Computational Science
Laboratory Technical Report
CSL-TR-19-2015
May 19, 2015

Ahmed Attia, Vishwas Rao, and Adrian Sandu

"A Hybrid Monte-Carlo Sampling Smoother for Four Dimensional Data Assimilation"

Computational Science Laboratory
Computer Science Department
Virginia Polytechnic Institute and State University
Blacksburg, VA 24060
Phone: (540)-231-2193
Fax: (540)-231-6075
Email: `sandu@cs.vt.edu`
Web: `http://csl.cs.vt.edu`

A Hybrid Monte-Carlo Sampling Smoother for Four Dimensional Data Assimilation

Ahmed Attia Vishwas Rao Adrian Sandu

May 19, 2015

Abstract

This paper constructs an ensemble-based sampling smoother for four-dimensional data assimilation using a Hybrid/Hamiltonian Monte-Carlo approach. The smoother samples efficiently from the posterior probability density of the solution at the initial time. Unlike the well-known ensemble Kalman smoother, which is optimal only in the linear Gaussian case, the proposed methodology naturally accommodates non-Gaussian errors and non-linear model dynamics and observation operators. Unlike the four-dimensional variational method, which only finds a mode of the posterior distribution, the smoother provides an estimate of the posterior uncertainty. One can use the ensemble mean as the minimum variance estimate of the state, or can use the ensemble in conjunction with the variational approach to estimate the background errors for subsequent assimilation windows. Numerical results demonstrate the advantages of the proposed method compared to the traditional variational and ensemble-based smoothing methods.

Contents

1 Introduction

Data assimilation (DA) is the process of combining information from predictions made by imperfect models, from noisy observations, and from priors to produce a consistent description of the state of a dynamical system. The application of DA to large scale systems such as the atmosphere is of great practical interest. Two approaches for solving large DA problems have gained widespread acceptance. The first approach, originating from control theory, are variational methods such as the three-dimensional variational (3D-Var) and four-dimensional variational (4D-Var) strategies [35]. The variational methods find a maximum a-posteriori (MAP) estimate of the true state of the system. The second approach are the ensemble-based statistical estimation methods. The most successful family of ensemble data assimilation algorithms includes the ensemble Kalman filter (EnKF) [19] and its variants, the square-root Kalman filters [42], the ensemble adjustment Kalman filter [6], the ensemble transform Kalman filter [11], and efficient implementations of Kalman Filter, such as [4], using the Sherman-Morrison formula. All variants of EnKF provide a minimum variance estimate of the state by approximating the expected value of the posterior distribution. Variational and statistical estimation methods yield identical estimates (only) in the case of linear dynamics, linear observations, and Gaussian errors.

Both 3D-Var and EnKF are filtering methods that estimate the true state of the system at the specific time instances where observations are available. For many oceanographic, meteorological, and hydrological applications, it is advantageous to employ smoothing methods such as 4D-Var and EnKS that simultaneously use information from all observations available at different time points within an assimilation time window. Strong-constraint 4D-Var updates the state of the system at the initial time of an assimilation window given a background estimate of the initial condition and a set of observations distributed through this interval. The ensemble Kalman smoother, on the other hand, estimates the posterior distributions of the state at time points in the window given all past, present, and future observations (in the assimilation window).

4D-Var requires the derivation and implementation of the tangent linear and the adjoint numerical models, a challenging and effort-intensive task for large-scale models. The 4D-Var algorithm provides a single analysis state, the best posterior estimate of the state of the system. The uncertainty in the estimated state is not inherently provided by the 4D-Var algorithm [14]. Previous work proposed to quantify the uncertainty in the 4D-Var analysis, by approximating the analysis error covariance matrix using an ensemble of simulations [24, 25]. These schemes provide an approximation of the analysis error covariance matrix that is inconsistent with the 4D-Var analysis itself because the covariance estimates are usually obtained from independent schemes such as EnKF. Approaches to quantify 4D-Var analysis uncertainty based on subspace error decompositions [14, 35] are statistically consistent but require additional computational effort.

The EnKS is optimal when the observation operators are linear and the errors are Gaussian. However, these assumptions are unlikely to hold for real applications. The analysis ensemble generated by the EnKS allows to find a minimum variance estimate (e.g., ensemble mean), as well as a measure of the analysis uncertainty (e.g., the analysis error covariance matrix). When the posterior distribution is nearly Gaussian EnKS offers an efficient practical algorithm. However, when the observation operators are nonlinear and the errors are non-Gaussian, the EnKS is not expected to yield good results.

The Markov Chain Monte Carlo (MCMC) family of algorithms provides a powerful foundation to sample from complicated probability distributions. These algorithms work in general by generating a Markov chain whose stationary distribution is the target probability distribution. MCMC sampling is considered to be the gold standard [26] in data assimilation. The main practical limitation of MCMC is the considerable computational cost required to achieve convergence, and to explore the entire state space in the case of high dimensional state spaces. Scalable and accelerated MCMC algorithms are being continuously developed to improve convergence and space exploration. Hybrid Monte Carlo (HMC), also known as Hamiltonian Monte Carlo, is an accelerated MCMC sampling algorithm that reduces the correlation between successive states by using Hamiltonian dynamics to generate proposal states [16]. Moreover, HMC targets states with high acceptance probability leading to fast convergence and fast space exploration. To the best of our knowledge, HMC was first considered in the context of DA in [8] to solve a nonlinear inverse problem by minimizing the residual between the solution and ill-posed boundary conditions. Posterior error statistics are approximated by sampling the nearby states to the optimal state after convergence. In [8] a simulated annealing strategy is used during the sampling process where the minimum is obtained at a low temperature and posterior samples are collected at high temperatures. Solving the weak-constraint 4D-Var problem using a gradient method then using HMC to estimate the analysis error statistics is also discussed in [23, Chapter 6].

This work develops a nonlinear non-Gaussian smoother to solve the four dimensional data assimilation problem. The new method uses a Hybrid Monte Carlo approach to sample the posterior distribution and is named the HMC smoother. This work extends the sampling filter, proposed in [7], to the four dimensional case where time-distributed observations are assimilated at once. HMC smoother provides an ensemble of states sampled from the posterior distribution of the state of the system at the initial time of the assimilation window. In a practical setting the smoothing step is carried out sequentially over consecutive assimilation windows. The generated ensemble encapsulates the uncertainty in the posterior distribution at the beginning of the current window; when propagated to the beginning of the next assimilation window it provides flow-dependent information about the background error distribution in the next assimilation cycle.

The use of HMC for solving smoothing problems was presented also in [5], where a generalized version of HMC is used in an attempt to reducing the number of chain steps required to ensure independence of the generated states. The underlying dynamical system in the generalized version of HMC is not Hamiltonian. There are several important differences between this work and [5].

In [5] the only source of uncertainty is model error, in form of additive random noise included in the dynamics. Here we work in the strong constraint 4D-Var framework and consider the model to be perfect; the system state is uncertain due to uncertainties in the initial conditions. While in [5] numerical experiments are carried out with a one-dimensional system, here we experiment with shallow water equations over the sphere, a moderately large multidimensional nonlinear model relevant for geophysical applications. Finally, we propose to use higher order symplectic integrators, as tested in [7], to efficiently sample from complex posterior distributions that arise when the observation operators and models are highly nonlinear.

The remaining part of the paper is organized as follows. Section 2 reviews the variational and the ensemble approaches for solving the data assimilation problem. The HMC smoother is presented in Section 3. Experimental settings and numerical results are discusses in Section 4. Conclusions and future directions are summarized in Section 5.

2 Data Assimilation

This section reviews the 4D-Var and the EnKS data assimilation schemes.

2.1 Four-dimensional variational data assimilation

4D-Var calculates the optimal initial condition for the state of the dynamical system over a specific assimilation time window, based on a background state and using all observations available within this time window [34]. The background initial state is usually the forecast produced by propagating the previous window analysis through the model dynamics. To be specific, let the current assimilation window be the time interval $[t_0\,, t_F]$. Given a background state $\mathbf{x}_0^{\mathrm{b}} = \mathbf{x}^{\mathrm{b}}[t_0]$, and a set of observations $\{\mathbf{y}_k = \mathbf{y}[t_k]\}_{k=0,1,\dots,\mathrm{N_{obs}}}$, available at the discrete time points $\{t_k\}_{k=0,1,\dots,\mathrm{N_{obs}}} \subset [t_0\,, t_F]$, the 4D-Var analysis is obtained by solving the following optimization problem:

$$
\begin{aligned}
\min_{\mathbf{x}_0} \mathcal{J}(\mathbf{x}_0) = {} & \frac{1}{2} \left\| \mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}} \right\|^2_{\mathbf{B}_0^{-1}} \\
& + \frac{1}{2} \sum_{k=0}^{\mathrm{N_{obs}}} \left\| \mathcal{H}_k(\mathbf{x}_k) - \mathbf{y}_k \right\|^2_{\mathbf{R}_k^{-1}} \,.
\end{aligned} \tag{1}
$$

Here $\mathcal{H}_k$ is the observation operator (generally nonlinear) that maps the model space into the observation space at time point t_k. The dimension of observation space m is usually much lower than the dimension of the state space, that is $\mathrm{m} \ll \mathrm{N_{var}}$. $\mathbf{B}_0$ is the background error covariance matrix, and $\mathbf{R}_k$'s are the observation error covariance matrices at each times t_k; $k = 1, \dots, \mathrm{N_{obs}}$. The background error covariance matrix determines how information from observed areas are extrapolated to unobserved regions or where observations are sparsely available [41]. The state $\mathbf{x}_k = \mathbf{x}[t_k]$, is produced by propagating the initial state

$\mathbf{x}_0$ through the model dynamics from time t_0 to point t_k

$$\mathbf{x}_k = \mathcal{M}_{0,k}(\mathbf{x}_0). \tag{2}$$

The model solution operator $\mathcal{M}$ represents the discretized partial differential equations that govern the evolution of the dynamical system. Realistic atmospheric and ocean models typically have $\mathrm{N_{var}} \sim 10^6 - 10^9$ state variables.

In this work we consider the strong-constraint 4D-Var case which assumes that the numerical model (2) is perfect. The methodology proposed here can be immediately extended to the weak-constraint 4D-Var framework [38], where model errors are accounted for by adding the corresponding model error terms to equation (1) [41].

Perturbations (small errors $\delta\mathbf{x}$) of the state of the system evolve according to the tangent linear model:

$$\delta\mathbf{x}_k = \mathbf{M}_{0,k}(\mathbf{x}_0)\cdot\delta\mathbf{x}_0\ ,\quad t_0 \le t \le t_F\ , \tag{3}$$

where $\mathbf{M}_{0,k} = \partial\mathcal{M}_{0,k}/\partial\mathbf{x}[t_0]$ is the Jacobian of the model solution operator.

In strong-constraint 4D-Var the model dynamics act as constraints to the optimization problem (2). The optimal initial condition obtained by solving the optimization problem (1) constrained by the model dynamics (2), is referred to as the analysis state $\mathbf{x}_0^{\mathrm{a}} = \mathbf{x}^{\mathrm{a}}[t_0]$. The gradient of the cost functional (1) is:

$$\begin{aligned}\nabla_{\mathbf{x}_0}\mathcal{J}(\mathbf{x}_0) &= \mathbf{B}^{-1}(\mathbf{x}_0 - \mathbf{x}_0^{\mathrm{b}}) \\ &\quad + \sum_{k=0}^{\mathrm{N_{obs}}} \mathbf{M}_{0,k}^T\mathbf{H}_k^T\mathbf{R}_k^{-1}(\mathcal{H}_k(\mathbf{x}_k) - \mathbf{y}_k)\,,\end{aligned} \tag{4}$$

where $\mathbf{M}_{0,k}^T$ is the adjoint of the tangent linear model operator (3), $\mathbf{H}_k = \partial\mathcal{H}_k/\partial\mathbf{x}_k$ is the Jacobian of the observation operator $\mathcal{H}_k$, and $\mathbf{H}_k^T$ is its adjoint. Gradient-based minimization using (4) requires the development of the tangent linear and the adjoint models, which is a challenging task for high-dimensional complex models of practical interest. The performance of the optimization can be improved by using the second order derivative information and adaptive observations as described in [3].

2.2 Bayesian interpretation of 4D-Var

The knowledge of the system state at the initial time t_0 prior to obtaining new observations is described by the background (prior) probability density $\mathcal{P}^b(\mathbf{x}_0)$. The "sampling model" gives the probability distribution of observations conditioned by the initial state
$\mathcal{P}(\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_{\mathrm{N_{obs}}}|\mathbf{x}_0)$, under the belief that the dynamical model $\mathbf{x}_k = \mathcal{M}_{0,k}(\mathbf{x}_0)$ perfectly represents reality. From Bayes' theorem:

$$\mathcal{P}^{\mathrm{a}}(\mathbf{x}_0) = \mathcal{P}(\mathbf{x}_0|\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_{\mathrm{N_{obs}}}) \tag{5a}$$

$$= \frac{\mathcal{P}(\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_{\mathrm{N_{obs}}}|\mathbf{x}_0)\,\mathcal{P}^{\mathrm{b}}(\mathbf{x}_0)}{\mathcal{P}(\mathbf{y}_0, \mathbf{y}_1, \ldots, \mathbf{y}_{\mathrm{N_{obs}}})}, \tag{5b}$$

The posterior (analysis) PDF $\mathcal{P}^{\rm a}(\mathbf{x}_0)$ is the probability distribution of the initial state after incorporating the new knowledge contained in the observations. The denominator in (5b) is the marginal density of the observations and acts as a normalization factor.

The background and observations errors are usually assumed to have Gaussian distributions:

$$\mathcal{P}^{\rm b}(\mathbf{x}_0) \propto \exp\left(-\frac{1}{2}\left\|\mathbf{x}_0-\mathbf{x}_0^{\rm b}\right\|_{\mathbf{B}_0^{-1}}^2\right), \tag{6a}$$

$$\mathcal{P}(\mathbf{y}_k|\mathbf{x}_k) \propto \exp\left(-\frac{1}{2}\left\|\mathcal{H}_k(\mathbf{x}_k)-\mathbf{y}_k\right\|_{\mathbf{R}_k^{-1}}^2\right), \tag{6b}$$

where $\mathbf{B}_0$ is the background error covariance matrix and $\mathbf{R}_k$'s are the observation error covariance matrices at times t_k; $k=1,\dots,\mathrm{N_{obs}}$. If the observation errors at different time points are independent, and the model is perfect, the joint sampling model can be written as

$$\mathcal{P}(\mathbf{y}_0,\mathbf{y}_1,\dots,\mathbf{y}_{\mathrm{N_{obs}}}|\mathbf{x}_0) \propto \exp\left(\sum_{k=0}^{\mathrm{N_{obs}}}\left(-\frac{1}{2}\left\|\mathcal{H}_k(\mathbf{x}_k)-\mathbf{y}_k\right\|_{\mathbf{R}_k^{-1}}^2\right)\right). \tag{6c}$$

Bayes' rule (5) yields the following posterior PDF

$$\mathcal{P}^{\rm a}(\mathbf{x}_0) \propto \exp\Big(-\mathcal{J}(\mathbf{x}_0)\Big), \tag{7a}$$

$$\mathcal{J}(\mathbf{x}_0) = \frac{1}{2}\left\|\mathbf{x}_0-\mathbf{x}_0^{\rm b}\right\|_{\mathbf{B}_0^{-1}}^2 + \frac{1}{2}\sum_{k=0}^{\mathrm{N_{obs}}}\left\|\mathcal{H}_k(\mathbf{x}_k)-\mathbf{y}_k\right\|_{\mathbf{R}_k^{-1}}^2 . \tag{7b}$$

For nonlinear models and nonlinear observation operators the posterior (7) is not Gaussian. The kernel of the posterior is $\exp(-\mathcal{J}(\mathbf{x}_0))$, where $\mathcal{J}(\mathbf{x}_0)$ is the cost functional of the 4D-Var problem. 4D-Var computes the analysis $\mathbf{x}_0^{\rm a}$ as the minimizer of $\mathcal{J}$. The 4D-Var solution is the MAP estimate of the initial state under the assumptions that the background and observation errors are Gaussian. For highly nonlinear observation operators and highly nonlinear models the posterior can have multiple modes. In this case the 4D-Var numerical solution can be trapped in a local minimum of the cost functional.

The posterior distribution (5) contains the complete characterization of the uncertain initial state of the dynamical system. However, calculating the full posterior with high dimensional models is infeasible in practice. A practical approach is to describe the posterior probability density by an ensemble of states, and to use it to estimate moments of the distribution. Sampling directly from the posterior PDF of the initial condition (7) acts as a smoother; the ensemble mean provides an estimate of the true state of the system, and the ensemble covariance an estimate of the posterior uncertainty that is totally consistent with the analysis state. In contrast, the current practice to use the analysis obtained from 4D-Var and the covariance obtained from EnKF or EnKS leads to inconsistent representations of uncertainty.

2.3 Ensemble Kalman filter and smoother

Filtering is the process of calculating the posterior distribution of the uncertain state of a dynamical system at a specific time given observations only available at that time instance. The ensemble Kalman filter [19] represents probability distributions by samples. Let $\{\mathbf{x}_k^{\rm b}(e)\}_{e=1,2,\ldots,\rm N_{ens}}$ be an ensemble of forecast states at time t_k, and $\mathbf{y}_k = \mathbf{y}[t_k]$ the observation vector (set of measurements) at t_k. If the forecast (background) ensemble is represented by the matrix $\mathbf{X}_k^{\rm b}$, whose columns are the forecast ensemble members, then the updated (analysis) ensemble matrix $\mathbf{X}_k^{\rm a}$ at the same time t_k is obtained as [20]

$$\mathbf{X}_k^{\rm a} = \mathbf{X}_k^{\rm b} \cdot \mathbf{T}_k \,, \tag{8}$$

where the matrix $\mathbf{T}_k \in \mathbb{R}^{\rm N_{ens} \times N_{ens}}$ is a nonlinear transformation constructed from the forecast ensemble and the observations at time t_k [20]. Square-root filters [42], the ensemble transform Kalman filter [11], and the ensemble adjustment Kalman filter [6] can all be written in the form (8) for specific choices of the transformation $\mathbf{T}_k$.

Smoothing is the process of calculating the posterior distribution of the uncertain states of a dynamical system given past, present, and future observations [13]. There are three approaches to smoothing: fixed-point, fixed-interval, and fixed-lag smoothing [13], with the second and third ones being the most popular [33]. Any scheme that can be used to solve any of the three smoothing problems can also be employed as a single fixed-interval smoothing scheme [13]. The ensemble smoother (ES) was introduced in [21] as a linear variance minimization algorithm. The ensemble Kalman smoother (EnKS) [22] employs an ensemble of states to describe distributions and obtains the posterior using the Kalman filter updates equations. EnKS is optimal in case of linear dynamics, Gaussian errors, and large number of ensemble members [23].

To construct EnKS [22] the EnKF update equations (8) are used repeatedly to develop a fixed-lag, and a fixed-interval smoothing algorithms. A fixed-point smoother can be written as [20]

$$\mathbf{X}_0^{\rm s} = \mathbf{X}_0^{\rm a} \cdot \prod_{k=0}^{\rm N_{obs}} \mathbf{T}_k \,. \tag{9}$$

The update equation (9) is used recursively for fixed-interval smoothing, where smoothed ensembles are obtained at specified set of times, and they are conditioned only on observations available at later times in the interval. Ravela and McLaughlin [33] presented efficient, fast versions of the fixed-interval and the fixed-lag EnKS. The fast fixed-interval smoother has a computational cost that scales linearly with respect to the length of the interval. In this work, we use the fast fixed-interval EnKS [33], with a single smoothing point (fixed-point smoother) chosen at the beginning of the time interval.

EnKS computes the minimum variance estimate of the state. This is not expected to be very accurate if the observations are highly nonlinear or if the Gaussianity assumptions are severely violated. As shown in Section 4, the HMC sampling smoother proposed herein is capable of handling nonlinear observation and model operators, and consequently produces posterior estimates that are more useful than the EnKS ensemble, and contain more information than the 4D-Var MAP analysis. Hybrid methods such as [31] make use of optimization over ensembles using the trust-region framework.

3 The hybrid Monte-Carlo sampling smoother

The most popular and successful class of sampling algorithms is the Markov chain Monte-Carlo (MCMC) [29], first introduced by Metropolis *et al.* [27]. MCMC algorithms sample from a general probability distribution $\mathcal{P}(\mathbf{x})$ by building a Markov chain whose invariant distribution is $\mathcal{P}(\mathbf{x})$. MCMC algorithms have an advantage of not requiring the normalization of target distributions. However, traditional MCMC samplers are often considered impractical for large dimensional problems due to the following drawbacks: The Markov chain may take a very long time to reach stationarity. A large number of (burn-in) states are generated and discarded before starting the sampling process, in order to guarantee that the collected samples are obtained from the true target PDF. The samples should be independent, however the Markov chain is not completely memoryless; in order to achieve independence of sampled states, the sampler usually drops some intermediate states between each selected state. Another drawback of most of Monte-Carlo sampling methods is the curse of dimensionality [29]: as the dimension of the state spaces grows, the number of sample members needed to represent the probability distribution, grows rapidly. The number of samples required to efficiently represent the probability distribution can be controlled if the sampler surveys sufficiently fast the entire state space. The sampler can become trapped in a high-probability mode of a multi-modal distribution, and fail to represent the other probability modes.

3.1 Hybrid Monte-Carlo

Hybrid/Hamiltonian Monte-Carlo (HMC) [16] follows an auxiliary-variable approach in order to alleviate the limitations of the traditional MCMC algorithms.

The phase space of a Hamiltonian dynamical system consists of points $(\mathbf{p},\mathbf{x}) \in \mathbb{R}^{2\,\mathrm{N_{var}}}$, where $\mathbf{x} \in \mathbb{R}^{\mathrm{N_{var}}}$ is the position variable, and $\mathbf{p} \in \mathbb{R}^{\mathrm{N_{var}}}$ is the momentum variable. The Hamiltonian dynamics is governed by the set of ordinary differential equations (ODEs):

$$\begin{aligned} \frac{d\mathbf{x}}{dt} &= \nabla_{\mathbf{p}}\, H(\mathbf{p},\mathbf{x})\,, \\ \frac{d\mathbf{p}}{dt} &= -\nabla_{\mathbf{x}}\, H(\mathbf{p},\mathbf{x})\,, \end{aligned} \tag{10a}$$

where the Hamiltonian function H describes the total energy of the system

$$H(\mathbf{p},\mathbf{x}) = \frac{1}{2}\,\mathbf{p}^T\mathbf{M}^{-1}\mathbf{p} + \mathcal{J}(\mathbf{x})\,. \tag{10b}$$

The first term of the Hamiltonian (10b) represents the potential energy of the system, while the second term corresponds to the kinetic energy. The exact (analytic) flow of the Hamiltonian system (10a) advances the solution in time from $t = 0$ to $t = T$:

$$\Phi_T : \mathbb{R}^{2\mathrm{N}_{\rm var}} \rightarrow \mathbb{R}^{2\mathrm{N}_{\rm var}}; \quad \Phi_T\big(\mathbf{p}[0], \mathbf{x}[0]\big) = \big(\mathbf{p}[T], \mathbf{x}[T]\big)\,. \tag{11}$$

This flow cannot be calculated exactly in practice, and has to be approximated by an equivalent numerical solution using a time reversible symplectic integrator [36, 37]. The most common symplectic integrator is leapfrog (Störmer–Verlet) [36, 37]. One step of the position Verlet algorithm advances the solution of the Hamiltonian equations (10a) from time t_j to time $t_{j+1} = t_j + h$ using:

$$\mathbf{x}_{j+1/2} = \mathbf{x}_j + \frac{h}{2}\,\mathbf{M}^{-1}\,\mathbf{p}_j\,, \tag{12a}$$

$$\mathbf{p}_{j+1} = \mathbf{p}_j - h\,\nabla_{\mathbf{x}}\mathcal{J}(\mathbf{x}_{j+1/2})\,, \tag{12b}$$

$$\mathbf{x}_{j+1} = \mathbf{x}_{j+1/2} + \frac{h}{2}\,\mathbf{M}^{-1}\,\mathbf{p}_{j+1}. \tag{12c}$$

The optimal time step size h must satisfy $h \propto (1/\mathrm{N}_{\rm var})^{1/4}$ [9], and careful empirical tuning of the step size is usually required for good performance [7]. Several other symplectic integrators with more stages and higher accuracy than Verlet have also been developed [12]. An infinite dimensional time integrator was also introduced in [10].

For practical considerations it is advisable to split the interval $[0, T]$ where the Hamiltonian system evolves into m smaller sub steps of length $h = T/m$. The flow of the numerical solution obtained by the symplectic integrator will be denoted by $\tilde{\Phi}_T$ and is an approximation of the exact flow Φ_T.

The key idea in HMC sampling is to add an auxiliary variable $\mathbf{p}$ to the target variable $\mathbf{x}$ and sample from the joint probability distribution of $(\mathbf{x}, \mathbf{p})$. The auxiliary variable is chosen such that the sampling procedure from the joint distribution is much faster than sampling from the marginal distribution of the target variable. In HMC sampling the target and the auxiliary variables are thought of as the position and momentum components of a Hamiltonian system, respectively. The Hamiltonian dynamics of the system serves as a transition kernel to the Markov chain.

The kernel of the stationary probability distribution of the Hamiltonian system (10) is [29]

$$\exp\left(-H(\mathbf{p}, \mathbf{x})\right) = \exp\left(-\frac{1}{2}\mathbf{p}^T\mathbf{M}^{-1}\mathbf{p} - \mathcal{J}(\mathbf{x})\right) \tag{13a}$$

$$= \exp\left(-\frac{1}{2}\mathbf{p}^T\mathbf{M}^{-1}\mathbf{p}\right) \cdot \pi(\mathbf{x})\,, \tag{13b}$$

where $\pi(\mathbf{x}) = \exp\Big(-\mathcal{J}(\mathbf{x})\Big)$ is the probability distribution of the position variable. The joint probability distribution of the state $(\mathbf{p}, \mathbf{x})$ in the phase space

$\mathbb{R}^{2\,\mathrm{N_{var}}}$ is the product of the marginal distributions of both the position and the momentum. This simply means that the two variables $\mathbf{x}$ and $\mathbf{p}$ are independent [36]. Independence of both position and momentum makes it possible to sample from the marginal distribution of each variable by sampling from their joint distribution. The marginal PDF of the momentum variable is a Gaussian distribution with zero mean and covariance matrix $\mathbf{M}$ (also known as the mass matrix), i.e., $\mathbf{p} \sim \mathcal{N}(0, \mathbf{M})$.

Let $\mathbf{x} \sim \pi(\mathbf{x})$ be a random variable that is the target of the MCMC sampling algorithm. View $\mathbf{x}$ as the position variable in the Hamiltonian system, and add the momentum $\mathbf{p}$ as an auxiliary variable. The symplectic integrator is used to propose a state that is either accepted or rejected using an acceptance/rejection rule based on the loss of energy. Algorithm 1 [36] summarizes the HMC steps to sample from the probability distribution $\pi(\mathbf{x})$. The loss of energy between the current and the proposed state is usually calculated as the difference between the Hamiltonians at the current and the proposed states:

Algorithm 1 HMCMC Sampling [36].

```
1: Initialize the Markov chain. Preferably $(\mathbf{p}_0, \mathbf{x}_0)$ should have high probability
   w.r.t. the target distribution.
2: At each step $k$ of the Markov chain draw the random auxiliary variable
   $\mathbf{p}_k \sim \mathcal{N}(0, \mathbf{M})$.
3: Use a symplectic numerical integrator (e.g. position Verlet) to advance the
   current state $(\mathbf{p}_k, \mathbf{x}_k)$ by a time increment $T$ to obtain a *proposal* state
   $(\mathbf{p}^*, \mathbf{x}^*)$:
       $(\mathbf{p}^*, \mathbf{x}^*) = \tilde{\Phi}_T\big((\mathbf{p}_k, \mathbf{x}_k)\big).$   (14)
4: Use the Hamiltonian (10b) to approximate the loss of energy $\Delta H$.
5: Calculate the acceptance probability:
       $a^{(k)} = 1 \wedge e^{-\Delta H}.$   (15)
6: Discard both $\mathbf{p}^*$ and $\mathbf{p}_k$.
7: *(Acceptance/Rejection)* Draw a uniform random variable $u^{(k)} \sim \mathcal{U}(0,1)$:
     i- If $a^{(k)} > u^{(k)}$ accept the proposal as the next sample: $\mathbf{x}_{k+1} := \mathbf{x}^*$;
     ii- If $a^{(k)} \le u^{(k)}$ reject the proposal and continue with the current state:
         $\mathbf{x}_{k+1} := \mathbf{x}_k$.
8: Repeat steps 2 to 7 until sufficiently many distinct samples are drawn.
```

$$\Delta H = H(\mathbf{p}^*, \mathbf{x}^*) - H(\mathbf{p}_k, \mathbf{x}_k)\,. \tag{16}$$

This equation (16) is valid for the Verlet (12), two-stage, three-stage, and four-stage symplectic integrators [12, 36]. See [7] for details on different symplectic integrators and corresponding expressions for energy. The length of the Hamiltonian trajectory T and the number of steps m are parameters to be tuned by the user [30]. Another user-tunable parameter is the mass matrix $\mathbf{M}$, a symmetric positive definite matrix that represents the covariance of the momentum variable. The choice of the mass matrix does not alter the fact that HMC sampling Algorithm 1 converges to the stationary distribution $\pi(\mathbf{x})$. However, a good choice of $\mathbf{M}$ can considerably improve sampling efficiency. One popular and simple choice is to take $\mathbf{M}$ a constant multiple of the identity matrix. Ideally, if the variances of the target distribution $\pi(\mathbf{x})$ are known (or can be approximated), the diagonal of $\mathbf{M}$ should be chosen as the corresponding precisions (reciprocals of these variances) [30]. We found that this choice results in a very fast convergence of the chain to stationarity.

HMC sampling Algorithm 1 tends to explore the state space faster than traditional MCMC, and the acceptance probability of all generated states is close to one. Several enhancements to the HMC sampling, such as parallel tempering [18, 40], have been proposed to guarantee that the algorithm escapes local modes of high probability.

3.2 Sampling smoother algorithm

We now present the HMC smoother (smoothing by sampling) that simultaneously accounts for all observations available within a specific assimilation window to obtain posterior estimates of the initial system state.

Consider the assimilation window $[t_0, t_F]$ with a set of observations available at times $t_0, t_1, \ldots, t_{\mathrm{N_{obs}}}$ inside the window, where $t_{\mathrm{N_{obs}}} \equiv t_F$. Under the assumptions discussed in Section 2.2 the posterior (analysis) probability distribution of the initial state $\mathbf{x}_0$ takes the form (7). We seek to sample from this posterior distribution using the HMC approach. For this we set the potential energy term in (10b) to be the 4D-Var cost functional (7b). Consequently the target probability distribution $\pi(\mathbf{x})$ coincides with the 4D-Var posterior distribution (7), i.e., $\pi(\mathbf{x}) = \exp(-\mathcal{J}(\mathbf{x}))$.

The smoother works sequentially *over consecutive assimilation windows* by applying the forecast and analysis (sampling) steps in succession. In the forecast step each state of the analysis ensemble is propagated in time to the end of the previous assimilation window (the beginning of the current window). The result of the forecast step is a forecast ensemble $\mathbf{X}^{\mathrm{b}}$ (or just a single background state $\mathbf{x}_0^{\mathrm{b}}$) at the beginning of the current time window, i.e. at t_0. One can just propagate the analysis state (e.g. the mean of the analysis ensemble) to obtain the current background state $\mathbf{x}_0^{\mathrm{b}}$. However, propagating the full analysis ensemble makes it possible to build an ensemble-based (flow-dependent) background error covariance matrix at the beginning of the current window. This background error covariance matrix includes the errors of the day, and can considerably enhance the quality of the analyses generated by a data assimilation scheme. We will assume herein that the full forecast ensemble is generated in the forecast step. In the analysis step, the HMC sampling strategy summarized in Algorithm 1 is applied to obtain the analysis ensemble at the initial time of the current assimilation window.

The HMC sampling smoother is detailed in Algorithm 2.

Algorithm 2 The Proposed Sampling Smoother

```
Analysis step: Given the background state and observations, draw an ensemble of initial states from the posterior distribution (7) as follows:
    i- Calculate an ensemble-based forecast error covariance matrix $\mathbf{B}_0^{\rm ens}$, and use it together with a fixed (modeled) matrix to construct the background error covariance matrix $\mathbf{B}_0$ [7]. (This step can be omitted by using the modeled background error covariance matrix; however, the use of forecast ensemble is expected to improve the quality of the generated analysis ensemble.)
    ii- Build the mass matrix $\mathbf{M}$ as a diagonal matrix such that $\operatorname{diag}(\mathbf{M}) = \operatorname{diag}\left(\mathbf{B_0^{-1}}\right)$.
    iii- Initialize the Markov chain to the best estimate of the current state available, e.g., the background state $\mathbf{x}_0^{\rm b}$, or a suboptimal 4D-Var solution. A good choice speeds up the convergence of the chain.
    iv- Follow the steps in Algorithm 1 to generate the chain and select ensemble members after the chain reaches stationarity. Dropping a small number (5 to 10) steps between each selected states helps to ensure the independence of the generated ensemble members.
Forecast step: Propagate each member of the analysis ensemble, using the full forward model, to the end of the current assimilation window (beginning the next assimilation window).
```

The generated ensemble of states $\{\mathbf{x}_0^{\rm a}(e)\}_{e=1,2,\dots,{\rm N_{ens}}}$, samples the posterior PDF $\mathcal{P}^{\rm a}(\mathbf{x}_0)$, and can be used to calculate the best estimate of the initial condition of the system (e.g., the mean $(\overline{\mathbf{x}}_0^{\rm a})$ of the ensemble), and to estimate the analysis error covariance matrix $\mathbf{A}_0$:

$$
\begin{aligned}
\overline{\mathbf{x}}_0^{\rm a} &= ({\rm N_{ens}})^{-1} \sum_{e=1}^{\rm N_{ens}} \mathbf{x}_0^{\rm a}(e)\,, && \text{(17a)} \\
\Delta\mathbf{X}_0^{\rm a} &= \left[\mathbf{x}_0^{\rm a}(1) - \overline{\mathbf{x}}_0^{\rm a}, \dots, \mathbf{x}_0^{\rm a}({\rm N_{ens}}) - \overline{\mathbf{x}}_0^{\rm a}\right], \\
\mathbf{A}_0 &= ({\rm N_{ens}} - 1)^{-1} \left(\Delta\mathbf{X}_0^{\rm a}\, (\Delta\mathbf{X}_0^{\rm a})^T\right)\,. && \text{(17b)}
\end{aligned}
$$

The forecast and analysis steps are repeated sequentially on subsequent assimilation windows. The propagated ensemble can be used to estimate the analysis covariance at the final time using (17b), such as to provide “flow-dependent” information for the background error covariance matrix used in the subsequent assimilation interval.

4 Numerical Experiments

The proposed HMC sampling smoother is tested against the EnKS and the 4D-Var schemes on two numerical models. We first illustrate the distinctive features of the HMC smoother with a simple one-dimensional model with a nonlinear observation operator and a bimodal posterior distribution. Next, we employ the shallow water on the sphere to test the sampling smoother on a problem relevant to geophysics, and to compare its performance against the conventional 4D-Var scheme.

4.1 A one-dimensional model

Consider the following model:

$$\frac{d\mathbf{x}}{dt} = -\frac{dV(\mathbf{x})}{d\mathbf{x}}\,, \tag{18a}$$

$$V(\mathbf{x}) = (\mathbf{x}+1)^2\,(\mathbf{x}-1)^2\,, \tag{18b}$$

that describes the position of a particle over the entire real line moving under the effect of the potential field (18b). This model is similar to the one used in [5]. The potential field has two local minima at ±1, which are expected to act as attractors for the particle. We set the reference initial condition to $\mathbf{x}_0^{\rm true} = -0.15$, and chose the background initial condition to be $\mathbf{x}_0^{\rm b} = 0.1$. Note that the true and the background initial conditions lie in the basins of attraction of different equilibria. The background errors are assumed to be normally distributed with zero mean and standard deviation $\sigma_{\mathbf{x}_0} = \sqrt{2}$.

Synthetic observations are obtained from the reference solution by applying the quadratic observation operator

$$\mathcal{H}(\mathbf{x}_k) = (\mathbf{x}_k)^2\,. \tag{19}$$

Observation errors are assumed to be Gaussian with zero mean and standard deviation $\sigma_{\rm obs} = 0.05$. The simulation time window is $[t_0, t_F] = [0, 0.12]$ (units), with equally spaced 12 observation points. The posterior distribution of the initial state reads

$$\begin{aligned}\mathcal{P}^{\rm a}(\mathbf{x}_0) \propto \exp\Bigg(&-\frac{1}{2}\left(\frac{\mathbf{x}_0-0.1}{1.41}\right)^2 \\ &-\frac{1}{2}\sum_{k=1}^{\rm N_{obs}=12}\left(\frac{\mathbf{x}_k^2-\mathbf{y}_k}{0.05}\right)^2\Bigg)\,,\end{aligned} \tag{20}$$

where $\mathbf{x}_k$ is obtained by propagating $\mathbf{x}_0$ forward in time, from $t_0 = 0$ to $t_k = k \times 0.01$ (units), using the model (18). The non-normalized posterior density (20) is illustrated in Figure 1. Traditional assimilation methods, like 4D-Var and EnKS, are expected to have difficulties capturing the bimodal nature of the posterior distribution of the initial condition. Since the prior PDF is

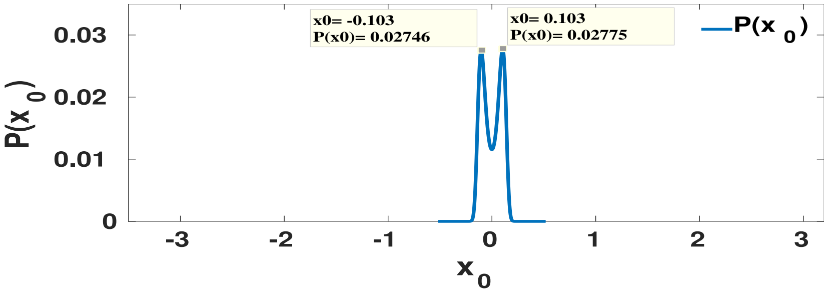


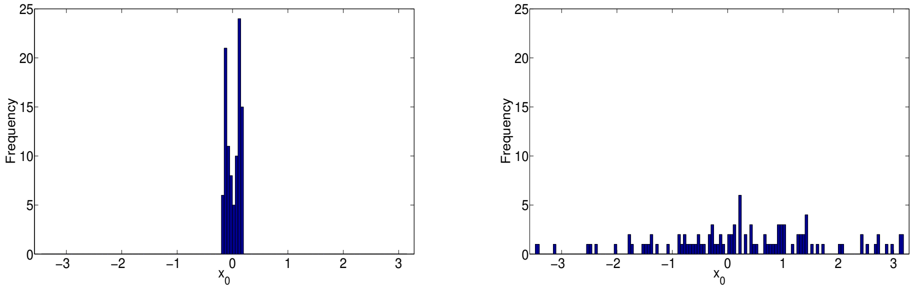
Figure 1: The non-normalized kernel of the posterior distribution (20). The right peak is slightly higher than than the left one as a result of the prior being a Gaussian distribution centered around $\mathbf{x}^{\rm b} = 0.1$ with small standard deviation ($\sigma_{\mathbf{x}_0} = \sqrt{2}$) Given the current settings, the right peak occurs at $\mathbf{x}_0 = 0.103$ with $P(\mathbf{x}_0) = 0.02775$ while the left peak occurs at $\mathbf{x}_0 = -0.103$ with $P(\mathbf{x}_0) = 0.02746$.

a Gaussian centered around the background state $\mathbf{x}^{\rm b} = 0.1$ with standard deviation $\sigma_{\mathbf{x}_0} = \sqrt{2}$, the right peak in Figure 1 is slightly taller than the left peak. With Gaussian background prior centered around one of the peaks, smaller standard deviation would damp the other peak. Capturing only that right peak completely misses the true solution, which is negative. Numerical results presented below show that the proposed HMC smoother is capable of building a representative ensemble from the bimodal posterior distribution. The analysis ensemble can then be used to draw more useful conclusions (e.g. statistics) than what can be obtained from analysis results obtained by the traditional methods.

4.1.1 Numerical results with the one-dimensional model

HMC smoothing was carried out to collect an ensemble of 100 members from the posterior (20). We tested several symplectic integrators [7], and found that all show similar behavior. We chose the position Verlet symplectic integrator due to its minimal computational cost for all our experiments. The Hamiltonian system step size is empirically tuned to $T = 0.1$, with step length $h = 0.01$, and number of steps $m = 10$. The number of burn-in steps is chosen to be 20 (for this simple model we already know that the forecast state 0.1 lies in the support of the posterior and the burn-in steps could be omitted; in general one can incorporate convergence tests to shorten the number of burn-in steps and ensure that the collected samples are from the target distribution). Four states are dropped between consecutive selected states at stationarity to guarantee the independence of the samples.

The histograms of the analysis ensembles obtained with the HMC smoother and EnKS are shown in Figure 2. The HMC smoother generates an analysis ensemble that matches the kernel shown in Figure 1, but EnKS fails to generate an accurate analysis ensemble. The most likely state seems to be located in



(a) HMC sampling smoother.,height=3.8cm

(b) Ensemble Kalman smoother.

Figure 2: Histograms of the analysis ensembles generated by HMC smoother, and EnKS. The number of ensemble members generated by each smoother is 100. For the HMC smoother the step of the symplectic integrator (12) is $T = 0.1$, with $h = 0.01$ and $m = 10$.

the correct place. A single analysis state (best estimate) in this case might be misleading. One needs to consider more than one analysis with certain probability to give better description of the true state of the system in case of multi-modal systems.

The 4D-Var algorithm is expected to be trapped in a local minimum of the posterior distribution. Since the background state is closer to +1 than to −1, and since the observations (19) are insensitive to the sign of solution, we expect the analysis to follow the behavior of the true solution but with the opposite sign. This is confirmed by results in Figure 3.

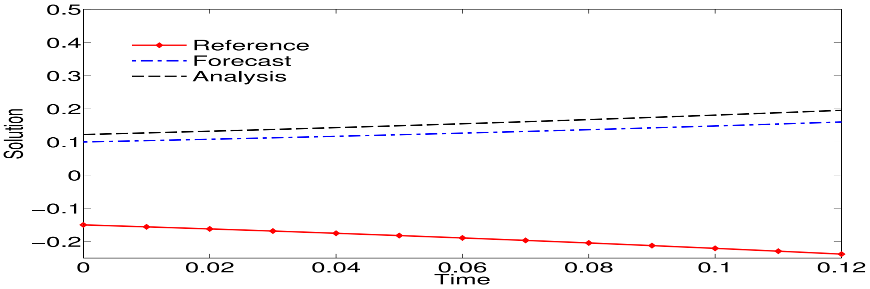


Figure 3: Data assimilation results using 4D-Var, together with the forecast and reference trajectories plotted over the assimilation window.

4.2 Shallow water model on the sphere

The shallow water equations have been used extensively as a simplified model of the atmosphere that contains the essential wave propagation mechanisms found in general circulation models (GCMs)[39]. The shallow water equations in spherical coordinates are [1]:

$$\begin{aligned}
&\frac{\partial u}{\partial t}+\frac{1}{a\cos\theta}\left(u\frac{\partial u}{\partial \lambda}+v\cos\theta\frac{\partial u}{\partial \theta}\right) \qquad\qquad (21a)\\
&\quad-\left(f+\frac{u\tan\theta}{a}\right)v+\frac{g}{a\cos\theta}\frac{\partial h}{\partial \lambda}=0,\\
&\frac{\partial v}{\partial t}+\frac{1}{a\cos\theta}\left(u\frac{\partial v}{\partial \lambda}+v\cos\theta\frac{\partial v}{\partial \theta}\right) \qquad\qquad (21b)\\
&\quad+\left(f+\frac{u\tan\theta}{a}\right)u+\frac{g}{a}\frac{\partial h}{\partial \theta}=0,\\
&\frac{\partial h}{\partial t}+\frac{1}{a\cos\theta}\left(\frac{\partial\,(hu)}{\partial \lambda}+\frac{\partial(hv\cos\theta)}{\partial \theta}\right)=0\,. \qquad (21c)
\end{aligned}$$

Here f is the Coriolis parameter, given by $f=2\Omega\sin\theta$, where Ω is the angular speed of the rotation of the Earth. In addition, h represents the height of the homogeneous atmosphere, u and v are the zonal and meridional wind components, respectively. The latitudinal and longitudinal directions are respectively denoted by θ and λ. The radius of the Earth is denoted by a and g is the acceleration due to gravity. The space discretization is performed using the unstaggered Turkel-Zwas scheme [28]. The discretization has nlon=72 nodes in longitudinal direction and nlat=36 nodes in the latitudinal direction. The semi-discretization in space leads to a system of ordinary differential equations:

$$\mathbf{x}'=f(t,\mathbf{x}),\quad \mathbf{x}(t_0)=\mathbf{x}_0;\quad t_0=0,\ t_F=9\,(\text{hours}). \qquad (22)$$

The vector $\mathbf{x}\in\mathbb{R}^n$ with $n=3\times\text{nlat}\times\text{nlon}$ contains discrete versions of the zonal wind, meridional wind, and the height variables. We perform the time integration using a 5^{th} order Runge-Kutta method. This time-integrator is part of the MATLODE suite [15], which also has sensitivity analysis capabilities.

4.2.1 Observations and background information

A reference initial condition is used to generate a reference trajectory. Synthetic linear observations are created from the reference trajectory by adding Gaussian noise with zero mean for each of the three components. The observation error standard deviation for height component is set to 1.5% of the average magnitude of the reference height component in the reference initial condition. The observation error standard deviation for wind components is set to 10% of the average magnitude of the reference wind component in the initial condition. The initial background state is created by perturbing the reference initial condition with a Gaussian error drawn from the distribution $\mathcal{N}(0,\mathbf{B}_0)$, with a modeled background error covariance matrix. The background error covariance $\mathbf{B}_0$ is modeled as follows:

- Start with a diagonal background error covariance matrix. The standard deviation of the background errors for the height component is 2% of the average magnitude of the reference height component in the reference initial condition. The standard deviation of the background errors for the wind components is 15% of the average magnitude of the reference wind component in the reference initial condition.

- Synthetic initial ensemble is created by adding zero-mean Gaussian noise to the reference initial condition with covariances set to the initial (diagonal) background error covariance matrix. Apply the ensemble Kalman filter for 48 cycles with observations obtained each hour. The uncertainties in observations are fixed to 1.5%, and 10% for the height and wind components respectively. The synthetic observations are obtained by adding Gaussian noise with zero mean and standard deviation equal to the uncertainty level multiplied by the average magnitude of the corresponding component (height and wind) in the initial condition.
- Decorrelate the ensemble-based covariances using a decorrelation matrix ρ with decorrelation distance $L = 1000\,km$.
- Calculate $\mathbf{B}_0$ by averaging the ensemble covariances over the last 6 hours with one matrix per hour. In this version the background noise levels are no longer 2% and 15%.

This method of creating a synthetic initial background error covariance matrix is empirical, but we found that the resulting background error covariance matrix performs well for several algorithms including 4D-Var. Enhancing the quality of this background error covariance matrix can be done by making use of the ensembles generated by the sampling smoother, an idea that we will investigate in future work.

Data assimilation experiments with this model were conducted for three consecutive assimilation windows. The time interval of the first assimilation window is $[0, 6]$ hours, the second window is $[6, 14]$ hours, and the third is $[14, 20]$ hours. The short first window can be regarded as a spin-off period for the data assimilation system. Hourly (synthetic) observations are available each of the three windows, with a total of 6 observation times in the first window, and 8 observation times in each of the last two windows.

Two experiments were conducted. In the first one the background error covariance matrix $\mathbf{B}_0$ is kept fixed for each of the three windows. In the second experiment $\mathbf{B}_0$ is updated with information from the generated ensemble according to the following expression:

$$\mathbf{B}_0^{\text{hybrid}} = \gamma \times \mathbf{B}_0^{\text{modeled}} + (1-\gamma) \times \mathbf{B}_0^{\text{ensemble}}\,, \tag{23}$$

where $\mathbf{B}_0^{\text{hybrid}}$ is the updated version of $\mathbf{B}_0$, and $\mathbf{B}_0^{\text{modeled}}$ is the fixed version used in the first experiment. The scalar weight γ is a number in the interval $[0, 1]$. Selecting $\gamma = 1$ ignores the error-of-the-day, while $\gamma = 0$ forces the use of only the flow-dependent background error covariance matrix obtained from the ensemble, possibly leading to a singular covariance matrix. In our experiments we chose $\gamma = 0.75$.

The error metric used to compare analyses against the reference solution is the root mean squared error (RMSE):

$$\mathbf{RMSE} = \sqrt{\frac{1}{\mathrm{N_{var}}} \sum_{i=1}^{\mathrm{N_{var}}} (\mathbf{x}_i - \mathbf{x}_i^{\mathrm{true}})^2}\,, \tag{24}$$

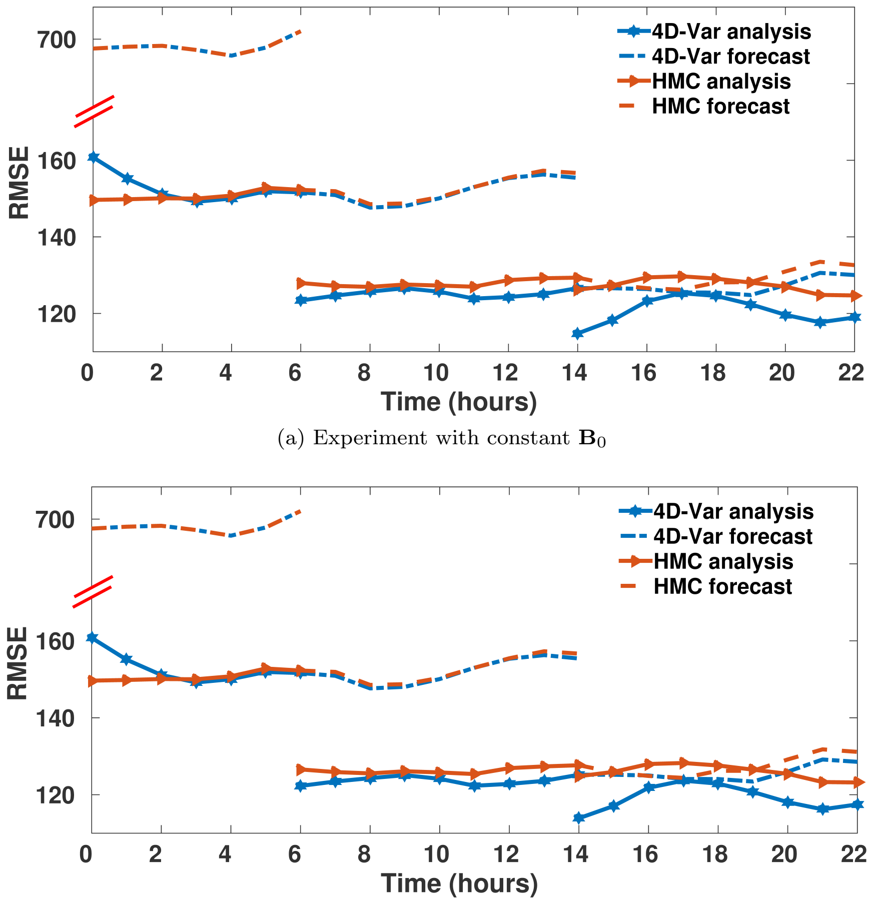
where $\mathbf{x}^{\mathrm{true}}$ is the reference state of the system. The RMSE is calculated hourly along the trajectory over each assimilation window.

4.2.2 Numerical results with the shallow water on the sphere model

The numerical optimization step in 4D-Var is carried out using the the LBFGS routine implemented in the Poblano optimization toolbox [17]. Here the optimization process is stopped when the norm of the gradient is $1e-10$ or when the relative function tolerance hits $1e-6$. The optimization process takes at least 45 iterations of LBFGS to converge for the experiment considered here; (see Table 1) .

The HMC sampling smoother is used to generate 100 ensemble members in each assimilation window. The symplectic integrator used is Verlet (12) with an empirically tuned step length of $T_* = 0.1$, with $h_* = 0.01$, and $m = 10$. A practically useful recipe is to perturb the step length of the symplectic integrator, a procedure that guarantees that the results obtained are not contingent on that specific selection of step settings [12, 30]. The step length h is perturbed with uniform random noise: $h := (1+r) \times h_*$, $r \sim \mathcal{U}(-0.2, 0.2)$. It is important to notice that the step h is pertubed only once at the beginning of the Hamiltonian trajectory and kept fixed for all the m steps. This actually means that the length of the Hamiltonian trajectory T is perturbed for each proposal state while keeping the number of steps m constant such that at each run the step size h scales accordingly

The number of burn-in steps is set to 30. We noticed that the HMC smoother converges to the posterior in much fewer steps $(5-10)$. Four generated states are discarded between each selected state in the ensemble to guarantee independence of the generated ensemble members.



(a) Experiment with constant $\mathbf{B}_0$

(b) Experiment with hybrid $\mathbf{B}_0$ using (23)

Figure 4: Data assimilation results for two scenarios using linear observations are shown. The first panel 4a shows RMSE with B_0 being fixed. The second panel 4b shows RMSE with B_0 being updated using (23). The symplectic integrator used in both cases is Verlet (12) with step $T = 0.1$, where $h = 0.01$, and $m = 10$. The number of dropped states between selected samples is 4.

The RMS errors for both HMC smoother and 4D-Var over the three assimilation windows are shown in Figure 4. Figure 4a reports the case where the background error covariance matrix $\mathbf{B}_0$ is kept fixed, and Figure 4b shows the case where $\mathbf{B}_0$ is updated, at the beginning of each assimilation window, according to equation (23). The quality of the analyses in both cases is very similar, however in the second case a slight reduction in RMSE is noticed along the entire trajectory. This is appreciated by Figure 5 zooming onto the RMSE results over the second assimilation window.

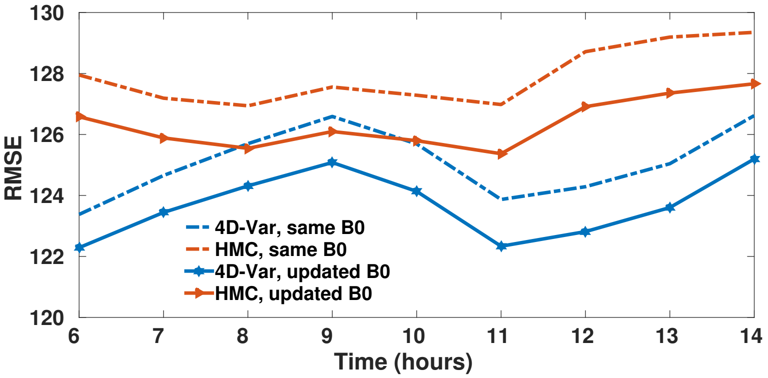
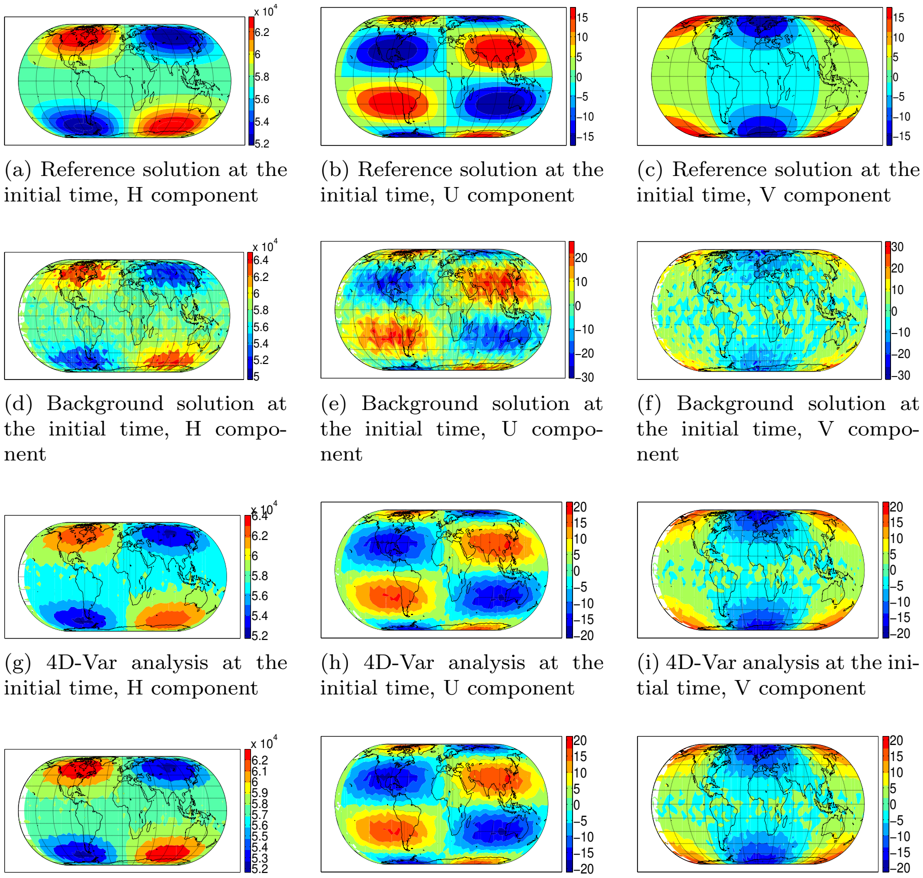


Figure 5: Same as results in Figure 4 with only RMS errors over the second window displayed for two scenarios. RMS errors obtained while keeping $\mathbf{B}_0$ fixed are plotted as dotted lines. RMS errors obtained with $\mathbf{B}_0$ being updated at the beginning of the assimilation window are plotted as dotted lines.

The HMC smoother can sample efficiently from the posterior distribution and resulting analysis competes in accuracy with that obtained using 4D-Var. Figure 6 shows the three components at the beginning of the first window for the reference solution, the background state, 4D-Var analysis, and HMC smoother analysis (ensemble mean). Results shown in The analysis recovered from the noisy background by both 4D-Var and HMC smoother are almost identical.

The assimilation results obtained over the next two windows with $\mathbf{B}_0$ kept fixed are shown in Figures 7 and 8. The performance of the two schemes, 4D-Var, and the HMC smoother is quite similar, and the HMC smoother analysis competes with the 4D-Var analysis.

Updating the background error covariance can, in principle, enhance the performance of both the 4D-Var and the HMC smoother. In the case of 4D-Var, updating $\mathbf{B}_0$ results in lower RMSE which indicates that in real applications, the analysis is expected to be closer to reality. In addition to more accurate prior kernel, updating $\mathbf{B}_0$ will result in a better update of the mass matrix $\mathbf{M}$ which in turn is expected to result in better performance of the smoother. In our experiments the update has a small positive impact on the performance of the two data assimilation schemes as explained in Figures 4, 5. The positive effect here is explained by reduction in the RMS errors. The resulting ensemble-based forecast error covariance matrix that is used to update $\mathbf{B}_0$ can be crucial for cases where observations are sparse or not uniformly distributed over the grid, and therefore well worth the computational overhead of the forward propagation of all the analysis ensemble members to build the full forecast ensemble. Results of the data assimilation process with hybrid (updated) background error covariance matrix on the next two windows are shown in Figures 9 and 10.



(a) Reference solution at the initial time, H component

(b) Reference solution at the initial time, U component

(c) Reference solution at the initial time, V component

(d) Background solution at the initial time, H component

(e) Background solution at the initial time, U component

(f) Background solution at the initial time, V component

(g) 4D-Var analysis at the initial time, H component

(h) 4D-Var analysis at the initial time, U component

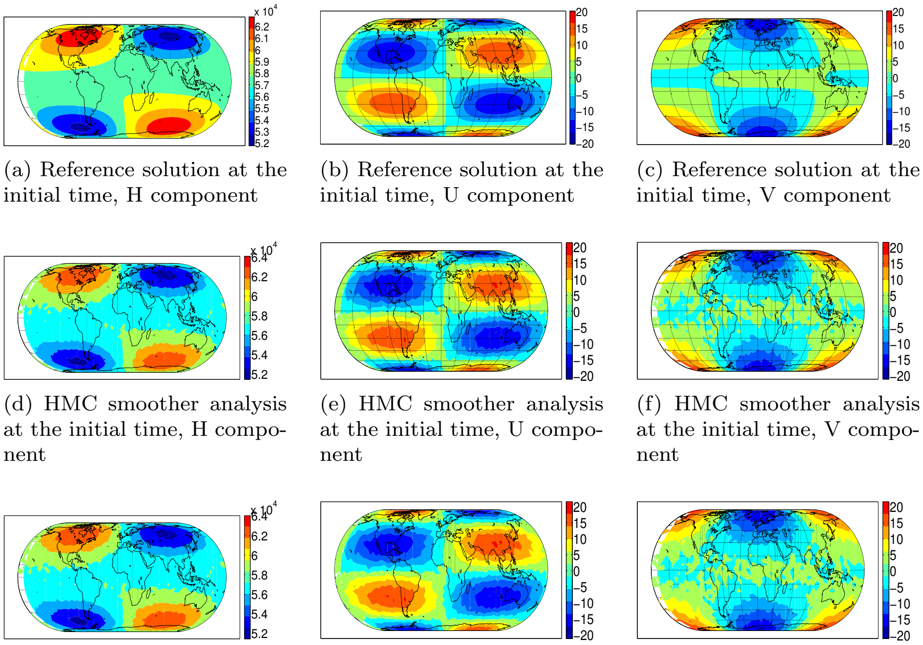
(i) 4D-Var analysis at the initial time, V component

(j) HMC smoother analysis at the initial time, H component

(k) HMC smoother analysis at the initial time, U component

(l) HMC smoother analysis at the initial time, V component

Figure 6: Four dimensional data assimilation results with linear observations. The initial condition solutions at the beginning of the first window are shown. The data assimilation scheme and the state components are indicated under each panel. The assimilation window length is 6 hours, with hourly observations. The background error covariance matrix $\mathbf{B}_0$ is kept fixed.



(a) Reference solution at the initial time, H component

(b) Reference solution at the initial time, U component

(c) Reference solution at the initial time, V component

(d) HMC smoother analysis at the initial time, H component

(e) HMC smoother analysis at the initial time, U component

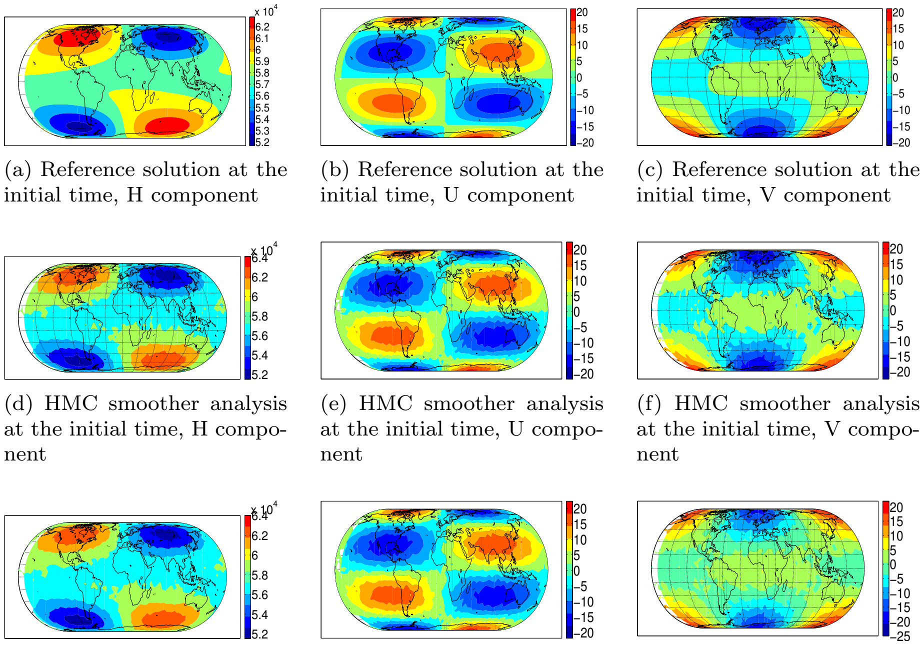
(f) HMC smoother analysis at the initial time, V component

(g) 4D-Var analysis at the initial time, H component

(h) 4D-Var analysis at the initial time, U component

(i) 4D-Var analysis at the initial time, V component

Figure 7: Four dimensional data assimilation results with linear observations. The initial condition solutions at the beginning of the second window are shown. The data assimilation scheme and the state components are indicated under each panel. The assimilation window length is 8 hours, with hourly observations. The background error covariance matrix $\mathbf{B}_0$ is not updated.



(a) Reference solution at the initial time, H component

(b) Reference solution at the initial time, U component

(c) Reference solution at the initial time, V component

(d) HMC smoother analysis at the initial time, H component

(e) HMC smoother analysis at the initial time, U component

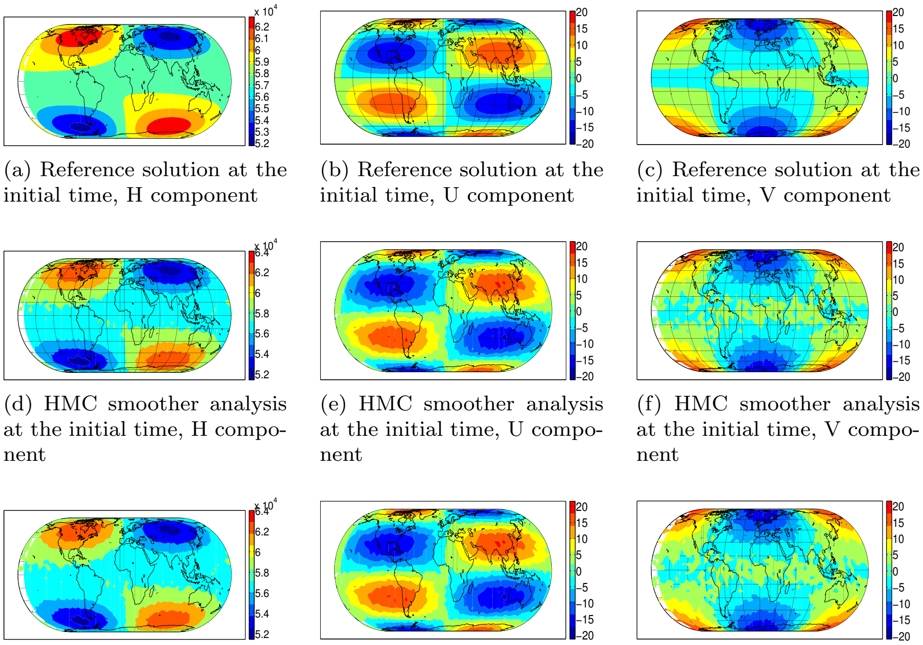
(f) HMC smoother analysis at the initial time, V component

(g) 4D-Var analysis at the initial time, H component

(h) 4D-Var analysis at the initial time, U component

(i) 4D-Var analysis at the initial time, V component

Figure 8: Four dimensional data assimilation results with linear observations. The initial condition solutions at the beginning of the third window are shown. The data assimilation scheme and the state components are indicated under each panel. The assimilation window length is 8 hours, with hourly observations. The background error covariance matrix $\mathbf{B}_0$ is kept fixed.



(a) Reference solution at the initial time, H component

(b) Reference solution at the initial time, U component

(c) Reference solution at the initial time, V component

(d) HMC smoother analysis at the initial time, H component

(e) HMC smoother analysis at the initial time, U component

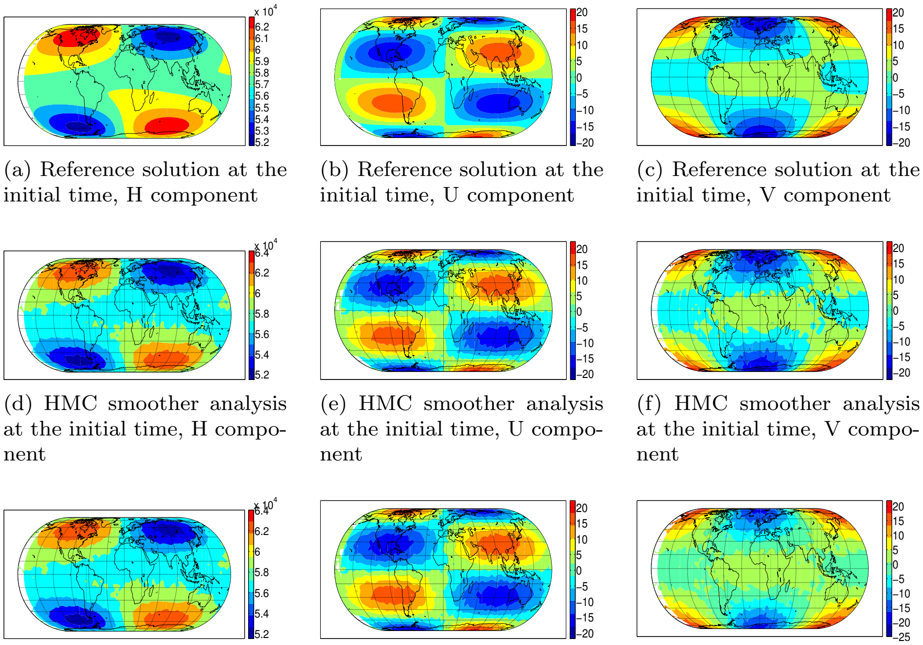
(f) HMC smoother analysis at the initial time, V component

(g) 4D-Var analysis at the initial time, H component

(h) 4D-Var analysis at the initial time, U component

(i) 4D-Var analysis at the initial time, V component

Figure 9: Four dimensional data assimilation results with linear observations. The initial condition solutions at the beginning of the second window are shown. The data assimilation scheme and the state components are indicated under each panel. The assimilation window length is 8 hours, with hourly observations. The background error covariance matrix $\mathbf{B}_0$ is updated using (23) for both schemes.



(a) Reference solution at the initial time, H component

(b) Reference solution at the initial time, U component

(c) Reference solution at the initial time, V component

(d) HMC smoother analysis at the initial time, H component

(e) HMC smoother analysis at the initial time, U component

(f) HMC smoother analysis at the initial time, V component

(g) 4D-Var analysis at the initial time, H component

(h) 4D-Var analysis at the initial time, U component

(i) 4D-Var analysis at the initial time, V component

Figure 10: Four dimensional data assimilation results with linear observations. The initial condition solutions at the beginning of the third window are shown. The data assimilation scheme and the state components are indicated under each panel. The assimilation window length is 8 hours, with hourly observations. The background error covariance matrix $\mathbf{B}_0$ is updated using (23) for both schemes.

4.3 Computational considerations

Both 4D-Var and HMC smoother require the same computational infrastructure, namely, an adjoint model that computes the gradient of the 4D-Var cost functional (7b). This gradient calculation is the computational bottleneck for both 4D-Var and HMC smoother. It requires forward propagation of the model, and a backward propagation of the adjoint model. In our shallow water test the cost of one adjoint model run is approximately equivalent to 2.5 times the cost of one forward model run. This makes the cost of gradient evaluation approximately equal to 3.5 the cost of a forward model run.

Table 1: Data assimilation scheme cost for the shallow water model with linear observations. Number of function evaluations (forward model runs), and the number of optimization iterstions (for 4D-Var) are listed. The cost of one adjoint run is approximately equal to 2.5 forward model runs in the current settings. Total cost is approximated in terms of number of forward model runs. The cost of the HMC sampling smoother is the same for the three assimilation windows.

Data assimilation scheme	**Cost**	**Assimilation window**				
		(1)	(2)		(3)	
		Fixed $\mathbf{B}_0$	Fixed $\mathbf{B}_0$	Hybrid $\mathbf{B}_0$	Fixed $\mathbf{B}_0$	Hybrid $\mathbf{B}_0$
4D-Var	Function evaluations	151	97	101	96	93
	Number of iterations	49	47	46	46	45
	Cost in equivalent forward model runs	322.5	261.5	262	257	250.5
HMC smoother	Number of proposed states	530				
	Cost per proposal	4.5				
	Cost in equivalent forward model runs	2,385				

The cost of the 4D-Var depends on the number of iterations and function evaluations required by the optimization algorithm. On the first window the number of iterations required by the LBFGS optimizer is 49, with 151 function evaluations. The total cost of the 4D-Var solution is then $151 + 49 \times 3.5 = 322.5$ equivalent forward model runs.

The cost of the HMC smoother depends on the configuration of the chain: the number of burn-in step, the number of dropped states at stationarity, and step-size settings of the symplectic integrator. The symplectic integrator itself controls the number of adjoint runs to evaluate the gradient of the cost functional in order to propose a new state to the chain. The size of the desired ensemble controls the length of the Markov chain and consequently the total cost of the analysis step by the HMC smoother. The Verlet integrator (12), used in the current experiments, requires a single adjoint run to propose a new state to the chain. The acceptance/rejection criterion requires an additional forward run to evaluate the loss of energy. This makes the cost of generating a proposal state to the Markov chain approximately equal to 4.5 the cost of a model run. On all assimilation windows the chosen ensemble size is 100. The number of burn-in states is 30, and 4 states are rejected between consecutive selected samples. The HMC sampling smoother, in this case generates $30 + 100 \times 5 = 530$ states to collect the analysis ensemble, with a total cost roughly equal to $530 \times 4.5 = 2,385$ forward model runs.

The computational cost of the two data assimilation schemes, 4D-Var and HMC smoother, on each assimilation window are summarized in Table 1. Notice that the total cost of DA schemes is given in terms of the total number of forward model runs. On the first window the cost of the HMC smoother is approximately 9 times the cost of the 4D-Var scheme. On the next two windows, the HMC smoother costs roughly 11 times the cost of the 4D-Var. A cost-reduction in 4D-Var is expected because the forecast state is closer to the MAP than the case on the first window. The higher computational cost of the HMC smoother can be handled more efficiently by parallelizing the sampling scheme and the gradient calculations. Another way to reduce the computational cost of the HMC smoother is to replace the burn-in steps with a suboptimal 4D-Var obtained using a small number of iterations. The computational cost of the proposed sampling smoother can of course by be reduced by decreasing the ensemble size, however this will result in higher sampling error. The impact of the sampling errors can be assessed by the techniques developed in [2, 32].

The increased cost of the HMC smoother could be acceptable in view of the additional useful information it provides: a sample estimate of the analysis probability distribution (and as immediate consequences an analysis error covariance matrix and a flow-dependent background error covariance matrix for the next cycle).

5 Conclusion and Future Work

A four-dimensional data assimilation smoother is proposed in this paper. The smoother samples from the posterior distribution using a Hybrid Monte-Carlo approach. The 4D-Var approach provides a MAP estimate of the true state, but it does not compute a measure of uncertainty of the analysis. The HMC smoother builds an ensemble approximating the posterior PDF. This can be used to estimate the true state together with the uncertainty in analysis, e.g., by calculating the ensemble mean and ensemble-based analysis error covariance matrix. Moreover, propagating the analysis ensemble to the beginning of the next assimilation window provides a forecast ensemble that can be used to construct a flow-dependent background covariance matrix for this new window. Unlike several popular hybrid approaches, the HMC smoother generates an analysis error covariance that is consistent with the analysis state – because both statistics are produced by one consistent data assimilation scheme.

The HMC smoother requires an adjoint of the simulation model, and runs on the same computational infrastructure as 4D-Var. The computational cost of the HMC smoother is - as of now - larger than that of 4D-Var. The efficiency issue must be addressed before the HMC smoother becomes fully practical. We are currently investigating several strategies to enhance the performance of the sampling smoother and to reduce its computational cost. Parallelizing the sampling smoother will be considered. We will also test the HMC smoother on the case of nonlinear observations, imperfect models, and non-Gaussian errors.

Acknowledgements

This work was partially supported by awards AFOSR FA9550–12–1–0293–DEF, NSF CCF–1218454, NSF DMS–1419003, and by the Computational Science Laboratory at Virginia Tech.

References

References

[1] Exshall: A Turkel-Zwas explicit large time-step FORTRAN program for solving the shallow-water equations in spherical coordinates. Computers and Geosciences **17**(9), 1311 – 1343 (1991)

[2] A-posteriori error estimates for variational inverse problems. SIAM/ASA Journal on Uncertainty Quantification **Submitted** (2014)

[3] An optimization framework to improve 4d-var data assimilation system performance. Journal of Computational Physics **275**(0), 377 – 389 (2014)

[4] An efficient implementation of the ensemble Kalman filter based on an iterative ShermanMorrison formula. Statistics and Computing **25**(3) (2015). DOI 10.1007/s11222-014-9454-4

[5] Alexander, F.J., Eyink, G.L., Restrepo, J.M.: Accelerated monte carlo for optimal estimation of time series. Journal of Statistical Physics **119**(5-6), 1331–1345 (2005)

[6] Anderson, J.: An ensemble adjustment Kalman filter for data assimilation. Monthly Weather Review **129**, 2884–2903 (2001)

[7] Attia, A., Sandu, A.: A hybrid Monte Carlo sampling filter for non-gaussian data assimilation. Quarterly Journal of the Royal Meteorological Society **Submitted** (2014)

[8] Bennett, A.F., Chua, B.S.: Open-ocean modeling as an inverse problem: the primitive equations. Monthly weather review **122**(6), 1326–1336 (1994)

[9] Beskos, A., Pillai, N., Roberts, G., Sanz-Serna, J.M., Stuart, A.: Optimal tuning of the hybrid Monte Carlo algorithm. Bernoulli **19**(5A), 1501–1534 (2013)

[10] Beskos, A., Pinski, F., Sanz-Serna, J.M., Stuart, A.: Hybrid Monte Carlo on Hilbert spaces. Stochastic Processes and their Applications **121**(10), 2201–2230 (2011)

[11] Bishop, C., Etherton, B., Majumdar, S.: Adaptive sampling with the ensemble transform Kalman filter. Part I: Theoretical aspects. Monthly Weather Review **129**, 420–436 (2001)

[12] Blanes, S., Casas, F., Sanz-Serna, J.M.: Numerical integrators for the hybrid Monte Carlo method. arXiv preprint arXiv:1405.3153 (2014)

[13] Briers, M., Doucet, A., Maskell, S.: Smoothing algorithms for state-space models. Annals of the Institute of Statistical Mathematics **62**(1), 61–89 (2010)

[14] Cheng, H., Jardak, M., Alexe, M., Sandu, A.: A hybrid approach to estimating error covariances in variational data assimilation. Tellus A **62**(3), 288–297 (2010)

[15] D'Augustine, A., Sandu, A.: MATLODE: A MATLAB ODE solver and sensitivity analysis toolbox. ACM Transactions on Mathematical Software (TOMS) **Submitted** (2015)

[16] Duane, S., Kennedy, A., B.J. Pendleton, J.B., Roweth, D.: Hybrid Monte Carlo. Physics Letters B **195**(2), 216–222 (1987)

[17] Dunlavy, D., Kolda, T., Acar, E.: Poblano v1. 0: A MATLAB toolbox for gradient-based optimization. Sandia National Laboratories, Tech. Rep. SAND2010-1422 (2010)

[18] Earl, D.J., Deem, M.W.: Parallel tempering: Theory, applications, and new perspectives. Physical Chemistry Chemical Physics **7**(23), 3910–3916 (2005)

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

[20] Evensen, G.: The ensemble Kalman filter: theoretical formulation and practical implementation. Ocean Dynamics **53** (2003)

[21] Evensen, G., van Leeuwen, P.: Assimilation of Geosat altimeter data for the Agulhas Current using the ensemble Kalman filter with a quasigeostrophic model . Monthly Weather Review **124**, 85–96 (1996)

[22] Evensen, G., van Leeuwen, P.: An ensemble Kalman smoother for nonlinear dynamics . Monthly Weather Review **128**, 1852–1867 (2000)

[23] G., E.: Data Assimilation: The ensemble Kalman filter. Springer, Berlin (2007)

[24] Gustafsson, N., Bojarova, J.: Four-dimensional ensemble variational (4D-En-Var) data assimilation for the high resolution limited area model (HIRLAM). Nonlinear Processes in Geophysics **21**(4), 745–762 (2014)

[25] Hunt, B.R., Kalnay, E., Kostelich, E., Ott, E., patil, D., sauer, T., Szunyogh, I., Yorke, J., Zimin, A.: Four-dimensional ensemble kalman filtering. Tellus **56**(4), 273–277 (2004)

[26] Law, K.J.H., Stuart, A.M.: Evaluating data assimilation algorithms. Monthly Weather Review **140**(11), 3757–3782 (2012)

[27] Metropolis, N., Rosenbluth, A., Rosenbluth, M., Teller, A., Teller, E.: Equation of state calculations by fast computing machines. The Journal of Chemical Physics **21**(6), 1087–1092 (1953)

[28] Navon, I.M., De Villiers, R.: The application of the Turkel-Zwas explicit large time-step scheme to a hemispheric barotropic model with constraint restoration. Monthly weather review **115**(5), 1036–1052 (1987)

[29] Neal, R.: Probabilistic inference using Markov chain Monte Carlo methods. Department of Computer Science, University of Toronto Toronto, Ontario, Canada (1993)

[30] Neal, R.: MCMC using Hamiltonian dynamics. Handbook of Markov chain Monte Carlo (2011)

[31] Nino Ruiz, E., Sandu, A.: A derivative-free trust region framework for variational data assimilation. Computational and Applied Mathematics **In print** (2015)

[32] Rao, V., Sandu, A.: A posteriori error estimates for dddas inference problems. Procedia Computer Science **29**, 1256–1265 (2014)

[33] Ravela, S., McLaughlin, D.: Fast ensemble smoothing. Ocean Dynamics **57**(2), 123–134 (2007)

[34] Sandu, A., Chai, T.: Chemical data assimilationan overview. Atmosphere **2**(3), 426–463 (2011)

[35] Sandu, A., Cheng, H.: A subspace approach to data assimilation and new opportunities for hybridization. International Journal for Uncertainty Quantification **In print** (2015)

[36] Sanz-Serna, J.: Markov chain Monte Carlo and numerical differential equations. In: Current Challenges in Stability Issues for Numerical Differential Equations, pp. 39–88. Springer (2014)

[37] Sanz-Serna, J., M-P.Calvo: Numerical Hamiltonian problems, vol. 7. Chapman & Hall London (1994)

[38] Sasaki, Y.: Some basic formalisms in numerical variational analysis. Monthly Weather Review **98**(12), 875–883 (1970)

[39] St-Cyr, A., Jablonowski, C., Dennis, J., Tufo, H., Thomas, S.: A comparison of two shallow water models with nonconforming adaptive grids. Monthly Weather Review **136**, 1898–1922 (2008)

[40] Swendsen, R.H., Wang, J.S.: Replica Monte Carlo simulation of spin-glasses. Physical Review Letters **57**(21), 2607 (1986)

[41] Trémolet, Y.: Accounting for an imperfect model in 4D-Var. Quarterly Journal of the Royal Meteorological Society **132**(621), 2483–2504 (2006)

[42] Whitaker, J., Hamill, T.M.: Ensemble data assimilation without perturbed observations. Monthly Weather Review **130**, 1913–1924 (2002)