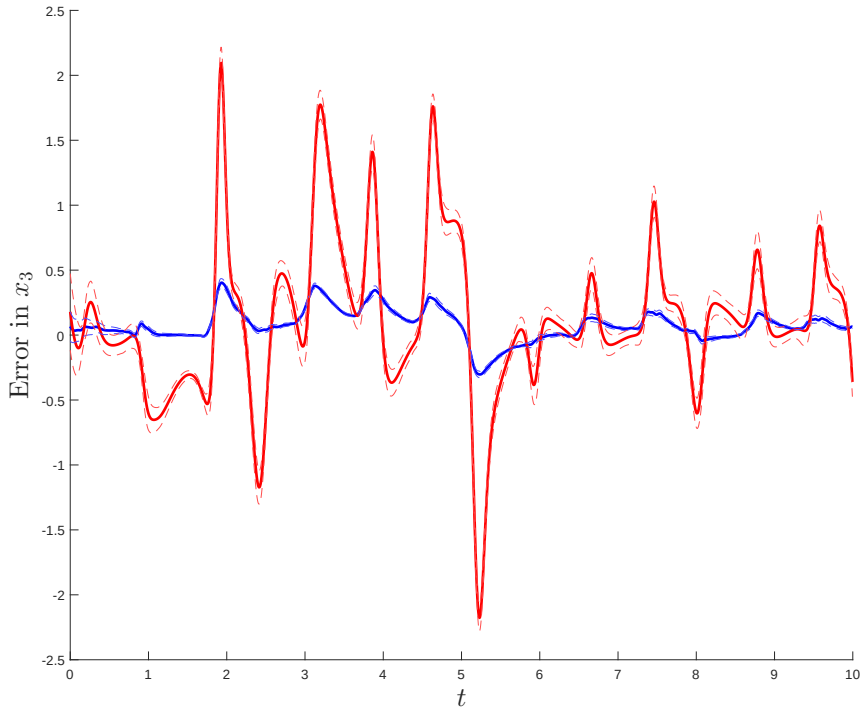

Figure 7: Error over time of an unobserved, but modeled, trajectory component using both shadowing and ew4DV. The reference shadowing result is in blue, and the w4DVar solution in red. Dotted lines are used to denote one standard deviation. The true model is given by (5.2)-(5.6) and the wrong model is given by (5.20)

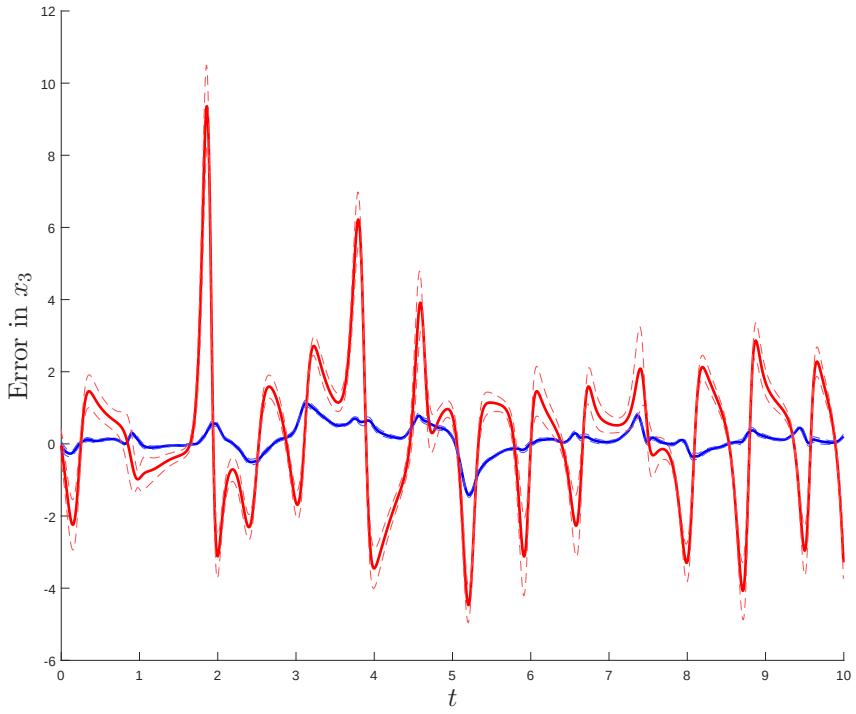

Figure 8: Same as figure 7, but with observational noise $\epsilon = 1$. Please note the scale of the y-axis is different from figure 7.

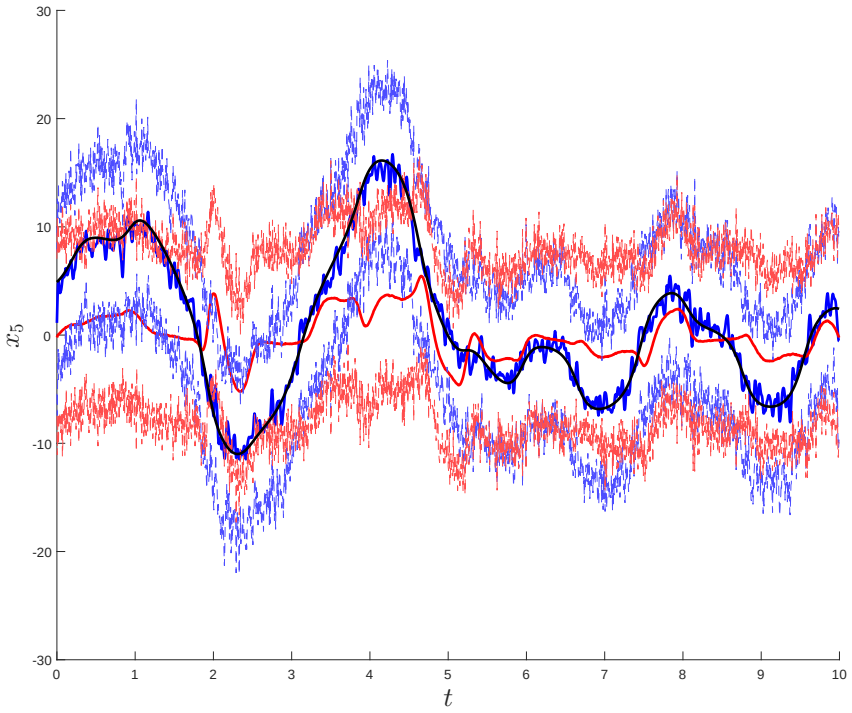

Figure 9: An estimate of an unobserved and unmodeled trajectory component using both shadowing and ew4DV. The truth is in black, the reference shadowing result is in blue, and the w4DVar solution in red. Dotted lines are used to denote one standard deviation $\epsilon = 0.1$. The true model is given by (5.2)-(5.6) and the wrong model is given by (5.20)

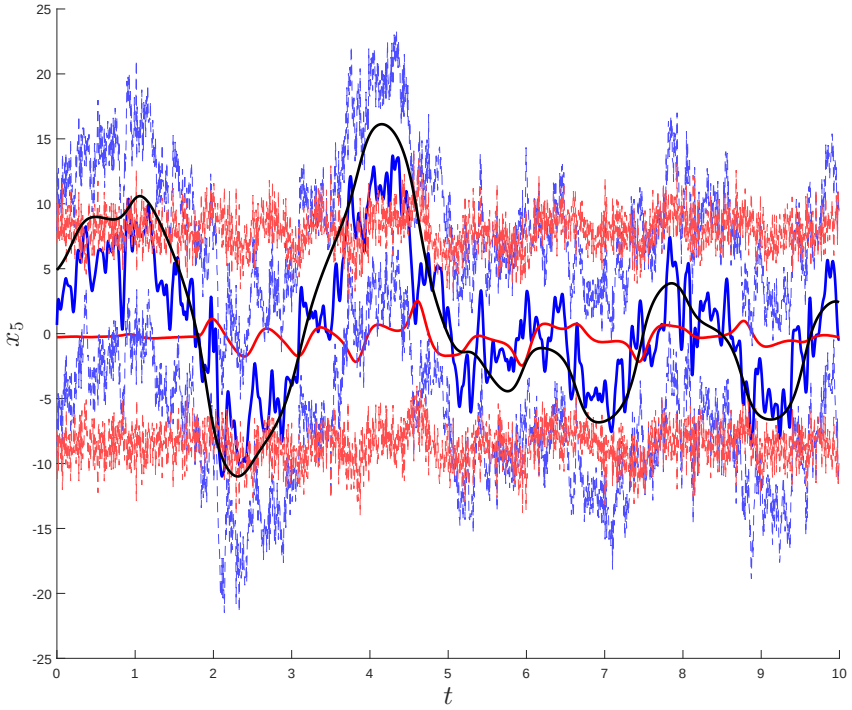

Figure 10: Same as figure 9, but with observational noise $\epsilon = 1$.Please note the scale of the y-axis is different from figure 9.

Ens.	x_1	x_2	x_3	x_4	x_5
5	0.034 (0.020)	0.033 (0.018)	0.017 (0.031)	6.9 (3.5)	6.4 (3.2)
10	0.036 (0.016)	0.034 (0.015)	0.020 (0.032)	7.2 (2.6)	6.6 (2.3)
20	0.036 (0.014)	0.034 (0.013)	0.021 (0.032)	7.3 (1.9)	6.7 (1.8)
40	0.037 (0.013)	0.035 (0.012)	0.020 (0.032)	7.4 (1.5)	6.8 (1.4)

Table 6: Spread and RMS distance between ensemble mean and reference, for different solution components and different shadowing ensemble sizes. The true model is given by (5.2)-(5.6) and the wrong model is given by (5.20). The entries first show spread and RMS distance in brackets behind it.

Ens.	x_1	x_2	x_3	x_4	x_5
5	0.95 (8.1)	0.94 (9.1)	0.92 (26)	263 (122)	269 (127)
10	0.98 (8.1)	0.97 (9.1)	0.97 (26)	269 (86)	276 (89)
20	0.99 (8.1)	0.98 (9.1)	0.98 (26)	273 (61)	281 (63)
40	1.0 (8.1)	1.0 (9.1)	1.0 (26)	275 (43)	282 (45)

Table 7: Spread and RMS distance between ensemble mean and reference, for different solution components and different shadowing ensemble sizes. The true model is given by (5.2)-(5.6) and the wrong model is given by (5.20). Ensembles are obtained using ensemble members with perturbed observations. The entries first show spread and RMS distance in brackets behind it.

both shadowing and ew4DV tend to underestimate the analysis error, but the shadowing analysis is much closer to the truth. This effect is even stronger for the unobserved x_3 component: the error for ew4DV is larger than the shadowing error and both methods underestimate the analysis error. For the unmodeled x_5-component, the estimation problem is much harder and hence the error with respect to the truth is larger. Here, the shadowing ensemble overestimates the analysis error (as in figure 2). The ew4DV ensemble correctly indicates that the ew4DV analysis suffers from large errors.

To further clarify the properties of using an ensemble, in table 6, for ensemble sizes 5, 10, 20 and 40 and for the different solution components, we show mean ensemble spreads and mean rms distances between ensemble mean and reference solutions. We remark that the shadowing method is a non-linear estimation method, and therefore we prefer to use the reference solution as the best estimate of the truth. We may contrast the results of table 6, with the results that arise from applying shadowing initialized with perturbed observations. The spread and distance between reference and mean for shadowing ensembles with perturbed observations is shown in table 7. It can be seen that the spread is severely over-estimated, especially for the wrongly modeled components.

5.A Stopping algorithmic time

We assume full observations and as linear model such that $G(u) = Au + b$. Then (5.14) becomes

$$\frac{du}{d\tau} = -C_{\mathrm{o}}A^T C_{\mathrm{m}}^{-1}(Au + b - \eta), \quad u(0) = \mathcal{X} + \xi. \tag{5.21}$$

Solving (5.21) and substituting $-\hat{\eta} = A\mathcal{X} + b$ yields

$$Au(\tau) = -(b - \eta) + e^{-\tau AC_{\mathrm{o}}A^T C_{\mathrm{m}}^{-1}}(A\xi - \hat{\eta} - \eta).$$

We note that at initial algorithmic time, $Au(0) + b = A\xi - \hat{\eta}$, which depends on the observation and the true model noise, and at infinite algorithmic time, $Au(\infty) + b = \eta$, which is the imposed model noise. The difference between the solution and the observations is given by

$$u(\tau) - y = A^\dagger \left(e^{-\tau AC_{\mathrm{o}}A^T C_{\mathrm{m}}^{-1}} - I\right)(A\xi - \hat{\eta} - \eta), \quad \text{where} \quad A^\dagger = (A^T A)^{-1}A^T.$$

The stopping algorithmic time $\mathcal{T}$ is defined by $\|u(\mathcal{T}) - y\|^2_{C_{\mathrm{o}}} = r^2 Nm$. Since $A = A(x)$, $C_{\mathrm{o}} = C_{\mathrm{o}}(x)$, $C_{\mathrm{m}} = C_{\mathrm{m}}(x)$, and ξ, $\hat{\eta}$, and η are uncorrelated, we have

$$\begin{aligned}\|u(\mathcal{T}) - y\|^2_{C_{\mathrm{o}}} = &\sum_x L_1(x,\mathcal{T}) \sum_{n=0}^{N} \xi^2(x,t_n) + \\ &\sum_x L_2(x,\mathcal{T}) \left[\sum_{n=0}^{N} \hat{\eta}^2(x,t_n) + \sum_{n=0}^{N} \eta^2(x,t_n)\right],\end{aligned}$$

where L_1 and L_2 depend on A, C_{o}, C_{m}, and algorithmic time. We note that

$$\frac{1}{N}\sum_{n=0}^{N} \xi^2(x,t_n) \approx C_{\mathrm{o}} \quad \text{and} \quad \frac{1}{N}\sum_{n=0}^{N} \hat{\eta}^2(x,t_n) \approx C_{\mathrm{m}},$$

which are the same for every ensemble member. While $\sum_{n=0}^{N} \eta^2(x,t_n)$ is different for each ensemble member, unless $N \to \infty$. This means that the stopping algorithmic time $\mathcal{T}$ is different for each ensemble member including the reference, for which $\sum_{n=0}^{N} \eta^2(x,t_n) = 0$.

5.B Accuracy of the ensemble shadowing data assimilation

We assume full observations in space. We focus on the case where $\hat{\eta}_n$ has known mean and covariance. Without essential loss of generality, we may assume that the known mean is zero. If the known mean would be non-zero, it is possible to improve the model by adding this known mean forcing as a constant term.

Theorem 4 *Assume the observation error is unbiased, then the analysis is unbiased with respect to the true solution for linear models if the number of observations $N \to \infty$.*

Proof 4 *For linear models $G(x) = Ax + b$ the gain* (5.3.2) *simplifies to*

$$K_\alpha := C_o A^T \left(AC_o A^T + \alpha C_m\right)^{-1}$$

and it is constant.

The difference between the truth $\boldsymbol{\mathcal{X}}$ and the analysis at the first iteration $\boldsymbol{u}^{(1)}$ is

$$\begin{aligned}\boldsymbol{\mathcal{X}} - \boldsymbol{u}^{(1)} &= \boldsymbol{\mathcal{X}} - \boldsymbol{\mathcal{X}} - \boldsymbol{\xi} + K_\alpha(A\boldsymbol{\mathcal{X}} + \boldsymbol{b} + A\boldsymbol{\xi} - \boldsymbol{\eta}) \\ &= -\boldsymbol{\xi} + K_\alpha(A\boldsymbol{\mathcal{X}} + \boldsymbol{b} + \hat{\boldsymbol{\eta}} - \hat{\boldsymbol{\eta}} + A\boldsymbol{\xi} - \boldsymbol{\eta}) \\ &= (K_\alpha A - I)\boldsymbol{\xi} - K_\alpha(\hat{\boldsymbol{\eta}} + \boldsymbol{\eta}). \end{aligned} \tag{5.22}$$

Then the expectation between the truth $\boldsymbol{\mathcal{X}}$ and the analysis at the first iteration $\boldsymbol{u}^{(1)}$ is zero. The expectation between the truth $\boldsymbol{\mathcal{X}}$ and the analysis at the second iteration $\boldsymbol{u}^{(2)}$ is

$$\begin{aligned}\mathbb{E}[\boldsymbol{\mathcal{X}} - \boldsymbol{u}^{(2)}] &= \mathbb{E}[\boldsymbol{\mathcal{X}} - \boldsymbol{u}^{(1)} + K_\alpha(A\boldsymbol{u}^{(1)} + \boldsymbol{b} + \hat{\boldsymbol{\eta}} - \hat{\boldsymbol{\eta}} - \boldsymbol{\eta})] \\ &= \mathbb{E}[\boldsymbol{\mathcal{X}} - \boldsymbol{u}^{(1)}] + K_\alpha A\mathbb{E}[\boldsymbol{u}^{(1)} - \boldsymbol{\mathcal{X}} + \boldsymbol{\mathcal{X}}] + K_\alpha\mathbb{E}[\boldsymbol{b} + \hat{\boldsymbol{\eta}} - \hat{\boldsymbol{\eta}} - \boldsymbol{\eta}] \\ &= K_\alpha A\mathbb{E}[\boldsymbol{u}^{(1)} - \boldsymbol{\mathcal{X}}] + K_\alpha\mathbb{E}[A\boldsymbol{\mathcal{X}} + \boldsymbol{b} + \hat{\boldsymbol{\eta}} - \hat{\boldsymbol{\eta}} - \boldsymbol{\eta}] \\ &= -K_\alpha\mathbb{E}[\hat{\boldsymbol{\eta}}] - K_\alpha\mathbb{E}[\boldsymbol{\eta}] = 0.\end{aligned}$$

For more iterations, the same argument holds.

Corollary 4.1 *We assume the model and observation error are uncorrelated and α is chosen a-priori. Then for a linear model $G(x) = Ax+b$ the covariance matrix of the error between the analysis and the true solution is*

$$C_{err} = K_\alpha(C_m + \tilde{C}_m)K_\alpha^T + (I - K_\alpha A)C_o(I - K_\alpha A)^T,$$

where $\tilde{C}_m$ is the ensemble approximation of the model covariance matrix. For the reference solution and $\alpha = 1$, this reduces to $C_{err} = (I - K_1 G)C_o$.

Proof 5 *The proof trivially follows from* (5.22).

5.C Consistency of the ensemble shadowing data assimilation

We assume full observations in space. We focus on the case where $\hat{\eta}_n$ has known mean and covariance. Without essential loss of generality, we may assume that the known mean is zero. If the known mean would be non-zero, it is possible to improve the model by adding this known mean forcing as a constant term.

Theorem 5 *We assume α is fixed for all ensemble members and the reference. Then the ensemble members are unbiased with respect to the reference for nonlinear models provided only one iteration is taken, and for linear models independent of number of iterations.*

Proof 6 *Since K_α and $\boldsymbol{\eta}$ are independent random variables, it trivially follows that*

$$\mathbb{E}[\boldsymbol{u}^{(1)} - \boldsymbol{u}^{0(1)}] = K_\alpha \mathbb{E}[\boldsymbol{\eta}] = 0. \tag{5.23}$$

If $G(x) = Ax + b$, then

$$\mathbb{E}[\boldsymbol{u}^{(2)} - \boldsymbol{u}^{0(1)}] = (I - K_\alpha A)\mathbb{E}[\boldsymbol{u}^{(1)} - \boldsymbol{u}^{0(1)}] - K_\alpha \mathbb{E}[\boldsymbol{\eta}] = 0. \tag{5.24}$$

For more iterations, the same argument holds.

Corollary 5.1 *We assume the model error is unbiased, α is fixed for all ensemble members, and the model G is linear. We denote by $\tilde{C}_m$ the ensemble approximation of the model covariance matrix. Then the ensemble covariance matrix after one iteration is*

$$C_{ens} = K_\alpha \tilde{C}_m K_\alpha^T,$$

which is identical for all iterations if $\alpha = 0$ and $C_o = \gamma I$.

Proof 7 *For one iteration, the proof trivially follows from* (5.23) *by substituting $\boldsymbol{u}^{0(1)}$ by $\mathbb{E}[\boldsymbol{u}^{(1)}]$. For the following iterations, the proof follows from remarking that for $G(x) = Ax + b$, $\alpha = 0$ and $C_o = \gamma I$, we have $K_0 A K_0 = K_0$.*

Corollary 5.2 *We assume the model error is unbiased, α is fixed for all ensemble members, and the model G is linear. Moreover we assume the initial guess consists of perturbed observations $\boldsymbol{y} + \boldsymbol{\xi}^j$, for $j = 1, \ldots, J$. We denote by $\tilde{C}_o$ the ensemble approximation of the observation covariance matrix. Then the ensemble covariance matrix after one iteration is*

$$C_{ens} = K_\alpha \tilde{C}_m K_\alpha^T + (I - K_\alpha G)\, \tilde{C}_o\, (I - K_\alpha G)^T,$$

which is identical for all iterations if $\alpha = 0$ and $C_0 = \gamma I$.

Proof 8 *The proof trivially follows from the proof of 5.1.*

Acknowledgments

We thank Dr. Amos S. Lawless for acquainting us with the coupled ocean-atmosphere model.

CHAPTER 6

Conclusion

In this thesis, various data assimilation algorithms have been developed. These algorithms are based upon shadowing.

The algorithms of chapter 2 are based upon a computational time dependent stable/unstable splitting. This algorithm is motivated by Assimilation in the Unstable Subspace (AUS) [12, 94, 78] and Pseudo-orbit Data Assimilation (PDA) [55, 54, 32]. The algorithm utilizes time dependent projections onto the non-stable subspace determined by employing computational techniques for Lyapunov exponents/vectors. In the unstable subspace, a shadowing algorithm is used, while in the stable subspace, synchronization is used. The method is extended to parameter estimation without changing the problem dynamics and we address techniques for adapting the method when (as is commonly the case) observations are not available in the full model state space. However, in the method of chapter 2, the assumption is that the model is perfect and the problem of observations not being available in the full model space is partly, but not fully addressed.

In chapter 3, a data assimilation problem for imperfect models was addressed. The model error was taken into account following a Levenberg-Marquardt regularization approach. It was also discussed how the proposed shadowing-based method is related to the weak constraint 4DVar method both analytically and numerically and demonstrated that the shadowing-based method respects the distribution of the data mismatch, while the weak constraint 4DVar does not, which becomes even more

pronounced with fewer observations. Moreover, sparse observations give weaker influence on unobserved variables for the shadowing-based method than for the weak constraint 4DVar.

In chapter 4, the shadowing refinement technique of chapter 2 was further developed to take partial observations into account. The method is based on a regularized Gauss-Newton method, similar to the one used in chapter 3. Local convergence to the solution manifold was proven and a lower bound on the algorithmic time step was provide. Numerical experiments with the Lorenz 63 and Lorenz 96 models illustrated convergence of the algorithm and show that the results compare favourably with a variational technique—weak-constraint four-dimensional variational method—and a shadowing technique–pseudo-orbit data assimilation. Numerical experiments show that a preconditioner chosen based on a cost function allows the algorithm to find an orbit of the dynamical system in the vicinity of the true solution.

In chapter 5, the algorithm of the previous chapters was developed yet further. The algorithm was extended to generate an ensemble of states, for estimating the uncertainties of the data assimilation algorithm using the concept of indistinguishable states. This chapter also includes some proofs regarding well-posedness, accuracy and consistency of the algorithm. The algorithm is applied to a non-perfect model to show how the unmodelled components of the model can be estimated using the data assimilation algorithm.

In all, throughout the thesis, it has been shown how data assimilation problems may be tackled by shadowing-based algorithms and that these algorithms can be adapted to many situations and compared to the related class of data assimilation algorithms known as 4D variatonal algorithms. Through these shadowing-based data assimilation methods, it is possible to take observations over relatively long time intervals into account, even though the underlying models are chaotic. Furthermore, the methods can deal with partial observations and imperfect models. It is possible to generalize the methods to ensemble methods, to obtain information on the accuracy of the state estimate and to provide a possibility of model improvement. Some future work could for example comprise combining the various methods, such as integrating the synchronization of chapter 2 into the algorithms developed in later chapters. Other possibilities include applying the method to larger-scale models, such as models that are used in weather forecasting, or at least "toy models" for weather forecasting. To make the methods applicable to larger scale models, an important challenge is to consider the computational cost and in particular the memory usage. These are currently comparable to w4dvar in state space formulation, which is however still to costly for large-scale models, especially over long time windows. So, this may be a possibility for future work.

Bibliography

[1] L. Ya. Adrianova. *Introduction to Linear Systems of Differential Equations*, volume 146 of *Translations of Mathematical Monographs*. American Mathematical Society, Providence, RI, USA, 1995. Translated from the Russian by Peter Zhevandrov.

[2] M. Badawy and E. Van Vleck. Perturbation theory for the approximation of stability spectra by QR methods for sequences of linear operators on a Hilbert space. *Linear Algebra Appl.*, 437(1):37–59, 2012.

[3] R. N. Bannister. A review of operational methods of variational and ensemble-variational data assimilation. *Quarterly Journal of the Royal Meteorological Society*, 143(703):607–633, 2017.

[4] A. Ben-Israel. A Newton-Raphson method for the solution of systems of equations. *Journal of Mathematical Analysis and Applications*, 15(2):243 – 252, 1966.

[5] G. Benettin, L. Galgani, A. Giorgilli, and J.-M. Strelcyn. Lyapunov exponents for smooth dynamical systems and for Hamiltonian systems: a method for computing all of them. *Meccanica*, 15:9–30, 1980.

[6] M.L. Berliner. Likelihood and Bayesian prediction for chaotic systems. *J. Am. Stat. Assoc.*, 86:938–952, 1991.

[7] W.J. Beyn. On the numerical approximation of phase portraits near stationary point. *SIAM Journal on Mathematical Analysis*, 24(5):1095–1113, 1987.

[8] S. Boccalettia, J. Kurths, G. Osipov, D.L. Valladares, and C.S. Zhou. The synchronization of chaotic systems. *Physics Reports*, 366:1–101, 2002.

[9] M. Brin and G. Stuck. *Introduction to Dynamical Systems*. Cambridge University Press, 2002.

[10] J. Bröcker. On variational data assimilation in continuous time. *Q.J.R. Meteorol. Soc.*, 136:1906–1919, 2010.

[11] J. Bröcker and U. Parlitz. Efficient noncausal noise reduction for deterministic time series. *Chaos*, 11(2):319–326, 2001.

[12] A. Carrassi, M. Ghil, A. Trevisan, and F. Uboldi. Data assimilation as a nonlinear dynamical systems problem: Stability and convergence of the prediction-assimilation system. *Chaos*, 18(023112):023112, 2008.

[13] A. Carrassi, A. Trevisan, L. Descamps, O. Talagrand, and F. Uboldi. Controlling instabilities along a 3DVar analysis cycle by assimilating in the unstable subspace: a comparison with the EnKF. *Nonlinear Process. Geophys.*, 15:503–521, 2008.

[14] X.-W. Chang. On the perturbation of the q-factor of the qr factorization. *Numer. Linear Algebra Appl.*, 19:607–619, 2012.

[15] Y. Chen and D.S. Oliver. Levenberg–marquardt forms of the iterative ensemble smoother for efficient history matching and uncertainty quantification. *Comput Geosci*, 17:689–703, 2013.

[16] S.-N. Chow and K. Palmer. A shadowing lemma with applications to semilinear parabolic equations. *SIAM Journal on Mathematical Analysis*, 20:547–557, 1989.

[17] S.-N. Chow and K. Palmer. On the numerical computation of orbits of dynamical systems: The higher dimensional case. *Journal of Complexity*, 8(4):398 – 423, 1992.

[18] M. Cullen. Analysis of cycled 4d-var with model error. *Quarterly Journal of the Royal Meteorological Society*, 139(675):1473–1480, 2013.

[19] B. de Leeuw and S. Dubinkina. Regularized shadowing-based data assimilation method for imperfect models and its comparison to the weak constraint 4DVar method. *arXiv e-prints*, page arXiv:1810.07064, Oct 2018.

[20] B. de Leeuw, S. Dubinkina, J. Frank, A. Steyer, X. Tu, and E. Van Vleck. Projected shadowing-based data assimilation. *SIAM Journal on Applied Dynamical Systems*, 17(4):2446–2477, 2018.

[21] E. de Sturler and M. Kilmer. A regularized gauss–newton trust region approach to imaging in diffuse optical tomography. *SIAM Journal on Scientific Computing*, 33(5):3057–3086, 2011.

[22] L. Dieci and C. Elia. The singular value decomposition to approximate spectra of dynamical systems. Theoretical aspects. *J. Differential Equations*, 230(2):502–531, 2006.

[23] L. Dieci, C. Elia, and E. Van Vleck. Exponential dichotomy on the real line: SVD and QR methods. *J. Differential Equations*, 248(2):287–308, 2010.

[24] L. Dieci, C. Elia, and E. Van Vleck. Detecting exponential dichotomy on the real line: SVD and QR algorithms. *BIT*, 51(3):555–579, 2011.

[25] L. Dieci, R.D. Russell, and E.S. Van Vleck. On the Computation of Lyapunov Exponents for Continuous Dynamical Systems. *SIAM J. Numer. Anal.*, 34:402–423, 1997.

[26] L. Dieci and E. Van Vleck. On the Error in Computing Lyapunov Exponents by QR Methods. *Numer. Math.*, 101:619–642, 2005.

[27] L. Dieci and E. Van Vleck. Perturbation theory for approximation of Lyapunov exponents by QR methods. *J. Dynam. Differential Equations*, 18(3):815–840, 2006.

[28] L. Dieci and E. Van Vleck. On the error in QR integration. *SIAM J. Numer. Anal.*, 46(3):1166–1189, 2008.

[29] L. Dieci and E Van Vleck. Lyapunov exponents: Computation. In B. Engquist, editor, *Encyclopedia of Applied and Computational Mathematics*. Springer-Verlag, 2015.

[30] J. Dieudonné. *Foundations of modern analysis*. Academic Press, 1960.

[31] H. Du. *Combining Statistical Methods with Dynamical Insight to Improve Nonlinear Estimation*. PhD thesis, London School of Economics and Political Science, 2009.

[32] H. Du and L. Smith. Pseudo-Orbit Data Assimilation. Part I: The Perfect Model Scenario. *Journal of the Atmospheric Sciences*, 71:469–482, 2014.

[33] H. Du and L. Smith. Pseudo-orbit data assimilation. Part II: Assimilation with imperfect models. *Journal of the Atmospheric Sciences*, 71(2):483–495, 2014.

[34] G. S. Duane, J. J. Tribbia, and J. B. Weiss. Synchronicity in predictive modelling: a new view of data assimilation. *Nonlinear Processes in Geophysics*, 13(6):601–612, 2006.

[35] M. Dubois and P. Yiou. Testing asynchronous coupling on simple "ocean-atmosphere" dynamic systems. *Climate Dynamics*, 15(1), 1999.

[36] A. El-Said. *Conditioning of the weak-constraint variational data assimilation problem for numerical weather prediction*. PhD thesis, University of Reading, 2015.

[37] S. Ershov and A. Potapov. On the concept of stationary Lyapunov basis. *Physica D: Nonlinear Phenomena*, 118(3–4):167 – 198, 1998.

[38] G. Evensen. *Data Assimilation: The Ensemble Kalman Filter*. Springer, 2006.

[39] M. Fisher, Y. Trémolet, H. Auvinen, D. Tan, and P. Poli. Weak-constraint and long-window 4d-var. *Techn. Rep*, 655, 2011.

[40] M. Fisher, Y. Trémolet, H. Auvinen, D. Tan, and P. Poli. Weak-constraint and long-window 4d-var. Technical report, ECMWF, 2011.

[41] G. Golub and C. Van Loan. *Matrix Computations*. The Johns Hopkins University Press, third edition, 1996.

[42] C. González-Tokman and B. Hunt. Ensemble data assimilation for hyperbolic systems. *Physica D: Nonlinear Phenomena*, 243(1):128–142, 2013.

[43] C. Grebogi, S. Hammel, J. Yorke, and T. Sauer. Shadowing of physical trajectories in chaotic dynamics: Containment and refinement. *Phys. Rev. Lett.*, 65:1527–1530, Sep 1990.

[44] S. Hammel, J. Yorke, and C. Grebogi. Do numerical orbits of chaotic dynamical processes represent true orbits? *Journal of Complexity*, 3(2):136 – 145, 1987.

[45] S. Hammel, J. Yorke, and C. Grebogi. Numerical orbits of chaotic processes represent true orbits. *Bulletin of the American Mathimatical Society*, 19(2):465–469, 1988.

[46] M. Hanke. A regularizing Levenberg-Marquardt scheme, with applications to inverse groundwater filtration problems. *Inverse problems*, 13(1):79, 1997.

[47] K. Hayden, E. Olson, and E. Titi. Discrete data assimilation in the Lorenz and 2D Navier–Stokes equations. *Physica D: Nonlinear Phenomena*, 240(18):1416 – 1425, 2011.

[48] K. E. Howes, A. M. Fowler, and A. S. Lawless. Accounting for model error in strong-constraint 4d-var data assimilation. *Quarterly Journal of the Royal Meteorological Society*, 143(704):1227–1240, 2017.

[49] M. Iglesias. Iterative regularization for ensemble data assimilation in reservoir models. *Computational Geosciences*, 19(1):177–212, 2015.

[50] M. Iglesias and C. Dawson. The regularizing Levenberg–Marquardt scheme for history matching of petroleum reservoirs. *Computational geosciences*, 17(6):1033–1053, 2013.

[51] A. Jazwinski. *Stochastic processes and filtering theory*. Mathematics in science and engineering. Academic press, New York, 1970.

[52] K. Judd. Forecasting with imperfect models, dynamically constrained inverse problems, and gradient descent algorithms. *Physica D: Nonlinear Phenomena*, 237(2):216–232, 2008.

[53] K. Judd. Fifty years of forecasting chaos and the shadow of imperfect models. *Nonlinear Theory and Its Applications*, 2016.

[54] K. Judd, C. Reynolds, T. Rosmond, and L. Smith. The geometry of model error. *Journal of the Atmospheric Sciences*, 65:1749–1772, 2008.

[55] K. Judd and L. Smith. Indistinguishable states I. Perfect model scenario. *Physica D*, 151:125–141, 2001.

[56] K. Judd and L. Smith. Indistinguishable states II: The imperfect model scenario. *Physica D: nonlinear phenomena*, 196(3-4):224–242, 2004.

[57] K. Judd and T. Stemler. Failures of sequential bayesian filters and the successes of shadowing filters in tracking of nonlinear deterministic and stochastic systems. *Physical Review E*, 79(6):066206, 2009.

[58] R. Kálmán. A New Approach to Linear Filtering and Prediction Problems. *Journal of Basic Engineering*, 82(1):35–45, 1960.

[59] E. Kalnay. *Atmospheric Modelling, Data Assimilation and Predictability.* Cambridge University Press, 2003.

[60] B. Kaltenbacher, A. Neubauer, and O. Scherzer. *Iterative regularization methods for nonlinear ill-posed problems*, volume 6. Walter de Gruyter, 2008.

[61] L. Kantorovich and G. Akilov. *Functional analysis in normed spaces.* Fizmatgiz, Moscow, 1959.

[62] A. Katok and B. Hasselblatt. *Introduction to the Modern Theory of Dynamical Systems*, volume 54 of *Encyclopedia of mathematics and its applications.* Cambridge University Press, 1995.

[63] K. Law, A. Stuart, and K. Zygalakis. *Data assimilation : a mathematical introduction*, volume 62 of *Texts in applied mathematics.* Springer, 2015.

[64] K.J.H. Law, D. Sanz-Alonso, A. Shukla, and A.M. Stuart. Controlling unpredictability with observations in the partially observed Lorenz '96 model. *ArXiv e-prints*, 2014.

[65] K. Levenberg. A method for the solution of certain non-linear problems in least squares. *Quart. Appl. Math.*, 2:164–168, 1944.

[66] J. Lewis and J. Derber. The use of adjoint equations to solve a variational adjustment problem with advective constraint. *Tellus*, 37A:309–322, 1985.

[67] A.C. Lorenc. Analysis methods for numerical weather prediction. *Quarterly Journal of the Royal Meteorological Society*, 112(474):1177–1194, 2000.

[68] A.C. Lorenc, S.P. Ballard, R.S. Bell, N.B. Ingleby, P.L.F. Andrews, D.M. Barker, J.R. Bray, A.M. Clayton, T. Dalby, D. Li, T.J. Payne, and F.W. Saunders. The met. office global three-dimensional variational data assimilation scheme. *Quarterly Journal of the Royal Meteorological Society*, 126(570):2991–3012, 2000.

[69] E. Lorenz. Deterministic Nonperiodic Flow. *Journal of Atmospheric Sciences*, 20:130–148, March 1963.

[70] E. Lorenz. Predictability - a problem partly solved. In Tim Palmer and Renate Hagedorn, editors, *Proceedings of seminar on Predictability*, volume 1, pages 1–18, Reading, UK, 1996. ECMWF, Cambridge University Press.

[71] J. Mandel, E. Bergou, S. Gürol, S. Gratton, and I. Kasanický. Hybrid levenberg–marquardt and weak-constraint ensemble kalman smoother method. *Nonlinear Processes in Geophysics*, 23(2):59–73, 2016.

[72] D. Marquardt. An algorithm for least-squares estimation of nonlinear parameters. *Journal of the Society for Industrial and Applied Mathematics*, 11(2):431–441, 1963.

[73] R. Miller, E. Carter, and S. Blue. Data assimilation into nonlinear stochastic models. *Tellus A: Dynamic Meteorology and Oceanography*, 51(2):167–194, 1999.

[74] R. Miller, M. Ghil, and F. Gauthiez. Advanced data assimilation in strongly nonlinear dynamical systems. *Journal of the Atmospheric Sciences*, 51(8):1037–1056, 1994.

[75] F. Molteni, L. Ferranti, T. Palmer, and P. Viterbo. A dynamical interpretation of the global response to equatorial pacific sst anomalies. *Journal of Climate*, 6(5):777–795, 1993.

[76] J. Nocedal and S. Wright. *Numerical Optimization*. Springer, New York, NY, USA, second edition, 2006.

[77] D. Oliver, A. Reynolds, and N. Liu. *Inverse Theory for Petroleum Reservoir Characterization and History Matching*. Cambridge University Press, 2008.

[78] L. Palatella, A. Carrassi, and A. Trevisan. Lyapunov vectors and assimilation in the unstable subspace: theory and applications. *Journal of Physics A: Mathematical and Theoretical*, 46(25):254020, 2013.

[79] L. Pecora and T. Carroll. Synchronization in chaotic systems. *Physical Review Letters*, 64(8):821–824, 1990.

[80] L. Pecora and T. Carroll. Driving systems with chaotic signals. *Physical Review A*, 44(4):2374–2383, 1991.

[81] L. Pecora, T. Carroll, G. Johnson, D. Mar, and J. Heagy. Fundamentals of synchronization in chaotic systems, concepts, and applications. *Chaos*, 7(4):520–543, 1997.

[82] C. Pires, R. Vautard, and O. Talagrand. On extending the limits of variational assimilation in nonlinear chaotic systems. *Tellus*, 48A:96–121, 1996.

[83] R. Rotunno and J.-W. Bao. A case study of cyclogenesis using a model hierarchy. *Monthly Weather Review*, 124:1051–1066, 1996.

[84] D. Sanz-Alonso and A. Stuart. Long-time asymptotics of the filtering distribution for partially observed chaotic dynamical systems. *SIAM/ASA Journal on Uncertainty Quantification*, 3(1):1200–1220, 2015.

[85] Y. Sasaki. Some basic formalisms in numerical variational analysis. *Mon. Wea. Rev.*, 98:875–883, 1970.

[86] C. Schillings and A. M. Stuart. Analysis of the ensemble kalman filter for inverse problems. *SIAM J Numerical Analysis*, 55(3):1264–1290, 2017.

[87] C. Schillings and A. M. Stuart. Convergence analysis of ensemble kalman inversion: The linear, noisy case. *Applicable Analysis*, 97:107–123, 2018.

[88] L. Smith, M. Cuéllar, H. Du, and K. Judd. Exploiting dynamical coherence: A geometric approach to parameter estimation in nonlinear models. *Physics Letters A*, 374(26):2618–2623, 2010.

[89] C. Snyder and T. Hamill. Leading Lyapunov vectors of a turbolent baroclinic jet in a quasigeostrophic model. *Journal of the Atmospheric Sciences*, 60:683–688, 2003.

[90] T. Stemler and K. Judd. A guide to using shadowing filters for forecasting and state estimation. *Physica D: Nonlinear Phenomena*, 238(14):1260–1273, 2009.

[91] O. Talagrand. Assimilation of observations, an introduction. *Journal of the Meteorological Society of Japan*, 75(1B):191–209, 1997.

[92] O. Talagrand and P. Courtier. Variational assimilation of meteorological observations with the adjoint vorticity equation. *Q.J.R. Meteorol. Soc.*, 113, 1987.

[93] Y. Trémolet. Accounting for an imperfect model in 4d-var. *Quarterly Journal of the Royal Meteorological Society*, 132(621):2483–2504, 2006.

[94] A. Trevisan, M. D'Isidoro, and O. Talagrand. Four-dimensional variational assimilation in the unstable subspace and the optimal subspace dimension. *Q.J.R. Meteorol. Soc.*, 136:487–496, 2010.

[95] E. Van Vleck. Numerical Shadowing Using Componentwise Bounds and a Sharper Fixed Point Result. *SIAM J. Sci. Comp.*, 22:787–801, 2001.

[96] E. Van Vleck. On the error in the product QR decomposition. *SIAM J. Matrix Anal. Appl.*, 31(4):1775–1791, 2009/10.

[97] D. Zupanski. A general weak constraint applicable to operational 4DVAR data assimilation systems. *Monthly Weather Review*, 125:2274–2292, 1997.

Samenvatting

Het combineren van banen van een model van een (chaotisch) dynamisch systeem met meetdata om tot een verbeterde schatting van de toestand van een fysisch systeem te komen staat bekend onder de naam data-assimilatie. Dit proefschrift gaat over verschillende algorithmes voor data-assimilatie. Deze algorithmes zijn gebaseerd op schaduwing. Schaduwing is een concept uit de theorie van dynamische systemen. Wanneer een dynamisch systeem de eigenschap heeft dat zich in een omgeving van iedere pseudobaan een exacte baan van het dynamisch systeem bevindt, dan schaduwt deze exacte baan de pseudobaan. Met behulp van schaduwing kan aangetoond worden dat een numerieke oplossing van een dynamisch systeem zich in een omgeving van een exacte oplossing bevindt. Schaduwingsverfijning is een numerieke techniek waarbij vanuit een pseudobaan een verbeterde benadering van een exacte oplossing gevonden wordt. Het is mogelijk om een schaduwingsverfijningstechniek te gebruiken voor data-assimilatie. Uitgaande van observaties wordt de methode van Newton toegepast om een nulpunt van een kostenoperator te benaderen, waarbij de kostenoperator kosten toekent aan afwijkingen van modeloplossingen.

De algorithmes van hoofdstuk 2 zijn gebaseerd op een numerieke tijdsafhankelijke splitsing tussen stabiele en instabiele richtingen. Het algorithme gebruikt tijdsafhankelijke projecties op de niet-stabiele deelruimte die bepaald wordt door het gebruik van Ljapoenovexponenten en Ljapoenovvectoren. In de instabiele deelruimte wordt gebruik gemaakt van een schaduwingsalgorithme, terwijl in de stabiele deelruimte gebruik gemaakt wordt van synchronisatie. De methode wordt voorts uitgebreid met parameterschatting en tot sommige gevallen waarin slechts gedeeltelijke observaties beschikbaar zijn.

In hoofdstuk 3 wordt data-assimilatie voor imperfecte modellen behandeld. Door middel van regularisatie volgens de Levenberg-Marquardt-methode

worden imperfecties in het model in beschouwing genomen. Ook wordt beschreven hoe de schaduwingsmethode zich, zowel analytisch als numeriek, verhoudt tot de 4DVar-methode met zwakke restricties en wordt getoond dat de schaduwingsmethode consistent is met de distributie van de meetfout, wat niet het geval is voor de 4DVar-methode met zwakke restricties. Dit effect is met name zichtbaar wanneer er minder observaties zijn. Wanneer er weinig observaties zijn hebben deze in de schaduwingsmethode bovendien een kleinere invloed op niet-geobserveerde variabelen dan in de 4DVar-methode met zwakke restricties.

In hoofdstuk 4 wordt de methode van hoofdstuk 2 uitgebreid tot andere gevallen van gedeeltelijke observaties, op een vergelijkbare wijze als in hoofdstuk 3. Lokale convergentie naar een oplossingsvariëteit wordt bewezen en er wordt voorzien in een ondergrens op een algorithmische tijdstap. Numerieke experimenten met de Lorenz-'63- en Lorenz-'96-modellen laten convergentie van het algorithme zien en laten voorts zien dat de methode gunstig is ten opzichte van de 4DVar-methode met zwakke restricties en een andere schaduwingsmethode genaamd pseudobaandata-assimilatie.

In hoofdstuk 5 wordt de methode van de voorgaande hoofdstukken verder doorontwikkeld. Het algorithme wordt uitgebreid tot een ensemble van toestanden voor het schatten van onzekerheden van het algorithme, gebaseerd op het concept van ononderscheidbare toestanden. Het hoofdstuk omvat ook enige bewijzen over eenduidigheid, accuraatheid en consistentie van het algorithme. Het algorithme wordt toegepast op een imperfect model om te tonen hoe de niet-gemodelleerde componenten van het model geschat kunnen worden met behulp van het data-assimilatie-algorithme.

Samenvattend wordt in dit proefschrift getoond hoe data-assimilatie met behulp van algorithmen gebaseerd op schaduwing gedaan kan worden. Ook wordt getoond hoe deze algorithmen aangepast kunnen worden aan verscheidene situaties en dat deze algorithmen vergeleken kunnen worden met geralateerde data-assimilatie-algorithmen, zoals de 4DVar-methode met zwakke restricties.

Curriculum Vitae

Work experience

- Trainee patent attorney at NLO, 2019-

  Eindhoven office, telecom, electronics and software sector. Drafting and prosecution of patent applications, advice work, litigation support

- PhD student at Centrum Wiskunde & Informatica, 2015-2019

  Conducting research in the Scientific Computing Group, under supervision of Svetlana Dubinkina & Jason Frank, leading to the present thesis.

  Including teaching the master level mathematics course "Laboratory class Scientific Computation" at Utrecht University for two semesters

- Student assistent at Utrecht University, 2014

  First year math course "Modellen en Simulaties"

  Second year physics course "Structuur van de Materie"

  Second year physics course "Statistische fysica theorie en experiment" (theoretical part)

Education

- Beroepsopleiding Octrooigemachtigden, Stichting Beroepsopleiding Octrooigemachtigden, 2020-2021

  Professional training to become a Dutch patent attorney

- Master Mathemetical Sciences, Utrecht University, 2012-2015

- Master Theoretical Physics, Utrecht University, 2012-2015

  Master thesis: On the numerical computation of electron transport through a topological crystalline insulator (for both masters)

- Bachelor Mathematics, Utrecht University, 2008-2012
- Bachelor Physics, Utrecht University, 2008-2012

  Bachelor thesis: Colloidal rods in electric fields (for both bachelors)

- Descartes College, Utrecht University, 2009-2011

  The Descartes College is an interdisciplinary honours programme of Utrecht University

- Erasmus Exchange program, Humboldt Universität zu Berlin, 2010-2011
- Christelijk Gymnasium Utrecht, 2002-2008
- Junior College Utrecht, Utrecht University, 2006-2008

  Junior College Utrecht is a selective science programme organised by Utrecht University and schools from the region.

Other

- Society for Industrial and Applied Mathematics Utrecht University student chapter: Treasurer 2017-2018
- Centrum Wiskunde & Informatica: Editor personnel magazine "We@CWI" 2017-2018
- Internal competition leader and board member at Schaakclub Woerden (2012-2016)
- Participant in the 106th and 126th European Study Group Mathematics with Industry