A Path Integral Method for Data Assimilation

Juan M. Restrepo [a]

[a] *Department of Mathematics and Department of Physics, University of Arizona Tucson, AZ 85721 U.S.A.*

Abstract

Described here is a path integral, sampling-based approach for data assimilation, of sequential data and evolutionary models. Since it makes no assumptions on linearity in the dynamics, or on Gaussianity in the statistics, it permits consideration of very general estimation problems. The method can be used for such tasks as computing a smoother solution, parameter estimation, and data/model initialization.

Speedup in the Monte Carlo sampling process is essential if the path integral method has any chance of being a viable estimator on moderately large problems. Here a variety of strategies are proposed and compared for their relative ability to improve the sampling efficiency of the resulting estimator. Provided as well are details useful in its implementation and testing.

The method is applied to a problem in which standard methods are known to fail, an idealized flow/drifter problem, which has been used as a testbed for assimilation strategies involving Lagrangian data. It is in this kind of context that the method may prove to be a useful assimilation tool in oceanic studies.

Key words: data assimilation, Lagrangian data assimilation, sampling, Markov Chain Monte Carlo, hybrid Monte Carlo.

PACS: 94.05sx, 92.60.Ry, 92.60.Wc, 93.65.+e, 92.70.-j, 02.50.-r

1 Introduction

Data assimilation refers to finding an estimator that derives its statistical nature from a model as well as data; the model or the data may have a stochastic component, which are referred to as model error and measurement error, respectively. The model error may be a stochastic parametrization of unresolved physics, for example. The stochastic component of both the model and the measurements are assumed to be known. The state vector for which an estimator is sought can consist of dynamic quantities (e.g., physical variables) as well as parameters. Hence the dimension of the state variable and the dynamic variable may be different. Furthermore, the dimension of the measurement vector may be different from that of the state vector; if the former is smaller the assimilation is sometimes called a "hidden variable" state estimation problem.

A straightforward way to combine the influence of model and data in the estimation is to make use of Bayesian ideas. In the method to be presented the model and data are used to propose the likelihood and prior. In what follows we will focus on the time-dependent problem and thus the estimators are conditional moments of the history of the state variables. Typically the mean and uncertainty of the history is sought. However, the method presented here

Email address: restrepo@math.arizona.edu (Juan M. Restrepo).

is sample-based and thus it is straightforward to compute sample moments of histories, without requiring added storage.

The conditional mean is the best estimate of the state, and the conditional covariance matrix provides a measure of its uncertainty. Of all estimators, the conditional mean is distinguished as the one which minimizes the trace of the conditional covariance matrix, i.e., the variance-minimizing estimator, or "smoother" estimate (a corresponding set of statistics using only the currently available set of measurements from prior times is called the "filter" estimate).

For linear dynamics and Gaussian error statistics an optimal smoother of the history and its uncertainty is provided by variance-minimizing least-squares, the variational data assimilation approach or the Kalman filter/smoother (see Wunsch (1996) for details on these). Two commonly used techniques in nonlinear and possibly non-Gaussian contexts are the extended Kalman filter/smoother, and the ensemble Kalman filter/smoother which uses a linear forecast but makes use of sampling techniques to update the uncertainty (see Evensen (1997) and references contained therein). Ensemble Kalman approaches and the variational approach (4DVar) are presently being evaluated in operational forecast models for weather and climate. The physics underlying these types of problems is generally nonlinear, and to a certain degree, non-Gaussian. If one ignores the issue of statistical convergence of the estimates, the remarkable thing is that these estimation methods work better than one would think is possible, at least in controlled numerical experiments and under the stipulation that only the mean and the variance are to be examined. (See Gilmour et al. (2001), Lawless et al. (2005) for relevant discussions). Not

withstanding, it is not surprising that linear methods are expected to produce poor estimates (for example, not be capable of minimizing the variance) or outright failing in getting the first moment. The propensity to failure can be significantly exacerbated when there are very few observations and/or when the confidence in the quality of the data is low.

This paper, along with the one by Alexander et al. (2005), presents a sampling-based strategy to data assimilation we call the *path integral Monte Carlo* approach (PIMC). By construction PIMC will yield the optimal estimate for the discretized nonlinear/non-Gaussian problem. As will be seen, whether PIMC can track the estimate is not the main issue standing in the way of its application, but rather, whether the computational expense is justified for a given problem. Obviously, the computational cost becomes less of an issue on problems that are amenable to this method and impossible to more conventional assimilation techniques. Hence, the method would hardly be of interest wherever a conventional least-squares based method will be suitable. There are a number of data assimilation strategies which specifically target problems in which nonlinearity and/or non-Gaussianity pose major challenges. Among them are: the optimal approach of Kushner (1962) (see as well Kushner (1967a), Kushner (1967b). Also, see Stratonovich (1960) and Pardoux (1982)); the mean-field variational strategy of Eyink et al. (2004) (see also Eyink and Restrepo (2000), and Eyink et al. (2002)); particle methods, such as is described in Leeuwen (2003) and Kim et al. (2003); direct Monte Carlo sampling (see Pham (2001)); and the Langevin approaches, such as that of Hairer et al. (2005). The method of Kushner (1962) yields an optimal filter/smoother and can be used as a benchmark for other methods (see Eyink

et al. (2004) for details of the methodology, its computational aspects, and how it was used as a benchmark in a simple problem). A common trait of the methods specifically developed for nonlinear/non-Gaussian problems is that they are computationally intensive and thus impractical for problems with a sufficiently large dimension, *e.g.* weather forecasting models. This dimensional/computational constraint, however, should not be construed as a practical failure; not every time dependent estimation problem of interest has dimensions comparable to those of the weather forecasting problem. Furthermore, these methods can be used to benchmark the results of operational methods for which optimality bounds are unavailable; seldom do test involve checking statistical convergence beyond the first two moments.

PIMC is considerably easier to implement than many of the nonlinear non-Gaussian methods alluded to above. Its efficiency is crucially tied to applying fast sampling methods and thus this study reports on preliminary efforts to test some of these fast samplers.

The data assimilation problem and a description of PIMC appear in Sections 2 and 3 –see Alexander et al. (2005) for more details and background. For time dependent problems PIMC can be briefly described as follows: the evolution equation for the state vector $\mathbf{x}(t)$ in its discretized form is used to construct a function proportional to the likelihood distribution of $\{\mathbf{x}(t_i)\}_{i=0}^{f}$ where $t_0 = 0 < t_1, ... < t_f$. The specific form of the action $U_{dynamics}$ depends on the statistical distribution of the model error. Another functional U_{obs}, associated with the discrete measurements $\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_M$, where the subscript labels the measurement times, is used to construct a function proportional to

the prior. Again, the functional associated with measurements takes its final form from the error distribution of the measurements. The distribution of the state variable, conditioned on observations, or the posterior distribution, is thus

$$P(\mathbf{x}(0), \mathbf{x}(1), ..., \mathbf{x}(t_f)|\mathbf{y}_1, \mathbf{y}_2, ..., \mathbf{y}_M) \propto \exp(-U), \tag{1}$$

where $U = U_{dynamics} + U_{obs}$. A connection between the estimation problem and the statistical mechanical behavior of a system of interacting particles proves inspiring. The connection between this estimation strategy and simulated annealing of interacting particles in a pinning potential is strong (see Ferreira and Toral (1993) and references contained therein): The vector particles are described by $\mathbf{x}(t_i)$, where $t_i = t_0, t_1, ..., t_f$, with action U (with $kT = 1$, here k is the Boltzmann constant and T the temperature), and the vectors $\mathbf{y}$ are then associated with pinning "sites." One can now use path integral ideas to obtain the moments of the posterior distribution (1) of the history of these interacting "particles." If the dynamics of this system are linear and the statistics are Gaussian then extremizing the action leads to the optimal mean history and variance, conditioned on observations. It is obtained by simple least-squares minimization of the variance, *e.g.* by the Kalman filter/smoother. For nonlinear non-Gaussian dynamics the situation is that Euler-Lagrange equations for the optimal history can be formally derived and these in turn are useful and practical if the action can be shown to be sufficiently convex. However, extremizing can lead to problems if there are multiple extremas. This motivates proposing sampling techniques to obtain the desired moments in PIMC. The system of particles is made to explore configuration space and from it,

statistics are drawn. Since (1) is not a proper probability distribution (the normalization is most likely unknown), a number of sampling techniques have been developed to circumvent this issue. One of these is Markov Chain Monte Carlo sampling scheme (MCMC) (see Liu (2002) and Chen et al. (2000) for some background on Monte Carlo and its use in the Bayesian setting). Simple MCMC is computationally expensive and the path integral method PIMC constructed with this sampler will be slow to converge statistically. It is for this reason for which three alternative accelerated sampling techniques are investigated here: gMC, a generalized Monte Carlo, and two hybrid Monte Carlo strategies. These accelerators, by the way, are not directly tied to PIMC or data assimilation. They are associated with MCMC and thus can be considered whenever acceleration in the application of MCMC is desired.

The gMC is an ad-hoc modification of the local MCMC, wherein the new proposal results from changing all elements of the state variable over all time steps randomly. That this strategy results in an acceleration is not at all obvious, and this study will not attempt to explore the scheme's accelerating qualities. Its appeal derives from the fact that, unlike the hybrid schemes, the gradient of the action is not required.

The hybrid Monte Carlo (HMC) strategy (see Doucet et al. (2002)) is one of the suggested ways to improve the efficiency of the sampling. Barth and Wunsch (1990), Bennett and Chua (1994), Evensen (1994, 1997) have used HMC in a similar geoscience setting, the context being inverse problems and data assimilation, but the Monte Carlo scheme was used to find the cost minimizer. Kruger (1993) applied it in its simulated annealing guise, but again,

it was employed in the extremization problem. The HMC makes use of an associated Hamiltonian problem which needs to be solved numerically in fictitious time in order to generate a new proposal for the MCMC. The acceptance rate of HMC decreases exponentially with increasing system size or increasing fictitious time step. This is due to discretization errors introduced by the numerical integrator.

Another sampling strategy is the generalized Hybrid Monte Carlo (gHMC) approach (see Ferreira and Toral (1993)). The gHMC sampler, proposed by Ferreira and Toral, was developed within the context of statistical mechanics and quantum field theory with the goal of obtaining a speedup in HMC. The strategy consists of using a matrix operator, designed to increase the proposal quality in the fictitious time integration stage of HMC by coupling or "mixing" more strongly the interacting "particles." As was shown by Alexander et al. (2005), the application of gHMC to a double well estimation problem lead to a significant speedup in the statistical convergence of the sampler.

The HMC uses gradient information in suggesting proposals and one is tempted to ask if there is a method that use Hessian information as well: this variant is called the Shadow Hybrid Monte Carlo (Hampton and Izaguirre (2004)) and beyond its mention, it will not be discussed in this paper.

The increased computational efficiency provided by the hybrid samplers comes with a cost: these sampling methods require extra software. The additional investment in coding and implementation of PIMC using the accelerated sampling methods involves producing a gradient code of the action functional. This is a reasonable added expense, not only for the increase in speedup, but

because there are at present efficient and robust gradient code generators (see Bischoff et al. (1992). See also Giering and Kaminski (1998) and Restrepo et al. (1998)). In the gHMC method, in addition to the expense of coding a gradient, there is a matrix-vector product: it is a simple thing to code up but here the issue is that the matrix/vector multiply can become an overwhelming computational expense in the method. This matrix-vector multiply can potentially lead to a significant increase in processor communication, making the prospect for speedup by its parallelized implementation less effective.

Which accelerating strategy should be used in conjunction with PIMC will depend largely on the nature of the problem and on the machine architecture used; more specifically, on the number of time steps in the dynamics, vis-a-vis the dimension of the state variable, and on the communication overhead (in parallel computations). A metric is suggested in this paper that could be used to help decide what strategy to follow.

A path integral formulation of the assimilation problem is not unique: PIMC distinguishes itself in that it is built upon the discretization of the model itself. This has an obvious practical advantage in that the user will most likely have an existing computer code that approximates the model. In some senses construction of the action in discrete form also makes numerical convergence studies consistent in both the estimation and the equation discretization. The disadvantage is that it ties the method's optimality to a particular discretization of the stochastic differential equation; in this study I have used the Euler-Mayurama scheme. In this paper it is shown how PIMC handles higher order integrators with a commensurate increase in computational expense. This is

done in Section 4. The Euler-Mayurama is ideally suited to handle legacy code in the construction of assimilation software. However, it leads to very small time steps in the assimilation: beyond the obvious issue that relates computational expense, for a given tolerance in the accuracy, the new dimension on this problem is that on finite precision machines it may have an effect on how the relative variances of the model and the data are balanced.

This paper is also meant to serve as a tutorial into data assimilation via PIMC; Section 5 presents practical issues related to the method's implementation. For specificity I focus on the implementation of the method for the case in which the number of time steps is larger than the dimension of the state variable, but the method does not distinguish between the dimension of time and the dimension of the state variable and thus it is possible to reformulate PIMC to take advantage of dimension size disparities.

In Section 6 it will be shown how PIMC performs on a problem with nonlinear dynamics and non-Gaussian statistics. The type of problem chosen for this purpose is often called a "hidden variable estimation problem." Field data and the error statistics are available for one of the components of the state variable and the aim is to estimate the mean and possibly higher moments of this component as well as the other components of the state variable, namely the hidden components. The example problem considered here consists of the assimilation of data of drifter positions in a two dimensional flow, driven by the dynamics of point vortices. This particular example problem has been used frequently by Ide et al. (2002), Kusnetsov et al. (2003), and Özgökmen et al. (2000), as a test case for what they refer to as "Lagrangian

data assimilation." The assimilation technique Kusnetsov and collaborators apply is otherwise known in the control theory community as the constrained extended Kalman Filter technique (see Simon and Chia (2002), and references contained therein). The point vortex problem is suited to showcase PIMC's ability to track the optimal estimate of the solution of the filter/smoother problem in the presence of nonlinearities in the dynamics and when the statistics are non-Gaussian. This is an important point, since technology based on the extended Kalman filter, developed specifically to deal with Lagrangian data assimilation, has been shown to fail in the presence of non-Gaussian or highly nonlinear dynamics (Ide et al. (2002), Kusnetsov et al. (2003) discuss this issue in some detail).

Lagrangian data assimilation has had a recent surge of interest, particularly in the oceanographical setting. Thousands of floats are in use at the moment and plans are presently being drawn up for the deployment of many more of these measuring devices, capable of sampling the ocean dynamics on paths, rather than on grids (see Veneziani et al. (2004) and references contained therein).

It should be noted that the example problem is entirely Lagrangian. As such it does not lend itself to considering the issue of relating Eulerian and Lagrangian frames and statistics (see L'vov et al. (1999) for how this plays out in the estimation problem. The reader might also want to consult Bennett (2006), as it discusses Lagrangian fluid dynamics as well as some material on the statistical problem. Also, see Özgökmen et al. (2001), and Mead and Bennett (2001)). The purely Lagrangian problem is still of significant practical importance: an important implication from the work of Kusnetsov and collaborators is that

assimilation of drifters and tracers in many complex flows can be performed by recasting or approximating the flow in terms of interacting point vortices thus potentially avoiding the mixture of Eulerian and Lagrangian frames. The combination of data assimilation and the application of highly developed analytical tools in ordinary differential equations and dynamical systems have the potential to yield insights into the complex dynamics of the flow and of inertial and passive drifters subjected to the flow. Typical of the more interesting oceanic and atmospheric problems in a Lagrangian frame is that they are highly nonlinear –a notable characteristic is that they can display "Lagrangian Chaos"– and far from Gaussian. If estimation techniques are to be used to complement the analysis of these types of problems, or deal sensibly with their sensitivity to initial conditions, an estimation technique that can handle nonlinear dynamics and/or non-Gaussian statistics would be of great utility.

2 Problem Statement

Considered here is the determination of the statistics of the state variable $\mathbf{x}(t)$, whose dimension is N_x, given incomplete and possibly imprecise observations of that system. Here t is an indexing parameter, taken to represent time.

The state vector $\mathbf{x}$ is assumed to satisfy

$$\begin{aligned} d\mathbf{x}(t) &= \mathbf{f}(\mathbf{x}(t), t)dt + (2D)^{1/2}(\mathbf{x}, t)d\mathbf{W}(t), \qquad t > t_0, \\ \mathbf{x}(t_0) &= \mathbf{x_0}, \end{aligned} \tag{2}$$

where $\mathbf{x}_0$ is the intial state. The deterministic part of the dynamics is given by

$\mathbf{f}$. The second term on the right hand side of (2) is the stochastic component of the model and it need not be Gaussian. The distribution of the stochastic component of the model is assumed known. Multiplicative noise is handled similarly: the logarithm of the model yields an equation of the form given by (2).

Stochasticity might be inherent in the system dynamics and/or might arise from parametrizations of unknown or unresolved physics, or from ignored degrees of freedom in the dynamics. The diffusion matrix is D, and the vector-value $d\mathbf{W}(t)$ represents an incremental standard Wiener process.

Observations at discrete times, denoted by the N_y-dimensional vector,

$$\mathbf{y}(t_m) \equiv \mathbf{y}_m, \qquad\qquad m = 1, 2, .., M_{obs},$$

where m labels each observation, are also presumed available. The relationship between the observations and the state variable at different times is given by

$$\mathbf{y}_m = \mathbf{h}(\mathbf{x}_m) + (2R)^{1/2}(\mathbf{x}_m, t_m)\boldsymbol{\epsilon}_m, \tag{3}$$

where $\mathbf{h} : \mathbf{R}^{N_x} \rightarrow \mathbf{R}^{N_y}$, and $\boldsymbol{\epsilon}_m$ is an N_y-dimensional noise vector with a known statistical distribution with matrix variance R. Typically the model noise and measurement errors are uncorrelated, but this need not be so.

The optimal estimate of the state history is then obtained by assimilating observations into the history of the state space statistics, *i.e.* by conditioning the statistics of the time series on those of the observations. The mean history of the state $\mathbf{x}$, conditioned on the measurements, for $t \geq t_0$ is

$$\mathbf{x}_S(t) = E[\mathbf{x}(t)|\mathbf{y}_1, ..., \mathbf{y}_M]. \tag{4}$$

It is the "best estimate" of the state, and the conditional covariance matrix

$$C_S(t) = E[(\mathbf{x}(t) - \mathbf{x}_S(t))(\mathbf{x}(t) - \mathbf{x}_S(t))^\top|\mathbf{y}_1, ..., \mathbf{y}_M] \tag{5}$$

quantifies its uncertainty ($\top$ denotes transpose). In PIMC these best estimates are made available only on the time lattice, $t_0, t_1, ..., t_f$.

3 The Path Integral Method

Equation (2) is discretized using an explicit Euler-Maruyama scheme (see Kloeden and Platen (1992)),

$$\begin{aligned}
\mathbf{x}_{k+1} &= \mathbf{x}_k + \mathbf{f}(\mathbf{x}_k, t_k)\delta t + (2D)^{1/2}(\mathbf{x}_k, t_k)(\mathbf{W}(t_k + \delta t) - \mathbf{W}(t_k)), \\
&\qquad k = 0, 1, 2, \ldots \\
\mathbf{x}_{k=0} &= \mathbf{x_0}.
\end{aligned} \tag{6}$$

The choice of the time discretization is not unique. The Euler-Maruyama discretization is the simplest and makes the presentation of the method clearest. More importantly, it is a convenient scheme with regard to adapting legacy code in the assimilation strategy. Section 4 will describe how higher order explicit discretizations of the stochastic differential equations (SDE's) are handled. The reader will find that PIMC easily handles this modification, and further, that the amount of work required to modify the method as presented here to handle higher order integrators is commensurate with the order and complexity of the numerical scheme used.

Without loss of generality it will be assumed that the discrete time steps are

equally spaced, and further, that the measurement times are commensurate with δt, the time step interval. Namely, $t_k = t_0 + k\delta t$ with $k = 0, 1, \ldots, T$, and $(t_f - t_0)/(T+1) = \delta t$.

In the absence of observations, the probability of the dynamics generating a given history is simply related to the probability that it experiences a certain noise history

$$\boldsymbol{\eta}(t_k) = (\mathbf{W}(t_k + \delta t) - \mathbf{W}(t_k)), \tag{7}$$

at times t_k, where $k = 0, 1, 2, \ldots, T$. Model error statistics are assumed known. For example, if the error is described as Gaussian uncorrelated noise, this probability $Prob\{\boldsymbol{\eta}(t), t = t_0, t_1, ..., t_T\} \sim \exp(-\frac{1}{2}\sum_k |\boldsymbol{\eta}|^2(t_k))$. Other noise statistics lead to a different probability expression.

For a Gaussian distribution the action functional associated with the likelihood $\propto \exp(-U_{dynamics})$ can be found by rearranging terms in (6), *i.e.*,

$$\begin{aligned} U_{dynamics} \equiv \sum_{k=0}^{T-1} \frac{\delta t}{4} \{ & [(\mathbf{x}_{k+1} - \mathbf{x}_k)/\delta t - \mathbf{f}(\mathbf{x}_k, t_k)]^\top D^{-1}(\mathbf{x}_k, t_k) \\ & [(\mathbf{x}_{k+1} - \mathbf{x}_k)/\delta t - \mathbf{f}(\mathbf{x}_k, t_k)] \}. \end{aligned} \tag{8}$$

Let

$$U_{obs} = \sum_{m=1}^{M_{obs}} \frac{1}{4} [\mathbf{h}(\mathbf{x}(t_m)) - \mathbf{y}(t_m)]^\top R^{-1} [\mathbf{h}(\mathbf{x}(t_m)) - \mathbf{y}(t_m)], \tag{9}$$

again, here presuming that the observation noise is Gaussian, for specificity. This cost function is associated with the prior distribution, $\propto \exp(-U_{obs})$.

The total action or cost functional associated with the distribution of the state

variable, conditioned on measurements, is then

$$U = U_{dynamics} + U_{obs}. \tag{10}$$

Rather than trying to minimize the cost functional (10) PIMC proposes to sample $\exp(-U)$, which is proportional to the conditional probability distribution. A standard local Markov-Chain Monte-Carlo (MC) based method (see Binder and Heermann (1997) and Brémaud (2001)) would be the simplest of sampling strategies: one picks a state variable at a randomly-chosen time slice t_k, and proposes a new configuration for it. The cost function of the new configuration is compared to the old cost function; the difference in the cost functions will determine whether to accept or reject using the standard Metropolis algorithm (see Liu (2002) for details on the Metropolis algorithm).

A slight modification of this sampling strategy, which is referred to here as gMC, would differ from the above in only one respect: the new proposal consists of a new configuration for all components of the state vector at every time slice.

PIMC using MC or gMC sampling is relatively simple to implement. The primary disadvantage of MC or gMC, however, is their slow convergence. PIMC using an accelerated sampling, such as HMC or gHMC, can achieve practical efficiency and thus extends the applicability of the method to significantly larger problems.

The two hybrid samplers considered next take the correspondence between the estimation problem and the mechanics of a multi-particle system further. The HMC proposes to use a Hamiltonian for the particles, over fictitious time, to generate new proposals. The state vector $\mathbf{x}_k$, where $k = 0, 1, ...T$, will be taken as the kth particle in an interacting chain. The particle position is denoted as $\mathbf{q}_k(\tau = 0) = \mathbf{x}_k$. Here, $\tau \geq 0$ is fictitious time. These particles are subject to a force $\mathbf{F}([\mathbf{q}]) = -grad\, U(\mathbf{q})$, of dimension $T + 1$. Here, $[\cdot] = (\cdot_0, \cdot_1, ..., \cdot_T)$. The force is such that $\mathbf{F}([\mathbf{q}](\tau = 0)) = \mathbf{f}([\mathbf{x}])$, where $\mathbf{f}$ is as in (6). To each of these generalized coordinates $\mathbf{q}_k$, a conjugate generalized momemtum $\mathbf{p}_k$, is assigned. The momentum $\mathbf{p}_k$ give rise to a kinetic energy $H_{Kin} = \sum_k \frac{|\mathbf{p}_k|^2}{2}$. At $\tau = 0$ the momenta take on values from a Gaussian, zero mean process, with variance 1.

The action of the system is then

$$\hat{H} = U + H_{Kin}. \tag{11}$$

Note that $P(\mathbf{x}, t|\mathbf{y}(t)) \propto \exp(-\hat{H})$: the $\exp(-H_{Kin})$ is multiplicative in the cost function, but this factor can be absorbed in the unknown normalization factor in the probability distribution.

The particles obey the following dynamics,

$$\begin{aligned} \frac{d\mathbf{q}_i}{d\tau} &= \mathcal{A}_{ij}\mathbf{p}_j \\ \frac{d\mathbf{p}_i}{d\tau} &= [\mathcal{A}_{ij}]^\top \mathbf{F}_j, \end{aligned} \tag{12}$$

where $0 \leq i, j \leq T$ and repeated indices imply summation. Here $\mathbf{F}_k = -\frac{\partial \hat{H}}{\partial \mathbf{q}_k}$ is the force on the k^{th} degree of freedom, and τ is fictitious time. $\mathcal{A}$ is a $(T+1)\times(T+1)$ matrix. When $\mathcal{A}$ is the identity matrix, the equations reduce to the familiar Hamilton's equations. In such case the sampling method is known as HMC. The evolution equations in (12) conserve energy, *i.e.* $d\hat{H}/d\tau = 0$, however, the system may no longer be Hamiltonian. Finding a matrix $\mathcal{A}$ that leads to a significant reduction in computation, particularly an optimal one, is an open problem; it is most generally dictated by the particular dynamical system and its discretization. Equally important, however, is to choose a matrix that does not increase the computational cost unreasonably: at worst (12) can increase the cost by $\mathcal{O}(T^2)$ due to the matrix/vector multiplies. This leads to consideration of matrices with a great deal of structure or sparse matrices with small bandwidth: an increase in computational steps linear in T, the number of real time steps, has to be a reasonable price to pay in exchange for a reduction in the number of MCMC trials. In Alexander et al. (2005) we used a circulant matrix for $\mathcal{A}$, Circ[$\exp(-j\alpha)$], where $j = 0, 1, .., T$. α is a real constant. In this instance the matrix/vector multiply can be efficiently done using the fast Fourier transform.

Proposals, *i.e.*, new configurations of the chain, for the Metropolis accept/reject procedure – a single accept/reject cycle will be referred to as an *MCMC trial*– are generated by running a discretization in τ of (12) for J steps. One hopes that J is a very small number. The approximate solution of the Hamiltonian system uses finite difference techniques developed in the molecular dynamics community. The gHMC proposes a dynamic over fictitious time that may no

longer be Hamiltonian, with the purpose of overcoming the torpor of standard Monte Carlo sampling.

In the examples presented later an integrator which can be schematically expressed as

$$q' = q + \delta\tau \mathcal{A} p + \frac{\delta\tau^2}{2} \mathcal{A}(\mathcal{A})^\top F([q])$$
and
$$p' = p + \frac{\delta\tau}{2}(\mathcal{A})^\top (F[q] + F[q']), \tag{13}$$

will be used for the approximation of the evolution equations (12). The discretization choice is not unique, however, detailed balance –in the stochastic sense, not in the geophysical fluid dynamics sense– with respect to the action must be preserved. For example, alternatives to (13) are a variety of "leap-frog" Verlet schemes that preserve phase space volume and obey reversibility (see Field (1999)). Detailed balance is achieved if the configuration obtained after evolving J steps is accepted with probability $\min[1, \exp \Delta\hat{H}]$, where

$$\Delta\hat{H} = \hat{H}([\mathbf{q}'], [\mathbf{p}']) - \hat{H}([\mathbf{q}], [\mathbf{p}]). \tag{14}$$

The update from $(\mathbf{q}_k, \mathbf{p}_k)$ to $(\mathbf{q}'_k, \mathbf{p}'_k)$, for $k = 0, 1, 2, \ldots, T$, will in general not conserve energy; the extent to which energy is not conserved is controlled by the step size $\delta\tau$. However, the Metropolis step corrects for τ-time discretization errors. The momentum variables are refreshed after every accept/reject stage. Since the part of the cost function corresponding to the momentum corresponds to particles having a Gaussian distribution, and a "kT" of 1, the momenta are refreshed to zero mean, variance 1, independent normal distributions. The rule of thumb in running the sampling algorithm is that the

acceptance rate should be about 40%. The parameters $d\tau$, and J –and if the circulant matrix is used, α– are used to achieve this rate of acceptance.

The resulting speedup in HMC or gHMC comes from solving the fictitious evolution system and may be understood as follows: the generalized coordinates will persistently move in the direction of the conjugate momenta during the integration in fictitious time and hence the state of the system will move a distance that goes linearly with the fictitious time step, rather than as the square root, which would be the case if the state moved through configuration space by a random walk. In the manner presented here the intermediate values, in fictitious time, for the conjugate position, are not saved. However, a slight modification of the strategy is to use some or all of the intermediate values as proposals for the Markov chain. Whether this modification leads to a speedup in the sampling is to be determined, but this alternative strategy is easily implementable.

The essential idea behind gHMC is to improve statistical mixing at all length scales (see Dyer and Greenhill (2000) and references contained therein for details on statistical mixing in the context of Markov chains). The increase in computational cost due to the vector/matrix multiplication needs to be weighted against the cost of the MC, gMc, or HMC, *i.e.* the mixing should lead to significant savings in the number of MCMC trials to be performed. Some reasonable choices for the matrix are, for example, one that mimics multigrid –the decorrelation has a frequency dependence, in general, that depends on the distance between the interacting conjugate variables–, or a circulant matrix such as the one used in Alexander et al. (2005), in which the parameter α is

used to make the interaction very nonlocal (small α) or more local (large α). In the examples a tridiagonal matrix Trid$[\exp(-\alpha), 1, \exp(-\alpha)]$ is used.

4 Higher Order SDE Integrators

Higher order methods for the solution of SDE's (see Langouche et al. (1978); Graham (1977); Langouche et al. (1979)) are at present not as popular as is the case in the deterministic counterpart [1] . Nevertheless, described in what follows is how to modify the PIMC assimilation methodology to handle integrators other than Euler-Mayurama (note that the point-vortex/drifter system is in fact one in which perhaps a higher order method would be more suitable in integrating the stochastic differential equations).

Explicit higher order SDE integrators (strongly convergent ones, as they usually avoid the requirement of computing or knowing derivatives) are easily handled within PIMC and the modifications on the method are commensurate with the extra complexity of the higher order of the integrator itself. A concrete example illustrates this: Let it be supposed that (2) is solved using a Heun method, *i.e.*,

[1] Consider the (Îto sense) Taylor series expansion of the jth component of the stochastic vector $\mathcal{X}(t, W_t^1, W_t^2, ..., W_t^m)$, with $j = 1, 2, ..., j_n$, to order dt:

$$\mathcal{X}_{t+dt}^j = \mathcal{X}_t^j + \frac{\partial \mathcal{X}_t^j}{\partial t} dt + \sum_{k=1}^{m} \frac{\partial \mathcal{X}_t^j}{\partial W_t^k} dW_t^k + \frac{1}{2} \sum_{k,l=1}^{m} \frac{\partial^2 \mathcal{X}_t^j}{\partial W_t^k W_t^l} dW_t^k dW_t^l.$$

Euler-Mayurama results from retaining the first three terms in the above expression. So even at lowest order other alternative discretizations can be quite complex.

$$
\begin{aligned}
\mathbf{x}^* &\equiv \mathbf{x}_k + \delta t \mathbf{f}(\mathbf{x}_k, t_k) + (2D)^{1/2}(\mathbf{x}_k, t_k)\Delta\mathbf{W}^* \\
\mathbf{x}_{k+1} &= \mathbf{x}_k + \frac{\delta t}{2}[\mathbf{f}(\mathbf{x}^*, t_k) + \mathbf{f}(\mathbf{x}_k, t_k)] + \\
&\quad \frac{1}{2}[(2D)^{1/2}(\mathbf{x}_k, t_k) + (2D)^{1/2}(\mathbf{x}^*, t_{k+1})]\Delta\mathbf{W}(t_k)), \\
&\quad k = 0, 1, 2, \ldots \\
\mathbf{x}_{k=0} &= \mathbf{x_0}.
\end{aligned} \tag{15}
$$

Here $\Delta\mathbf{W}^*$ and $\Delta\mathbf{W}(t_k)$ are distinguished as independent processes with the same statistical distribution. The modifications required are in the definition of the cost function, relevant to the accept/reject stage, and the Hamiltonian in the molecular dynamics stage (see Section 3.1). Assume, for illustration that the noise is Gaussian and that D is a constant, then the probability distribution associated with the predictor is, by definition, capable of generating samples independent of the (Gaussian) distribution for the corrector changes (8). In other words the use of a higher order explicit method can be accomodated by making the state vector be composed of $\mathbf{x}_k$ and $\mathbf{x}^*$, thus, $U_{dynamics}$ is in this case

$$
\begin{aligned}
&\sum_{k=0}^{T-1} \frac{\delta t}{4}\{[(\mathbf{x}_{k+1} - \mathbf{x}_k)/\delta t - \overline{\mathbf{f}}]^\top D^{-1}[(\mathbf{x}_{k+1} - \mathbf{x}_k)/\delta t - \overline{\mathbf{f}}]\} \\
&+\frac{\delta t}{4}\{[(\mathbf{x}^* - \mathbf{x}_k)/\delta t - \mathbf{f}(\mathbf{x}_k, t_k)]^\top [(\mathbf{x}^* - \mathbf{x}_k)/\delta t - \mathbf{f}(\mathbf{x}_k, t_k)]\},
\end{aligned} \tag{16}
$$

where $\overline{\mathbf{f}} = [\mathbf{f}(\mathbf{x}_k, t_k) + \mathbf{f}(\mathbf{x}^*, t_k)]/2$. Obviously, no changes are required in the U_{obs} contribution to the cost function. With regard to the Hamiltonian in the molecular dynamics equations associated with HMC or gHMC the changes should be obvious and they come in when calculating $\mathbf{F}_k$.

5 Implementation Details

Presuming that the model is already in the form of (6) what is required is a code that implements the MCMC trials (consisting of a scheme to generate proposals, calculate the cost function, and accept/reject the proposal within a Markov Chain Monte Carlo context). The accept/reject scheme chosen here is the Metropolis algorithm. Proposals are generated via random walks of single degrees of freedom in MC and of sets of state vectors and time steps gMC, or via molecular dynamics runs in HMC and gHMC. If the noise statistics for either the model or the data are not Gaussian, the cost function $\hat{H}$ needs to properly reflect this fact. Within the MCMC trials loop care must be taken when computing $\Delta\hat{H}$ (see (14)): If $N_x \times T$ is very large it is possible to degrade the computation due to loss-of-precision errors in subtracting large numbers from one another. This is seldom an issue using local MC, but entirely possible with any of the other samplers considered here.

The computation of the sample mean and the uncertainty (and for that matter, whatever higher order moments are required) can be performed during the computations as running-sample averages and thus demand negligible storage overhead. In solving the Hamiltonian system of HMC there is an increase in storage, when compared with MC, say, by the Verlet integrator. However, the storage overhead can be traded for extra computation. In implementing gHMC there is an extra storage cost associated with the matrix $\mathcal{A}$, but in general this cost can be no worse than $\mathcal{O}(T)$ for matrices with limited bandwidth or the circulant matrix.

The choice of a first proposal for the sampler is arbitrary. However, the length of the burn-in period can be significantly affected by choosing a good starting point (see Binder and Heermann (1997) for a discussion of the system memory with regard to its initial state in simple spin systems). If the data being assimilated was produced by (6) that a good starting guess would be produced by solving the SDE itself with no noise. In the next section data that has *not* been generated using the SDE will be used. This would be a situation that is closer to the practical situation. It turns out that the no-noise SDE solution was better than a random noise trial, from the point of view of reducing the burn-in period.

5.1 The MCMC Trial Loop

The MCMC trial section of the code, in its simplest guise, might look as in Algorithm 1. Far more elegant and compact version of this algorithm are possible but the one presented here is easy to follow. The while loop turns out to be a convenient construct from the point of view of monitoring progress in the MCMC loop. The while loop can be implemented with added diagnostic tools that can examine the statistics of the cost function itself.

5.2 Acceleration by Fictitious-Time Molecular Dynamics

The parameters $d\tau$ and J are used to change the ratio of accepted trials to total trials. As in any Markov Chain Monte Carlo method, if the change in the cost function is too large the rejection rate becomes unacceptable; if too

again = 1; acce = 0; count = 0; construct $\mathcal{A}$; input σ, $d\tau$, J, n_{max} ;

q position = initialize using zero-noise SDE; ;

while *again = 1* **do**

count++ ;

MCMC Trial Loop ;

for $n = 1$ **to** n_{max} **do**

q =q_{old} ;

if *MD==NO* **then**

q =q_{old} + σ(2 rand(0,1)-1) *gMC* ;

end

if *MD==YES* **then**

HMC or gHMC ;

p = randn(0,1); (q,p)=molecular dynamics(q,p,$d\tau$, J) ;

end

(dH,H)=dHcalc(H, q, p) *calculate energy difference* ;

accept/reject ;

if *rand(0,1)*$<$ exp($-dH$) **then**

q_{old}=q. acce++ ;

end

mean(q_{old}) var(q_{old}) *update desired moments* ;

end

$th_{times} = n_{max}$count ;

acceptance percentage = 100 acce/th_{times}; Check Results(*set again =0 to quit*) ;

end

Algorithm 1: Pseudo-code for the accept/reject loop. The molecular dynamics call is used to propose new state in HMC or gHMC. gMC results from not using the molecular dynamics routine.

small, i.e. moving in a random walk around the probability space, it will take many steps to sample effectively. A key requirement of HMC and gHMC is to know $\mathbf{F}_k = -\frac{\partial U}{\partial \mathbf{q}_k}$ in the form of a code. When the model is simple the subroutine representing $\mathbf{F}_k$ can be built by hand. However, for large problems one could use ADIFOR/ADIC (see Bischoff et al. (1992)): the actual output of this processor is a subroutine of the required $\mathbf{F}_k$. In using gHMC the choice of matrix will also have a bearing on the accept/reject ratio.

The following are rough estimates of the operational count of the various methods:

- MC: $\mathcal{O}(N_{mc})$
- gMC: $\mathcal{O}(M\, N_{gmc})$
- HMC: $\mathcal{O}(2M(1+J)\, N_{hmc})$
- gHMC: $\mathcal{O}(2M(1+JaT)\, N_{ghmc})$

$M := N_x\, T$, where N_x is the dimension of the state variable, T the number of time steps. J the number of fictitious time steps, $N_{\{.\}}$ the MCMC trials required by each of the sampling methods. In local MC a single entry in the state vector at a single instance in time is changed. Referring to the single entry as an element, the difference in energy, before and after the change in the element is computed, in order to accept/reject sample. In gMC every element is changed (which can be done at the same time (8) is evaluated in order to avoid significant increases in the computational cost). When M is large it is very possible that taking the difference between the energy before and after the move can lead to significant errors on a finite-precision machine; the difference of 2 possibly large numbers is taken. In order to ameliorate

the loss-of-precision error the computation of the energy difference is done element by element, increasing the computational overhead by M. In HMC there are conjugate momenta associated with each element. The energy change calculation is then twice as expensive as in gMC. The system is integrated in fictitious time for J steps and thus an extra $\mathcal{O}(M \times J)$ calculations. The parameter $1 \leq a \leq T$ represents the computational overhead in gHMC due to the acceleration matrix. Naive matrix/vector multiply involving $\mathcal{A}$ in gHMC would lead to $a = T$. Using something like a tridiagonal matrix in gHMC would, on the other hand, lead to $a = 5$ and a circulant matrix, using Fast Fourier Transforms, to $a = \mathcal{O}(\log T)$.

Efficiency is the ultimate goal of the accelerator and thus a metric would be useful in assessing this aspect of the code. An important thing to note from the operation count estimate is that without knowledge of the MCMC trials $N_{\{\cdot\}}$ required for the various methods, it is not clear which one is the most efficient. The acceleration of HMC over MC should lead to $N_{mc} \gg 2N_x T N_{hmc}$, and so on: $N_{\{\cdot\}}$ is such an important contribution to the efficiency in the method the operational count is not a good metric of the method efficiency unless it is tied to some notion of statistical convergence. The most important consideration with regard to efficiency is deciding how many samples are required for each of the methods to be statistically equivalent, *i.e.*, for a number of moments, as computed with each method, to be within some asymptotically small tolerance. The correlation time $n^*(t)$ in the action is the proposed metric.

First, the normalized correlation of the unbiased cost function $\overline{H}(n) = \hat{H}(n) - <$

$H >$ is calculated using

$$c(n) = \mathcal{F}^{-1}[|\mathcal{F}(\overline{H})|^2](n)/\mathcal{F}^{-1}[|\mathcal{F}(\overline{H})|^2](0), \tag{17}$$

$\mathcal{F}$ indicates the Fourier transform. Here n is the MCMC trial index. Markovian processes engender their correlation length/time with specific structure that make it a robust and telling quantity to check. See Gardiner (2004). Next, the number of trial steps n^* required for the correlation to reach $1/e$ is found. This defines a "correlation length" for the method –which, incidentally, will pin down the value of $N_{\{\cdot\}}$: statistical stability of the results was obtained with $N_{\{\cdot\}}$ in the order of 8 to 10 times the correlation length. The correlation time is a wall-clock estimate of the amount of time required to compute n^* steps with a particular method. Use of this metric to compare the samplers will be demonstrated in Section 6.

5.3 Testing the Outcomes

Debugging the code is greatly facilitated by the relatively simple structure of the algorithm: the gMC strategy can be checked by turning off the molecular dynamics routine. Once the sampler is working HMC can be checked by turning on the molecular dynamics routine with $\mathcal{A}$ set to the identity matrix and seeking agreement with gMC. Finally gHMC can be tested by checking for agreement with gMC and HMC.

There are a number of tests that can be performed, both as diagnostics of the code as well as for diagnostics of the results. A useful test, when comparing the

implementation of MC against gMC, HMC and gHMC, consists in comparing the single-time statistics of the methods: it is required that the statistics of U for all methods agree, provided the number of sample trials is sufficiently large.

Monitoring the spectrum of the cost function is also very useful. A histogram is usually a telling but very rough way to monitor whether the statistics of the cost function reflect the assumptions on the statistics of model and data.

Another test requires generating the data $\mathbf{y}(t_m)$ by solving the SDE. Here t_m are the measurement times and the vector $\mathbf{y}$ is composed of the components that make up the drifter and the vortical centers. Let $\hat{\mathbf{x}}_j(t_k)$, $k = 0, 1, \ldots, T-1$ be the j^{th} solution of the SDE (each of these is with independent random noise. The mean history is

$$< \mathbf{X}(t_k) > = \frac{1}{Z} \sum_{j=1}^{L} e^{-E_j} \hat{\mathbf{x}}_j(t_k) \qquad \text{where}$$
$$Z = \sum_{r=1}^{L} \exp(-E_j), \tag{18}$$

where L is the number of independent samples and

$$E_j = \sum_{m=0}^{N_y} [\mathbf{y}(t_m) - \hat{\mathbf{x}}(t_m)]^\top [\mathbf{y}(t_m) - \hat{\mathbf{x}}(t_m)].$$

The estimate $\hat{\mathbf{x}}(t_k)$ should be close to $< \mathbf{X}(t_k) >$, for all $k = 0, 1, \ldots, T-1$. The mean history for the different methods can be compared with the following estimate:

$$\left| 1 - \frac{< \mathbf{X}(t_m) >}{\hat{\mathbf{x}}(t_m)} \right|,$$

where $m = 1, 2, \ldots, M$ is the index of the measurements, and none of the $\hat{\mathbf{x}}(t_m)$ are zero. For MCMC trials, in the order of 10 million, it was found that the mean relative error in all methods to be used in the next section to evaluate the samplers was very small.

6 Example Calculations

The example featured here consists of producing the conditional mean history and the uncertainty of a system comprised of N_p passive drifters and N_v point vortices (see Friedrichs (1966)). In compact form, the dynamics of the mth point vortex in space (x, y) and time t can be written in terms of $z(t) := x(t) + iy(t)$ as

$$\frac{dz_m}{dt} = \frac{i}{2\pi} \sum_{l=1, l\neq m}^{N_v} \frac{\Gamma_l}{z_m^* - z_l^*} + \eta^V(t), \tag{19}$$

and the nth passive tracer will have dynamics given by

$$\frac{d\zeta_n}{dt} = \frac{i}{2\pi} \sum_{l=1}^{N_p} \frac{\Gamma_l}{\zeta_n^* - z_l^*} + \eta^P(t). \tag{20}$$

In the above equations, $\eta^{V,P} = \sigma^{V,P}(1+i)$ are the stochastic terms, representing unresolved processes for either the point vortex or the tracer, $(\cdot)^*$ denotes its complex conjugate. The above point-vortex/passive-drifter system is very well understood and has been used by Kusnetsov et al. (2003); Ide et al. (2002) as a testbed for a Lagrangian data assimilation scheme, based on constrained extended Kalman filter.

In the calculations $N_p = 1$, and $N_v = 2$, therefore, the resulting dimension of the state variable is 6. Like Kusnetsov et al. (2003), the vortex data is ignored and the drifter data is supplied in the assimilation stage. This is the hidden variable problem. The initial drifter position is $0.3 - i0.6$, and the vortices are at 1 and -1, respectively. The final time is 2.2. Figure 1 shows the drifter and vortex paths, as given by the model, in the absence of noise.

The data was created synthetically in a manner similar to that in Kusnetsov et al. (2003): instead of solving the SDE they compute the drifter/vortex system using MATLAB's ODE45 routine, an explicit adaptive Runge-Kutta of order 4-5, adding noise to the computed paths. An explicit Runge-Kutta 4, with time steps of 0.01 is used here. As they did, noise is added at each time step by appending to the Runge-Kutta solution a term $\sqrt{2\,sdt}\mathcal{N}(0,1)$, where $\mathcal{N}(0,1)$ is a Gaussian with zero mean and variance of 1. In the calculations $s = 0.05$. The independent noise terms are added to the x and y components of each dynamic variable, all with the same s.

The correlation time will be used to gauge the performance of the samplers. In order to confidently capture the correlation length and obtain reasonable statistical convergence of the first 3 moments of the history the number of MCMC trials was taken to be very large.

Kusnetsov et al. (2003) report that their constrained extended Kalman filter yields good results provided that the noise in the data is not too large and the number of time steps too large, *i.e.*, paths longer than 2 or 3 times the intrinsic rotational period lead to failures when the noise was significant. For them failure is defined as an inordinate deviation of the path from its pressumed

mean. They attribute failure in the estimation, mainly, to repeated crossings of saddle points by the drifter. No evidence of this or any other type of failure was found when PIMC was run, on a data stream that was up to 4 revolutions long, with parameters otherwise similar to those considered below.

The estimated histories, shown in the following figures, were computed via HMC with $J = 1$. MC, gMC, and gHMC produced results that were virtually identical. Figure 2 show the outcome of running HMC on the hidden variable problem. Figure 2a shows with circle tracks the noisy drifter path which is taken here as given measurements for use in the estimation. The vortical paths that accompanies this drifter data did not form part of the observations (the vortices are "hidden variables"). These vortical paths are shown in Figure 2c as circles. The parameters for this run were as follows: $\sigma^{V,P}$ were both set to 0.05. The value for R, related to our confidence in the data, was 0.001; this particular estimation problem represents the situation wherein there is significant confidence in the observations. The data was read every 4 time steps. The predicted standard deviation for the drifter position is plotted as a function of time in Figure 2b. The estimated vortex paths are superimposed on the data in Figure 2c. In Figure 2c appear the estimated vortical paths, as connected dots, the connected circles are the vortical paths that were not used in the assimilation process but produced by the numerical run that generated the drifter data.

Table 1 summarizes the computational efficiency of the methods in obtaining the results portrayed in Figure 2. HMCJ refers to HMC using J fictitious time steps. For the various cases, (J=1, $d\tau = 0.0012$), (J=2, $d\tau = 0.0007$), (J=3, $d\tau = 0.0006$), (J=4, $d\tau = 0.0005$); for gHMCT, with a tridiagonal matrix $\text{Trid}[\exp(-\alpha), 1, \exp(-\alpha)]$ and (J=1, $d\tau = 0.0012$, $\alpha = 2.0$). Absent from the table are the results from gHMC with a circulant decorrelating matrix, but comments on this run appear in what follows.

The correlation length was computed using (17). The time of each run is quoted in seconds, all of the cases were run using 10 million MCMC trials. This number of trials is far larger than what is required to see adequate convergence in the mean and in the uncertainty of the estimates in any of the sampling methods; it is instead chosen to insure relatively good statistical convergence.

The correlation time, in seconds, for gMC is only about 1.5 times better than MC and higher than a couple of the HMC entries, however, this strategy does not require a gradient. The HMC with $J = 1$ beats all methods, with a correlation time that is about 3 times smaller than MC.

The gHMC with a tridiagonal matrix entry shows a correlation length that is competitive when compared to MC or gMC. However, the HMC methods turn out to be more efficient. The circulant matrix gHMC was also tried on this specific problem; with (J=1, $d\tau = 0.0012$), the run time was 52712 seconds. In order for this method to have been competitive on this specific

problem the correlation length would have to be approximately 353, a correlation length that seemed impossible to achieve, though trial and error was the only strategy available to test this conjecture computationally. The more feasible route to improving the performance of the gHMC would seem to be to bring down significantly the run time. However, it should be noted that in the gHHMC calculations FFTW was used (see Frigo and Johnson (2005)) and thus matrix-vector multiplies are performed in $4T(1 + \log T))$ per conjugate position/momenta using an already optimized FFT code. Hence, significant gains in computational efficiency would be difficult to obtain by optimizing the remaining parts of the code.

Alexander et al. (2005) significant gains were achieved in sampling speed using gHMC and it seems at first that the results presented here contradict the findings in that paper. In that work we considered a double-well problem with a state variable of dimension 1. When the correlation length is the only metric, the gHMC outcomes are better than other methods tried in both studies; the metric in the double-well problem was the correlation length and here the correlation time is chosen instead and is suggested as being more meaningful in applications. Hence, it is the use of the temporal metric in the Lagrangian problem that makes gHMC less appealing. The outcome, however, is problem specific: it is not clear what is the best decorrelation matrix for either the double-well or the Lagrangian problems. It is also possible that the circulant matrix was well matched to the double well problem, but not as well matched to the Lagrangian problem, leading to much more impressive results in the use of gHMC on the double-well problem than in the Lagrangian problem, using the correlation length metric.

6.2 Assimilating Discretized-SDE Data

PIMC should deliver highly accurate estimates when the discretzed SDE that was used in deriving the action is used to produce the data and the uncertainty in the data is small. This should still be the case in the hidden variable case. To illustrate this case drifter data was generated by actually solving the SDE via (6), with all parameters equal to those used in generating the results in Figure 2. Figure 3a shows the drifter path data as well as its estimated mean, and Figure 3b shows the assimilated vortical paths. Granted, the data was considerably smoother and less noisy when compared to the results in Figure 2a as it enters the creation of the data in a different way.

6.3 Uncertainties, Data Insertion Frequency

Considered here are two more examples that further illustrate how the method deals with changes in data parameters. HMC1 was used in the computations. The data was the same as that used in generating Figure 2. The first case will correspond to dropping the insertion rate from every 4 to every 15 time steps. In the second case the confidence in the drifter position is decreased by changing $R = 0.001$ to $R = 0.01$. Figure 4 shows how the results are modified by a significantly lower contribution in the cost function from observations. The insertion interval of every 15 steps was been chosen because it is not commensurate with the inherent frequencies of motion of the dynamical system (a commensurate interval would either follow the twists and turns of the drifter data more faithfully or will suppress completely the higher frequency

motions. This points out to the fact that in assimilating data it is important to understand the dynamics of the problem as much as possible). In Figure 5 decreasing the certainty in the data leads, not surprisingly, to an uncertainty estimate is substantially higher as compared to prior cases. The model plays a more important role in the estimate in both of these cases, as compared to the case illustrated by Figure 2. No failures in the estimation were observed when changing the insertion frequency or the certainty in the data, even if the number of time steps was increased significantly.

7 Conclusions

PIMC is a data assimilation scheme which makes use of the discretized model in the formulation of the cost function. The cost function itself is an action functional, built upon the specific form of the probability distribution underlying the stochasticity in the model and the data. The preferred outcomes of the assimilation process are moments of histories, conditioned on data and thus the structure of the action is also influenced by a Bayesian inter-relation between model and data. For time dependent problems the method, as presented here, is global rather than sequential. It yields a filter/smoother solution in the form of explicit moments. It is generally applicable and capable of handling nonlinear and/or non-Gaussian problems.

The non-Gaussian statistics and/or nonlinear dynamics can produce action functionals for which seeking a least-action extremizer is analytically or practically difficult to obtain. Hence the moments of the posterior probability

distribution are alternatively obtained via sampling. Sampling, however, is notoriously expensive, and thus an effort must be made to apply accelerated samplers in problems which have state variables of significant dimension..

One of the proposed acceleration schemes is based on solving in fictitious time a molecular dynamics problem of an associated Hamiltonian system. This procedure yields proposals with greater acceptance rate than standard Markov Chain Monte Carlo. This hybrid Monte Carlo can be turned into the generalized hybrid Monte Carlo by applying a nonlocal decorrelation matrix to the Hamiltonian system. The resulting fictitious time dynamics problem is no longer Hamiltonian, in general, but permits extra degrees of freedom in the molecular dynamics integration, in addition to its time step and the number of time steps. The matrix is chosen to improve the decorrelation of particles in the fictitious time integration, especially of particles that are not simple nearest-neighbors. The choice of matrix is most likely problem-dependent, and it has to take into account that in its practical implementation the matrix/vector multiplication does not increase substantially the computational overhead. In Alexander et al. (2005) we tested PIMC with gHMC applying it to a simple double well model. When compared to other accelerated Monte Carlo schemes it delivered impressive increases in computational efficiency (we did not take into account the amount of computer time required to achieve the result, however). On the Lagrangian assimilation problem used to illustrate PIMC gHMC did not fare as well.

In order to exploit the computational savings that the hybrid Monte Carlo methods afford, a code for the gradient of the action is needed. In principle

one could use very robust and straightforward automatic differentiation tools to obtained the required gradient, nevertheless, an alternative sampler is a simple, gradient-free, general Monte Carlo method.

A generally fair and useful estimate of the efficiency of a particular PIMC implementation is the correlation time, the shorter the time the more cost efficient the implementation becomes. The correlation is a useful metric in estimating how many proposals are required, roughly, for statistical stability. When the correlation time is used, instead of the correlation length, in gauging the performance of the different sampling schemes the gHMC faired poorly on problems where the product of the number of time steps and the dimension of the state vector is large. gHMC had a higher computational overhead than other methods explored here.

The path integral method is expensive computationally, when compared to assimilation methods that are based on least-squares. The experience of practitioners of data assimilation is that when the estimation problem has very mild nonlinearities and/or near-Gaussian statistics, linear or linearized estimation methods still deliver useful outcomes. For such problems the path integral method will not be the tool of choice for estimation. When linear and linearized methods fail, the matter of efficiency becomes a moot point, and the feasibility of getting an estimate, particularly one that is optimal or near-optimal, make the methods developed for nonlinear/non-Gaussian problems viable. For these problems the KSP method (see Eyink et al. (2004), Eyink et al. (2002) for details) the computational overhead for the discretized estimation problem is $\mathcal{O}(N_x \times D_p)$, where N_x is the state vector dimension, D_p

is the cost of solving numerically a partial differential equation for each state variable over T time steps for the forward and the adjoint fields. PIMC has an overall computational cost of $\mathcal{O}(N_x \times T \times N)$, where N is the number of samples required by the particular choice of sampler.

How can the path integral contribute to practical forecasting in geophysical applications? It can serve as a benchmark for testing outcomes in other methods; it is not that hard to implement. Clearly, as a viable data assimilation method in problems which have a small $N_x \times T$, whenever nonlinearity and non-Gaussianity are strong enough to deem linearized methods inapplicable. A problem that has these characteristics and is of interest in ocean dynamics and forecasting is Lagrangian estimation. Lagrangian data assimilation has received increased attention recently, since it is one of the practical ways in which oceanic data is obtained; there are plans to significantly increase the use of passive and active moving platforms in an effort to improve oceanic data coverage. The hope of significant advances into understanding flows in general and mixing processes, both theoretically and practically is reasonable, even from looking at a few Lagrangian tracks, if one couples nonlinear/NonGaussian estimation to powerful tools in the analysis of ordinary differential equations.

Acknowledgements

The author is most thankful to F. Alexander: his comments and suggestions had a great deal of impact in the writing of this paper. The anonymous reviewers help was considerable in making the paper more friendly to the geophysics

audience. I also gratefully acknowledges the helpful discussions on the topic of Lagrangian data assimilation with C.K.R.T. Jones and K. Ide. This work was performed while JMR was a visiting fellow at SAMSI; its staff's kind hospitality is acknowledged. This work was funded by NSF DMS0327642 and ONR N00014-04-1-0215.

References

Alexander, F. J., Eyink, G. L., Restrepo, J. M., 2005. Accelerated Monte-Carlo for optimal estimation of time series. Journal of Statistical Physics 119, 1331–1345.

Barth, N., Wunsch, C., 1990. Oceanographic experiment design by simulated annealing. Journal of Physical Oceanography 20, 1249–1263.

Bennett, A. F., 2006. Lagrangian Fluid Dynamics. Cambridge University Press, Cambridge.

Bennett, A. F., Chua, B. S., 1994. Open-ocean modeling as an inverse problem: the primitive equations. Monthly Weather Review.

Binder, K., Heermann, D. W., 1997. Monte Carlo Simulation in Statistical Physics. Springer, Berlin.

Bischoff, C. H., Carle, A., Corliss, G. F., Griewank, A., 1992. ADIFOR: Automatic differentiation in a source translation environment. In: Wang, P. S. (Ed.), Proceedings of the International Symposium on Symbolic and Algebraic Computation. ACM Press, New York, pp. 294–302.
URL `http://www-fp.mcs.anl.gov/autodiff/`

Brémaud, P., 2001. Markov Chains, Gibbs Fields, Monte Carlo Simulation, and Queues. Springer, New York.

Chen, M.-H., Shao, Q.-M., Ibrahim, J. G., 2000. Monte Carlo Methods in Bayesian Computation. Springer-Verlag, New York, Berlin.

Doucet, A., de Freitas, N., Gordon, N., 2002. Sequential Monte Carlo Methods in Practice. Springer-Verlag, New York.

Dyer, M. E., Greenhill, C. S., 2000. On markov chains for independent sets.

J. Algorithms 35 (1), 17–49.

Evensen, G., 1994. Inverse methods and data assimilation in nonlinear ocean models. Physica D 77, 108–129.

Evensen, G., 1997. Advanced data assimilation for strongly nonlinear dynamics. Monthly Weather Review 125, 1342–1354.

Eyink, G. L., Restrepo, J. M., Alexander, F. J., 2002. Reducing computational complexity using closures in a mean field approach in data assimilation, *submitted*.

Eyink, G. L., Restrepo, J. M., Alexander, F. J., 2004. A mean field approximation in data assimilation for nonlinear dynamics. Physica D 194, 347–368.

Eyink, G. L., Restrepo, J. R., 2000. Most probable histories for nonlinear dynamics: tracking climate transitions. J. Stat. Phys. 101, 459–472.

Ferreira, A., Toral, R., 1993. Hybrid Monte Carlo method for conserved-order-parameter systems. Phys. Rev. E 47, 3848–3851.

Field, M. J., 1999. A Practical Introduction to the Simulation of Molecular Systems. Cambridge University Press, Cambridge, UK.

Friedrichs, K. O., 1966. Special Topics in Fluid Dynamics. Gordon and Breach, New York.

Frigo, M., Johnson, S. G., 2005. The design and implementation of FFTW3. Proceedings of the IEEE 93 (2), 216–231, special issue on "Program Generation, Optimization, and Platform Adaptation".

Gardiner, C. W., 2004. Handbook of Stochastic Methods. Springer, Berlin.

Giering, R., Kaminski, T., 1998. Recipes for adjoint code construction. ACM Transactions on Mathematical Software 24 (4), 437–474.

Gilmour, I., Smith, L. A., Buizza, R., 2001. On the duration of the linear

regime: is 24 hours a long time in synoptic weather forecasting? Journal of Atmospheric Science 58, 3525–2539.

Graham, R., 1977. Path integral formulation of general diffusion processes. Zeitschrift fur Physik 26, 281–290.

Hairer, M., Stuart, A. M., Voss, J., 2005. A Bayesian approach to data assimilation. Physica D, –.

Hampton, S., Izaguirre, J., 2004. Improved sampling for biological molecules using shadow hybrid monte carlo.
URL citeseer.ist.psu.edu/hampton04improved.html

Ide, K., Kusnetsov, L., Jones, C., 2002. Lagrangian data assimilation for point vortex systems. Journal of Turbulence 3, 053.

Kim, S., Eyink, G. L., Restrepo, J. M., Alexander, F. J., Johnson, G., 2003. Ensemble filtering for nonlinear dynamics. Monthly Weather Review 131, 2586–2594.

Kloeden, P., Platen, E., 1992. Numerical Solution of Stochastic Differential Equations. Springer-Verlag, Berlin.

Kruger, J., 1993. Simmulated annealing: a tool for data assimilation into an almost steady model state. Journal of Physical Oceanography 23, 679–688.

Kushner, H. J., 1962. On the differential equations satisfied by conditional probability densities of Markov processes, with applications. J. SIAM Control, Ser.A 2, 106–119.

Kushner, H. J., 1967a. Approximation to optimal nonlinear filters. IEEE Trans. Auto. Contr. 12, 546–556.

Kushner, H. J., 1967b. Dynamical equations for optimal nonlinear filtering. J. Diff. Eq. 3, 179–190.

Kusnetsov, L., Ide, K., Jones, C., 2003. A method for assimilation of Lagrangian data. Monthly Weather Review 131, 2247–2260.

Langouche, F., Roeckaerts, D., Tirapegui, E., 1978. On the most probable path for diffusion processes. J. Phys. A 11, L263–L268.

Langouche, F., Roeckaerts, D., Tirapegui, E., 1979. Functional integral methods for stochastic fields. Physica 95A, 252–274.

Lawless, A. S., Gratton, S., Nichols, N. K., 2005. Approximate iterative methods for variational data assimilation. International Journal of Numerical Methods in Fluids 47, 1129–1135.

Leeuwen, P. J. V., 2003. A variance minimizing filter for large scale applications. Monthly Weather Review 131, 2071–2084.

Liu, J. S., 2002. Monte Carlo Strategies in Scientific Computing. Springer-Verlag, New York.

L'vov, V. S., Pomyalov, A., Procaccia, I., 1999. Temporal surrogates of spatial turbulent statistics: The Taylor hypothesis revisited. Physical Review E 60, 4175–4184.

Mead, Bennett, A. F., 2001. Towards regional assimilation of Lagrangian data: the Lagrangian form of the shallow water model and its inverse. Journal of Marine Sciences 29, 365–384.

Özgökmen, T. M., Griffa, A., Mariano, A. J., Piterbarg, L. I., 2000. On the predictability of Lagrangian trajectories in the ocean. Journal on Atmospheric and Oceanic Technology 17, 366–383.

Özgökmen, T. M., Piterbarg, L. I., Mariano, A. J., EH, E. H. R., 2001. Predictability of drifter trajectories in the tropical Pacific Ocean. Journal of Physical Oceanography 31, 2691–2720.

Pardoux, E., 1982. Équations du filtrage non linéaire de la prédiction et du lissage. Stochastics 6, 193–231.

Pham, D. T., 2001. Stochastic methods for sequential data assimilation in strongly nonlinear systems. Mon. Wea. Rev. 129, 1194–1207.

Restrepo, J. M., Leaf, G. K., Griewank, A., 1998. Circumventing storage limitations in variational data assimilation studies. SIAM J. Sci. Comput. 19, 1586–1605.

Simon, D., Chia, T., 2002. Kalman filtering with state equality constraints. IEEE Transactions on Aerospace and Electronic Systems 39, 128–136.

Stratonovich, R. L., 1960. Conditional Markov processes. Theor. Prob. Appl. 5, 156–178.

Veneziani, M., Griffa, A., Reynolds, A. M., Mariano, A. J., 2004. Oceanic turbulence and stochastic models from subsurface Lagrangian data for the North-West Atlantic Ocean. Journal of Physical Oceanography 34, 1884–1906.

Wunsch, C., 1996. The Ocean Circulation Inverse Problem. Cambridge University Press, Cambridge, UK.

List of Figures

List of Tables

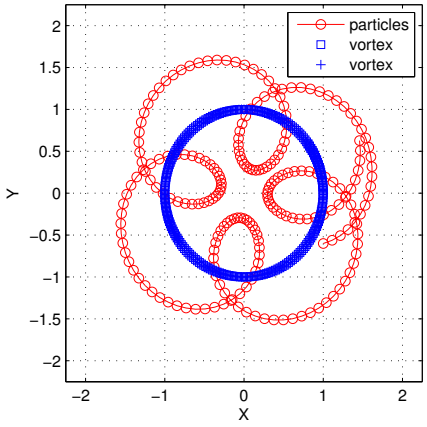

Fig. 1. Drifter and vortex paths in the absence of noise. The vortices rotate counter-clockwise on a shared circular path.

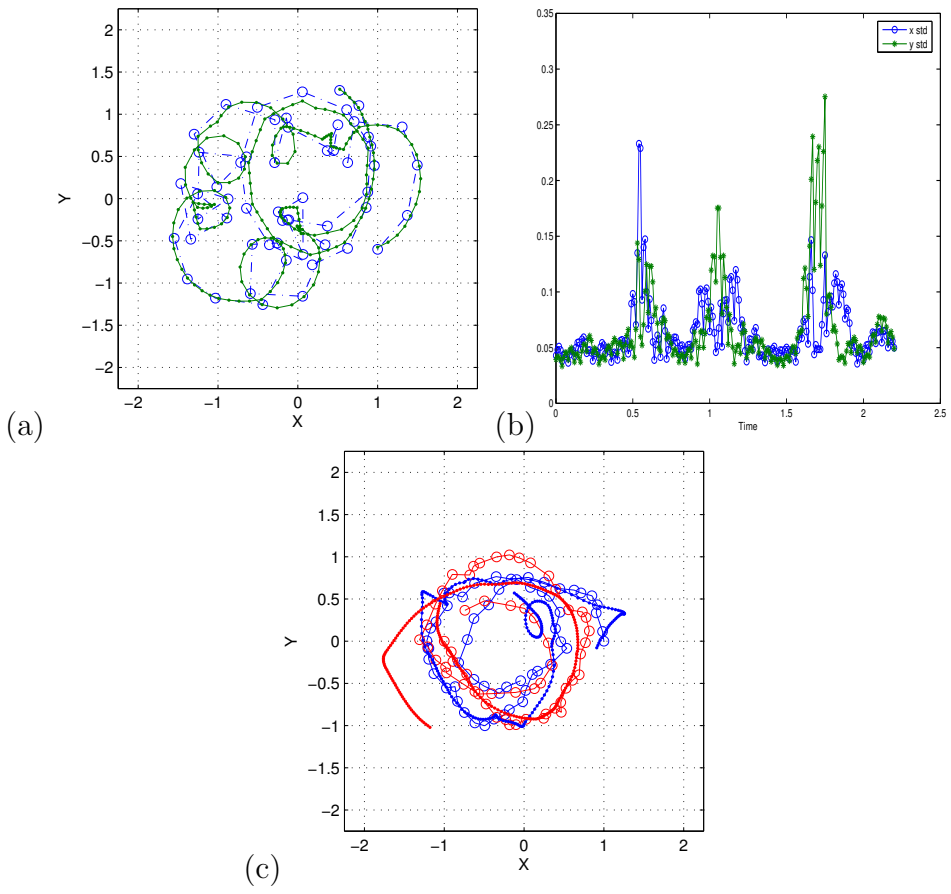

Fig. 2. Smoother results on hidden variable problem. $R = 0.001$; the data was read every 4 time steps. (a) Drifter path with noise which served as data (connected circles). Estimated drifter mean history (connected dots). (b) Standard deviation estimate for the two components x and y describing the drifter path, as a function of time. (c) Estimated vortical tracks (connected dots). The circles represent the vortical tracks that were generated with the drifter data. In the assimilation process the data represented by circles is not used.

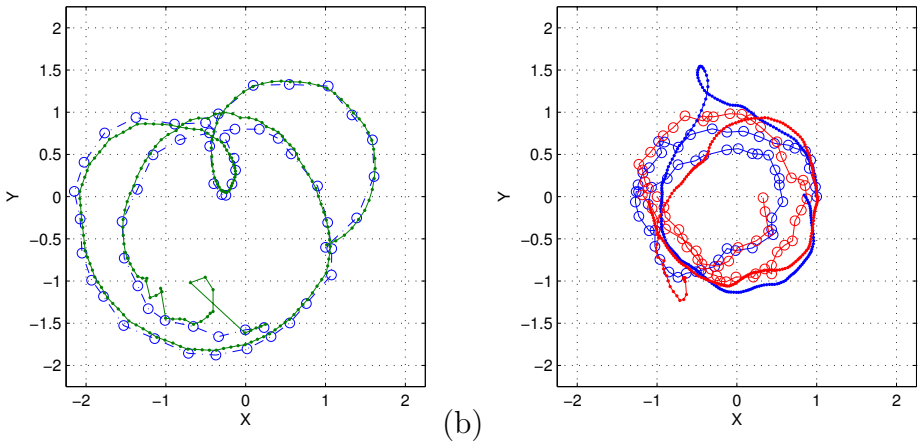

Fig. 3. Optimal estimation. Assimilation of data generated with the discretization that went into building the action. $R = 0.001$, data inserted every 4 time steps. (a) Drifter and (b) vortex path estimates, as given by HMC. Other sampling schemes give nearly identical outcomes. The drifter data (circles in a) was obtained by using (6). All parameters are the same as those used in computing Figure 2.

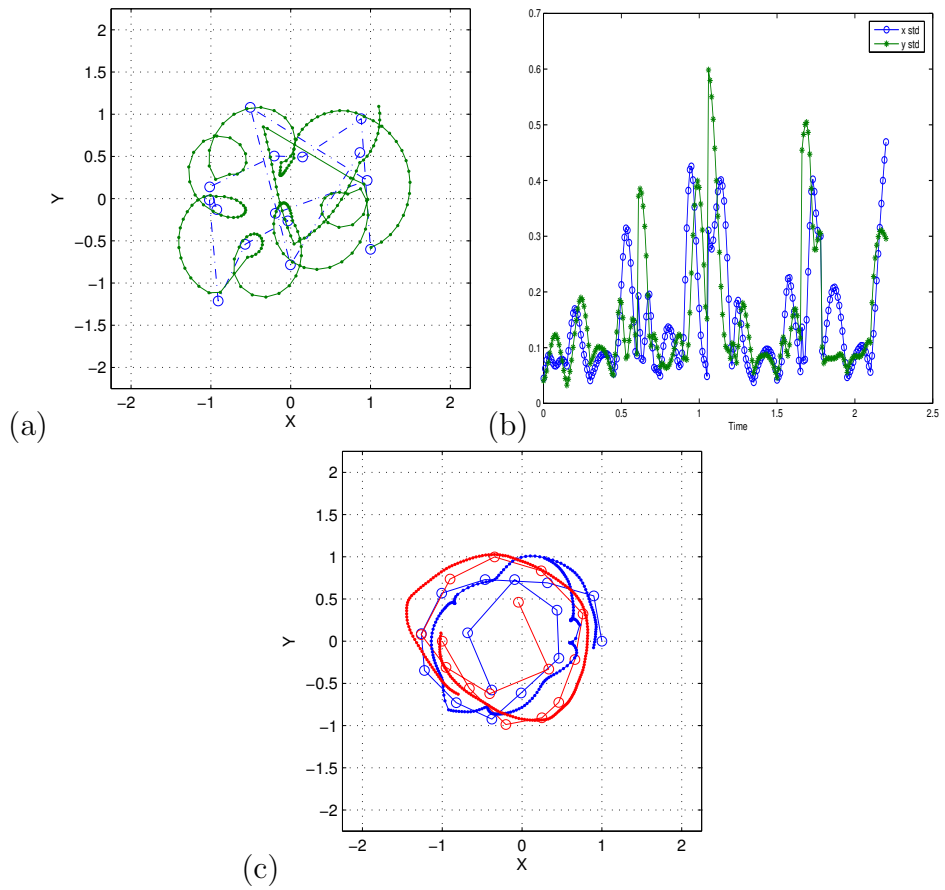

Fig. 4. Effect of decreasing data stream insertion frequency. $R = 0.001$, data read every 15 time steps. (a) Data (connected circles) and drifter mean history estimate (connected dots); (b) standard deviation estimate for the drifter path as a function of time, x and y components; (c) estimated vortical tracks (dots). The vortical tracks, shown at measurement times, which was generated with the drifter path which served as data, is shown in circles. This (circle) data was not used in the assimilation process.

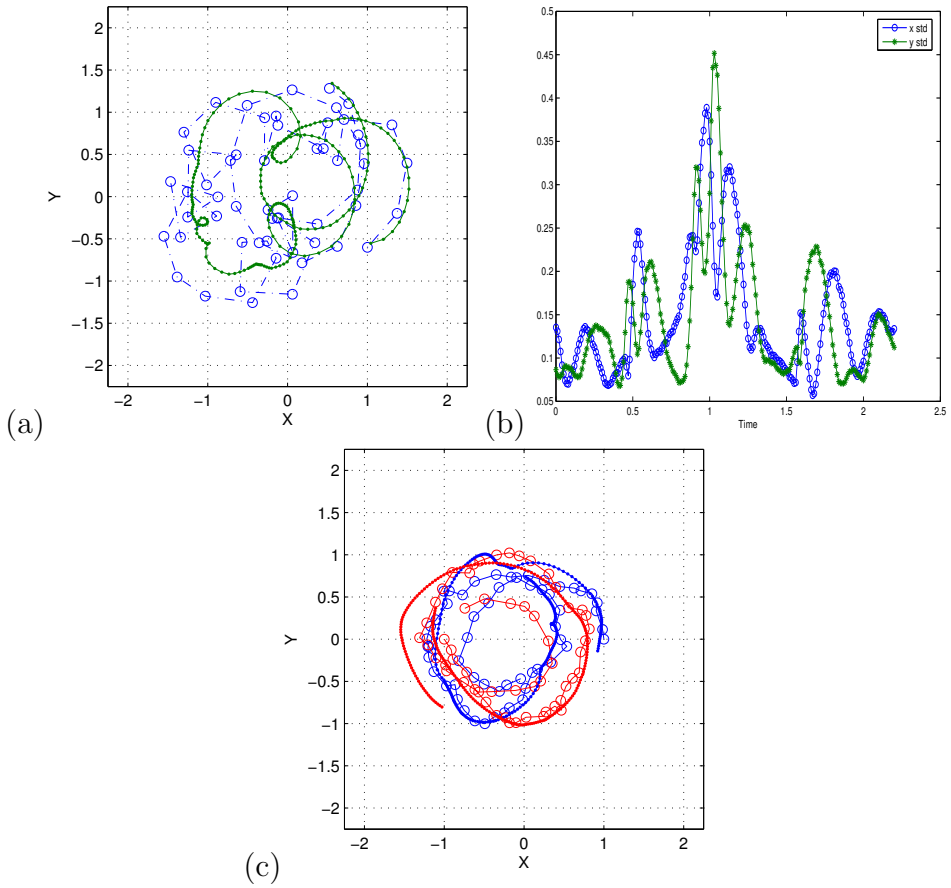

Fig. 5. Effect of decreasing confidence in the data stream. $R = 0.01$, data read every 4 time steps. (a) Data (connected circles) and drifter mean history estimate (connected dots); (b) standard deviation estimate for the drifter path as a function of time, x and y components; (c) estimated vortical tracks (dots). The vortical tracks at measurement times which accompanied the drifter path which served as data is shown in circles. This (circle) data was not used in the assimilation process.

Table 1

Computational efficiency comparison for the various sampling strategies. MC is local Monte Carlo, gMC is general Monte Carlo and the HMCJ refers to the hybrid Monte Carlo, where J refers to the number of τ steps. gHMCT is the generalized hybrid Monte Carlo with a tridiagonal matrix. All cases were run using 10^7 MCMC trials.

method	run time	corr length	corr time
MC	8402	5202	4.37
gMC	8487	3339	2.83
HMC1	13397	1182	1.58
HMC2	13572	1370	1.86
HMC3	23397	1246	2.92
HMC4	28764	1180	3.39
gHMCT	27851	1367	3.81