

Projected Shadowing-based Data Assimilation

Bart de Leeuw* Svetlana Dubinkina† Jason Frank‡ Andrew Steyer‡
 Xuemin Tu§ Erik Van Vleck¶

July 31, 2017

Abstract

In this article 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) [10, 53, 41] and Pseudo-orbit Data Assimilation (PDA) [32, 31, 24]. 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.

Key words: Data assimilation; tangent space decomposition; shadowing; synchronization;

AMS subject classifications: 62M20; 37C50; 34D06; 37M25

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. In this paper we develop a data assimilation method in the context of a discrete deterministic model

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

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

$$\dot{x} = f(t, x), \quad x(t) \in \mathcal{R}^d, \quad (2)$$

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

Let the sequence¹ $\{\mathcal{X}_0, \mathcal{X}_1, \dots, \mathcal{X}_N\}$ be a distinguished orbit of (1), referred to as the *true solution* of the model, and presumed to be unknown. Suppose we are given a sequence of noisy observations

*Centrum Wiskunde & Informatica, PO Box 94079, 1090 GB Amsterdam, Netherlands, (b.m.de.leeuw@cwi.nl, s.dubinkina@cwi.nl).

†Utrecht University, Mathematical Institute, P.O. Box 80010, 3508 TA Utrecht, Netherlands, (j.e.frank@uu.nl).

‡Sandia National Laboratories, PO Box 5800 Albuquerque, NM 87185, (asteyer@sandia.gov).

§University of Kansas, Dept. of Mathematics, 405 Snow Hall, Lawrence, KS 66405, (xuemin@ku.edu, erikvv@ku.edu).

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

y_n related to \mathcal{X}_n via

$$y_n = H\mathcal{X}_n + \xi_n, \quad y_n \in \mathcal{R}^b, \quad n = 0, \dots, N, \quad (3)$$

where $H : \mathcal{R}^d \rightarrow \mathcal{R}^b$, $b \leq d$, 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 2.1) $\mathbf{u} = \{u_0, u_1, \dots, u_N\}$, $u_n \in \mathcal{R}^d$, of the model (1), such that the differences $\|y_n - Hu_n\|$, $n = 0, \dots, N$ 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 [51, 36]. For instance, well known four-dimensional variational data assimilation (4DVar) aims at finding the optimal initial condition u_0 of 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 (see e.g. [48, 37, 52, 51] and references therein). One drawback of variational data assimilation is that the number of local minima of the cost function increases dramatically with N [4, 40, 45]. 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*

$$G(\mathbf{u}) = \begin{pmatrix} G_0(\mathbf{u}) \\ G_1(\mathbf{u}) \\ \vdots \\ G_{N-1}(\mathbf{u}) \end{pmatrix}, \quad G_n(\mathbf{u}) = u_{n+1} - F_n(u_n), \quad n = 0, \dots, N-1, \quad (4)$$

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 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 6.1 we demonstrate this idea using direct insertion of noisy partial observations into the iteration 1.

Recent attempts [10, 53, 26, 41, 35, 47] 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). 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 [53, 41]. On the other hand, the stable subspace may also be exploited. Research by Pecora & Carroll [42] 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. Their work has triggered a substantial body of research on the idea of synchronization of chaos (see review articles by Pecora et al. [44] and Boccaletti et al. [6]).

Motivated by the above, in this paper 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 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 4. The updates can then be synchronized in the stable directions as shown by the analysis in A. We provide details of our implementation in 5. Finally, in 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 7.

2 Background

In 2.1 we review concepts from numerical shadowing, in 2.2 we describe the computation of tangent space splitting used in this paper, and in 2.3 we review synchronization of chaos.

2.1 Numerical shadowing

An ε -pseudo-orbit is a sequence $\mathbf{u} = \{u_0, u_1, \dots, u_N\}$ satisfying $\|G_n(\mathbf{u})\| < \varepsilon$, $n = 0, \dots, N - 1$. 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 \mathbf{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 [34]) 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 $n = 0, \dots, N$. Rigorous bounds on the global error of numerical integrations with respect to a shadowing orbit can be proved by applying the Newton-Kantorovich theorem to Newton’s iteration for $G(\mathbf{x}) = 0$ with starting data given by the numerical iterates \mathbf{u} on a time interval that is long relative to the characteristic Lyapunov time [5, 29, 28, 13, 14, 54]. Shadowing is an important analysis technique for obtaining global error bounds on the numerical approximation to the solution of differential equations exhibiting chaos. 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 \mathbf{u} δ -shadow a desired orbit of (2). Shadowing refinement (see, e.g., [27]), employs the pseudo-orbit as an initial guess for $G(\mathbf{u}) = 0$ 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 [32, 31, 24] and references therein. For the PDA approach a cost function

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

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 $G(\mathbf{u}) = 0$, that is, if \mathbf{u} is any model trajectory. The approach in [24] approximately 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 $n = 0, \dots, N$, 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 [31], parameter estimation [49] and as a method for finding reference trajectories for ensemble forecasting [24].

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 paper. Let $\{x_n; n = 0, \dots, N\}$ denote an orbit of 1. The fundamental matrix equation associated with $\{x_n\}$ is a matrix valued difference equation

$$X_{n+1} = F'_n(x_n)X_n, \quad n = 0, \dots, N-1, \quad (5)$$

where $X_n \in \mathcal{R}^{d \times d}$. The iterates of 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

$$Q_{n+1} R_{n+1} = F'_n(x_n) Q_n \quad \text{for } n = 0, \dots, N-1, \quad (6)$$

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 [22]:

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

The method of construction ensures $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_p$. 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 [15, 16, 17]). The iteration 5 is a generalized power iteration. For each $\ell = 1, \dots, p$, one finds in the limit $n \rightarrow \infty$ that $\text{span}\{V_n^{(1)}, \dots, V_n^{(\ell)}\} = \text{span}\{Q_n^{(1)}, \dots, Q_n^{(\ell)}\}$, where $Q_n^{(i)}$ denotes the i th column of Q_n [22, 19, 1, 3]. 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 $Q_n^u = (Q_n^{(1)}, \dots, Q_n^{(p)})$ provides an orthonormal basis for the non-stable tangent space at X_n . In practice we may obtain Q_n^u using a thin QR-factorization,

$$Q_{n+1}^u R_{n+1}^u = F'_n(x_n) Q_n^u \quad \text{for } n = 0, \dots, N-1, \quad (7)$$

where $Q_0^u \in \mathcal{R}^{d \times p}$ and $R_n^u \in \mathcal{R}^{p \times p}$ (please note R_n^u is the upper left $p \times p$ block of R_n). 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 $R_n^u \in \mathcal{R}^{p \times p}$ with positive diagonal elements and, importantly, preserves the ordering of the columns of the Q_n^u . We note here that by approximating the non-stable subspace we obtain information (see [26]) 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 than the total dimension. In Carrassi et al. [11] it is shown that the AUS-framework gives good results for a quasi-geostrophic model described in [46]. This model is of dimension 14784, while the unstable subspace has a dimension of 24 [50].

We will use the computed factors $Q_n \in \mathcal{R}^{d \times p}$ to construct projection operators onto the non-stable tangent space. The Q_n 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). In particular, the results in [20, 21, 55, 2] show that the Q_n 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. This is characterized by the integral separation or integral separation structure (see also [1]) which is closely related to the continuity of Lyapunov exponents with respect to perturbations of $F'(x_n)$. In our context this ensures the time dependent projection operators $P_n = Q_n^u Q_n^{uT}$ are robust.

2.3 Synchronization

Pecora & Carroll [42] 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 [42, 30, 26].

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

$$x_{n+1} = F_n(x_n), \quad (8a)$$

$$z_{n+1} = P_n x_{n+1} + (I - P_n) F_n(z_n), \quad (8b)$$

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 $w_n = z_n - x_n$, $n = 0, 1, \dots$, the transverse dynamics with respect to \mathcal{S} is given by

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

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 [43]. 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 . 1 illustrates synchronization of the Lorenz 96 model (see [39] and 6.2) using the driver-response system (8a)–(8b) 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 .

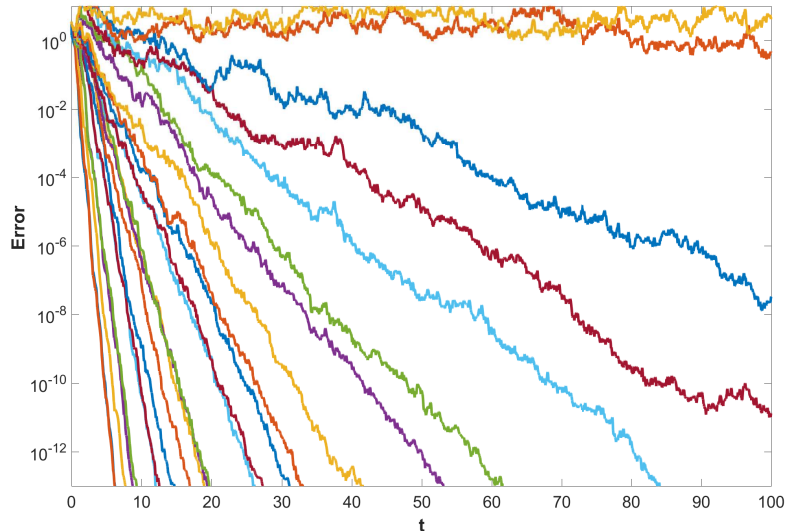


Figure 1: Synchronization in the L96 model 22 [39] with 13 positive Lyapunov exponents in 36 dimensions. We plot the ℓ^∞ -norm of the difference between the true solution and the synchronization approximation as a function of time. Forcing is done with projections of the true solution onto the non-stable space and the different graphs are for $k = 12, 13, \dots, 35$, colored with the Matlab default color order starting from 35. It can be observed that after a transient time and for sufficiently large k (i.e. $k \geq 14$), convergence to the true solution is exponential.

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 [9]. An important property of Newton's method is its local nature: when the initial guess is sufficiently close to a zero, the iterates converge to that zero at a quadratic rate. This statement is made formal in the Kantorovich Theorem [33].

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

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

where G is defined in (4) and starting data is provided by the noisy observations $\{y_n : n = 0, \dots, N\}$ with observation operator the identity $\{H\mathcal{X}_n = \mathcal{X}_n : n = 0, \dots, N\}$. (an assumption that can be relaxed, see 6).

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

$$G(\mathbf{u}^{(k)} + \delta^{(k)}) = 0. \quad (9)$$

We then update using $\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \delta^{(k)}$. The solution to (9) is approximated by iterating

$$G'(\mathbf{u}^{(k)})\delta^{(k)} = -G(\mathbf{u}^{(k)}), \quad \mathbf{u}^{(k+1)} := \mathbf{u}^{(k)} + \delta^{(k)} \quad (10)$$

Table 1: Application of Newton’s method to L63. $C(\cdot)$ and MSE are defined in equations (12) and (11) respectively.

Property	Value
Observation error $C(\{\mathcal{X}_n\})$	2.9853
Observational discrepancy of estimated trajectory $C(\mathbf{u})$	2.9844
MSE between estimation and the truth	8.1921e-04

to convergence. We remark that the function $G(\mathbf{u})$ has a zero for every orbit of the model (1). The function $G : \mathcal{R}^{dN} \rightarrow \mathcal{R}^{d(N-1)}$ has a $d(N-1) \times dN$ Jacobian with block structure

$$G'(\mathbf{u}) = \begin{bmatrix} -F'_0(u_0) & I & & & \\ & -F'_1(u_1) & I & & \\ & & \ddots & \ddots & \\ & & & -F'_{N-1}(u_{N-1}) & I \end{bmatrix}.$$

We 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. To distinguish this method from the projected method to be described in 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. Initializing it with observations, we can expect to find a trajectory close to observations, provided the initial observational error is not too large [9, 8].

The convergence of this approach with Newton’s method can be demonstrated with a numerical example. Using the Lorenz 63 model [38] (see also 6.1) the true trajectory $\{X_n\}$ was integrated and perturbed² as specified in (3) with noise covariance $E = I$. The perturbed data was used as a starting guess for Newton’s method. As shown in 2, convergence to a model trajectory can be observed, and the mean square error (MSE) defined as

$$\text{MSE} = \frac{1}{N} \sum_{n=1}^N (u_n - \mathcal{X}_n)^T (u_n - \mathcal{X}_n) \quad (11)$$

is equal to 8.1921e-04. Next we examine errors with respect to the observation operator

$$C(\{x_n\}) = \frac{1}{N} \sum_{n=1}^N (y_n - Hx_n)^T (y_n - Hx_n). \quad (12)$$

The mean squared error in the observations (mean noise variance) is given by $C(\{\mathcal{X}_n\}) = 2.9853$. By comparison, the mean observation discrepancy of the Newton’s solution is equal to $C(\mathbf{u}) = 2.9844$. 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.

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 $G(\mathbf{u})$ defined in (4) and replace $G_n(\mathbf{u})$ with $G_n(\mathbf{u}; \boldsymbol{\alpha}) = u_{n+1} - F_n(u_n; \boldsymbol{\alpha})$ where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \dots, \alpha_q)$. Linearization with

²For this simple demonstration, we compute just a single realization of the noise process. Later in 6.1 we include results for an ensemble.

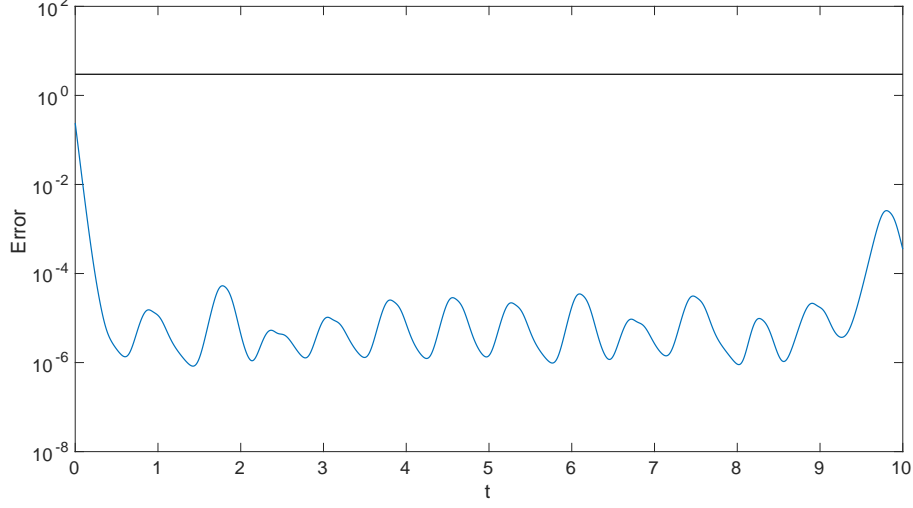


Figure 2: Application of Newton's method to the L63 model 20. The mean observational error is in black, the error over time of the estimation is in blue.

respect to α_j takes the form

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

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

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

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

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

so that $G'_\alpha G'_\alpha^T$ is a rank q perturbation of the block tridiagonal matrix $G'_\mathbf{u} G'_\mathbf{u}^T$. This allows for the use of Sherman-Morrison-Woodbury formulas and a solver for $G'_\mathbf{u} G'_\mathbf{u}^T$ to solve linear systems with matrix $G' G'^T$. In the case of time dependent parameters α_n , the diagonal blocks of $G'_\mathbf{u} G'_\mathbf{u}^T$ are modified and the overall block tridiagonal structure is maintained in $G' G'^T$. In 6.4 we illustrate this approach with numerical experiments.

4 Tangent space splitting of Newton's method

In the previous section we demonstrated that Newton's method applied to the residual 4 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 p 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 (9) into the equivalent system

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

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

We propose to modify the Newton's iteration as follows. Instead of computing the update $\delta^{(k)}$ 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 k th iteration, we first approximate the update in the range of $\widehat{\mathcal{P}}^{(k)}$, neglecting the term $(I - \widehat{\mathcal{P}}^{(k)})\delta^{(k)}$ in the first equation above and solving

$$\mathcal{P}^{(k)}G(\mathbf{u}^{(k)} + \widehat{\mathcal{P}}^{(k)}\delta^{(k)}) = 0 \quad (13)$$

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

$$(I - \mathcal{P}^{(k)})G(\mathbf{u}^{(k)} + \widehat{\mathcal{P}}^{(k)}\delta^{(k)}) + (I - \widehat{\mathcal{P}}^{(k)})\delta^{(k)} = 0 \quad (14)$$

for $\delta_{\perp}^{(k)} = (I - \widehat{\mathcal{P}}^{(k)})\delta^{(k)}$. Then the update is computed as $\mathbf{u}^{(k+1)} = \mathbf{u}^{(k)} + \delta_{||}^{(k)} + \delta_{\perp}^{(k)}$. Expressions 13 and 14 are solved approximately for the components $\delta_{||}^{(k)}$ and $\delta_{\perp}^{(k)}$ as described below.

4.1 Computation of projection matrices

The basis Q_n^u , $n = 0, \dots, N$, 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 k th 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 6 linearized along the most recently updated pseudo-orbit $\mathbf{u}^{(k)}$. That is, we take $x_n = u_n^{(k)}$, $n = 0, \dots, N$, in 6. For the first iteration we use the observations: $u_n^{(0)} = y_n$, $n = 0, \dots, N$.

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 [18].

4.2 Newton's step on the unstable space

Linearization of (13) yields a projected linear system for the update $\delta_{||}^{(k)} = \widehat{\mathcal{P}}^{(k)}\delta^{(k)}$:

$$\mathcal{P}^{(k)}G'(\mathbf{u}^{(k)})\widehat{\mathcal{P}}^{(k)}\delta^{(k)} = -\mathcal{P}^{(k)}G(\mathbf{u}^{(k)}). \quad (15)$$

Suppressing the iteration index k for the moment, define block matrices $\mathcal{Q} = \text{blockdiag}(Q_1^u, \dots, Q_N^u)$ and $\widehat{\mathcal{Q}} = \text{blockdiag}(Q_0^u, \dots, Q_N^u)$, 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 $\boldsymbol{\mu} = \widehat{\mathcal{Q}}^T\delta = \widehat{\mathcal{Q}}^T\widehat{\mathcal{P}}\delta$, $\widetilde{G}' = \mathcal{Q}^TG'(\mathbf{u})\widehat{\mathcal{Q}}$ and $\mathbf{b} = \mathcal{Q}^TG(\mathbf{u})$. Then the linear system for the update $\boldsymbol{\mu}$ may be written as

$$\widetilde{G}'\boldsymbol{\mu} = -\mathbf{b}, \quad (16)$$

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, $\tilde{G}' \in \mathcal{R}^{Np \times (N+1)p}$. We solve (16) using the right pseudoinverse $\tilde{G}'^\dagger = \tilde{G}'^T (\tilde{G}' \tilde{G}'^T)^{-1}$ and define the intermediate update

$$\bar{\mathbf{u}}^{(k)} = \mathbf{u}^{(k)} + \hat{\mathcal{P}}^{(k)} \boldsymbol{\delta}^{(k)} = \mathbf{u}^{(k)} + \hat{\mathcal{Q}}^{(k)} \boldsymbol{\mu}^{(k)}. \quad (17)$$

4.3 Synchronization step in the stable space

We next turn to the treatment of 14. Inserting the definition 17 into 14 yields the relation

$$(I - \mathcal{P}^{(k)})G(\bar{\mathbf{u}}^{(k)} + (I - \hat{\mathcal{P}}^{(k)})\boldsymbol{\delta}^{(k)}) = 0, \quad (18)$$

whose solution for $\boldsymbol{\delta}_\perp^{(k)} = (I - \hat{\mathcal{P}}^{(k)})\boldsymbol{\delta}^{(k)}$ we wish to approximate. Again dropping the iteration index k for the moment, we expand (18) component-wise over the time index n :

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

The second equation is rewritten in the form

$$(I - P_{n+1})\delta_{n+1} = (I - P_{n+1}) (F_n(\bar{u}_n + (I - P_n)\delta_n) - \bar{u}_{n+1}), \quad n = 0, \dots, N-1.$$

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

$$\bar{u}_{n+1} + (I - P_{n+1})\delta_{n+1} = P_{n+1}\bar{u}_{n+1} + (I - P_{n+1})F_n(\bar{u}_n + (I - P_n)\delta_n), \quad n = 0, \dots, N-1,$$

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

$$u_{n+1}^{(k+1)} = P_{n+1}\bar{u}_{n+1}^{(k)} + (I - P_{n+1})F_n(u_n^{(k+1)}), \quad n = 0, \dots, N-1. \quad (19)$$

The form of this iteration is identical to that of the receiver equation (8b) in the synchronization process. In other words, given the update $\bar{\mathbf{u}}^{(k)}$, corrected in the non-stable subspace 17, the correction to the stable subspace can be implemented through a forward synchronization integration 19. In the Appendix, we prove that under suitable assumptions, if after k iterations, the error in $\bar{\mathbf{u}}^{(k)}$ 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 19 converges exponentially to \mathcal{X}_n as $n \rightarrow \infty$.

To summarize, the complete iteration step consists of:

1. Compute the approximate basis Q_n^u , $n = 0, \dots, N$, for the tangent bundle along the pseudo-trajectory $\{u_n^{(k)}\}$.
2. Solve the linear system 15 for the update $\boldsymbol{\delta}_\perp^{(k)}$ in the non-stable subspace, and compute the intermediate update $\bar{\mathbf{u}}^{(k)}$ from 17.
3. Synchronize in the stable subspace using the forward iteration 19 to obtain $\mathbf{u}^{(k+1)}$.

The Newton's step in the unstable subspace is based upon residual ($r_n := u_{n+1} - F_n(u_n)$) correction with both the residual and the correction projected into the unstable subspace. If $P_n r_n = r_n$ 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.

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 ??). 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 [31] or by just inserting climatological means for missing observations [23, 24]. In 6.1, we demonstrate an alternative preprocess motivated 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 paper 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 , $m = 0, \dots, M + 1$, 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 m th 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 19 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 (19) we can define a unique solution (this analysis point of view is also taken up in [30] and [26]). 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 3) on a short window and start the forced system (19) from there. This also improves the approximation of the unstable directions at the beginning of the window at which the projected method is started.

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 Gaussian noise as in equation 3 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 $\frac{\|\mathbf{b}\|_2}{\|\mathbf{u}\|_2} < 10^{-15}$, where \mathbf{b} is the projected residual in 16.

6.1 Dependence on projector in the L63 model

The well-known Lorenz attractor [38] is a chaotic dynamical system commonly used as a test problem for data assimilation algorithms. The L63 model is

$$\dot{x}_1 = \sigma(x_2 - x_1), \quad \dot{x}_2 = x_1(\rho - x_3) - x_2, \quad \dot{x}_3 = x_1x_2 - \beta x_3 \quad (20)$$

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

In 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 [42, 30], 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 (17) and (19). Errors (11)–(12) are given in 2, 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 [42, 30].

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

Table 2: Application of the algorithm to L63 with $P = P_{x_1}$. Results are unsatisfactory. Please recall C and MSE are defined in equations (12) and (11) respectively.

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

Table 3: 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 (12), (11) and (21) respectively.

Property	Value
Observation error $C(\{X_n\})$	11.9
Distance between Estimate and Observations $C(\mathbf{u})$	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

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 3, where in addition to errors we also display a measure of discontinuity at window boundaries defined as

$$D = \frac{1}{N} \sum_{n=1}^N \max |G_n(\mathbf{u})| \quad (21)$$

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 ??, where it can be seen that the algorithm 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 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 6, observations of the full model state are not feasible and thus

Table 4: Application of the algorithm to L63 with P_2 . Good results are obtained. Please recall C , MSE and D are defined in equations (12), (11) and (21) 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 $C(\mathbf{u})$	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

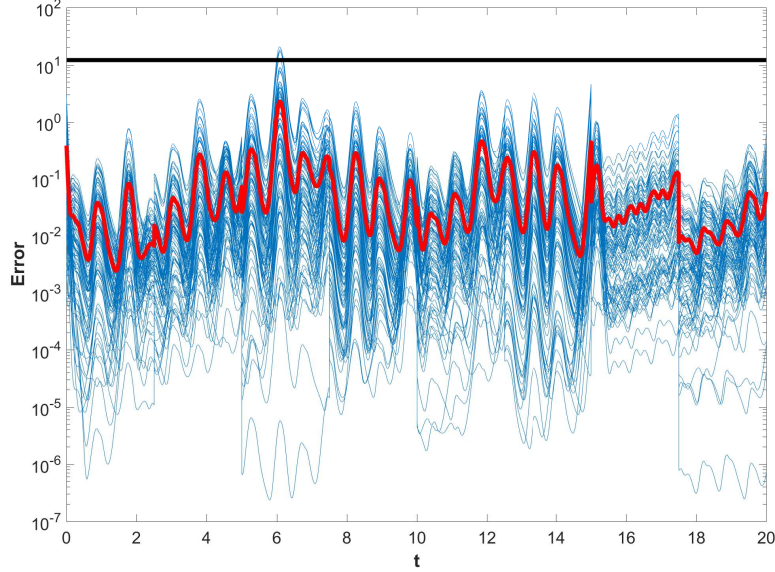


Figure 3: Application of the algorithm to L63 with P_2 . We compare the time-averaged observational error (black) with the error of the estimations (blue) over time. We used 100 observational noise realizations. The average estimation error over time is shown in red.

Table 5: 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 (12), (11) and (21) respectively.

Property	Value
Observation error $C(\{X_n\})$	3.97
Distance between Estimate and Observations $C(u)$	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

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 (8b) with observations of x_1 as driving signal and $H\mathcal{X}_n = P_{x_1}\mathcal{X}_n$, $n = 0, \dots, N$, 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 5, 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.

6.2 Dependence on window length in the L96 model

Lorenz [39] proposed the following model as an example of a simple one-dimensional model with features of the atmosphere. The L96 model is

$$\dot{x}_l = -x_{l-2}x_{l-1} + x_{l-1}x_{l+1} - x_l + \mathcal{F}, \quad (l = 1, \dots, d), \quad (22)$$

Table 6: Application of the algorithm to L96 with P_{15} . Please recall C , MSE and D are defined in equations (12), (11) and (21) respectively.

Property	Value							
Window length	5		2.5			1.25		0.75
Initial window	15	5	15	5	2.5		1.25	0.75
Observation error $C(\{X_n\})$	3.24							
Distance between Estimate and Observations $C(\mathbf{u})$	4.70	3.65	3.58	3.36	3.35	3.22	3.25	3.26
Error between Estimate and the Truth MSE	1.47	0.41	0.40	0.18	0.18	0.09	0.12	0.20
Discontinuity measure at window boundaries D	0.96	0.51	0.48	0.35	0.35	0.21	0.24	0.28
Average number of iterations #	11.1	10.4	8.8	8.5	8.6	7.5	7.5	7.0

Table 7: Application of the algorithm to L96 with P_{25} . Please recall C , MSE and D are defined in equations (12), (11) and (21) respectively.

Property	Value							
Window length	5		2.5			1.25		0.75
Initial window	15	5	15	5	2.5		1.25	0.75
Observation error $C(\{X_n\})$	3.24							
Distance between Estimate and Observations $C(\mathbf{u})$	4.01	3.94	3.38	3.26	3.26	3.16	3.16	3.09
Error between Estimate and the Truth MSE	0.82	0.74	0.22	0.11	0.12	0.09	0.10	0.16
Discontinuity measure at window boundaries D	0.89	0.87	0.37	0.33	0.32	0.27	0.27	0.30
Average number of iterations #	9.1	9.1	8.0	8.0	8.0	7.0	7.0	6.9

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 $E = 0.3^2 I$. 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) 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 $G(\mathbf{u}) = 0$, 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 in all experiments. In ?? and ?? 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 $p = 15$ results in better estimation, also see 4.

In 8 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 [53, 41] The values that are 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

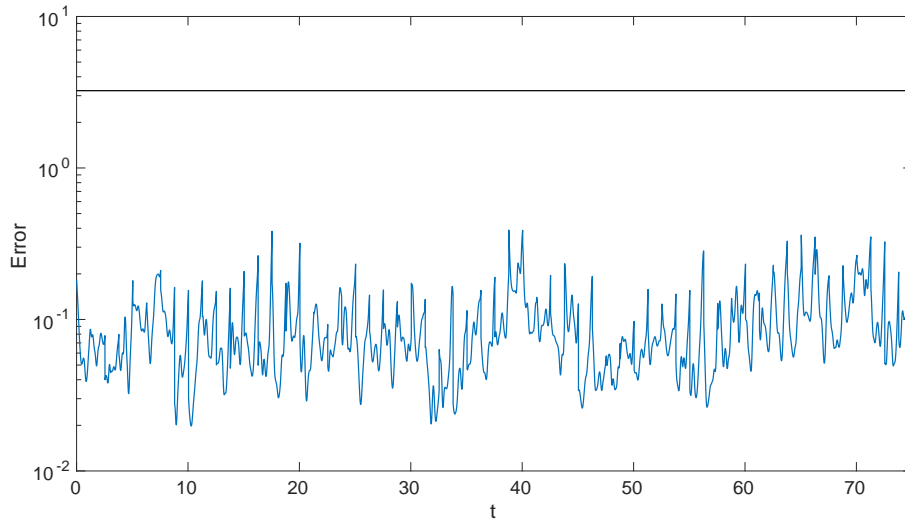


Figure 4: The best estimation obtained by the algorithm applied to L96 ($p = 15$, initialization window of 2.5, following windows of 1.25).

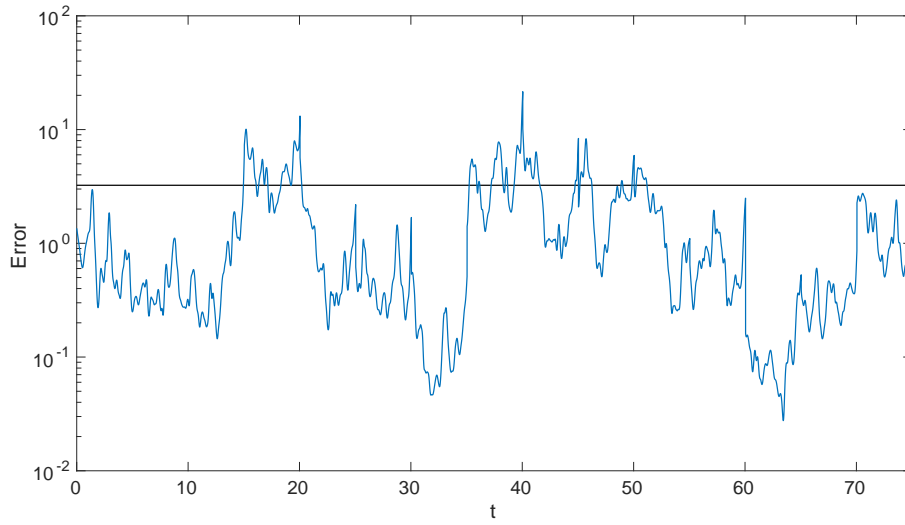


Figure 5: The algorithm can also be applied to L96 with very long windows ($p = 15$, initialization window of 15, following windows of 5). Trajectories are found, but if error reduction is the main purpose longer windows do not necessarily result in smaller errors.

is used (i.e. $p = 40$) and at the subsequent three windows p is as specified in 8; $C(\mathbf{u})$ is then computed over all 4 windows.

Table 8: Distance between Estimate and Observations $C(\mathbf{u})$ for L96.

p	5	10	13	15	20	30	40
$C(\mathbf{u})$	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.

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 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. Some results are shown in ??.

From this test it can be seen that 4DVar and our method give comparable results in this test case, but that our 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.

We remark that one iteration of our 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

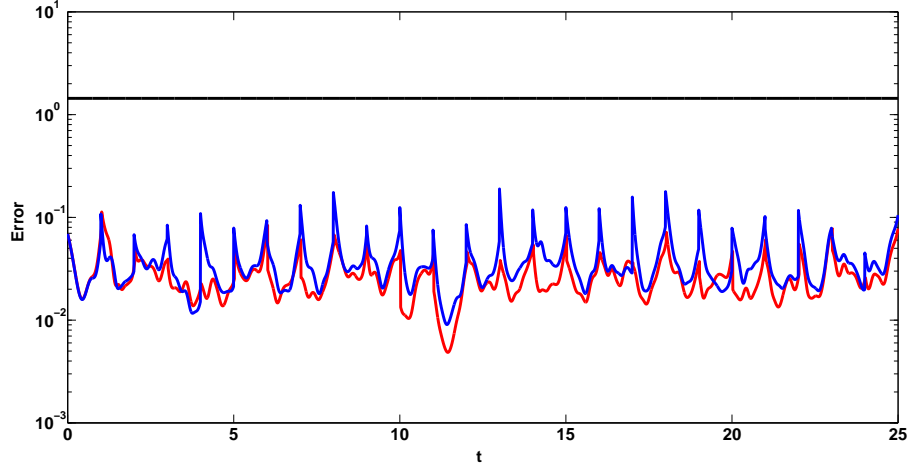


Figure 6: The error between the results of the approximation methods and the truth over time, measured using the ∞ -norm. The error of 4DVar is shown in blue, the error for our shadowing method is shown in green.

Table 9: 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
Observation error $C(\{X_n\})$	1.43	
Distance between Estimate and Observations $C(u)$	1.40	1.39
Error between Estimate and the Truth MSE	0.027	0.037
Discontinuity measure at window boundaries D	0.14	0.17
Average number of iterations #	6.3	418.3

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 $\mathcal{O}(pNc) + \mathcal{O}(Nd(p)^2)$. 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.

We can see in 9 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 [53].

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 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 [4, 40, 45]. 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.

6.4 Parameter estimation

As described in 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 10, 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 11, we show σ estimations obtained by 4DVar using a window length of 0.25. Instead of using the method of 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 12 this method fails for σ estimations, though performs sufficiently well for ρ (see 13) 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 10: Estimation of σ by shadowing-based data assimilation methodology from 3. The true value is 10.

Property	Initial guess for σ			
Initial σ	5	10	15	20
Observation error $C(\{X_n\})$	2.97			
Distance between Estimate and Observations $C(\mathbf{u})$	2.98	2.96	2.96	2.98
Error between Estimate and the Truth MSE	0.03	0.02	0.03	0.07
Estimated σ	10.08	10.03	10.05	10.06

Table 11: Estimation of σ by 4DVar parameter estimation. The true value is 10.

Property	Initial guess for σ			
Initial σ	5	10	15	20
Observation error $C(\{X_n\})$	2.97			
Distance between Estimate and Observations $C(\mathbf{u})$	2.98	2.97	3.04	3.06
Error between Estimate and the Truth MSE	0.12	0.12	0.17	0.18
Estimated σ	9.92	9.95	9.91	9.94

Table 12: 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 between Estimate and Observations $C(\mathbf{u})$	360	5.50	∞	∞
Error between Estimate and the Truth MSE	355	2.77	∞	∞
Estimated σ	8.85	9.83	∞	∞

Table 13: 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 between Estimate and Observations $C(\mathbf{u})$	35.7	2.95	5.52	161
Error between Estimate and the Truth MSE	32.7	0.02	2.74	159
Estimated ρ	26.7	28.0	28.4	37

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. 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. 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.

A Convergence of the synchronization update in the stable subspace

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 than 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

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

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

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

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

$$v_n := \mathcal{X}_n - \bar{u}_n - w_n, \quad (23)$$

and denote the projection on the κ -th column of Q_n by v_n^κ , for $\kappa = p+1, \dots, d$. 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 $\hat{\lambda}_k := \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{l=0}^n \log(R_l^{(k,k)} + \delta^2)$, for $k = p+1, \dots, d$.

Assumption 1 *There exists a positive constant $\hat{\epsilon} > 0$ and a positive integer \mathcal{N} such that for all integers $\mathcal{N}' \geq \mathcal{N}$ and all $n > 0$,*

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

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

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

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

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

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

Proof of Theorem 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:

$$(I - P_{n+1})v_{n+1} = (I - P_{n+1})[Q_{n+1}R_{n+1}(\mathcal{X}_n - \bar{u}_n - w_n)]. \quad (24)$$

We now split the right side of (24) between the range and the kernel of P_n :

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

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

$$Q_{n+1}R_{n+1} = I + \Delta_n. \quad (25)$$

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 [12]. 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 [54]. We remark that R_n encodes the local approximation to the Lyapunov exponents [25]. 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

$$\begin{aligned} \|(I - P_{n+1})v_{n+1}\|_2 &< \|(I - P_{n+1})F'_n \{(I - P_n)v_n\}\|_2 + \zeta \\ &\leq \|(I - P_n)R_{n+1} \{(I - P_n)v_n\}\|_2 \\ &\quad + \|((I - P_{n+1})Q_{n+1} - (I - P_n))R_{n+1} \{(I - P_n)v_n\}\|_2 + \zeta \\ &< \|(I - P_n)R_n \{(I - P_n)v_n\}\|_2 \\ &\quad + \left(2 \frac{\sqrt{2}\|\Delta_{n+1}\|_F}{1 - \|\Delta_{n+1}\|_2} \Delta + \left[\frac{\sqrt{2}\|\Delta_{n+1}\|_F}{1 - \|\Delta_{n+1}\|_2} \right]^2 (1 + \Delta) \right) \|(I - P_n)v_n\|_2 + \zeta \\ &\leq \|(I - P_n)R_n \{(I - P_n)v_n\}\|_2 + \delta^2 \|(I - P_n)v_n\|_2 + \zeta. \end{aligned} \quad (26)$$

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 $m_2 > m_1$,

$$|v_{m_2}^{(d)}| \leq \prod_{l=m_1}^{m_2-1} ((R_{dd})_l + \delta^2) |v_{m_1}^{(d)}| + \sum_{l=m_1+1}^{m_2} \zeta \prod_{j=l+1}^{m_2} ((R_{dd})_j + \delta^2). \quad (27)$$

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

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

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

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

We remark the first two terms of the above formula are the same as those in Eq.(27). This finishes the proof of the theorem.

□

Corollary 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 $m_1 = m_2 - \mathcal{N} - 1$.

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

From which it immediately follows that

$$|\overline{v_{m_2}^{(d)}}| < \frac{1}{1 - e^{(\hat{\lambda}_d + \hat{\epsilon}_d)\mathcal{N}}} \left(1 + \frac{1}{|\hat{\lambda}_d|} (1 - e^{\hat{\lambda}_d \mathcal{N}}) \right) \zeta. \quad (28)$$

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

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

From which it immediately follows that

$$|\overline{v_{m_2}^{(\kappa)}}| < \frac{1}{1 - e^{(\hat{\lambda}_\kappa + \hat{\epsilon}_\kappa)\mathcal{N}}} \left(1 + \frac{1}{|\hat{\lambda}_\kappa|} (1 - e^{\hat{\lambda}_\kappa \mathcal{N}}) \right) \left(\zeta + \sum_{\iota=\kappa+1}^d \Delta |\overline{v_{m_2}^{(\iota)}}| \right). \quad (29)$$

Combining Eqs. (27) and (29) and using that $\zeta = \mathcal{O}(\Delta)$, we conclude that

$$|\overline{v_{m_2}^{(\kappa)}}| < \frac{1}{1 - e^{(\hat{\lambda}_\kappa + \hat{\epsilon}_\kappa)\mathcal{N}}} \left(1 + \frac{1}{|\hat{\lambda}_\kappa|} (1 - e^{\hat{\lambda}_\kappa \mathcal{N}}) \right) \zeta + \mathcal{O}(\Delta^2). \quad (30)$$

□

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 $P_n \tilde{v} \in K_\alpha^u(\mathcal{X}_n)$, where $K_\alpha^u(\mathcal{X}_n)$ is the non stable cone [34, 7] of size α at \mathcal{X}_n , and $(I - P_n)\tilde{v} \in K_\alpha^s(\mathcal{X}_n)$, where $K_\alpha^s(\mathcal{X}_n)$ is the stable cone [34, 7] of size $\tilde{\alpha}$ at \mathcal{X}_n , then

$$\|v_{n+1}\| < \epsilon_2 + K_2 \epsilon_1^2 + 2\alpha(K_0 \epsilon_2 + K_2 \epsilon_2^2) + (\tilde{\lambda} + \delta) \epsilon_1. \quad (31)$$

Proof of Theorem 2.

Throughout this proof we use results of [34, 7, 26]. 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

$$\|v_{n+1} - F'_n(\mathcal{X}_n)v_n\| = \|F_n(\mathcal{X}_n) - F_n(\mathcal{X}_n - v_n) - F'_n(\mathcal{X}_n)v_n\| \leq K_2 \|v_n\|^2. \quad (32)$$

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 (32) we obtain

$$\|v_{n+1}\| < \epsilon_2 + K_2 \epsilon_1^2 + \|(I - P_{n+1})F'_n(\mathcal{X}_n)v_n\|. \quad (33)$$

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 $F'_n(\mathcal{X}_n)K_\alpha^u(\mathcal{X}_n) \subset \text{int}(K_\alpha^u(\mathcal{X}_{n+1})) \cup \{0\}$ (it follows for example from Proposition 5.4.1 of [7] or Lemma 6.2.10 of [34]). Hence we have that $F'_n(\mathcal{X}_n)P_n v_n \in \text{int}(K_\alpha^u(\mathcal{X}_{n+1})) \cup \{0\}$. 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 $\|F'_n(\mathcal{X}_n)P_n v_n\| \leq K_0 \|P_n v_n\| + K_2 \|P_n v_n\|^2$, where $K_0 = \sup |F_n(\mathcal{X}_n)|$. However, the only part of $F'_n(\mathcal{X}_n)P_n v_n$ of interest is the component $(I - P_{n+1})F'_n(\mathcal{X}_n)P_n v_n$. We have that

$$\|(I - P_{n+1})F'_n(\mathcal{X}_n)P_n v_n\| < 2\alpha \|F'_n(\mathcal{X}_n)P_n v_n\| < 2\alpha(K_0 \|P_n v_n\| + K_2 \|P_n v_n\|^2). \quad (34)$$

For the part $(I - P_n)v_n$ in the stable cone we can use [7] Proposition 5.4.2 or [34] 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 $(I - P_n)v_n \in K_{\tilde{\alpha}}^s(\mathcal{X}_n)$, then

$$\|F'_n(\mathcal{X}_n)(I - P_n)v_n\| < (\tilde{\lambda} + \delta) \|(I - P_n)v_n\|. \quad (35)$$

Collecting the estimates (33)–(35), we find that

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

□

Acknowledgments

Andrew Steyer and Erik Van Vleck acknowledge support from NSF grant DMS-1419047. Xuemin Tu acknowledges support from NSF grant DMS-1419069. Bart de Leeuw acknowledges this work is part of the research programme Mathematics of Planet Earth 2014 EW with project number 657.014.001, which is financed by the Netherlands Organisation for Scientific Research (NWO).

References

- [1] L. Y. ADRIANOVA, *Introduction to Linear Systems of Differential Equations*, vol. 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 (2012), pp. 37–59, <https://doi.org/10.1016/j.laa.2012.01.036>, <http://dx.doi.org/www2.lib.ku.edu/10.1016/j.laa.2012.01.036>.
- [3] 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 (1980), pp. 9–30.
- [4] M. BERLINER, *Likelihood and Bayesian prediction for chaotic systems*, J. Am. Stat. Assoc., 86 (1991), pp. 938–952.
- [5] W. BEYN, *On the numerical approximation of phase portraits near stationary-point*, SIAM Journal on Mathematical Analysis, 24 (1987), pp. 1095–1113, <https://doi.org/10.1137/0724072>.
- [6] S. BOCCALETIA, J. KURTHS, G. OSIPOV, D. VALLADARES, AND C. ZHOU, *The synchronization of chaotic systems*, Physics Reports, 366 (2002), pp. 1–101.
- [7] M. BRIN AND G. STUCK, *Introduction to Dynamical Systems*, Cambridge University Press, 2002.
- [8] J. BRÖCKER, *On variational data assimilation in continuous time*, Q.J.R. Meteorol. Soc., 136 (2010), pp. 1906–1919.
- [9] J. BRÖCKER AND U. PARLITZ, *Efficient noncausal noise reduction for deterministic time series*, Chaos, 11 (2001), pp. 319–326.
- [10] 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 (2008), p. 023112.
- [11] 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 (2008), pp. 503–521.
- [12] X.-W. CHANG, *On the perturbation of the q-factor of the qr factorization*, Numer. Linear Algebra Appl., (2012), pp. 607–619, <https://doi.org/10.1002/nla.787>.
- [13] S. CHOW, X. LIN, AND K. PALMER, *A shadowing lemma with applications to semilinear parabolic equations*, SIAM Journal on Mathematical Analysis, 20 (1989), pp. 547–557, <https://doi.org/10.1137/0520038>.

- [14] S.-N. CHOW AND K. J. PALMER, *On the numerical computation of orbits of dynamical systems: The higher dimensional case*, Journal of Complexity, 8 (1992), pp. 398 – 423, [https://doi.org/http://dx.doi.org/10.1016/0885-064X\(92\)90004-U](https://doi.org/http://dx.doi.org/10.1016/0885-064X(92)90004-U), <http://www.sciencedirect.com/science/article/pii/0885064X9290004U>.
- [15] L. DIECI AND C. ELIA, *The singular value decomposition to approximate spectra of dynamical systems. Theoretical aspects*, J. Differential Equations, 230 (2006), pp. 502–531, <https://doi.org/10.1016/j.jde.2006.08.007>, <http://dx.doi.org.www2.lib.ku.edu/10.1016/j.jde.2006.08.007>.
- [16] L. DIECI, C. ELIA, AND E. VAN VLECK, *Exponential dichotomy on the real line: SVD and QR methods*, J. Differential Equations, 248 (2010), pp. 287–308, <https://doi.org/10.1016/j.jde.2009.07.004>, <http://dx.doi.org.www2.lib.ku.edu/10.1016/j.jde.2009.07.004>.
- [17] L. DIECI, C. ELIA, AND E. VAN VLECK, *Detecting exponential dichotomy on the real line: SVD and QR algorithms*, BIT, 51 (2011), pp. 555–579, <https://doi.org/10.1007/s10543-010-0306-0>, <http://dx.doi.org.www2.lib.ku.edu/10.1007/s10543-010-0306-0>.
- [18] L. DIECI, R. RUSSELL, AND E. VAN VLECK, *On the Computation of Lyapunov Exponents for Continuous Dynamical Systems*, SIAM J. Numer. Anal., 34 (1997), pp. 402–423.
- [19] L. DIECI AND E. S. VAN VLECK, *On the Error in Computing Lyapunov Exponents by QR Methods*, Numer. Math., 101 (2005), pp. 619–642.
- [20] L. DIECI AND E. S. VAN VLECK, *Perturbation theory for approximation of Lyapunov exponents by QR methods*, J. Dynam. Differential Equations, 18 (2006), pp. 815–840, <https://doi.org/10.1007/s10884-006-9024-3>, <http://dx.doi.org.www2.lib.ku.edu/10.1007/s10884-006-9024-3>.
- [21] L. DIECI AND E. S. VAN VLECK, *On the error in QR integration*, SIAM J. Numer. Anal., 46 (2008), pp. 1166–1189, <https://doi.org/10.1137/06067818X>, <http://dx.doi.org.www2.lib.ku.edu/10.1137/06067818X>.
- [22] L. DIECI AND E. S. VAN VLECK, *Lyapunov exponents: Computation*, in Encyclopedia of Applied and Computational Mathematics, B. Engquist, ed., Springer-Verlag, 2015.
- [23] H. DU, *Combining Statistical Methods with Dynamical Insight to Improve Nonlinear Estimation*, PhD thesis, London School of Economics and Political Science, 2009.
- [24] H. DU AND L. A. SMITH, *Pseudo-Orbit Data Assimilation. Part I: The Perfect Model Scenario*, Journal of the Atmospheric Sciences, 71 (2014), pp. 469–482.
- [25] S. V. ERSHOV AND A. B. POTAPOV, *On the concept of stationary Lyapunov basis*, Physica D: Nonlinear Phenomena, 118 (1998), pp. 167 – 198, [https://doi.org/http://dx.doi.org/10.1016/S0167-2789\(98\)00013-X](https://doi.org/http://dx.doi.org/10.1016/S0167-2789(98)00013-X), <http://www.sciencedirect.com/science/article/pii/S016727899800013X>.
- [26] C. GONZÁLEZ-TOKMAN AND B. R. HUNT, *Ensemble data assimilation for hyperbolic systems*, Physica D: Nonlinear Phenomena, 243 (2013), pp. 128–142.
- [27] C. GREBOGI, S. M. HAMMEL, J. A. YORKE, AND T. SAUER, *Shadowing of physical trajectories in chaotic dynamics: Containment and refinement*, Phys. Rev. Lett., 65 (1990), pp. 1527–1530, <https://doi.org/10.1103/PhysRevLett.65.1527>, <http://link.aps.org/doi/10.1103/PhysRevLett.65.1527>.
- [28] S. HAMMEL, J. YORKE, AND C. GREBOGI, *Numerical orbits of chaotic processes represent true orbits*, Bulletin of the American Mathematical Society, 19 (1988), pp. 465–469, <https://doi.org/10.1090/S0273-0979-1988-15701-1>.

- [29] S. M. HAMMEL, J. A. YORKE, AND C. GREBOGI, *Do numerical orbits of chaotic dynamical processes represent true orbits?*, Journal of Complexity, 3 (1987), pp. 136 – 145, [https://doi.org/http://dx.doi.org/10.1016/0885-064X\(87\)90024-0](https://doi.org/http://dx.doi.org/10.1016/0885-064X(87)90024-0), <http://www.sciencedirect.com/science/article/pii/0885064X87900240>.
- [30] K. HAYDEN, E. OLSON, AND E. S. TITI, *Discrete data assimilation in the Lorenz and 2D Navier–Stokes equations*, Physica D: Nonlinear Phenomena, 240 (2011), pp. 1416 – 1425, <https://doi.org/http://dx.doi.org/10.1016/j.physd.2011.04.021>, <http://www.sciencedirect.com/science/article/pii/S016727891100114X>.
- [31] K. JUDD, C. A. REYNOLDS, T. E. ROSMOND, AND L. A. SMITH, *The geometry of model error*, Journal of the Atmospheric Sciences, 65 (2008), pp. 1749–1772.
- [32] K. JUDD AND L. SMITH, *Indistinguishable states I. Perfect model scenario*, Physica D, 151 (2001), pp. 125–141.
- [33] L. V. KANTOROVICH AND G. P. AKILOV, *Functional analysis in normed spaces*, Fizmatgiz, Moscow, 1959.
- [34] A. KATOK AND B. HASSELBLATT, *Introduction to the Modern Theory of Dynamical Systems*, vol. 54 of Encyclopedia of mathematics and its applications, Cambridge University Press, 1995.
- [35] K. LAW, D. SANZ-ALONSO, A. SHUKLA, AND A. STUART, *Controlling unpredictability with observations in the partially observed Lorenz '96 model*, ArXiv e-prints, (2014), <https://arxiv.org/abs/1411.3113>.
- [36] K. LAW, A. STUART, AND K. ZYGALAKIS, *Data assimilation : a mathematical introduction*, vol. 62 of Texts in applied mathematics, Springer, 2015, <https://doi.org/http://dx.doi.org/10.1007/978-3-319-20325-6>.
- [37] J. LEWIS AND J. DERBER, *The use of adjoint equations to solve a variational adjustment problem with advective constraint*, Tellus, 37A (1985), pp. 309–322.
- [38] E. N. LORENZ, *Deterministic Nonperiodic Flow*, Journal of Atmospheric Sciences, 20 (1963), pp. 130–148, [https://doi.org/10.1175/1520-0469\(1963\)020<0130:DNF>2.0.CO;2](https://doi.org/10.1175/1520-0469(1963)020<0130:DNF>2.0.CO;2).
- [39] E. N. LORENZ, *Predictability - a problem partly solved*, in Proceedings of seminar on Predictability, T. Palmer and R. Hagedorn, eds., vol. 1, Reading, UK, 1996, ECMWF, Cambridge University Press, pp. 1–18.
- [40] R. N. MILLER, M. GHIL, AND F. GAUTHIEZ, *Advanced data assimilation in strongly nonlinear dynamical systems*, Journal of the Atmospheric Sciences, 51 (1994), pp. 1037–1056.
- [41] 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 (2013), p. 254020, <http://stacks.iop.org/1751-8121/46/i=25/a=254020>.
- [42] L. M. PECORA AND T. L. CARROLL, *Synchronization in chaotic systems*, Physical Review Letters, 64 (1990), pp. 821–824.
- [43] L. M. PECORA AND T. L. CARROLL, *Driving systems with chaotic signals*, Physical Review A, 44 (1991), pp. 2374–2383.
- [44] L. M. PECORA, T. L. CARROLL, G. A. JOHNSON, D. J. MAR, AND J. F. HEAGY, *Fundamentals of synchronization in chaotic systems, concepts, and applications*, Chaos, 7 (1997), pp. 520–543.
- [45] C. PIRES, R. VAUTARD, AND O. TALAGRAND, *On extending the limits of variational assimilation in nonlinear chaotic systems*, Tellus, 48A (1996), pp. 96–121.

- [46] R. ROTUNNO AND J.-W. BAO, *A case study of cyclogenesis using a model hierarchy*, Monthly Weather Review, 124 (1996), pp. 1051–1066.
- [47] D. SANZ-ALONSO AND A. M. STUART, *Long-time asymptotics of the filtering distribution for partially observed chaotic dynamical systems*, SIAM/ASA Journal on Uncertainty Quantification, 3 (2015), pp. 1200–1220, <https://doi.org/10.1137/140997336>.
- [48] Y. SASAKI, *Some basic formalisms in numerical variational analysis*, Mon. Wea. Rev., 98 (1970), pp. 875–883.
- [49] L. A. SMITH, M. C. CUÉLLAR, H. DU, AND K. JUDD, *Exploiting dynamical coherence: A geometric approach to parameter estimation in nonlinear models*, Physics Letters A, 374 (2010), pp. 2618–2623.
- [50] C. SNYDER AND T. M. HAMILL, *Leading Lyapunov vectors of a turbulent baroclinic jet in a quasigeostrophic model*, Journal of the Atmospheric Sciences, 60 (2003), pp. 683–688.
- [51] O. TALAGRAND, *Assimilation of observations, an introduction*, Journal of the Meteorological Society of Japan, 75 (1997), pp. 191–209.
- [52] O. TALAGRAND AND P. COURTIER, *Variational assimilation of meteorological observations with the adjoint vorticity equation*, Q.J.R. Meteorol. Soc., 113 (1987).
- [53] 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 (2010), pp. 487–496, <https://doi.org/10.1002/qj.571>.
- [54] E. S. VAN VLECK, *Numerical Shadowing Using Componentwise Bounds and a Sharper Fixed Point Result*, SIAM J. Sci. Comp., 22 (2001), pp. 787–801.
- [55] E. S. VAN VLECK, *On the error in the product QR decomposition*, SIAM J. Matrix Anal. Appl., 31 (2009/10), pp. 1775–1791, <https://doi.org/10.1137/090761562>, <http://dx.doi.org/www2.lib.ku.edu/10.1137/090761562>.