Particle filters for data assimilation based on reduced order data models†

John Maclean[a]*, Erik S. Van Vleck[b]

[a]*School of Mathematical Sciences, University of Adelaide, South Australia.*

`http://www.adelaide.edu.au/directory/john.maclean`

[b]*Department of Mathematics, University of Kansas, USA.* `http://people.ku.edu/~erikvv/`

*Correspondence to: `John.Maclean@adelaide.edu.au`

We introduce a framework for Data Assimilation (DA) in which the data is split into multiple sets corresponding to low-rank projections of the state space. Algorithms are developed that assimilate some or all of the projected data, including an algorithm compatible with any generic DA method. The major application explored here is PROJ-PF, a projected Particle Filter. The PROJ-PF implementation assimilates highly informative but low-dimensional observations. The implementation considered here is based upon using projections corresponding to Assimilation in the Unstable Subspace (AUS). In the context of particle filtering, the projected approach mitigates the collapse of particle ensembles in high dimensional DA problems while preserving as much relevant information as possible, as the unstable and neutral modes correspond to the most uncertain model predictions. In particular we formulate and numerically implement a projected Optimal Proposal Particle Filter (PROJ-OP-PF) and compare to the standard optimal proposal and to the Ensemble Transform Kalman Filter.

Key Words: Data Assimilation, Numerical Analysis, Dimension Reduction

Received . . .

1. Introduction

Many data assimilation techniques were developed based on extending assumptions of linearity in the state space and data models and under the assumption of Gaussian errors. Several techniques have proven to be successful in weakening these assumptions, while other techniques have been developed to explicitly overcome these obstacles. Important among these are particle filters (Doucet *et al.* 2000), a key subject of this paper. Particle filters have proven to be successful for low dimensional assimilation problems but tend to have difficulty with higher dimensional problems. Different variants of particle filters have been developed to combat these difficulties, including implicit

†JM, ONR, grant: N00014-18-1-2204; ARC, grant DP180100050; EVV, NSF, DMS-1714195 and DMS-1722578.

particle filters, proposal density methods, the optimal proposal, etc. (Chorin *et al.* 2010; Snyder *et al.* 2008; van Leeuwen 2010; Snyder 2011; Morzfeld *et al.* 2012). Recent work has often focused on the issue of localization (Farchi and Bocquet (2018), e.g.), and two localised particle filtering algorithms (Poterjoy and Anderson 2016; Potthast *et al.* 2019) have been applied in an operational geophysical framework. In the localised particle filter of Potthast *et al.* (2019) observations are projected onto the subspace spanned by the ensemble of model forecasts to reduce the dimension of the observations.

Our contribution in this paper is to develop a framework for data assimilation schemes in which the data are constrained by an arbitrary projection to lie in some subspace of observation or model space. We explicitly obtain a form for the reduction in data dimension, and an expression that determines how much the posterior of the Bayesian DA scheme is affected by use of the projection. While the projection is not specified, the key idea is that some physically based reduction technique can then be employed in concert with a DA scheme. In such a way the assimilation step is performed in a space of very low dimension.

The derivation in this paper was motivated in large part by assimilation in the unstable subspace (AUS) techniques. These techniques have largely focused on projection in the tangent space of the nonlinear model using Lyapunov vectors while employing the original data or observational model. The techniques and framework developed in this paper allow for combinations of (time dependent) projected and unprojected physical and data models, and their formulation is independent of the source of the projections. The framework and techniques lead to several natural applications. In particular, we develop a new particle filter algorithm that makes use of the original, unprojected physical and observation models for the particle update together with a weight update employing the projected observation model and a resampling scheme that restricts perturbations to the projected space.

We now discuss the historical antecedents of the projections in this manuscript, and connect them to other recent filtering approaches. The AUS techniques (Carrassi *et al.* 2008a; Trevisan *et al.* 2010; Palatella *et al.* 2013) 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. 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 (Trevisan *et al.* 2010; Palatella *et al.* 2013). These techniques are based on updating in the unstable portion of the tangent space and may be interpreted in terms of projecting covariance matrices during the assimilation step. Motivated by these techniques for assimilation in the unstable subspace, in de Leeuw *et al.* (2018) a new method is developed for data assimilation that utilizes distinct treatments of the dynamics in the stable and non-stable directions. The key piece of de Leeuw *et al.* (2018) related to this work is the following projected model update. For a smooth discrete time model $u_{n+1} = F_n(u_n)$ and projection Π_n, and for $\{u_n^{(0)}\}_{n=0}^N$ any reference solution, solve for $\{d_n\}_{n=0}^N$:

$$u_{n+1}^{(0)} + d_{n+1} = \Pi_{n+1} F_n(u_n^{(0)} + d_n), \;\; n = 0, ..., N-1. \quad (1)$$

Unlike most past work related to AUS our primary focus is on developing a systematic approach to confining the data, not the model, to the unstable subspace. In some of the initial works on AUS (Carrassi *et al.* 2007, 2008b), either target observations at the location where the unstable mode attains its maximum value, or only the observations falling in the vicinity of the maximum, were assimilated. Albeit empirical, that choice already signified using only data projected on an approximation of the unstable subspace, that was obtained by Breeding on the Data Assimilation Cycle (BDAS). Furthermore, González-Tokman and Hunt (2013); Bocquet *et al.* (2017); Grudzien *et al.* (2018b,a); Frank and Zhuk (2018); Reddy *et al.* (2020); Tranninger *et al.* (2020) are all at least in part devoted to discussing the necessary and/or sufficient criteria for filter stability in terms of the projection of the observations into the unstable/neutral/weakly stable directions

and this is directly related to the choice of adaptive observation operators in Law *et al.* (2016).

Another branch of projected DA schemes use the 'Dynamically Orthogonal' (DO) formulation (Sapsis and Lermusiaux 2009; Sapsis 2010), in which the forecast model is broken into a partial differential equation governing the mean field and a number of stochastic differential equations describing the evolution of components in a time-dependent stochastic subspace of the original differential equation. The DO approach was used to assimilate with different DA schemes in the subspace and mean field space in Sondergaard and Lermusiaux (2013); Majda *et al.* (2014); Qi and Majda (2015). These techniques use both a projected and mean field model to make a forecast, similar to using (1).

Projection-based DA schemes have been developed to assimilate coherent structures (Maclean *et al.* 2017) or features (Morzfeld *et al.* 2018) in the data. These approaches have used likelihood-free sequential Monte Carlo methods, or an ad hoc 'perturbed observations' approach, to deal with the difficulty of calculating the likelihood function for a coherent structure. The derivation in this paper may lead to an explicit likelihood for data-derived coherent structures/features obtained via a projection.

This paper is organized as follows. Data assimilation is reviewed in section 2 and projected DA is formulated in section 3. Algorithms for using the new projected data are introduced (section 4) and applied in several numerical experiments (section 5). A discussion (section 7) and bibliography conclude the paper.

2. Data Assimilation

Data assimilation methods combine orbits from a dynamical system model with measurement data to obtain an improved estimate for the posterior probability density function (pdf) of a physical system. In this paper we develop a data assimilation method in the context of the discrete time stochastic model

$$u_{n+1} = F_n(u_n) + \sigma_n, \;\; n = 0, 1, ... \tag{2}$$

where $u_n \in \mathbb{R}^N$ are the state variables at time n and $\sigma_n \sim \mathcal{N}(0, \mathbf{Q})$, i.e., drawn from a normal distribution with mean zero and model error covariance $\mathbf{Q}$. Let the sequence $\{u_0^t, u_1^t, \dots\}$, be a distinguished orbit of this system, referred to as the *true solution* of the model, and presumed to be unknown. As each time t_n is reached we collect an observation y_n related to u_n^t via

$$y_n = \mathbf{H} u_n^t + \eta_n, \qquad y_n \in \mathbb{R}^M \tag{3}$$

where $\mathbf{H} : \mathbb{R}^N \to \mathbb{R}^M$, $M \leq N$, is the observation operator, and the noise variables η_n are drawn from a normal distribution $\eta_n \sim \mathcal{N}(0, \mathbf{R})$ with zero mean and known observational error covariance matrix $\mathbf{R}$. In general the observation operator can be nonlinear.

We formulate DA under the ubiquitous Bayesian approach. Consider the assimilation of a single observation, y_n, at time step n. Given a prior estimate $p(u_n)$ of the state, Bayes' Law gives $p(u_n|y_n) \propto p(y_n|u_n)p(u_n)$. Using (3) the likelihood function is, up to a normalization constant,

$$p(y_n|u_n) \propto \exp\left[-\frac{1}{2}\left(y_n - \mathbf{H}u_n\right)^T \mathbf{R}^{-1}\left(y_n - \mathbf{H}u_n\right)\right]. \tag{4}$$

This procedure, which we have written for the assimilation of data at a single observation time, readily extends to the sequential assimilation of observations at multiple times under the assumptions that the state is Markovian and the observation errors at different times are conditionally independent (see for example Budhiraja *et al.* (2017)).

In the following we introduce some key DA schemes. Not much detail is given here, but the interested reader is referred in particular to three recent books on DA, (Reich and Cotter 2015; Law *et al.* 2015; Asch *et al.* 2016).

2.1. Kalman Filtering

The Kalman Filter and later extensions are ubiquitous in DA, and are now briefly described. For a linear model, i.e. where (2)

is

$$u_{n+1} = \mathbf{A}_n u_n + \sigma_n, \tag{5}$$

and for the linear observation operator $\mathbf{H}$, the Kalman Filter calculates the exact posterior $u_n|y_n \sim \mathcal{N}(u_n^a, \mathbf{P}_n^a)$, where the *analysis* variables are

$$u_n^a = u_n^f + \mathbf{K}_n(y_n - \mathbf{H}u_n^f)\ , \tag{6}$$

$$\mathbf{P}_n^a = (\mathbf{I} - \mathbf{K}_n\mathbf{H})\,\mathbf{P}_n^f. \tag{7}$$

The weight matrix $\mathbf{K}_n$ is the Kalman gain matrix

$$\mathbf{K}_n = \mathbf{P}_n^f\mathbf{H}^T\left(\mathbf{H}\mathbf{P}_n^f\mathbf{H}^T + \mathbf{R}\right)^{-1}\ . \tag{8}$$

The superscript f is reserved for *forecast* variables, obtained at time n by using (5) to update $\{u_{n-1}^a, \mathbf{P}_{n-1}^a\}$,

$$u_n^f = \mathbf{A}_{n-1}u_{n-1}^a + \sigma_{n-1}\ , \tag{9}$$

$$\mathbf{P}_n^f = \mathbf{A}_{n-1}\mathbf{P}_{n-1}^a\mathbf{A}_{n-1}^T + \mathbf{Q}\ . \tag{10}$$

Two extensions of the Kalman Filter are prevalent in nonlinear DA, the Extended Kalman Filter (EKF) and Ensemble Kalman Filter (EnKF). Neither give the exact posterior for a nonlinear model.

2.1.1. *Extended Kalman Filter*

The nonlinear model (2) is used to make the forecast u_n^f, and then the update of the covariance (10) is applied using the linearisation

$$\mathbf{A}_n = \left.\frac{\partial F_n}{\partial u}\right|_{u_n^a}\ . \tag{11}$$

If the observation operator is a nonlinear function $h()$, the linearization

$$\mathbf{H}_n = \left.\frac{dh}{du}\right|_{u_n^f} \tag{12}$$

is used everywhere except to compute the *innovation* $y_n - h(u_n^f)$ in the calculation of y_n^a.

The EKF is suitable for low dimensional nonlinear filtering, but the required linearizations are nontrivial for high-dimensional filtering. The EnKF by contrast is well suited to high dimensions.

2.1.2. *Ensemble Kalman Filter*

The Ensemble Kalman Filter is a Monte Carlo approximation of the Kalman Filter that is well suited to high dimensional filtering problems, introduced in Evensen (1994); Burgers *et al.* (1998). An ensemble of forecasts $u_n^{f,i}$ are made at time t_n, i from 1 to L. Then the forecast covariance $\mathbf{P}_n^f$ is approximated by the sample covariance of the ensemble, and the analysis ensemble $u_n^{a,i}$ is obtained in such a way that its mean $\bar{u}_n^a = \frac{1}{L}\sum_i u_n^{a,i}$ satisfies (6) and its sample covariance satisfies (7). In this paper we will use analysis updates corresponding to the Ensemble Transform Kalman Filter (ETKF) (Bishop *et al.* 2001). For more details and a modern introduction to the Ensemble Kalman Filter, see e.g. Evensen (2009) and Carrassi *et al.* (2018).

2.2. *The Particle Filter*

Particle Filters (PF) are a collection of particle based data assimilation schemes that do not rely on linearization of the dynamics or Gaussian representations of the posterior; see Doucet *et al.* (2001) for a comprehensive review. The basic idea is to represent the prior distribution $p(u_n)$, previously the forecast, and the posterior distribution $p(u_n|y_n)$, previously the analysis, by discrete probability measures. Suppose that at time $n-1$ we have the posterior distribution (u_{n-1}^i, w_{n-1}^i), supported on points $u_{n-1}^1, \ldots u_{n-1}^L$ and with weights $w_{n-1}^1, \ldots w_{n-1}^L$. Each $w_{n-1}^i \geq 0$ and $\sum_{i=1}^L w_{n-1}^i = 1$. Here L is the number of particles that are used to approximate the distribution. The two key steps in the Particle Filter are as follows:

Prediction step. Propagate each of the particles $u_{n-1}^i \mapsto u_n^i$. One simple choice, the bootstrap PF, is to use the state dynamics (2) to forecast each particle.

This gives the forecast probability distribution as a discrete probability measure concentrated on L points $\{u_n^i\}_{i=1}^L$ with weights $\{w_{n-1}^i\}_{i=1}^L$.

Filtering step. Update the weights $\{w_{n-1}^i\}_{i=1}^L$ using the observation y_n. In the bootstrap PF the update is

$$w_n^i = c\, w_{n-1}^i p(y_n|u_n^i), \tag{13}$$

where c is chosen so that $\sum_{i=1}^{L} w_n^i = 1$.

This scheme is easy to implement but suffers from severe degeneracy, especially for high dimensional observations: even ten independent observations are sufficient to produce degenerate weights. That is, after a few time steps all the weight tends to concentrate on a few particles. A partial remedy is to monitor the Effective Sample Size (ESS) and resample when the ESS drops below some threshold in order to refresh the particle cloud; see e.g. Doucet *et al.* (2001); Budhiraja *et al.* (2017).

2.2.1. The Optimal Proposal

The optimal proposal particle filter (OP-PF) (Snyder *et al.* 2008; Doucet *et al.* 2000; Snyder 2011; Van Leeuwen 2012, e.g.) attempts to address the degeneracy issue in particle filters with the aim of ensuring that all posterior particles have similar weights. The 'proposal' is the distribution used to update the particles from one time step to the next. In the prediction step in the basic particle filter above, the particles are updated using the model, so the proposal density in that approach is (compare (2)) $u_n^i | u_{n-1}^i \sim \mathcal{N}(F_{n-1}(u_{n-1}^i), \mathbf{Q})$.

The optimal proposal density is $p(u_n^i | u_{n-1}^i, y_n)$. Given the additive noise of the model (2) and a linear observation operator $\mathbf{H}$, the optimal proposal update in each particle is Gaussian with $u_n^i | u_{n-1}^i, y_n \sim \mathcal{N}(m_n^i, \mathbf{Q}_p)$, and we obtain the explicit update

$$u_n^i = m_n^i + \phi, \;\; \phi \sim \mathcal{N}(0, \mathbf{Q}_p) \tag{14}$$

where

$$m_n^i = F_{n-1}(u_{n-1}^i) + \mathbf{Q}\mathbf{H}^T \left(\mathbf{H}\mathbf{Q}\mathbf{H}^T + \mathbf{R} \right)^{-1} I_n^i \,, \tag{15}$$

$$\mathbf{Q}_p^{-1} = \mathbf{Q}^{-1} + \mathbf{H}^T \mathbf{R}^{-1} \mathbf{H} \,, \tag{16}$$

and $I_n^i := y_n - \mathbf{H} F_{n-1}(u_{n-1}^i)$. The mean of the particle m_n^i is obtained by a Kalman filter step, albeit with $\mathbf{P}_n^f = \mathbf{Q}$.

Two applications of Bayes' law (e.g. in Snyder (2011)) show that the weight update for the i-th particle drawn from this proposal satisfies $w_n^i \propto p(y_n | u_{n-1}^i) w_{n-1}^i$ with a Gaussian likelihood function,

$$w_n^i \propto \exp\left[-\frac{1}{2} (I_n^i)^T \left(\mathbf{H}\mathbf{Q}\mathbf{H}^T + \mathbf{R} \right)^{-1} (I_n^i) \right] w_{n-1}^i \,. \tag{17}$$

As mentioned in the previous section, degeneracy - characterised by a single particle with weight of approximately 1 - is a common problem in the PF. In Snyder *et al.* (2015) it is shown that, of all PF schemes that obtain u_n^i using u_{n-1}^i and y_n, the 'optimal proposal' above has the minimum variance in the weights. That is, it suffers the least from weight degeneracy. In van Leeuwen *et al.* (2019) this result is extended to any PF scheme that obtains u_n^i using $u_{n-1}^{1:L}$ and y_n.

However, in Snyder (2011) it is shown that the optimal proposal requires an ensemble size L satisfying $\log L \propto N \times M$ for a linear model, or will suffer from filter degeneracy. That is, filter degeneracy is intimately connected to model and observation dimension, and is a fundamental obstacle to Particle Filtering in high dimensional problems.

3. Projected Data Models

We now develop an approach to decompose the observations using projections defined in state space. A wealth of techniques from dynamical systems theory can then be used to obtain low-dimensional data models.

Suppose that at time n a dynamically significant rank p orthogonal projection $\Pi_n \in \mathbb{R}^{N \times N}$ is available, as well as data $y_n \in \mathbb{R}^M$. We next derive a projected data model consisting of a projected obeservation operator, projected observation, and associated observation error covariance.

Step One: lift the data into model space

In order to apply the projection Π_n to data, we first need to find an equivalent representation of the data in model space.

Assuming $\mathbf{H}$ has full row rank, we define an N-dimensional vector $\tilde{y}_n = \mathbf{H}^\dagger y_n$ where $\mathbf{H}^\dagger = \mathbf{H}^T (\mathbf{H}\mathbf{H}^T)^{-1}$, the pseudo inverse of $\mathbf{H}$. The data model for $\tilde{y}_n$ is

$$\tilde{y}_n = \mathbf{H}^\dagger y_n = \Pi_{\mathbf{H}} u_n^t + \mathbf{H}^\dagger \eta_n = \Pi_{\mathbf{H}} u_n^t + \psi_n \tag{18}$$

where $\Pi_{\mathbf{H}} = \mathbf{H}^\dagger\mathbf{H}$ is an orthogonal projection, and $\psi_n \sim \mathcal{N}(0, \mathbf{H}^\dagger\mathbf{R}(\mathbf{H}^\dagger)^T)$.

Using that $\mathbf{H}\mathbf{H}^\dagger = \mathbf{I}$ one readily confirms that $\mathbf{H}\tilde{y}_n = y_n = \mathbf{H}u_n^t + \eta_n$. That is, the observation operator collapses $\tilde{y}_n$ onto the standard data model. The transformation through $\mathbf{H}^\dagger$ has not affected the output of a DA scheme, as $p(\tilde{y}_n|x) = p(y_n|x)$; however $\tilde{y}_n$ is of compatible dimension with Π_n.

Step Two: project the data into a rank p subspace

We now make use of the orthogonal projection Π_n. The idea is to formulate a new data model, along the lines of $\Pi_n\tilde{y}_n = \Pi_n\Pi_{\mathbf{H}}u_n^t + \Pi_n\mathbf{H}^\dagger\eta_n$, that contains only the components of the observation that align with the projection. The projected data models that are developed here may be considered as generalizations of the construction of observation operators (see Grudzien *et al.* (2018a) Def. 13 and Law *et al.* (2016)).

Define $y_n^p = \Pi_n\tilde{y}_n = \Pi_n\mathbf{H}^\dagger y_n \in \mathbb{R}^N$, the projected observation. The data model is

$$y_n^p = \Pi_n\mathbf{H}^\dagger y_n = \Pi_n\Pi_{\mathbf{H}}u_n^t + \xi_n \tag{19}$$

where $\xi_n \sim \mathcal{N}(0, \Pi_n\mathbf{H}^\dagger\mathbf{R}(\mathbf{H}^\dagger)^T\Pi_n)$. The data model y_n^p has a singular normal distribution with support in the p-dimensional subspace of model space spanned by the projection Π_n. Some information from the observations is typically lost in this step by applying the projection. At the end of Section 3 we derive a data model for the orthogonal data (that is discarded in (19)), and in Section 4.1 we sketch a result that establishes the difference between assimilating with the full data (3) and with the projected data.. The likelihood $y_n^p|u$ of this distribution has an explicit form using the pseudo-inverse (see e.g. Tsukuma and Kubokawa (2015)) as

$$p(y_n^p|u) \propto \exp\left(-\frac{1}{2}(I_n^p)^T\left(\Pi_n\mathbf{H}^\dagger\mathbf{R}(\mathbf{H}^\dagger)^T\Pi_n\right)^\dagger I_n^p\right) \tag{20}$$

where $I_n^p := y_n^p - \Pi_n\Pi_{\mathbf{H}}u$.

Remark 1. *The product $\Pi_n\Pi_{\mathbf{H}}$ is not generally an orthogonal projection, and in some circumstances it might be desired to instead identify the projection $\Pi_n^{\mathbf{H}}$ that is the intersection of Π_n and $\Pi_{\mathbf{H}}$. This projection $\Pi_n^{\mathbf{H}}$ may be approximated by Von Neumann's algorithm or Dykstra's projection algorithm. The projection $\Pi_n^{\mathbf{H}}$ should only be used if the transversality condition $p + M - N > 0$ is satisfied; otherwise there is no guarantee of any intersection between Π_n and $\Pi_{\mathbf{H}}$. If the transversality condition is satisfied, e.g., with a high dimensional observation space, then we can replace the product of projections $\Pi_n\Pi_{\mathbf{H}}$ with the projection into the intersection $\Pi_n^{\mathbf{H}}$. Since in most applications this will not be satisfied, throughout the rest of the paper we focus on the use of the product of projections.*

Step Three: reduce the projected data to a p-vector

To make explicit the reduction in the data dimension that has been obtained by y_n^p we introduce a low dimensional data model. Denote by $\mathbf{U}_n$ the matrix with orthonormal columns satisfying $\Pi_n = \mathbf{U}_n\mathbf{U}_n^T$. This matrix may be already known (in the examples in Section 3 $\mathbf{U}_n$ is obtained first, and then Π_n is calculated from $\mathbf{U}_n\mathbf{U}_n^T$), or $\mathbf{U}_n$ may be found via the singular value or Schur decompositions.

Define $y_n^q = \mathbf{U}_n^T y_n^p \equiv \mathbf{U}_n^T\tilde{y}_n \in \mathbb{R}^p$, with the associated data model

$$y_n^q = \mathbf{H}_n^q u_n^t + \gamma_n\ , \tag{21}$$

where $\mathbf{H}_n^q = \mathbf{U}_n^T\Pi_{\mathbf{H}}$, $\gamma_n \sim \mathcal{N}(0, \mathbf{R}_n^q)$, and $\mathbf{R}_n^q = \mathbf{U}_n^T\mathbf{H}^\dagger\mathbf{R}(\mathbf{H}^\dagger)^T\mathbf{U}_n$.

The transformations between, and dimensions of, the different data variables defined in this section are illustrated in Figure 1.

3.1. Properties of the projected data

Theorem 3.1 (Equivalence of y_n^p and y_n^q)**.** *For the data models associated with y_n^p and y_n^q given by (19) and (21), respectively, $p(y_n^q|u) = p(y_n^p|u)$.*

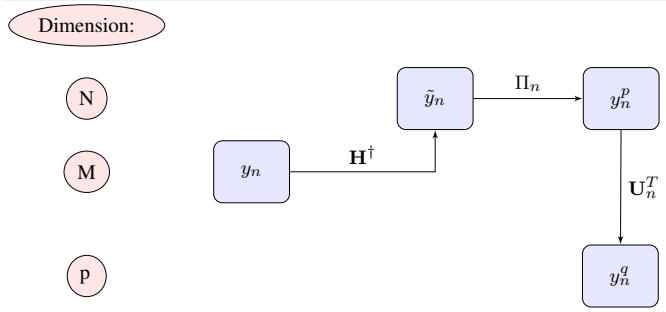

Figure 1. The progression from the original data y_n to low-dimensional, projected data y_n^q. The rectangular boxes contain data, or data-derived constructs. The vertical placement of each box corresponds to the dimension of the data at each step: $N \geq M \geq p$. Note that in practice one does not need to compute $\tilde{y}_n$ or y_n^p.

Proof. The matrix $\mathbf{U}_n$ has orthonormal columns, so $\mathbf{U}_n^\dagger = \mathbf{U}_n^T$ and for any matrix $\mathbf{B}$

$$(\mathbf{U}_n\mathbf{B})^\dagger = \mathbf{B}^\dagger\mathbf{U}_n^\dagger = \mathbf{B}^\dagger\mathbf{U}_n^T \,, \tag{22}$$

$$\left(\mathbf{B}\mathbf{U}_n^T\right)^\dagger = (\mathbf{U}_n^T)^\dagger\mathbf{B}^\dagger = \mathbf{U}_n\mathbf{B}^\dagger \,. \tag{23}$$

Applying these results to (20), and using that $\Pi_n = \mathbf{U}_n\mathbf{U}_n^T$, $y_n^p = \mathbf{U}_n y_n^q$, $\mathbf{U}_n^T\mathbf{U}_n = \mathbf{I}$, $I_n^p := y_n^p - \Pi_n\Pi_{\mathbf{H}}u$, and $I_n^q := y_n^q - \mathbf{U}_n^T\Pi_{\mathbf{H}}u$,

$$\begin{aligned} p(y_n^p|u) = &\ \tfrac{1}{c}\exp\left(-\tfrac{1}{2}(I_n^p)^T\left(\Pi_n\mathbf{H}^\dagger\mathbf{R}(\mathbf{H}^\dagger)^T\Pi_n\right)^\dagger I_n^p\right) \\ = &\ \tfrac{1}{c}\exp\left(-\tfrac{1}{2}\left(\mathbf{U}_n I_n^q\right)^T(\tilde{\mathbf{R}}_n)^\dagger\left(\mathbf{U}_n I_n^q\right)\right) \\ = &\ \tfrac{1}{c}\exp\left(-\tfrac{1}{2}(I_n^q)^T\mathbf{U}_n^T\mathbf{U}_n\left(\mathbf{R}_n^q\right)^\dagger\mathbf{U}_n^T\mathbf{U}_n I_n^q\right) \\ = &\ \tfrac{1}{c}\exp\left(-\tfrac{1}{2}(I_n^q)^T\left(\mathbf{R}_n^q\right)^\dagger I_n^q\right) \\ = &\ p(y_n^q|u) \end{aligned} \tag{24}$$

where $\tilde{\mathbf{R}}_n := \mathbf{U}_n\mathbf{U}_n^T\mathbf{H}^\dagger\mathbf{R}(\mathbf{H}^\dagger)^T\mathbf{U}_n\mathbf{U}_n^T$ and c is a normalising constant. □

If in addition $p \leq M$ (or $0 < p + M - N \leq M$ for $\Pi_n \equiv \Pi_n^{\mathbf{H}}$), and if $\mathbf{H}\mathbf{U}_n$ is full rank, then the covariance matrix $\mathbf{R}_n^q$ of y_n^q is invertible and y^q has a standard normal distribution. More generally for $(\mathbf{H}\mathbf{H}^T)^{-1}\mathbf{R}(\mathbf{H}\mathbf{H}^T)^{-1} = \mathbf{L}^T\mathbf{L}$, the Cholesky factorization, consider the SVD of $\mathbf{L}\mathbf{H}\mathbf{U}_n = \mathbf{S}\mathbf{\Sigma}\mathbf{V}^T$. The rank of the covariance matrix $\mathbf{R}_n^q = \mathbf{U}_n^T\mathbf{H}^\dagger\mathbf{R}(\mathbf{H}^\dagger)^T\mathbf{U}_n = \mathbf{V}\mathbf{\Sigma}^T\mathbf{\Sigma}\mathbf{V}^T$ is equal to the number of non-zero singular values of $\mathbf{\Sigma}$.

Theorem 3.1 provides a blueprint for any DA system with a linear observation operator to be efficiently implemented with projected observations, involving the following changes: the observation y_n is replaced with y_n^q, the observation operator $\mathbf{H}$ is replaced with $\mathbf{H}_n^q$, and the assumed measurement covariance $\mathbf{R}$ is replaced with $\mathbf{R}_n^q$.

3.2. The orthogonal data model

Though the focus of this paper is on the projected data, a data model for the complementary orthogonal projection $\mathbf{I} - \Pi_n$ is easy to write down. Define

$$y_n^{q\perp} = \left(\mathbf{U}_n^\perp\right)^T\tilde{y}_n \in \mathbb{R}^{N-p} \,, \tag{25}$$

where $\mathbf{U}_n^\perp(\mathbf{U}_n^\perp)^T = \mathbf{I} - \Pi_n$. The two projected data models are not independent in general and have joint distribution

$$\begin{bmatrix} y_n^q \\ y_n^{q\perp} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{H}_n^q\, u_n^t \\ \mathbf{H}_n^{q\perp} u_n^t \end{bmatrix}, \begin{bmatrix} \mathbf{R}_n^q & \mathbf{R}_{12,n}^q \\ \mathbf{R}_{21,n}^q & \mathbf{R}_n^{q\perp} \end{bmatrix}\right) , \tag{26}$$

where $\mathbf{H}^{q\perp} = (\mathbf{U}_n^\perp)^T\Pi_{\mathbf{H}}$, $\mathbf{R}^{q\perp} = (\mathbf{U}_n^\perp)^T\mathbf{H}^\dagger\mathbf{R}(\mathbf{H}^\dagger)^T\mathbf{U}_n^\perp$, and the off-diagonal covariances are $\mathbf{R}_{12,n}^q = \mathbf{U}_n^T\mathbf{H}^\dagger\mathbf{R}(\mathbf{H}^\dagger)^T\mathbf{U}_n^\perp$ and $\mathbf{R}_{21,n}^q = \left(\mathbf{R}_{12,n}^q\right)^T$.

The joint distribution (26) is not used in this manuscript. The cross-covariance term $\mathbf{R}_{12,n}^q$ measures the information about the projected subspace that is lost by not assimilating the orthogonal component $y^{q\perp}$. In ongoing work we are developing approaches to factorise the posterior into two components, via (26), and apply different DA methods to each component, incorporating the cross-covariance terms.

4. Algorithms for Projected DA

In this section we discuss how some combination of the standard/projected forecast models (2), (1) and data models (3), (21), (25)–(26) may be used to form a 'projected DA scheme'.

A projected data model changes the innovation, the observation operator, and the observation error covariance. A projected physical model changes the prior and model error covariances. We want combinations of physical models, data models, and DA techniques that optimize the assimilation, particularly of the Particle Filtering schemes discussed in Section 2.2.

We identify the following approaches to assimilating with projected data using the results of this paper:

Algorithm 1 (Project data only, and discard the orthogonal component)**.** *Apply a standard DA scheme using the unprojected forecast model* (2)*, but replace the standard data* (3) *with the projected data* y_n^q *of* (21)*. The observation operator is replaced by* $\mathbf{H}_n^q$*, and the covariance matrix of the observations is replaced by* $\mathbf{R}_n^q$*.*

By careful construction, the unique component of 'Algorithm 1' is actually a modified Bayesian posterior. This construction of a modified DA scheme as a modification to the posterior enables modularity: any DA scheme suitable for nonlinear filtering may be implemented with projected data models. In particular, this projected DA algorithm is fully compatible with localized DA schemes.

PROJ-PF uses the standard forecast model (2) to update the particles, but computes the weight update with

$$w_n^i \propto \exp\left[-\frac{1}{2}\left(y_n^q - \mathbf{H}_n^q u_n\right)^T \left(\mathbf{R}_n^q\right)^{-1} \left(y_n^q - \mathbf{H}_n^q u_n\right)\right] w_{n-1}^i \,. \tag{27}$$

Another algorithm to be described is a novel, efficient PF scheme taking advantage of the Optimal Proposal PF described in section 2.2.1.

Algorithm 2 (PROJ-OP-PF: Blend projected and unprojected data in the assimilation step)**.** *This algorithm describes a Particle Filter, PROJ-OP-PF, that uses the typical optimal proposal equations* (14)–(15) *for the particle update. The weight update for each particle is computed using the projected data model only, i.e. using the projected form of* (17)*,*

$$w_n^i \propto \exp\left[-\frac{1}{2}(I_n^q)^T \left(\mathbf{H}_n^q \mathbf{Q} (\mathbf{H}_n^q)^T + \mathbf{R}_n^q\right)^{-1} (I_n^q)\right] w_{n-1}^i \,. \tag{28}$$

where $I_n^q \equiv I_n^q(u_{n-1}^i) := y_n^q - \mathbf{H}_n^q F_{n-1}(u_{n-1}^i)$.

Algorithm 2 uses the original observation error covariance and the original observation operator for the particle update but employs the projected observation error covariance and the projected observation operator for the weight update. This strategy will be tested on the chaotic Lorenz-96 system in Section 6. One major advantage of this approach is that it requires no modification of the numerical simulation used to obtain the forecast. A second advantage is its efficiency; the full data are used for the particle update step, over which the update is straightforward and the dimension of the data does not lead to filter degeneracy; and only the projected data are used to avoid filter degeneracy in the weight update step. The scheme will prove to be more accurate than either, OP-PF or an Algorithm 1 implementation of OP-PF, in numerical tests.

Particle Filters can benefit from adding noise on resampling, particularly with deterministic forecast models. The correct way to do this is to generate noise sampled from a Markov chain that leaves the target pdf unchanged, see e.g. Doucet *et al.* (2000). For example, one can implement an accept-reject step for the particle. For simplicity we either do not add noise, or add noise sampled from $\mathcal{N}(\mathbf{0}, \omega^2\mathbf{I})$, where $\omega \in \mathbb{R}$ must be tuned. We consider an algorithm for projecting the noise on resampling.

Algorithm 3 (PROJ-RESAMP: Resampling in the Unstable Subspace)**.** *When adding noise* ω *to particles after resampling, multiply this random vector by* $\alpha\Pi_n + (1-\alpha)\mathbf{I}$ *for some* $\alpha \in [0, 1]$.

When $\alpha = 0$ this algorithm is no different to the normal resampling approach, but for $\alpha > 0$ some proportion of the uncertainty in resampling is constrained to lie in the space spanned by the columns of $\mathbf{U}_n$. If employed in concert with an accept-reject step, this may improve the chances of acceptance. For AUS the resampling scheme should add more noise in the directions of greatest uncertainty in the forecast model, which provides one advantage; a second advantage is that the algorithm does not shift particles as far off the attractor.

4.1. Convergence results for projected algorithms

A normal line of inquiry for a new DA algorithm is to quantify the conditions under which it will well represent the posterior distribution, which neglecting time subscripts we write as $p(u|y)$. The projected algorithms above do not generally converge to $p(u|y)$, and so there are two questions: 'Does the algorithm

converge to a known distribution?', and 'How different is that distribution to the actual posterior?'.

Algorithm 1 clearly implements an approximation of the distribution $p(u|y^q)$. That is, a Particle Filter implementation would converge to $p(u|y^q)$ in the limit as the number of particles approaches infinity. The distribution approximated by Algorithm 2 is a blending of $p(u|y)$ and $p(u|y^q)$ that is non-trivial to obtain in closed form.

We now quantify how the Algorithm 1 distribution $p(u|y^q)$ relates to the standard posterior $p(u|y)$. For this we will employ the Hellinger distance: given two probability measures μ and μ', with associated probability distributions ρ and ρ', the Hellinger distance between the two is

$$d_H(\mu,\mu') = \left[\frac{1}{2}\int\left(\sqrt{\rho(u)}-\sqrt{\rho'(u)}\right)^2 du\right]^{1/2} . \tag{29}$$

To bound this distance for Algorithm 1 we write $\rho(u) = p(u|y)$ and $\rho'(u) = p(u|y^q)$. The second distribution is written as

$$p(u|y^q) = p(u|y)\frac{p(y|y^q)}{p(y|u,y^q)} , \tag{30}$$

obtained via Bayes' law in the form $p(u) = p(u|y)\,p(y)/p(y|u)$, conditioning on y^q, and using $p(u|y,y^q) = p(u|y)$. Factorising both numerator and denominator, we obtain the final form

$$p(u|y^q) = p(u|y)\frac{p(y^{q\perp}|y^q)}{p(y^{q\perp}|u,y^q)} . \tag{31}$$

Substituting into (29) we obtain a bound for the consistency of Algorithm 1 with the original posterior $p(u|y)$,

$$\begin{aligned} d_H(\mu,\mu') &= \left[\frac{1}{2}\int\left(1-\sqrt{\frac{p(y^{q\perp}|y^q)}{p(y^{q\perp}|u,y^q)}}\right)^2 \rho(u)\,du\right]^{1/2} \\ &= \left[\frac{1}{2}\mathbb{E}^{\mu}\left(1-\sqrt{\frac{p(y^{q\perp}|y^q)}{p(y^{q\perp}|u,y^q)}}\right)^2\right]^{1/2} . \end{aligned} \tag{32}$$

An intuitive example of the above bounds in practice is a slow-fast system with a slow manifold onto which the fast variables are attracted. Choosing Π_n to identify the slow variables will lead to a small value of $d_H(\mu,\mu')$ for Algorithm 1 data assimilation schemes, since knowledge of the slow variables is sufficient to constrain the fast variables. In the case where there are few slow variables and many fast variables, then, an Algorithm 1 Particle Filter will be a much less degenerate implementation of the Particle Filter that converges close to the desired posterior $p(u|y)$. The bounds (29)–(32) at present help to build intuition rather than providing a practical tool: they only apply at a single assimilation time, and it is difficult to infer a bound for the difference between the Algorithm 1 posterior and the standard posterior over multiple time steps.

5. Application: Assimilation in the Unstable Subspace

For the remainder of the paper we will study the case where the projection identifies the most unstable modes in the forecast model. To determine these modes we employ the discrete QR algorithm (Dieci and Van Vleck 2007, 2015). For the discrete time model $u_{n+1} = F_n(u_n) + \sigma_n$ with $u_n \in \mathbb{R}^N$, let $\mathbf{U}_0 \in \mathbb{R}^{N\times p}$ $(p \le N)$ denote a random matrix such that $\mathbf{U}_0^T\mathbf{U}_0 = \mathbf{I}$,

$$\mathbf{U}_{n+1}\mathbf{T}_n = F_n'(u_n)\mathbf{U}_n \approx \frac{1}{\epsilon}[F_n(u_n+\epsilon\mathbf{U}_n) - F_n(u_n)], \;\; n = 0,1,... \tag{33}$$

where $\mathbf{U}_{n+1}^T\mathbf{U}_{n+1} = \mathbf{I}$ and $\mathbf{T}_n$ is upper triangular with positive diagonal elements. With a finite difference approximation the cost is that of an ensemble of size p plus a reduced QR via modified Gram-Schmidt to re-orthogonalize. Time dependent orthogonal projections to decompose state space are $\Pi_n = \mathbf{U}_n\mathbf{U}_n^T$ and $\mathbf{I} - \Pi_n = \mathbf{I} - \mathbf{U}_n\mathbf{U}_n^T$. In order to apply (33) to an ensemble DA method, $\mathbf{U}_0$ must be specified and we must choose how to obtain u_n from the ensemble of particles at time t_n. We initialise $\mathbf{U}_0$ from a modified Gram-Schmidt orthonormalization of a random $N \times p$ matrix, and choose $u_n := \sum_i w_n^i u_n^i$, the weighted particle mean.

5.1. Projected approach and classical AUS techniques

This somewhat technical section establishes the relationship between existing AUS algorithms and the projected data approach. We consider the EKF-AUS (Trevisan and Palatella 2011; Palatella

et al. 2013, e.g.). EKF-AUS is a modified EKF in which the forecast covariance matrix $\mathbf{P}_n^f$ is replaced by the projected matrix $\Pi_n \mathbf{P}_n^f \Pi_n$, leading to the Kalman gain

$$\mathbf{K}_n = \Pi_n \mathbf{P}_n^f \Pi_n \mathbf{H}^T \left[\mathbf{H} \Pi_n \mathbf{P}_n^f \Pi_n \mathbf{H}^T + \mathbf{R} \right]^{-1} , \quad (34)$$

where the EKF forecast covariance matrix $\mathbf{P}_n^f$ and observation operator $\mathbf{H}_n \equiv \mathbf{H}$ are described in Section 2.1.1. It is clear that the EKF-AUS Kalman gain can be written as a combination of the columns of $\mathbf{U}_n$.

For comparison, we write down the Kalman gain associated with the data model (19),

$$\mathbf{K}_n = \mathbf{P}_n^f \Pi_{\mathbf{H}} \Pi_n \left[\Pi_n \mathbf{H}^\dagger \left(\mathbf{H} \mathbf{P}_n^f \mathbf{H}^T + \mathbf{R} \right) (\mathbf{H}^\dagger)^T \Pi_n \right]^\dagger . \quad (35)$$

We choose this form to most closely resemble EKF-AUS; the arguments of Theorem 3.1 guarantee that (35) is identical to the Algorithm 1 implementation of the EKF.

The difference between the two Kalman gains is essentially that (35) interchanges the position of $\mathbf{H}$ and Π_n, requiring the use of $\mathbf{H}^\dagger$ in order to do so, but manages to project all terms in the covariance-weighting inverse instead of only the forecast covariance matrix. Unlike the classical AUS gain (34), (35) does not restrict the analysis increment to the unstable subspace. The innovation is $y_n - \mathbf{H} u_n^f$ in classical AUS, but with (35) would be $y_n^p - \Pi_n \Pi_{\mathbf{H}} u_n^f$.

That is, classical AUS uses the full data but restricts the assimilation update to the unstable subspace via (34); Algorithm 1 restricts the innovation to the unstable subspace but the assimilation update can distribute this innovation across the whole of model space. The comparison between these algorithms here is pedagogical, not competitive; the advantages of the EKF-AUS algorithm are well established, while Algorithm 1 effects a reduction in data dimension that we will explore for Particle Filters, not the EKF.

Finally we obtain a form of EKF associated with the projected model (1) and unprojected data. This is essentially a re-derivation of EKF-AUS from the projected framework employed in this paper, confirming that the two are compatible. Consider a linear or linearized physical model $u_{n+1} = \mathbf{A}_n u_n + \sigma_n$. The projected model has the form $v_{n+1} = \mathbf{B}_n v_n + \Pi_{n+1} \sigma_n$ with $\mathbf{B}_n := \Pi_{n+1} A_n \Pi_n$, $v_n = \Pi_n u_n$, model error covariances $\mathbf{Q}_{n+1} := \Pi_{n+1} \mathbf{Q} \Pi_{n+1}$, and observation operators $\mathbf{H}_n := \mathbf{H} \Pi_n$. The reduced dimensional projected model has the form $w_{n+1} = \mathbf{T}_n w_n + \mathbf{U}_{n+1}^T \sigma_n$ with $\mathbf{T}_n := \mathbf{U}_{n+1}^T A_n \mathbf{U}_n$, $w_n = \mathbf{U}_n^T u_n$, model error covariances $\mathbf{Q}_{n+1} := \mathbf{U}_{n+1}^T \mathbf{Q} \mathbf{U}_{n+1}$, and observation operators $\mathbf{H}_n := \mathbf{H} \mathbf{U}_n$. Let $\tilde{\mathbf{P}}_n^f$ and $\hat{\mathbf{P}}_n^f$ denote the forecast covariance matrices for the projected and reduced order models, respectively. Initialising with $\tilde{\mathbf{P}}_0^f = \Pi_0 \mathbf{P}_0^f \Pi_0$ and $\hat{\mathbf{P}}_0^f = \mathbf{U}_0^T \mathbf{P}_0^f \mathbf{U}_0$ we see that by projecting and appropriate modfication to the observation operators we obtain projected and reduced order EKF-AUS forecast covariance matrices, respectively.

We will now explore the benefits of the projected data algorithms in an AUS framework, using (33) to calculate the projections, and present examples from the Lorenz 96 system.

6. Numerics

Consider the system of ordinary differential equations introduced in Lorenz (1996),

$$\dot{u}_i = (u_{i+1} - u_{i-2})\, u_{i-1} - u_i + F , \quad (36)$$

for $i = 1, ..., 40$ and $F = 8$. This 'Lorenz-96' system is chaotic with 14 positive and 1 neutral Lyapunov exponents. We present experiments in which the deterministic part of the model (2) is given by an integration of (36) for a fixed time.

The primary focus of this section is Algorithm 2, PROJ-OP-PF, compared to the OP-PF and ETKF.

6.1. Methods

The following five DA methods are compared.

1. PROJ-OP-PF, adding no noise on resampling. We compute with each p from 1 to 20.

2. PROJ-OP-PF, adding Gaussian noise on resampling and employing Algorithm 3 with $\alpha = 0.99$. We consider each

p from 1 to 20, and consider ten values of the noise on resampling ω, linearly spaced between 10^{-4} and 10^{-1}.

3. OP-PF, adding no noise on resampling.

4. OP-PF, adding Gaussian noise on resampling. We consider ten values of the noise on resampling ω, linearly spaced between 10^{-4} and 10^{-1}.

5. ETKF. We consider ten values of multiplicative inflation for the forecast ensemble between 1 and 1.1, and additionally employ additive inflation of $\mathbf{Q}$ on the forecast covariance matrix.

The first and third DA methods are benchmark schemes that, compared to the second and fourth scheme, clarify the contribution of noise on resampling to the good performance of OP-PF and PROJ-OP-PF.

The goal of these experiments is to improve on the Optimal Proposal Particle Filter; the ETKF is present as an example of a very good DA method for this model. As outlined in Section 2, the OP-PF is the least degenerate of a wide class of Particle Filters, and further improving on its performance within the Particle Filtering class of algorithms is a significant achievement. The major advantage of PROJ-OP-PF over the ETKF is, as with any Particle Filter, when filtering nonGaussian prior/posterior distributions.

6.2. *Experimental setup*

We perform identical twin experiments: the true system state is generated by the model update equation (2), and the role of DA is to correct for errors in the initial condition and the realizations of the model noise. All experiments fix the following.

- the number of ensemble members, $L = 20$.
- the initial conditions for each ensemble sampled from a Gaussian with spread equal to the model noise, centred on the true initial condition.
- model and true system dynamics are simulated with the fourth order Runge-Kutta scheme with time steps of 0.01, repeated until an observation time is reached.
- the methods are spun up by computing and then discarding 1000 analysis steps; performance is then measured over the following $10,000$ analysis steps.
- each method is repeated twenty times at each of the possible combinations of the tuning parameters listed above.

We present results for six distinct scenarios, or experiments. The key parameters in each experiment are the model noise $\mathbf{Q}$, the observation covariance $\mathbf{R}$, the time between observations $\Delta \equiv t_n - t_{n-1}$, and the proportion of the state that is observed. The first experiment produces a well-known test regime for the Ensemble Kalman Filter. The final experiment is chosen according to recommendations in Majda *et al.* (2014):

> "Demanding tests for filter performance are the regimes of spatially sparse, infrequent in time, high-quality (low observational noise) observations for a strongly turbulent dynamical system."

Our implementation of this test is less extreme than in Majda *et al.* (2014), but we only employ $L = 20$ particles (compared to $10,000$ in the cited paper).

Table 1 displays the key experimental parameters.

The first two experiments involve small values of $\mathbf{Q}$. For these the OP-PF essentially reduces to a bootstrap Particle Filter, which is not accurate for the high-dimensional Lorenz-96 system (RMSE around 5). In these two experiments *only*, we additively inflate the value of $\mathbf{Q}$ used in the OP-PF and PROJ-OP-PF, (14)–(17) and (28), by $0.3\mathbf{I}_{40}$. This modification retains some of the optimal proposal update and stabilises both OP-PF and PROJ-OP-PF (though the best performing PROJ-OP-PF, in experiment 3, does not use it).

6.3. *Results*

On average out of all experiments, PROJ-OP-PF has 13% less RMSE than OP-PF, and resamples about half as often (49% of the time). The advantages of PROJ-OP-PF are more pronounced in Experiments 1-3, but there is some improvement over OP-PF in all cases. These statistics are computed by comparing the best-tuned PROJ-OP-PF to the best-tuned OP-PF (each averaged over 10^4

	Q	R	Δ	observation spacing
Exp. 1	$(0.01)^2\mathbf{I}_{40}$	$\mathbf{I}_{40}$	0.05	all vars observed
Exp. 2	$(0.01)^2\mathbf{I}_{40}$	$(0.5)^2\mathbf{I}_{40}$	0.05	all vars
Exp. 3	$(0.1)^2\mathbf{I}_{40}$	$(0.5)^2\mathbf{I}_{40}$	0.05	all vars
Exp. 4	$(0.1)^2\mathbf{I}_{40}$	$(0.5)^2\mathbf{I}_{20}$	0.05	every second var
Exp. 5	$(0.1)^2\mathbf{I}_{40}$	$(0.1)^2\mathbf{I}_{20}$	0.05	every second var
Exp. 6	$(0.1)^2\mathbf{I}_{40}$	$(0.1)^2\mathbf{I}_{20}$	0.1	every second var

Table 1. Key parameters for the Lorenz-96 experiments in Section 6. Each row is a different experiment and the columns show the model noise, observation noise, time interval between observations and proportion of state space observed. The first experiment resembles a classic Lorenz-96 filtering experiment. The final experiment has larger model uncertainty, more accurate observations (harder for Particle Filters to avoid degeneracy), and infrequent, partial observations. Only one parameter changes between each experiment; in the online version of this article, that parameter is colored blue.

	OP-PF:		PROJ-OP-PF:	
	RMSE	Resamp	RMSE	Resamp
Exp. 1	0.71	59%	0.53	8%
Exp. 2	0.42	57%	0.35	5%
Exp. 3	0.42	58%	0.36	5%
Exp. 4	1.78	58%	1.68	49%
Exp. 5	0.81	62%	0.72	53%
Exp. 6	0.57	61%	0.53	57%

Table 2. Summary, for each of the experiments described in Section 6.2, of the RMSE and percentage of assimilation steps that trigger resampling. Results only listed for the best tuned (minimal RMSE) implementations of OP-PF and PROJ-OP-PF. The projected scheme beats OP-PF in all cases, ranging from a modest (7%) improvement in both RMSE and resampling in Experiment 6, to a 25% improvement in RMSE and 85% reduction in the number of resampling steps in Experiment 1.

time steps and over 20 repetitions of each experiment, at all tuning parameters described in section 6.1). Full details for the tuning and error scaling are presented in detail in the following subsection. The best-tuned results for each experiment are summarised in Table 2.

6.4. *Detailed results*

The optimally tuned parameters for each experiment are reported in Table 3. Figures 2–7 show how varying the projection rank p affects PROJ-OP-PF, compared to optimally tuned implementations of the OP-PF and ETKF, in each experiment. For each plot the tuned ETKF shows roughly what is achievable with an ensemble of size 20, while OP-PF displays the poor filtering performance of Particle Filters in high dimensions. PROJ-OP-PF reduces, by up to half, the gap in RMSE between OP-PF and ETKF.

	OP-PF:	PROJ-OP-PF:	
	ω	p	ω
Exp. 1	0	1	0
Exp. 2	0	1	0
Exp. 3	0	1	0
Exp. 4	0.02 (−1%)	5	0.01 (−1%)
Exp. 5	0.01 (−4%)	7	10^{-4} (−5%)
Exp. 6	10^{-4} (−2%)	9	10^{-4} (−3%)

Table 3. Summary, for each of the experiments described in Section 6.2, of the optimal tuning parameters for OP-PF and PROJ-OP-PF. Numbers in parentheses show the percentage reduction of the RMSE by employing noise on resampling, compared to the benchmark OP-PF and PROJ-OP-PF schemes. The optimal inflation for ETKF was 1.09 or 1.1 in all cases.

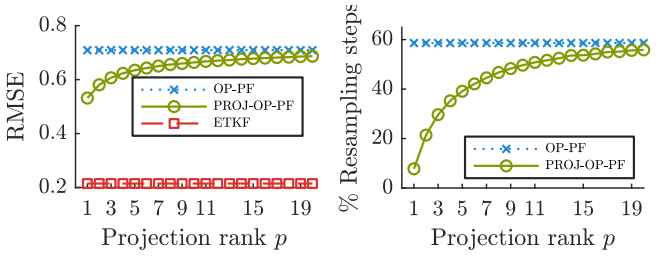

Figure 2. PROJ-OP-PF results scaling p for experiment 1 (compare Tables 1, 2 and 3). The best tuned PROJ-OP-PF covers 36% of the gap from the OP-PF error to the, probably optimal, ETKF error, and reduces the proportion of resampling to almost none.

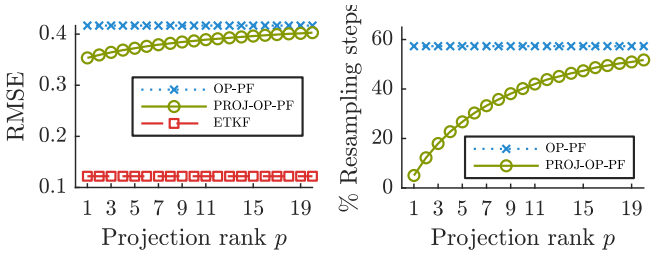

Figure 3. PROJ-OP-PF results scaling p for experiment 2. The best tuned PROJ-OP-PF scheme covers 21% of the gap from the best tuned OP-PF to the ETKF.

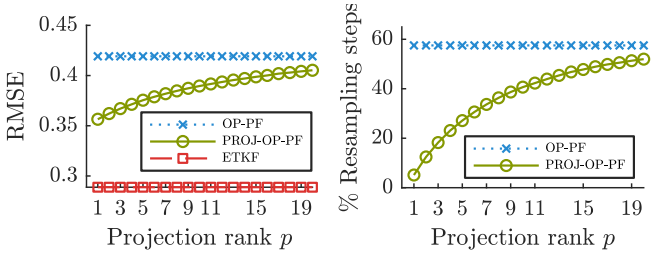

Figure 4. PROJ-OP-PF results scaling p for experiment 3. The best tuned PROJ-OP-PF scheme covers 48% of the gap from the best tuned OP-PF to the ETKF.

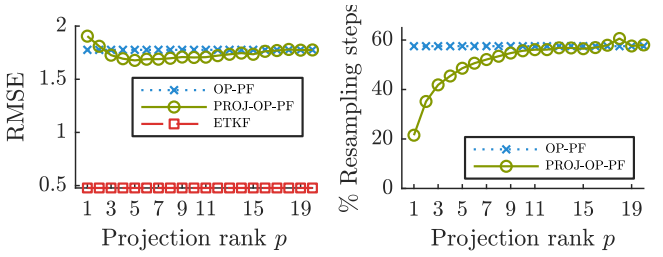

Figure 5. PROJ-OP-PF results scaling p for experiment 4. This experiment is by far the most challenging for PROJ-OP-PF, which at best has RMSE 1.68 (and resamples on 49% of assimilation steps) compared to RMSE 1.78 (58%) for OP-PF. One can instead ($p = 2$) reduce resampling by a factor of about half, corresponding to a much less tight ensemble and better higher order statistics, without greatly affecting the RMSE.

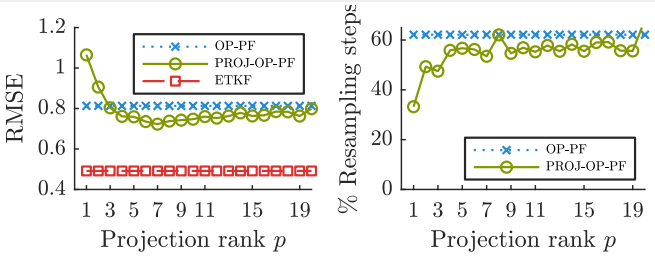

Figure 6. PROJ-OP-PF results scaling p for experiment 5. Compared to OP-PF one can tune PROJ-OP-PF for RMSE, covering 28% of the gap from the best tuned OP-PF to the ETKF, or ($p = 3$) reduce resampling without affecting the RMSE.

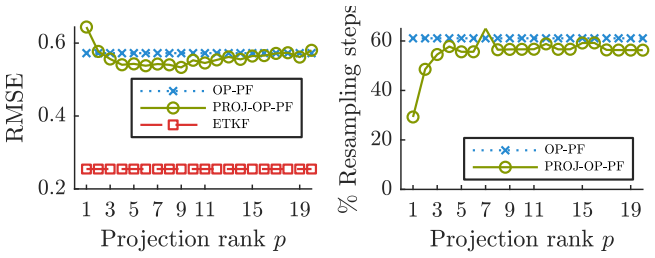

Figure 7. PROJ-OP-PF results scaling p for experiment 6. Compared to OP-PF one can tune PROJ-OP-PF for RMSE, covering 12% of the gap from the best tuned OP-PF to the ETKF, or ($p = 2$) reduce resampling without affecting the RMSE.

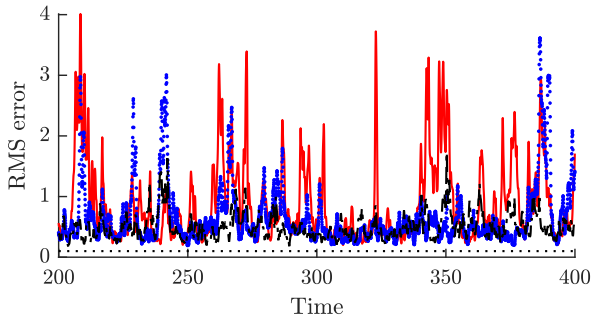

Figure 8. RMSE over time of OP-PF (solid line, red in the online version), PROJ-OP-PF (dots, blue in the online version) and ETKF (dashed, black in the online version). OP-PF frequently spikes in error, while PROJ-OP-PF spikes less frequently. For visualisation we plot the rolling mean RMSE of every five assimilation steps and restrict the plot to a subset of assimilation times.

The RMSE over time from all methods from a subset of one run from experiment 5 is visualised in Figure 8. We see that both Particle Filter methods have similar *minimum* RMSE to the ETKF, but also frequently spike in error. The advantage of PROJ-OP-PF is that the error spikes are much less frequent, and smaller in amplitude. The more severe degeneracy of OP-PF, compared to PROJ-OP-PF, is visible in Figure 9, where an ensemble from experiment 5 is shown by plotting six state variables at the 6000-th assimilation step.

The ETKF is generally stable with low RMSE for multiplicative inflation equal to or larger than 1.02 (1.04 in experiments 4-6). However it may be under-dispersive with spread lower than RMSE, and for this reason the optimal inflation was 1.1 in all cases. For each experiment in order, the spread in the ETKF was

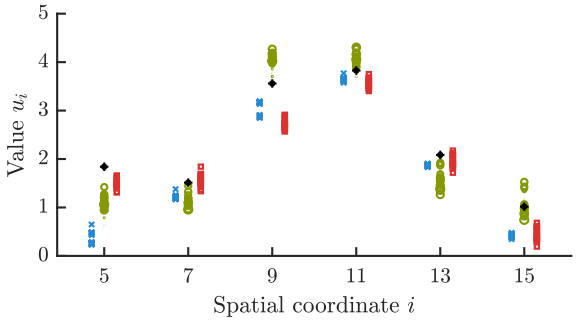

Figure 9. Details of all ensembles after an assimilation step, plotting six spatial coordinates. Coloured splotches show the ensembles, each plotted slightly apart in the horizontal direction for visualisation. From left to right in each coordinate: OP-PF (blue crosses in the online version), which has just resampled, is tightly spread in four of the six plotted coordinates, and has significant error in many; PROJ-OP-PF (green circles) has a larger spread and tends to be closer to the truth (black star). The EnKF (red squares) is closest to the truth in four of six coordinates. Note that none of the variables plotted in this figure were observed.

0.20, 0.10, 0.13, 0.19, 0.06, 0.06. The final three spreads can be widened to approximately equal the RMSE by taking a much larger value of the multiplicative inflation, 1.3. The additive inflation of $\mathbf{Q}$ is needed in experiments 4-6, else the ETKF suffers large RMSE values (> 3).

6.5. *Compete with EnKF by including model error*

The previous experiments outline, over diverse scenarios, the advantage of PROJ-OP-PF over OP-PF. We now adapt Harlim and Hunt (2007)'s example scenario in which non-Gaussian filters can compete with the EnKF, and confirm the advantage of PROJ-OP-PF in that case. The idea is to introduce model error: all DA methods will obtain forecasts using forcing $F = 6$ in (36), whereas the true system state is generated with the original forcing $F = 8$. Experiments in this section will otherwise replicate the prior sections: experimental setup as given in Section 6.2, and key parameters for each experiment given in Table 1. The principal result is that PROJ-OP-PF outperforms the tuned ETKF (which, as before, performs best with larger multiplicative inflation) in three of six experiments, with roughly equal performance in a fourth. Given that the posterior for the Lorenz-96 system is generally almost Gaussian, this is a remarkable result. Figure 10 shows the relative performance of all three schemes for Experiment Three. Summary statistics are provided for all DA schemes in Table 4.

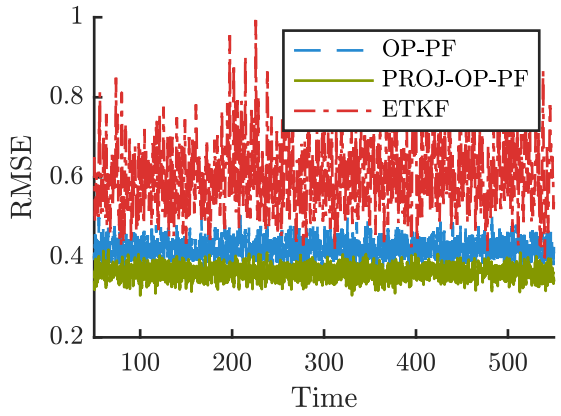

Figure 10. RMSE over time for a single realization of Experiment Three with model error, described in Section 6.5. For visualisation purposes we plot the rolling mean of the RMSE over ten assimilation steps. Our algorithm PROJ-OP-PF now out-performs the ETKF, and maintains its advantage over OP-PF.

	OP-PF:		PROJ-OP-PF:		ETKF:
	RMSE	Res	RMSE	Res	RMSE
Exp. 1	0.73	58%	0.60	10%	1.29
Exp. 2	0.42	57%	0.36	5%	1.25
Exp. 3	0.42	57%	0.36	6%	0.61
Exp. 4	1.97	58%	1.88	50%	1.25
Exp. 5	1.40	84%	1.17	100*%	1.16
Exp. 6	1.25	95%	1.15	100*%	0.91

Table 4. Statistics for experiments with model error (compare Table 2 without model error). Tuned parameters are given in Table 3, except for Experiments 5 and 6 in which PROJ-OP-PF uses $\omega = 0.1$. The 100% resampling for PROJ-OP-PF in those experiments can be reduced if ω is smaller (at a commensurate cost to RMSE.) For three experiments, including the standard benchmark for the EnKF, Experiment One, PROJ-OP-PF significantly outperforms both OP-PF and the ETKF, with drastically reduced resampling compared to OP-PF.

6.6. *Outlook*

We now place the results of our experiments in more context. A key difference between our tuned PROJ-OP-PF and related works is the optimal selection of the subspace dimension: for our Experiments One to Three, the optimal dimension was $p = 1$, and the largest value considered was $p = 9$. This is in stark contrast with AUS: in (Trevisan and Palatella 2011; Palatella *et al.* 2013, e.g.) the optimal subspace dimension is typically the number of positive and stable Lyapunov exponents (14 in our experiments), and Grudzien *et al.* (2018b,a) incorporate model error and conclude the optimal subspace dimension is larger still. Quinn *et al.* (2020) consider reduced rank DA in three coupled Lorenz-63 systems and conclude that the optimal subspace dimension is the Kaplan-Yorke dimension, again larger than the dimension of the unstable-neutral subspace. In some manner, all of the above papers are considering lowering the dimension of the model—whether for reduced order modelling or to represent some ensemble in model space. By contrast our approach projects the data, and preserves the full rank model forecast.

We consider three reasons for the smaller subspace dimension. First note that the PROJ-OP-PF algorithm uses the data in two ways: the original, non-projected form is used in an EnKF-like step to update the particles, then the reduced order data is used to update the particle weights. So we can understand that our method may still perform well with small p if the *combination* of the full rank particle update with the low rank weight update is sufficient to approximate the posterior distribution.

Secondly, we conjecture that the effect of reduced data may relate to synchronization. Pecora and Carroll (1990, 1991); Pecora *et al.* (1997) show that if one inserts values for certain variables into a dynamical system, and solves the dynamical system with essentially arbitrary initial conditions in the remaining variables, then the solution synchronizes with the true solution. In this work we are forcing with projected data that for each time is a linear combination of the model variables. We conjecture that this may provide a synchronizing effect even if the dimension of the projected data is lower than the number of positive Lyapunov exponents.

Lastly we note that there is a reason to anticipate that the optimal dimensions for projected data and projected models may differ. In the reduced order physical models, the optimal subspace dimension has more to do with sufficiently resolving solutions so that particles provide good approximation of the true solution. If we are projecting the data (only), then the particles can well approximate the true solution and we only need to determine their importance, i.e., their weights. The projected data models may, due to containing less information, update the weights more slowly, but they also avoid the degeneracy issues with large data dimension (see, e.g., Bengtsson *et al.* (2008); Snyder *et al.* (2008); Van Leeuwen (2009)).

This manuscript provides a framework for the use of projections in reduced order data models. The framework empowers future research to investigate optimal projections for various classes of problems. Some contemporary works provide insight: Beeson

and Sri Namachchivaya (2020) find best results using projections calculated from future time intervals, and (Carrassi *et al.* 2020, Section 5.1) argue that there is limited benefit to the projected approach if the model is deterministic. The second reference includes an explicit proof that, for a perfect deterministic model with uncorrelated observation errors ($\mathbf{R} = \mathbf{I}$), projecting to a lower dimension does not reduce filter degeneracy; therefore we learn that the benefit of employing projections is related to model error, model noise, and/or correlated observation errors. Along these lines, Albarakati *et al.* (2021) extend the PROJ-OP-PF scheme to project both the model and data and consider other reduced order modeling techniques in addition to AUS. Numerical results with a 38100-dimensional Shallow Water Equation show that the projected model dimension should be at least the Kaplan-Yorke dimension, but the projected data dimension can be much smaller.

7. Discussion

In this work a new approach to DA has been derived that allows for dimension reduction of the data using a projection defined in state space. The chief application has been Particle Filters Assimilating in the Unstable Subspace, which the classical AUS approach is unsuitable for because ensemble methods already project the forecast strongly into the unstable subspace (Bocquet and Carrassi 2017). By contrast the new approach sharply reduces filter degeneracy in a predictable fashion, improves filter accuracy and allows one to construct a sensible resampling scheme that adds more noise in more uncertain directions. Algorithms resting on the projected DA approach were tested on the chaotic Lorenz 96 system that provides a challenging scenario for particle filters. The main algorithm tested was a particle filter, PROJ-OP-PF that mixes projected and unprojected data based on the optimal proposal. The discrete QR technique used to find the unstable subspace in this work is rigorously justified and the additional cost incurred by it is proportional to employing an ensemble size of the dimension of the projected subspace. The PROJ-OP-PF scheme makes only the most basic use of the data in the space orthogonal to whatever projection is employed. Projecting the data to lower the dimension of the observations generally reduces both the RMSE and the resampling as compared to OP-PF. Resampling is essentially always better in our experiments and the RMSE is only worse for PROJ-OP-PF with extremely low rank projections in experiments 4 and 5.

Interesting extensions to these techniques include the use of nonlinear observation operators and the application of spatial localization techniques. Due to the modular approach we have taken by introducing projected observations, existing localization techniques for particle filters can be applied as they would with the original unprojected observations. In addition, the combination of projected physical model and projected observational model allow for a combination of localized and unlocalized systems to be employed much like the way projected and unprojected data models are used in Proj-OP-PF. In this work we have considered full rank, linear observation operators. There are some straightforward extensions to nonlinear observation operators both when using state space based projections and observation space based projections. For state space based projections, the full rank assumption is not necessary, only the existence of $\mathbf{H}^\dagger$ which can be formed using the SVD. For nonlinear observation operators, the linearization of $\mathbf{H}$ could be used to form $D\mathbf{H}(x)^\dagger$. Observation space projections, e.g., using PCA, remove the need for $\mathbf{H}^\dagger$ or $D\mathbf{H}(x)^\dagger$ and can be applied directly to obtain projected data models for both linear and nonlinear observation operators.

Future work will generalise the projected DA approach to employ two assimilation methods, one in the projected and one in the orthogonal space. Such manipulations are done in Majda *et al.* (2014); Slivinski *et al.* (2015), for example, and formulated for model error in AUS in section 3.2 of Grudzien *et al.* (2018b). The methods formulated in this paper have also been extended to particle filters that project both model and data; the resulting scheme was successfully applied to a high-dimensional shallow water model in Albarakati *et al.* (2021).

Acknowledgements

JM acknowledges the support of ONR grant N00014-18-1-2204, NSF grant DMS-1722578, and the Australian Research Council Discovery Project DP180100050. EVV acknowledges the support of NSF grants DMS-1714195 and DMS-1722578. The

authors are grateful to Alberto Carrassi for helpful feedback on an early version of this work. This work was supported with supercomputing resources provided by the Phoenix HPC service at the University of Adelaide.

References

Albarakati A, Budišić M, Crocker R, Glass-Klaiber J, Iams S, Maclean J, Marshall N, Roberts C, Van Vleck ES. 2021. Model and Data Reduction for Data Assimilation: Particle Filters Employing Projected Forecasts and Data with Application to a Shallow Water Model. *arXiv:2101.09252 [nlin, physics:physics]* ArXiv: 2101.09252.

Asch M, Bocquet M, Nodet M. 2016. *Data assimilation: methods, algorithms, and applications*, vol. 11. SIAM.

Beeson R, Sri Namachchivaya N. 2020. Particle filtering for chaotic dynamical systems using future right-singular vectors. *Nonlinear Dynamics* **102**: 679–696.

Bengtsson T, Bickel P, Li B. 2008. *Curse-of-dimensionality revisited: Collapse of the particle filter in very large scale systems*. Institute of Mathematical Statistics, ISBN 978-0-940600-74-4. Pages: 316-334 Publication Title: Probability and Statistics: Essays in Honor of David A. Freedman.

Bishop CH, Etherton BJ, Majumdar SJ. 2001. Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects. *Monthly weather review* **129**(3): 420–436.

Bocquet M, Carrassi A. 2017. Four-dimensional ensemble variational data assimilation and the unstable subspace. *Tellus A: Dynamic Meteorology and Oceanography* **69**(1): 1304 504.

Bocquet M, Gurumoorthy KS, Apte A, Carrassi A, Grudzien C, Jones CKRT. 2017. Degenerate Kalman filter error covariances and their convergence onto the unstable subspace. *SIAM/ASA J. Uncertain. Quantif.* **5**(1): 304–333.

Budhiraja A, Friedlander E, Guider C, Jones CK, Maclean J. 2017. Data assimilation; inference for linking physical and probabilistic models for complex nonlinear dynamic systems. In: *Handbook of Environmental and Ecological Statistics*, Gelfand AE, Fuentes M, Hoeting JA, Smith RL (eds), CRC Press, 1 edn, pp. 687–708.

Burgers G, Van Leeuwen PJ, Evensen G. 1998. Analysis scheme in the ensemble Kalman filter. *Monthly weather review* **126**(6): 1719–1724.

Carrassi A, Bocquet M, Bertino L, Evensen G. 2018. Data assimilation in the geosciences: An overview of methods, issues, and perspectives. *WIREs Climate Change* **9**(5): e535.

Carrassi A, Bocquet M, Demaeyer J, Grudzien C, Raanes P, Vannitsem S. 2020. Data assimilation for chaotic dynamics. *arXiv preprint arXiv:2010.07063* .

Carrassi A, Ghil M, Trevisan A, Uboldi F. 2008a. Data assimilation as a nonlinear dynamical systems problem: Stability and convergence of the prediction-assimilation system. *Chaos* **18**(2).

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

Carrassi A, Trevisan A, Uboldi F. 2007. Adaptive observations and assimilation in the unstable subspace by breeding on the data-assimilation system. *Tellus A: Dynamic Meteorology and Oceanography* **59**(1): 101–113.

Chorin A, Morzfeld M, Tu X. 2010. Implicit particle filters for data assimilation. *Communications in Applied Mathematics and Computational Science* **5**(2): 221–240.

de Leeuw B, Dubinkina S, Frank J, Steyer A, Tu X, Van Vleck E. 2018. Projected shadowing-based data assimilation. *SIAM Appld. Dyn. Sys.* .

Dieci L, Van Vleck ES. 2007. Lyapunov and Sacker-Sell spectral intervals. *J. Dynam. Differential Equations* **19**(2): 265–293.

Dieci L, Van Vleck ES. 2015. Lyapunov exponents: Computation. In: *Encyclopedia of Applied and Computational Mathematics*, Engquist B (ed). Springer-Verlag, pp. 834–838.

Doucet A, De Freitas N, Gordon N. 2001. An introduction to sequential Monte Carlo methods. In: *Sequential Monte Carlo methods in practice*, Doucet A, De Freitas N, Gordon N (eds), Springer, pp. 3–14.

Doucet A, Godsill S, Andrieu C. 2000. On sequential Monte Carlo sampling methods for Bayesian filtering. *Statistics and computing* **10**(3): 197–208.

Evensen G. 1994. Sequential data assimilation with a nonlinear quasi-geostrophic model using Monte Carlo methods to forecast error statistics. *Journal of Geophysical Research: Oceans* **99**(C5): 10 143–10 162.

Evensen G. 2009. *Data assimilation: the ensemble Kalman filter*. Springer Science & Business Media.

Farchi A, Bocquet M. 2018. Comparison of local particle filters and new implementations. *Nonlinear Processes in Geophysics* **25**(4): 765–807.

Frank J, Zhuk S. 2018. A detectability criterion and data assimilation for nonlinear differential equations. *Nonlinearity* **31**(11): 5235–5257.

González-Tokman C, Hunt BR. 2013. Ensemble data assimilation for hyperbolic systems. *Physica D: Nonlinear Phenomena* **243**(1): 128–142.

Grudzien C, Carrassi A, Bocquet M. 2018a. Asymptotic forecast uncertainty and the unstable subspace in the presence of additive model error. *SIAM/ASA J. Uncertain. Quantif.* **6**(4): 1335–1363.

Grudzien C, Carrassi A, Bocquet M. 2018b. Chaotic dynamics and the role of covariance inflation for reduced rank kalman filters with model error. *Nonlinear Processes in Geophysics* **25**(3): 633–648.

Harlim J, Hunt BR. 2007. A non-Gaussian Ensemble Filter for Assimilating Infrequent Noisy Observations. *Tellus A: Dynamic Meteorology and Oceanography* **59**(2): 225–237.

Law K, Stuart A, Konstantinos Z. 2015. *Data assimilation: A mathematical introduction*, vol. 62. Springer Texts in Applied Mathematics, ISBN 978-3-319-20324-9.

Law KJH, Sanz-Alonso D, Shukla A, Stuart AM. 2016. Filter accuracy for the Lorenz 96 model: fixed versus adaptive observation operators. *Phys. D* **325**: 1–13.

Lorenz EN. 1996. Predictability - a problem partly solved. In: *Proceedings of seminar on Predictability*, vol. 1, Palmer T, Hagedorn R (eds). ECMWF, Cambridge University Press: Reading, UK, pp. 1–18.

Maclean J, Santitissadeekorn N, Jones CKRT. 2017. A coherent structure approach for parameter estimation in Lagrangian data assimilation. *Phys. D* **360**: 36–45.

Majda AJ, Qi D, Sapsis TP. 2014. Blended particle filters for large-dimensional chaotic dynamical systems. *Proc. Natl. Acad. Sci. USA* **111**(21): 7511–7516.

Morzfeld M, Adams J, Lunderman S, Orozco R. 2018. Feature-based data assimilation in geophysics. *Nonlinear Processes in Geophysics* **25**(2): 355–374.

Morzfeld M, Tu X, Atkins E, Chorin AJ. 2012. A random map implementation of implicit filters. *J. Comput. Phys.* **231**(4): 2049–2066.

Palatella L, Carrassi A, Trevisan A. 2013. Lyapunov vectors and assimilation in the unstable subspace: theory and applications. *Journal of Physics A: Mathematical and Theoretical* **46**(25): 254 020.

Pecora LM, Carroll TL. 1990. Synchronization in chaotic systems. *Physical Review Letters* **64**(8): 821–824. Publisher: American Physical Society.

Pecora LM, Carroll TL. 1991. Driving systems with chaotic signals. *Physical Review A* **44**(4): 2374–2383. Publisher: American Physical Society.

Pecora LM, Carroll TL, Johnson GA, Mar DJ, Heagy JF. 1997. Fundamentals of synchronization in chaotic systems, concepts, and applications. *Chaos: An Interdisciplinary Journal of Nonlinear Science* **7**(4): 520–543. Publisher: American Institute of Physics.

Poterjoy J, Anderson JL. 2016. Efficient assimilation of simulated observations in a high-dimensional geophysical system using a localized particle filter. *Monthly Weather Review* **144**(5): 2007–2020.

Potthast R, Walter A, Rhodin A. 2019. A localized adaptive particle filter within an operational nwp framework. *Monthly Weather Review* **147**(1): 345–362.

Qi D, Majda AJ. 2015. Blended particle methods with adaptive subspaces for filtering turbulent dynamical systems. *Phys. D* **298/299**: 21–41.

Quinn C, O&amp TJ, apos, Kane, Kitsios V. 2020. Application of a local attractor dimension to reduced space strongly coupled data assimilation for chaotic multiscale systems. *Nonlinear Processes in Geophysics* **27**(1): 51–51. Publisher: Copernicus GmbH.

Reddy AS, Apte A, Vadlamani S. 2020. Asymptotic properties of linear filter for deterministic processes. *Systems Control Lett.* **139**: 104 676, 8.

Reich S, Cotter C. 2015. *Probabilistic forecasting and Bayesian data assimilation*. Cambridge University Press.

Sapsis T. 2010. Dynamically orthogonal field equations. PhD thesis, Massachusetts Institute of Technology, Department of Mechanical Engineering.

Sapsis TP, Lermusiaux PFJ. 2009. Dynamically orthogonal field equations for continuous stochastic dynamical systems. *Phys. D* **238**(23-24): 2347–2360.

Slivinski L, Spiller E, Apte A, Sandstede B. 2015. A hybrid particle–ensemble kalman filter for lagrangian data assimilation. *Monthly Weather Review* **143**(1): 195–211.

Snyder C. 2011. Particle filters, the "optimal" proposal and high-dimensional systems. In: *Proceedings of the ECMWF Seminar on Data Assimilation for atmosphere and ocean*. pp. 1–10.

Snyder C, Bengtsson T, Bickel P, Anderson J. 2008. Obstacles to high-dimensional particle filtering. *Monthly Weather Review* **136**(12): 4629–4640.

Snyder C, Bengtsson T, Morzfeld M. 2015. Performance bounds for particle filters using the optimal proposal. *Monthly Weather Review* **143**(11): 4750–4761.

Sondergaard T, Lermusiaux PF. 2013. Data assimilation with Gaussian mixture models using the dynamically orthogonal field equations. Part I: Theory and scheme. *Monthly Weather Review* **141**(6): 1737–1760.

Tranninger M, Seeber R, Steinberger M, Horn M. 2020. Uniform detectability of linear time varying systems with exponential dichotomy. *IEEE Control Systems Letters* : 1–1.

Trevisan A, D'Isidoro M, Talagrand O. 2010. Four-dimensional variational assimilation in the unstable subspace and the optimal subspace dimension. *Q.J.R. Meteorol. Soc.* **136**: 487–496.

Trevisan A, Palatella L. 2011. On the Kalman filter error covariance collapse into the unstable subspace. *Nonlinear Processes in Geophysics* **18**(2): 243–250.

Tsukuma H, Kubokawa T. 2015. Estimation of the mean vector in a singular multivariate normal distribution. *J. Multivariate Anal.* **140**: 245–258.

Van Leeuwen PJ. 2009. Particle Filtering in Geophysical Systems. *Monthly Weather Review* **137**(12): 4089–4114. Publisher: American Meteorological Society.

van Leeuwen PJ. 2010. Nonlinear data assimilation in geosciences: an extremely efficient particle filter. *Quarterly Journal of the Royal Meteorological Society* **136**(653): 1991–1999.

Van Leeuwen PJ. 2012. Particle filters for the geosciences. *Advanced Data Assimilation for Geosciences: Lecture Notes of the Les Houches School of Physics: Special Issue* : 291.

van Leeuwen PJ, Künsch HR, Nerger L, Potthast R, Reich S. 2019. Particle filters for high-dimensional geoscience applications: A review. *Quarterly Journal of the Royal Meteorological Society* **145**(723): 2335–2365.