A Statistical-Mechanical Approach to Data Assimilation for Nonlinear Dynamics III. Numerical Algorithms

Gregory L. Eyink and Juan M. Restrepo
Department of Mathematics and Department of Physics
University of Arizona, Tucson, AZ 85721, U.S.A.

Francis J. Alexander
C C S 3, Los Alamos National Laboratory
Los Alamos, NM, 87545, U.S.A.
(Dated: January 13, 2003)

In two previous papers, a new probabilistic approach has been developed for data assimilation into nonlinear dynamical systems. The new method employs a mean-field approximation to the conditioning upon observational data and moment-closure approximation to the evolution of probabilities between measurements. In this final paper of the series, fast, robust, and efficient algorithms are described for practical application of the method, exploiting its underlying variational structure. Direct minimization of the convex cost function by a conjugate-gradient algorithm is guaranteed to converge. Such descent methods are used as part of fast, hybrid algorithms which switch over to solving Euler-Lagrange equations by Newton relaxation. These algorithms are illustrated on a simple nonlinear model with bimodal statistics. In the simple test problem, the new statistical-mechanical variational methods give results much faster than empirical-ensemble methods. Storage requirements are also very substantially reduced. It is argued that these advantages of the mean-field closure methods are likely to persist for large-scale geophysical systems, particularly if first-order moment closures are employed.

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

I. INTRODUCTION

This paper is the third and final in a series presenting a new probabilistic approach to data assimilation and inverse modeling for large-scale nonlinear stochastic dynamical systems. Data from observations change the prior distributions calculated from such a model. The probabilities conditioned on the data then represent the optimal information one can hope to have about the state of the system. The calculation of the conditional probabilities involves two steps: the evolution of the probabilities by the dynamics between measurements and the determination of posterior probabilities given the data (and an error model) by Bayes formula. The latter is traditionally called the analysis step in data assimilation for meteorological forecasting. In our first paper, Eyink et al. [1] hereafter denoted I, we discussed this general framework and, primarily, developed a mean-field approximation to the analysis or conditioning step of the calculation. It was shown that this mean-field conditional analysis can be obtained by the minimization of nonnegative, convex cost functions, either the multi-time relative entropy, which depends upon the state only at the measurement times, or the effective action, which is a functional of the entire state-history continuous in time. In the second paper of the series, Eyink et al. [2] hereafter denoted as II, we considered instead approximations to the dynamical evolution between measurements. The method most specifically developed there was a moment-closure or Rayleigh-Ritz scheme based upon a variational formulation of the Kolmogorov equations to evolve the probabilities. We discussed in general how to calculate the multi-time relative entropy and the effective action within moment-closure schemes which make trial guesses for the solutions of the Kolmogorov equations. Especially attractive features were shown to exist for entropy-based closures which preserve an exact H-theorem for the (single-time) relative entropy.

In this final paper of the series we discuss in detail the numerical implementation of the new methods advanced in I and II, based upon moment-closure approximation of the dynamical evolution and the mean-field conditional analysis. In fact, the methods cannot be considered to be formulated completely without a specification of the numerical algorithms necessary to implement them on the computer. The latter determine not only the accuracy of the results, but also the speed and efficiency of the computation. These factors are decisive in distinguishing between a method which is useful only in principle and one which can, in fact, be affordably applied. This is especially true for the large-scale, spatially-extended systems that are relevant in the geosciences. For real-time forecasting in meteorology the speed of the computation is essential. For other applications, such as inverse modeling in physical oceanography or climatology, the constraints on speed may be less severe, but still a result should be obtainable in at most a matter of weeks, not years. Efficiency of the calculation in terms of requirements on memory and—in a parallel implementation—communication, are also relevant factors. Variational as-

similation methods of the type currently practiced for large-scale geophysical applications already threaten to overwhelm even the largest storage capabilities of computer facilities [see 3, 4]. It is important that the variational methods proposed here impose storage requirements at least no greater than those conventionally employed.

For this work we have therefore developed and utilised algorithms with an eye toward eventual application of the methods to large-scale geophysical systems. The test problem studied here is the same one-variable stochastic ODE considered in I and II, a double-well model with bimodal statistics considered in Miller et al. [5] and Miller et al. [6]). In this case, brute force methods could be employed that would not be feasible in a genuine application. However, we developed instead faster and more efficient methods that could be applied to more realistic systems. For example, in our variational approach, the variance of the estimated time-history is given by the diagonal elements of the inverse Hessian of the minimized cost function (relative entropy or effective action). This Hessian is a matrix of order at most only a few hundred in the simple test problem, and it could be easily calculated and inverted. However, in a realistic application its order, if it were calculated naively, would be the total number of spacetime points used in the computation of the model and that could be enormous. We have used an important theorem of large deviations theory, the Contraction Principle [see 7], to calculate the desired variance in a simpler way. First, we can reduce the order of the Hessian to just the number of spacetime points at which the variance is required, not the full number used in the computation of the model. Second, we can avoid any matrix inversion whatsoever by performing instead a sequence of constrained minimizations. (See section 2.2.1). This is just one example of the type of tailored algorithm that we have employed to make our method more feasible to apply. These include also an efficient minimax algorithm, simplifying variable transformations, adjoint algorithms for calculating derivatives, etc. These are described in the text in sufficient detail that a reader should be able to reproduce all of the results obtained by them which are reported here and in II.

Our new methods are compared in this paper for speed and efficiency with the empirical-ensemble methods that were also the basis of comparison in II (there for the accuracy of the computed conditional statistics). The ensemble methods have been extensively developed in the last several years by G. Evensen and his collaborators: see Evensen [8], van Leeuwen and Evensen [9], Burgers et al. [10], Evensen and van Leeuwen [11]. See Tippett et al. [12] for more recent developments. These methods approximate the dynamical evolution of the statistics by integrating the primitive equations of motion for N-sample ensembles, with N on the order of 100 or so. As currently practiced, the ensemble methods employ a linear analysis borrowed from Kalman filtering and smoothing, but which is only an approximation for nonlinear dynamics with non-Gaussian statistics. We shall present evidence to convince the reader that our new methods are competitive, at least, with the ensemble methods in terms of computational speed and efficiency. In certain respects the present methods are even superior and these advantages are scalable in the size of the computation. We expect therefore that the present methods can be a useful complement to the existing ensemble methods, and will allow an independent check of results where both can be applied. Since both methods make some uncontrolled approximations in calculating the conditional statistics, using such methods in parallel would provide the most reliable results.

The detailed contents of this paper are as follows: In Section 2 we discuss the numerical implementation of our methods, in several versions. First, in subsection 2.1 we consider two methods which use the exact conditional analysis but a second-order moment-closure for the dynamics. One of these, KSP-2LL, employs a linear ansatz for the solution of the backward Kolmogorov equation and the second, KSP-2DE, employs exponential ansätze for the solutions of both forward and backward equations. It is shown that the first of these leads to a realizability breakdown, but the corresponding filtering approximation, KS-2E, is sound. Next, in subsection 2.2 we discuss the various methods based upon the mean-field variational analysis, both with first-order closure (MFV-1LL,MFV-1DE) and second-order closure (MFV-2LL,MFV-2DE). The methods using the linear ansatz for the solution of the backward equation (MFV-1LL,MFV-2LL) are also found to suffer from realizability problems, but less severe than for the KSP closure. In the subsection 2.3 we briefly discuss the implementation of the ensemble methods, the filter EnKF and smoother EnKS. In Section 3 of the paper we present our comparisons of the methods in terms of run times, rates of convergence, and storage requirements. Section 4 contains our final discussion and summary. In an Appendix we consider some technical, but important, details of the computational algorithms: use of dual thermodynamic variables, cumulants and their generating functions, and variational-based discretization schemes.

II. ESTIMATION SCHEMES AND ALGORITHMS

Moment-closure approximations to the estimation problem were discussed in II for two different approaches to the analysis step: exact conditional analysis and a mean-field conditional analysis. In this section, we give a brief summary of the various approximate estimation schemes that were proposed and studied in II. For each method we give a complete description of the computational problem involved in calculating the estimated history and its variance, as well as details of how we implemented that method numerically.

Integrating the forward-backward closure equations

II(8),(9) or II(25),(26) is the base-level problem that must be solved in all of our algorithms. In the Appendix we shall give some details of how we accomplished that integration. Aside from relatively straightforward numerical analysis, there were several less obvious steps that were taken. First, the equations were reformulated in a new set of variables, rather than the variables $(\boldsymbol{\alpha}, \boldsymbol{\mu})$ in terms of which they were written in II. Second, the individual terms in the equations were calculated by a method that exploits the cumulant-generating function for the exponential PDF ansatz. Third, the numerical integration scheme for the equations was generated from a discretization of the associated action functional, guaranteeing a structurally consistent method. Each of these matters is discussed in turn in the Appendix.

In addition, we shall describe briefly in this section our numerical implementation of ensemble methods which employ the linear analysis of I, section 3.2. These are the primary methods of comparison for our moment-closure schemes in II and in section 3 of this paper.

A. KSP Closure Methods

Methods of this type keep the exact conditional analysis of the jump conditions, I(8), (10), but employ a moment-closure of the KSP forward-backward evolution equations I(7),(9). Such methods yield approximations to the functions $\mathcal{P}_F(\mathbf{x}, t)$ and $\mathcal{A}_S(\mathbf{x}, t)$ in I. Therefore, the conditional mean and variance are obtained simply from

$$\begin{aligned} \mathbf{x}_*(t) &= \int d\mathbf{x}\ \mathbf{x}\mathcal{P}_S(\mathbf{x}, t), \\ \sigma_*^2(t) &= \int d\mathbf{x}\ |\mathbf{x} - \mathbf{x}_*(t)|^2 \mathcal{P}_S(\mathbf{x}, t), \end{aligned} \tag{1}$$

where $\mathcal{P}_S(\mathbf{x}, t) = \mathcal{A}_S(\mathbf{x}, t)\mathcal{P}_F(\mathbf{x}, t)$. As discussed in II, this approach requires a moment closure of at least second-order. Therefore, we only consider the second-order exponential PDF closure introduced in II,Section 3.2.

2.1(a) Left-Linear Ansatz (KSP-2LL)

One can consider such a second-order closure with the left-linear ansatz given by II(5),(10). This yields the set of forward-backward equations, (12),(13) and the jump conditions given by (15), (17) in the Appendix. As discussed there, it is more convenient to use a formulation in $(\boldsymbol{\gamma}, \boldsymbol{\lambda})$-variables, because it avoids performing a minimization at each timestep. Since the Euler-Lagrange equations are now partially uncoupled, the forward equation can be integrated first from steady-state initial conditions for the closure and this can be followed by an integration of the backward equation using stored values from the forward integration.

However, this method does not yield a useful estimator for the model problem. We have implemented the method very simply, using the explicit Euler scheme discussed in the Appendix. For all of the datasets A-E considered in II, this method gave a very poor approximation to the exact conditional statistics. In Figure 1(a) show the resulting mean history just for dataset E. Comparing with the results in I,Figure 2 we see that this method does yield a poor result for $x_*(t)$. Even worse, the conditional standard deviation $\sigma_*^2(t)$ does not exist for the left-linear ansatz, because the approximate variance is negative over most of the time-interval. In Figure 1(b) we show the variance for the dataset E. The negative values observed there imply a failure of statistical realizability of the left-linear approximation. In fact, the PDF computed with that ansatz becomes negative over a broad range. It is not surprising, because the left trial function II(5) can easily develop negative values when the coefficients $\boldsymbol{\alpha}$ become order one. Only for conditional averages very close to unconditioned ones—e.g. for extremely inaccurate measurements —will the $\boldsymbol{\alpha}$ coefficients be small and will statistical realizability hold. For all of the datasets considered, observation error R was already too small to allow for realizable results. Therefore, no comparisons were made of this method with other schemes in II.

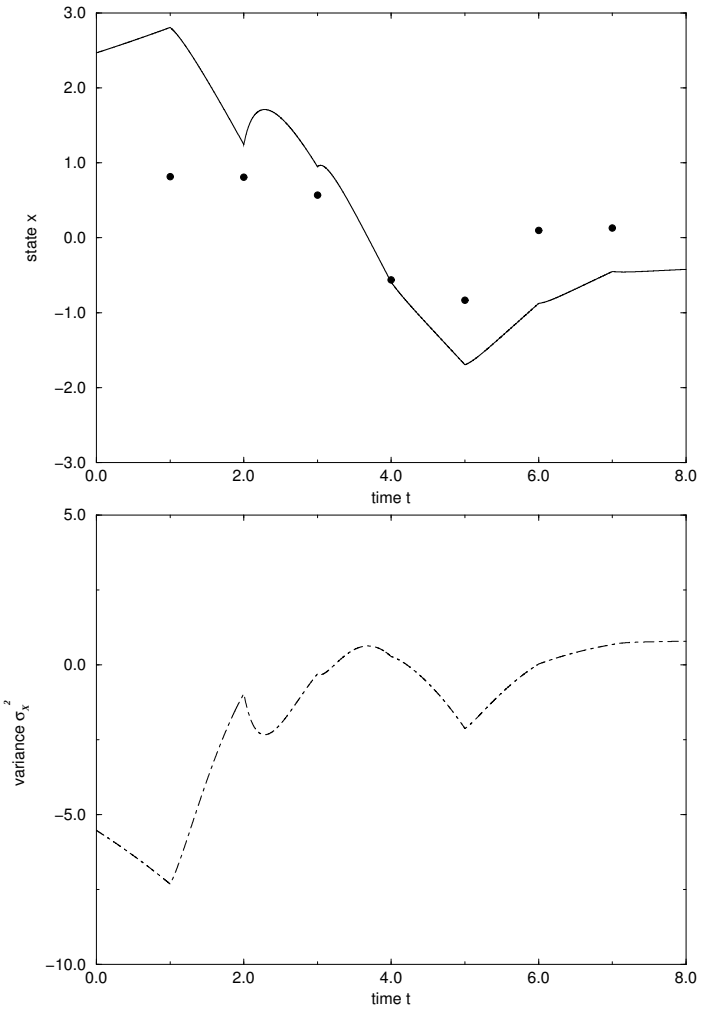

FIG. 1: KSP-1LL closure, dataset E (60 % rms error). (a) mean history, (b) variance.

Despite this serious failure of the left-linear ansatz approximation to the KSP smoother, the approximate filter KS-2E, given by the forward equation (14) and the jump condition (12) for the second-order exponential PDF closure, is a good approximation to the optimal KS filter. This method implements exactly the conditional analysis step, and only the evolution between measurements is approximated by closure. Furthermore, the exponential PDF ansatz is guaranteed to be nonnegative and to give a realizable result. Comparisons were thus made in II for this second-order closure of the KS filter.

2.1(b) Double-Exponential Ansatz (KSP-2DE)

We also consider the second-order KSP closure with the double exponential ansatz, II(18),(19). Because the left and right trial functions are now both nonnegative, we expect to avoid the problems due to the left-linear ansatz in the previous subsection. However, the Euler-Lagrange equations for this ansatz are now (23),(24) in the Appendix, along with the jump conditions (26) for the solution of backward equation. These are a fully coupled set of ODE's satisfying an initial-value for the forward equation and a final value for the backward equation. They must be solved together as a two-time boundary-value problem.

We have attempted to solve this coupled nonlinear system by a Newton relaxation method. However, the domain of convergence appears to be very small and for no initial guess of the solution that we tried did the Newton iteration converge. Therefore, we have resorted to a cruder method, minimizing a cost function constructed by integrating the squares of the equations over time. After tens of millions of steps of a conjugate-gradient algorithm, we could reduce the rms residual of the equations only to about 10%. In Figure 2 we show the results for dataset A. Considering the poor quality of our numerical solution the results are surprisingly close to those of the exact KSP equations shown in I, Figure 1. Because of the difficulty in obtaining solutions, no results for this method were shown in II. However, if good solution algorithms can be found, the method appears very promising.

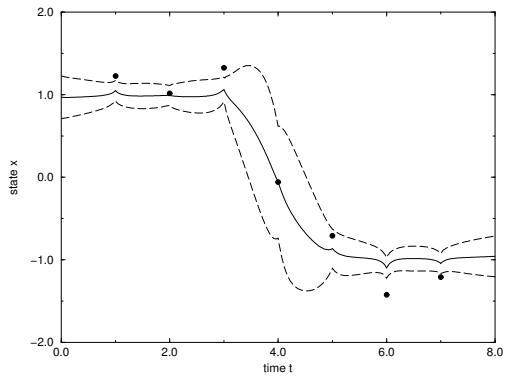

FIG. 2: KSP-2DE closure, dataset A (20 % rms error) represented by filled circles. Solid line: mean, dashed line: mean ± standard deviation.

B. Mean-Field Variational Methods

1. First-Order Closure

We have also considered methods based upon the mean-field conditioning presented in I. One of the principal advantages of the mean-field approach is that it permits the application of *first-order closures*. In realistic, spatially-extended systems, this is an important consideration, because second-order closures will involve the square of the number of variables as for a first-order closure. Although equations for covariances have been solved for realistic systems, e.g. in the context of Kalman filtering approaches, it is best to avoid solving such equations, if at all possible. The first-order closure methods we consider do so by providing models of the required second-order moments in terms of the first-order ones.

The mean-field approach leads also to a variational formulation of the estimation problem. Therefore, the problem of calculating the optimal history $x_*(t)$ becomes one of minimizing the multi-time entropy, or of solving the corresponding Euler-Lagrange equations with initial and final conditions. To obtain the conditional variance $\sigma_*^2(t)$, we could calculate the Hessian matrix $(\mathbf{\Gamma}_*)_{mm'} = \partial^2 H_*/\partial x_m \partial x_{m'}(x_1^*, ..., x_M^*)$ and then get the variance from the diagonal elements of the inverse Hessian matrix $\sigma_*^2(t_m) = (\mathbf{\Gamma}_*^{-1})_{mm}$. Here $x_m^* = x_*(t_m)$ is the conditional mean history at the m^{th} measurement time, $m = 1, ..., M$. This is the approach that was used in I. However, in realistic applications this involves the inversion of the Hessian matrix, which may be very large. We use here an alternative "constrained minimization method" based upon the Contraction Principle, which avoids this inversion. It is discussed in detail in Eyink [13]. Briefly, this method uses the fact that $\sigma_*^2(t_m) = 1/H_*''(x_m^*; t_m)$, now just a scalar inverse. Here

$$H_*(x_m; t_m) = \min_{x_n, n \neq m} H_*(x_1, ..., x_M) \tag{2}$$

is the single-time relative entropy, and

$$H_*'(x_m; t_m) = \lambda_m(x_1, ..., x_M)|_{x_n = \bar{x}_n(x_m), n \neq m}\,, \tag{3}$$

where $\{\bar{x}_n(x_m), n \neq m\}$ achieves the minimum in (2). To obtain the second-derivative, we can use a finite-difference approximation such as $H_*''(x_m^*; t_m) \approx H_*'(x_m^* + \delta; t_m)/\delta$ for some small δ. An algorithm based upon these ideas performs the minimization in (2) with $x_m = x_m^* + \delta$ held fixed, successively for each measurement time t_m. Each such constrained minimization yields $\sigma_*^2(t_m)$ at the corresponding time. To obtain $\sigma_*^2(t)$ at selected times t intermediate to the measurements, we insert a set of "pseudo-measurements" at those times with infinite variance.

There are two versions of these first-order closure methods that we consider:

2.2.1(a) Left-Linear Ansatz (MFV-1LL)

When using the left-linear ansatz, there is no closed-form expression for the multi-time entropy or the effective action. As in I, the problem of calculating the optimal history reduces to solving the minimax problem

$$H_*(x_1^*,...,x_M^*) = \min_{x_1,...,x_M} \max_{\lambda_1,...,\lambda_M} \left\{ \sum_{m=1}^{M} x_m\lambda_m - F_X(\lambda_1,...,\lambda_M) + \sum_{m=1}^{M} \frac{(x_m-y_m)^2}{2R_m} \right\}, \tag{4}$$

where $y_m,\ m=1,...,M$ are the measured values and $R_m,\ m=1,...,M$ are the error variances of those measurements. For the first-order closure, $\mathbf{M}(x)=(x)$. Therefore, one obtains $F_X(\lambda_1,...,\lambda_M)$ as discussed in II,Section 2.1.1, by one forward integration of (12) with the jump conditions (14). To perform the inner maximization in (4) by a conjugate-gradient algorithm, one requires the derivatives $x_m = \partial F_X/\partial\lambda_m$ as well. These are obtained as discussed in I,Section 2.1.1 and the Appendix, from one backward integration of (13) with the jump conditions (15). To obtain the outer minimization in (4) one may apply a conjugate-gradient algorithm yet again, as was done in I. Although the minimization is over only the variables $x_1,...,x_M$ at the measurement times, the optimal history $x_*(t)$ is obtained at all times from the formula (16) in the Appendix. This fact gives a significant reduction in the computational labor involved in carrying out the minimization. It is another consequence of the Contraction Principle of large deviations theory.

However, performing two conjugate-gradient optimizations in such a nested fashion as described above gives the optimal history to only a quarter of working precision and requires the square of the computing time as for one minimization. Therefore, it is advantageous to reformulate the minimax problem above as a single minimization:

$$H_*(x_1^*,...,x_M^*) = \min_{\lambda_1,...,\lambda_M} \left\{ \sum_{m=1}^{M} x_m(\lambda_1,...,\lambda_M)\lambda_m - F_X(\lambda_1,...,\lambda_M) + \sum_{m=1}^{M} \frac{(x_m(\lambda_1,...,\lambda_M)-y_m)^2}{2R_m} \right\}. \tag{5}$$

Here

$$x_m(\lambda_1,...,\lambda_M) = \frac{\partial F_X}{\partial\lambda_m}(\lambda_1,...,\lambda_M), \quad m=1,...,M. \tag{6}$$

To carry out the minimization in (5) by a conjugate-gradient algorithm requires knowing the gradient of the function inside the bracket. Calling this function $\tilde{H}_*(\lambda_1,...,\lambda_M)$, one can see that for $m=1,...,M$

$$\frac{\partial\tilde{H}_*}{\partial\lambda_m}(\lambda_1,...,\lambda_M) = \sum_{n=1}^{M} \left(\lambda_n + \frac{x_n(\lambda_1,...,\lambda_M)-y_n}{R_n} \right) \frac{\partial x_n}{\partial\lambda_m}(\lambda_1,...,\lambda_M). \tag{7}$$

With this gradient we can perform a single conjugate-gradient minimization, but the price to be paid is that we must obtain the Hessian $\partial x_n/\partial\lambda_m = \partial^2 F_X/\partial\lambda_n\partial\lambda_m$. To calculate this, we used the algorithm discussed in the Appendix. In this scheme, one calculates the above derivatives, for each fixed $m=1,...,M$ by integrating a forward equation for times $> t_m$ and a backward equation for times from t_f to t_i. Only the gradient in (7) needs to be stored and it can be built up by adding in successively the M terms of the sum. Therefore, this method gives in M forward-backward integrations the M components of the gradient in (7), with storage requirements of the same order as those in the double-minimization algorithm. Hence, we have employed this efficient minimax algorithm for our computations shown in II (with thanks to M. Anitescu for suggesting the approach).

Given the experience with the left-linear ansatz for the KSP closure in section 2.1(a), one may worry that

there will be similar realizability failures in the present case as the negative variances which occurred there. It turns out that there is a realizability failure using the left-linear ansatz in the mean-field variational approach for the double-well problem, but it is more subtle and less pernicious than the negative variances found in the KSP closure. A simple realizability condition on the multi-time functions $F_X(\lambda_1, ..., \lambda_M)$ and $H_X(x_1, ..., x_M)$ is that they should be convex functions of their arguments. We have found by numerical computations with the algorithms outlined above that these functions are not convex when calculated using our first-order closure with the left-linear ansatz. For example, the function $F_X(\lambda_1, \lambda_2)$ for $M = 2$ has a small "ridge" along which it is concave, not convex. This can be clearly seen in the section of its graph in Figure 3(a). The dual surface, which is

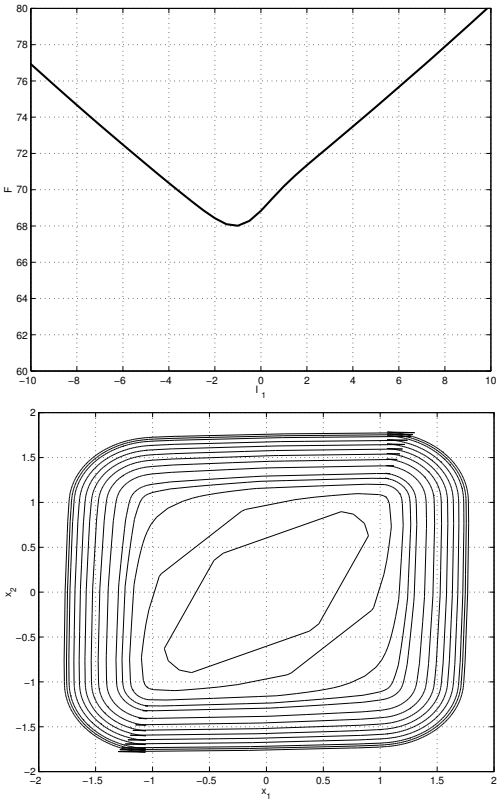

FIG. 3: MFV-1LL closure. (a) two-time free-energy function, cross-section at $\lambda_2 = 50$. Note concavity for $0 < \lambda_1 < 2$ (b) two-time entropy function, contour plot. Swallowtails are in the lower left and upper right.

the graph of the Legendre transform $H_X(x_1, x_2)$, should then contain a swallowtail caustic by well-known results in singularity theory. The swallowtails are easily seen as the dark "wedges" in the contour plot of $H_X(x_1, x_2)$ in Figure 3(b). Hence, the approximate two-time entropy is multi-valued.

However, this is a less severe breakdown of realizability than what occurred in the KSP closure. It is quite analogous to the breakdown that appears in the equilibrium statistical mechanics of the van der Waals gas, where the Helmholtz free energy $A(T, V)$ exhibits a similar non-convexity in volume V and then its Legendre transform, the Gibbs free energy $G(T, P)$ develops a swallowtail singularity (see [14], Section 2.3 for the van der Waals gas and Pippard [15] for an interesting discussion from the point of view of singularity theory.) A pragmatic solution to this realizability failure would be replace $F_X(\lambda_1, ..., \lambda_M)$ by its convex hull, the exact analogue of the Maxwell construction used in the van der Waals gas. However, we have not attempted to implement such a fix because, in practice, we have encountered no difficulty in employing the multi-valued entropy $H_X(x_1, ..., x_M)$ for any of our five datasets A-E. This is perhaps due to the fact that the swallowtails are not very large and their vertices are located at a point somewhat beyond the points $(\pm 1, \pm 1)$ in the x_1x_2-plane for $M = 2$. In fact, the minimizing histories shall rarely enter this region of multi-valuedness unless there are extremely accurate measurements which force the history there. In any case, our algorithm as formulated originally, without any Maxwell-type construction, successfully converged for all of the datasets A-E and results for those were presented in II. It is interesting that the left-linear ansatz led to a less dire breakdown of realizability for the first-order closure of the mean-field variational equations than it did for the second-order closure of the KSP equations.

2.2.1(b) Double-Exponential Ansatz (MFV-1DE)

Another first-order closure method for the mean-field variational approach uses the double-exponential ansatz. In that case, there is a closed form for the effective action $\Gamma_X[x]$, given by (21) in the Appendix with $x(t) = m(t) = \mu(\lambda(t))$. Again it is more convenient to use the (γ, λ)-formulation, because it avoids minimizations at each timestep to find λ given μ. It is much easier to obtain $\mu(\lambda)$ once λ is known. As in I(30), the M-time entropy is obtained from the effective action by minimization over the history at intermediate times:

$$H_X(x_1, ..., x_M) = \min_{\{x: x(t_m) = x_m, m = 1, ..., M\}} \Gamma_X[x], \qquad (8)$$

Notice that the action in (21) is always single-valued, non-negative and locally convex near its global minimum, the solution of the first-order moment-closure equation. The relative entropy function obtained from the Contraction Principle in (8) will share these properties. As expected, the multi-valuedness which mars the multi-time entropy calculated from the left-linear ansatz is cured by using the double-exponential ansatz. There is no sign of the swallowtail singularities, which appeared in Figure 3(b) for the left-linear ansatz, in a contour plot of the two-time entropy $H_X(x_1, x_2)$ calculated from (8) for the double-well model using the first-order closure with the double-exponential ansatz.

The algorithms to calculate the conditional mean $x_*(t)$ and variance $\sigma_*^2(t)$ are the same as before, and reduce to minimizations of the effective action $\Gamma_X[x]$ (with or without constraints). For this purpose we found it

most efficient to use a hybrid algorithm which combined conjugate-gradient minimization with direct solution of the Euler-Lagrange equations (12),(13) for the jump conditions (14), (15). The solution method of the latter equations was the same as that used for the KSP closure equations in section 2.1.1(b), namely, a Newton relaxation algorithm of the discretized equations. We found the same small domain of convergence of the Newton method here as we found earlier for the KSP closure equations. The difference is that here the equations plus jump conditions are variational, so that the Newton relaxation may be initialized with the output of a conjugate-gradient minimization of the action (21). The latter is rather slowly converging, but robust, since convergence is guaranteed. The output of the conjugate-gradient minimization yields a good guess for the solution $\lambda(t)$ and the latter may be input into (13) which is then solved backward to yield a corresponding guess for $\gamma(t)$. By switching over to Newton relaxation at successive stages of convergence of the conjugate-gradient algorithm, we developed an algorithm with the robustness of conjugate-gradient but the speed and accuracy of Newton. This hybrid algorithm was used to generate all results with the first-order closure, double-exponential ansatz in II.

2. *Second-Order Closure*

Although first-order closures are the simplest and most widely applicable, second-order closures are conceivably practical. As mentioned earlier, solution of equations for covariances are also required for Kalman filtering and smoothing schemes. Therefore, the second-order closure described in II,Section 3.2 is also applied within the mean-field variational approach. It is interesting to see whether addition of further moments to the closure improves or degrades performance.

Most details for the mean-field analysis with the second-order closure are the same as for the first-order closures, discussed in section 2.2.1. Therefore, we shall only mention here the essential differences. First, since now the vector of moments is $(M_1(x), M_2(x)) = (x, x^2)$, we obtain the action for the state-history $\mathsf{x}(t)$ from

$$\Gamma_X[\mathsf{x}] = \min_{m_2} \Gamma_M[m_1, m_2], \tag{9}$$

where $(m_1, m_2) = (\mathsf{x}, X)$. This leads to rather obvious modifications in the algorithms discussed earlier. For example, the formulas (2)-(8) in the preceding section 2.2.1 now hold for $F_M(\boldsymbol{\lambda}_1, ..., \boldsymbol{\lambda}_M), H_M(\mathbf{m}_1, ..., \mathbf{m}_M)$ rather than for $F_X(\lambda_1, ..., \lambda_M), H_X(\mathsf{x}_1, ..., \mathsf{x}_M)$ directly. Results for the latter must be obtained by applying the Contraction Principle, in the form of (9). Another obvious change is that the forward-backward equations from the Appendix that must be solved are now 2-component vector equations. It is no longer a mere convenience, but in fact a necessity, to use the $(\boldsymbol{\gamma}, \boldsymbol{\lambda})$-formulation of the Appendix in any conjugate-gradient minimizations. In fact, the alternative $(\boldsymbol{\alpha}, \boldsymbol{\mu})$-formulation introduces a moment realizability constraint, $\mu_2 - \mu_1^2 \geq 0$, which is not respected by a conjugate-gradient algorithm. Instead the $(\boldsymbol{\alpha}, \boldsymbol{\mu})$-formulation can only be used in conjunction with optimization methods involving inequality constraints [22]. When calculating the conditional variance $\sigma_*^2(t)$ using the $(\boldsymbol{\gamma}, \boldsymbol{\lambda})$-formulation, the constrained minimization in (2) must be carried out with a Lagrange multiplier. In our calculations for II we used that method, discussed in detail in Eyink [13].

2.2.2(a) Left-Linear Ansatz (MFV-2LL)

By means of such algorithms as outlined above, we implemented the second-order closure of the variational mean-field equations, with the left-linear ansatz. Except for the changes discussed above, we used the same efficient minimax algorithm discussed in section 2.2.1(a) for the first-order closure. In fact, we encountered the same realizability breakdown for the second-order closure as we did previously for the first-order one, i.e. multi-valuedness of the multi-time entropy in the form of swallowtail singularities. However, for the second-order closure these were even broader, and invade the region where $x_1 \approx \pm 0.5, x_2 \approx \pm 1.5$. See Figure 4. In practice, this

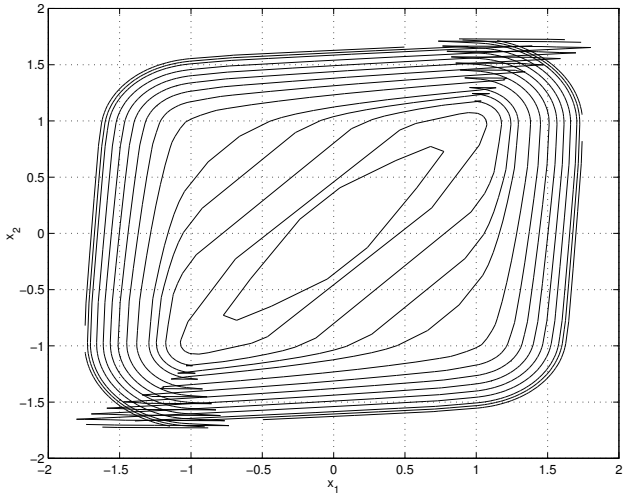

FIG. 4: MFV-2LL closure, two-time entropy function, contour plot. Swallowtails are seen at the lower left and upper right.

prevented the convergence of the conjugate-gradient minimization algorithms for any of the datasets A-E. Rather than converge, such methods jumped from sheet to sheet of the multi-valued function. Therefore, no results from this method were presented in II. It is interesting that the realizability failure was more severe for the second-order closure than the first-order one. It is still not as severe as the breakdown for the second-order closure of the KSP equations, and might be effectively dealt with by a "Maxwell construction", as discussed in section 2.2.1(a). However, we did not attempt such a remedy in this work.

2.2.2(b) Double-Exponential Ansatz (MFV-2DE)

We also implemented the second-order closure of the variational mean-field equations, with the double-exponential ansatz. With changes as indicated above, we employed the same hybrid algorithm as discussed in section 2.2.1(b) for the first-order closure. This is also for the second-order closure a fast and robust algorithm. As expected, the double-exponential ansatz avoids the realizability problems with the left-linear ansatz, discussed above. The cost function is single-valued, nonnegative, and convex. There are neither swallowtail caustics nor any other evident realizability failures encountered in our calculations.

C. Empirical Ensemble Methods

For comparison with our moment-closure methods, we have also implemented two ensemble approximations to the statistical evolution, as discussed in II,section 2.2.

2.3(a) Ensemble Kalman Filter (EnKF)

The first method we considered is the Ensemble Kalman Filter (EnKF). This method was originally proposed and tested by Evensen 1994 for a nonlinear quasigeostrophic model. We also incorporated the important correction to the analysis step in the original EnKF method, which was noted in Burgers et al. [10] and Houtekamer and Mitchell [16]. For details of the method itself, we refer the reader to those orginal references. We integrated our model stochastic ODE II,(33) by the Euler-Maruyama method with $\Delta t = 0.01$ [see 17]. We employed $N = 10,000$ samples in calculating empirical moment-averages according to II(32).

2.3(b) Ensemble Kalman Smoother (EnKS)

The second method we considered is the Ensemble Kalman Smoother (EnKS) proposed in Evensen and van Leeuwen (2000). Note that this is a sequential smoothing method, distinct from the ensemble smoothing algorithm earlier introduced by those same authors in van Leeuwen and Evensen [9]. The latter is now referred to simply as the Ensemble Smoother (EnS) method. In fact, we employed both methods, EnKS and EnS, for our datasets A-E. However, in all cases the original EnS method performed considerably worse than did the EnKS method, so that only results of the latter are reported in II. Our numerical implementation of the EnKS method was the same as that described above for EnKF (which, in fact, forms a subalgorithm of EnKS).

III. COMPUTATIONAL EFFICIENCY

In this section we shall present our comparisons of the moment-closure estimation schemes with the empirical ensemble methods, in terms of computational speed and efficiency.

TABLE I: Wall-clock time for the KS and EnKF methods.

Method	Parameters	Time (seconds)
KS-2E	$\Delta t = 0.01$	0.49
EnKF	$\Delta t = 0.01,\ N = 1600$	15.58

A. Comparison of Filtering Schemes

First, we consider the approximate filters. The methods we shall compare are the KS-2E closure of section 2.1(a) and the EnKF method of section 2.3(a). It is not as interesting to consider the optimal KS method of I, which uses the forward Kolmogorov equation, because it cannot be applied to realistic geophysical systems with many degrees of freedom. Therefore, we shall systematically compare just the suboptimal methods which may be applied practically to such systems. All of these methods calculate an approximation to the entire filter distribution, $\mathcal{P}_F(\mathbf{x}, t)$, from which the first two moments presented in II, $x_F(t)$ and $\sigma_F^2(t)$, were obtained by integration. Since in II the results on filters were given only for dataset A, we shall consider here as well only that dataset. For a fair comparison, we ran the codes for both approximate methods on the same 200 MHz machine, a Linux-based Intel Pentium Pro II. In each case, we made a choice of parameters in the method so that a tolerance of 5% was achieved, compared with a fully converged calculation. For KS-2E the only parameter is the time-step Δt used in the integration of the closure equations. For EnKS there is, in addition to the time-step Δt, also the number of samples N. The total time to calculate both $x_F(t)$ and $\sigma_F^2(t)$ to 5% accuracy appears in Table I, along with the parameter values required, for both of these methods: We shall also just mention that the computation with the optimal KS method of I, using $\Delta x = 0.09375$ and $\Delta t = 0.01$, required 7.64 seconds.

In terms of computational speed, the KS-2E closure is the fastest of all three methods considered. The KS-2E closure solving ODE's is about fifteen times faster than the exact KS filter solving PDE's for this simple test case. Of course, as the dynamical equations become more and more complicated, the closure algorithm would be expected to gain even a greater advantage in speed (but with possible loss in accuracy). EnKF was slowest of all the methods, which is not surprising for a Monte Carlo algorithm. We expect that the KS closure methods, as exemplified by KS-2E, will continue to be faster than the the EnKF method, or at least be no slower, also for more realistic problems. The reason is that, in EnKF, one must solve the full dynamics, analogous to II(33), for N separate samples, where N is at least 100. However, in KS closure methods one must solve equations for means and covariances, such as II(8),(44), but only once. Furthermore, the solutions of the closure equation represent ensemble-averaged quantities. Hence, they shall be smoother in space and time than individual realizations,

and thus require a less fine resolution in spacetime to compute numerically.

It was already shown in II that, for our simple test problem, II(33), the KS-2E closure gives also a more accurate approximation to the optimal KS results than does EnKF. Whether this shall remain true for more complicated dynamical systems, depends, of course, on the quality of the statistical ansätze which are input into the moment-closure scheme. For the simple test problem, the KS-2E closure reduces as well the storage requirements of EnKF. The latter needs to save the current values of the N sample solutions $x^{(n)}(t),\ n = 1, ..., N$ of the equation II(33). In the KS-2E closure, one must store only the current values of the two moments $\mu_1(t), \mu_2(t)$. However, in more realistic, spatially-extended systems, this means that equations for space fields and also a space-space covariance function must be stored in a typical KS closure scheme. Let S_C be the number of space points used in the computation of the closure equations, S_M the (greater) number required for the solution of the primitive model, and N the number of samples required in the calculation of ensemble-averages. As long as $S_C^2 < S_M \cdot N$, then the storage requirements for a KS-closure will be less than that for the EnKF method, on a rough order of magnitude.

It should be noted that these considerations apply only for the most standard forms of closure, in which all the first-order moments, and possibly also the second-order moments, of the state variables of the system are used in constructing the closure equation. However, there is nothing that restricts to one to such conventional closures. One can expect more substantial computing economies to arise from more ingenious and insightful closures, which incorporate more knowledge of the statistics and dynamics of the system.

B. Comparison of Smoothing Schemes

Now we shall compare the approximate smoothers. As for the filters, we do not make any systematic comparisons for the methods KSP and MFV that use the forward-backward Kolmogorov equations. Instead, we shall compare just the suboptimal methods which may be applied practically to large-scale systems, namely, MFV-1LL, MFV-1DE, MFV-2DE, and EnKS. Since calculations for all of the datasets gave rather similar results in regard to computing time, we shall only report the results here for dataset C. In each case, we used the most efficient algorithm that we have in hand, as discussed in Section 2, although we have not attempted to optimize the codes themselves for performance. For a fair comparison, we ran all four codes on the same 200 Mhz Pentium Pro II as used for the filters. In each case, we made a choice of parameters to obtain 5% accuracy in the results. For the EnKS method, there are two parameters, time-step Δt and number of samples N. For the MFV methods, the parameters involved are time-step Δt, the tolerance TOL requested in the conjugate-gradient minimization, and the discretization δ used in the finite-difference approximation of the Hessian. Since we used a first-order finite-differencing, the optimal value of the latter is $\delta \sim \sqrt{TOL}$ and we have always made that choice. The minimizations in the MFV methods were carried out with a standard conjugate-gradient-type algorithm, the TOMS 500 algorithm CONMIN [see 18]. The total time to calculate both $x_*(t)$ and $\sigma_*^2(t)$ to 5% accuracy for each of these methods is shown in Table II. We shall also mention that exact KSP with $\Delta x = 0.09375,\ \Delta t = 0.01$ took 17.97 seconds to compute the conditional distribution for our simple double-well model.

TABLE II: Wall-clock time for the various methods.

Method	Parameters	Time (seconds)
MFV-1LL	$\Delta t = 0.01,\ TOL = 10^{-3}$	376.07
MFV-1DE	$\Delta t = 0.1,\ TOL = 10^{-2}$	0.77
MFV-2DE	$\Delta t = 0.1,\ TOL = 10^{-3}$	5.79
EnKS	$\Delta t = 0.01,\ N = 3600$	42.16

At first sight, it is perhaps surprising that MFV-1LL performs so badly in comparison to the other methods (even exact KSP). There are two reasons for this. First, a smaller time step $\Delta t = 0.01$ was required than for the other MFV methods. Use of a larger value of Δt gave rise to divergent results, due a realizability breakdown similar to what was found in section 2.2.2(a) with MFV-2LL at finer resolution. Second, there is no closed-form expression for the cost function in the MFV-1LL method. Using the most efficient minimax algorithm discussed in section 2.2.1(a), still the entire code relies on the conjugate-gradient algorithm for convergence. While conjugate-gradient is robust and reliable, it is also very slow. The hybrid algorithm discussed in section 2.2.1(b) that was employed for MFV-1DE and MFV-2DE allows a switch over to a fast Newton algorithm to solve the Euler-Lagrange equations once conjugate-gradient has produced a sufficiently good initial approximation. For these reasons, the MFV-1LL method turns out not only to be much less accurate, but also far slower than MFV-1DE and MFV-2DE. The latter two methods, when implemented by the efficient hybrid algorithm are also both much faster than EnKS. One of the reasons for this is the ten times larger timestep Δt allowed by the MFV closure methods (including MFV-1LL when there was no realizability breakdown, as, for example, for dataset E). This is an example of the reduced resolution requirements expected in working with equations for smooth statistical moments rather than for individual realizations, which are rougher. Another reason is just the well-known slow rate of convergence of Monte Carlo schemes, in which statistical errors decay only $\sim 1/\sqrt{N}$. We expect that the MFV-2DE closure method will at least equal the EnKS method in speed also for more realistic problems, for similar reasons as in the case of filtering. The MFV-1DE

method will likely be much faster even than MFV-2DE—and therefore also EnKS—because it requires the solution of only first-order moment equations and not equations for second-order moments or covariances.

For the test problem II(33) all the MFV-closure methods give a more accurate approximation to the optimal KSP results than does EnKS. See II. As in the filtering problem, this shall remain true for more complicated systems only if sufficiently good guesses of the statistics are input into the moment-closure schemes. The storage requirements of the various methods may also be briefly compared. In MFV-1LL one must store, in addition to the current conjugate-gradient iterate $(\lambda_1, ..., \lambda_M)$, also the corresponding solution histories $\lambda(t)$ of (12) and $\partial\lambda(t)/\partial\lambda_m$ of (19) for $m = 1, ..., M$ at times $t \in [t_i, t_f]$. Note, however, that the latter derivatives need not all be stored simultaneously, but only the contribution to the gradient of the cost function needs to be stored. The storage requirements of the latter and the conjugate-gradient iterate are of order $S_C \cdot M$, with notations as above. However, the storage of the solution histories requires order $S_C^2 \cdot T_C$ numbers, where T_C is the number of time points used in the computation of the closure equations. For the MFV-1DE method, the storage of the current conjugate-gradient iterate requires a storage of order $S_C \cdot T_C$ and a similar storage for the current Newton iterate in the solution of the Euler-Lagrange equations. For the MFV-2DE method this storage requirement increases in both cases to $S_C^2 \cdot T_C$. For the EnKS method, N solution histories $\mathbf{x}^{(n)}(t)$, $t \in [t_i, t_f]$, $n = 1, ..., N$ of the model equation must be stored, a set of numbers of order $S_M \cdot N \cdot T_M$, where T_M is the number of time points required in the computation of the primitive model equations. As long as $S_C^2 \cdot T_C < S_M \cdot N \cdot T_M$, then the storage requirements for MFV-1LL and MFV-2DE will be less than that for the EnKS method. The storage for MFV-1DE will then usually be far less than that required for EnKS. Some reductions can also be achieved for EnKS by storing past histories only for a finite memory-interval or correlation-time prior to the current state. Details on this strategy appear in Evensen and van Leeuwen [11].

IV. DISCUSSION AND SUMMARY

In this paper we have described efficient numerical algorithms to implement various estimation or data assimilation schemes based on moment-closure approximations to the dynamics and either the exact or a mean-field conditional analysis. It was found that the methods using a linear ansatz for the left trial function, or solution of the backward Kolmogorov equation, are subject to failure of realizability. This is particularly true when using the exact conditioning. However, the methods employing the mean-field variational analysis are generally realizable. In terms of computational speed and efficiency, the mean-field variational closures were found to perform better than empirical ensemble methods in a simple test problem. It is expected that the closure methods will be at least competitive with ensemble methods in more realistic applications in the geosciences, involving one or several partial differential equations.

Here let us add just a few summary remarks about the individual strengths of these two methods:

The ensemble method has an advantage that it is systematically convergent in the limit as $N \to \infty$. By contrast, moment closure approximations are generally not so, and it is usually not clear how to improve them in a straightforward way. (An exception to this statement is closure achieved by expansion of the trial functions in orthogonal polynomials; e.g. see [19].) On the other hand, ensemble methods are very slowly converging, suffering the $\frac{1}{\sqrt{N}}$ errors typical of Monte Carlo methods. This means, for example, that a 10% accuracy in calculating moments will generally require no less than N=100 independent samples. Therefore, the empirical-average approximation will, in practical applications, often be far from converged to the correct evolution, even though convergence is achievable in principle. Methods to accelerate convergence of Monte Carlo integration, such as stratified sampling [20] and Richardson extrapolation [21], can improve this situation.

In terms of convergence of space-time discretizations, the closure method will generally have some advantage. Because it integrates moment-closure equations which correspond to statistical averages, its solutions will generally be smoother and require less numerical resolution. On the other hand, the empirical ensemble method integrates equations for individual realizations which may contain sharp gradients or rapid oscillations that need to be resolved. This advantage of moment-closure schemes will increase with the size of the system and the number of active scales of motion. On the other hand, the moment-closure method has its own convergence problems. If fast iterative solution of Euler-Lagrange equations fails, it may require direct minimization of the convex cost function. Descent algorithms for this purpose are suitably robust but also quite slow. Another important issue for computational efficiency is the storage requirements of each method. Implementations of ensemble smoother schemes require storage of N entire sample histories, at least for times less than some characteristic correlation time of the system. In contrast, the closure schemes require storage only of the current approximation to the conditional mean history, and, possibly, the conditional covariance.

A. APPENDIX: THE FORWARD-BACKWARD EQUATIONS

A. Dual Thermodynamic Fields

To integrate the forward equations II(8) or II(25), one must carry out the maximization in II(41) at each timestep in order to determine the value of $\boldsymbol{\lambda}$ corre-

sponding to the current value of $\boldsymbol{\mu}$. This is necessary in order to calculate averages with respect to the exponential PDF II(10). However, this maximization at each timestep adds an additional computational cost to the integration. As we show here, it is possible to eliminate this maximization by formulating the closure dynamics directly in terms of the dual "thermodynamic fields" $\boldsymbol{\lambda}$. The equation for the dual fields is

$$\dot{\boldsymbol{\lambda}} = \mathbf{W}(\boldsymbol{\lambda}, t). \tag{10}$$

The vector $\mathbf{W}(\boldsymbol{\lambda}, t) = \boldsymbol{\Gamma}(\boldsymbol{\lambda}, t)\mathbf{V}(\boldsymbol{\mu}(\boldsymbol{\lambda}, t), t)$, where $\mathbf{V}(\boldsymbol{\mu}, t)$ is the dynamical vector for the moments themselves, defined in II(7). The second-order irreducible correlation function $\boldsymbol{\Gamma}(\boldsymbol{\lambda}, t) = [\mathbf{C}(\boldsymbol{\lambda}, t)]^{-1}$, where $\mathbf{C}$ is the covariance matrix of the moment-variables. Since this equation is obtained from II(8) by a mere change of variables, it retains all of the good properties, such as an H-theorem. The H-function or relative entropy may also be expressed directly in terms of the dual $\boldsymbol{\lambda}$-variables, via the formula $H_M(\boldsymbol{\lambda}, t) = \boldsymbol{\mu}^\top(\boldsymbol{\lambda}, t)\boldsymbol{\lambda} - F_M(\boldsymbol{\lambda}, t)$.

The action functional II(6) that arises from left-linear ansatz may also be expressed in terms of the dual thermodynamic fields $\boldsymbol{\lambda}$ along with a corresponding set of adjoint variables $\boldsymbol{\gamma} = \mathbf{C}\boldsymbol{\alpha}$. One has, corresponding to II(6), the reduced action

$$\Gamma[\boldsymbol{\gamma}, \boldsymbol{\lambda}] = \int_{t_i}^{t_f} dt\ \boldsymbol{\gamma}^\top(t)[\dot{\boldsymbol{\lambda}}(t) - \mathbf{W}(\boldsymbol{\lambda}(t), t)]. \tag{11}$$

The Euler-Lagrange equations are the closure equation

$$\dot{\boldsymbol{\lambda}} = \mathbf{W}(\boldsymbol{\lambda}, t) \tag{12}$$

and its adjoint equation

$$\dot{\boldsymbol{\gamma}} + \left(\frac{\partial \mathbf{W}}{\partial \boldsymbol{\lambda}}\right)^\top (\boldsymbol{\lambda}, t)\boldsymbol{\gamma} = \mathbf{0}. \tag{13}$$

The mean-field jump conditions in these variables are quite simple, forward in time:

$$\boldsymbol{\lambda}_m^+ = \boldsymbol{\lambda}_m^- + \boldsymbol{\lambda}_m \tag{14}$$

and backward in time:

$$\boldsymbol{\mu}(\boldsymbol{\lambda}_m^+, t_m) + \boldsymbol{\gamma}_m^+ = \boldsymbol{\mu}(\boldsymbol{\lambda}_m^-, t_m) + \boldsymbol{\gamma}_m^-, \tag{15}$$

for $m = 1, ..., M$. The former are obviously equivalent to I(12), while the latter follow from the requirement of continuity of

$$\mathbf{m}(t) = \boldsymbol{\mu}(\boldsymbol{\lambda}(t), t) + \boldsymbol{\gamma}(t) \tag{16}$$

at the measurement times. Cf. II(14),(15). The KSP jump conditions in a second-order closure with $\boldsymbol{\lambda} = (\boldsymbol{\ell}, \boldsymbol{\Lambda})$, in the notations of II, are forward in time:

$$\boldsymbol{\ell}_m^+ = \boldsymbol{\ell}_m^- + \mathbf{R}_m^{-1}\mathbf{y}_m, \quad \boldsymbol{\Lambda}_m^+ = \boldsymbol{\Lambda}_m^- - \frac{1}{2}\mathbf{R}_m^{-1}, \quad m = 1, ..., M \tag{17}$$

and backward in time the same as (15) above.

For the purpose of the minimax algorithm discussed in Section 2.2.1(a), one requires also the components of the derivatives $\partial \mathbf{m}_{n'}/\partial \boldsymbol{\lambda}_n$, $n, n' = 1, ..., M$. These are obtained by, first, differentiating (16) to get

$$\frac{\partial \mathbf{m}(t)}{\partial \boldsymbol{\lambda}_n} = \mathbf{C}(\boldsymbol{\lambda}(t), t)\frac{\partial \boldsymbol{\lambda}(t)}{\partial \boldsymbol{\lambda}_n} + \frac{\partial \boldsymbol{\gamma}(t)}{\partial \boldsymbol{\lambda}_n} \tag{18}$$

and setting $t = t_{n'}$. Note that, by causality, the first term on the righthand side does not appear for $t < t_n$. We may obtain the partial derivatives in (18) by integrating forward in time

$$\frac{d}{dt}\frac{\partial \boldsymbol{\lambda}(t)}{\partial \boldsymbol{\lambda}_n} = \frac{\partial \mathbf{W}}{\partial \boldsymbol{\lambda}}(\boldsymbol{\lambda}(t), t)\frac{\partial \boldsymbol{\lambda}(t)}{\partial \boldsymbol{\lambda}_n} \tag{19}$$

for $t > t_n$, starting with initial condition $\partial \lambda_{n'}(t_n+)/\partial \lambda_n = \delta_{n,n'}$, and equation

$$\frac{d}{dt}\frac{\partial \boldsymbol{\gamma}(t)}{\partial \boldsymbol{\lambda}_n} + \boldsymbol{\gamma}^\top(t)\frac{\partial^2 \mathbf{W}}{\partial \boldsymbol{\lambda}\partial \boldsymbol{\lambda}}(\boldsymbol{\lambda}(t), t)\frac{\partial \boldsymbol{\lambda}(t)}{\partial \boldsymbol{\lambda}_n} + \left(\frac{\partial \mathbf{W}}{\partial \boldsymbol{\lambda}}\right)^\top (\boldsymbol{\lambda}(t), t)\frac{\partial \boldsymbol{\gamma}(t)}{\partial \boldsymbol{\lambda}_n} = \mathbf{0} \tag{20}$$

backward in time for $t_f > t > t_n$, with final condition $\partial \gamma_{n'}(t_f)/\partial \lambda_n = 0$. Again, the solutions of the forward equation (19) must be stored for use in the backward equation (20).

In the double-exponential ansatz, the effective action can be written as

$$\Gamma_M[\boldsymbol{\lambda}] = \frac{1}{4}\int_{t_i}^{t_f} dt\ [\dot{\boldsymbol{\lambda}}(t) - \mathbf{W}(\boldsymbol{\lambda},t)]^\top \mathbf{S}^{-1}(\boldsymbol{\lambda},t)[\dot{\boldsymbol{\lambda}}(t) - \mathbf{W}(\boldsymbol{\lambda},t)] + H_M(\boldsymbol{\mu}(\boldsymbol{\lambda}(t_i)),t_i) \tag{21}$$

using $\mathbf{m}(t) = \boldsymbol{\mu}(\boldsymbol{\lambda}(t),t)$ in (20) of II. Here

$$\mathbf{S}(\boldsymbol{\lambda},t) = \boldsymbol{\Gamma}(\boldsymbol{\mu},t)\mathbf{Q}(\boldsymbol{\mu},t)\boldsymbol{\Gamma}(\boldsymbol{\mu},t)|_{\boldsymbol{\mu}=\boldsymbol{\mu}(\boldsymbol{\lambda},t)}\,. \tag{22}$$

Using the same adjoint variable as above, the Euler-Lagrange equations for this functional are

$$\dot{\boldsymbol{\lambda}} = \mathbf{W}(\boldsymbol{\lambda},t) + 2\mathbf{S}(\boldsymbol{\lambda},t)\boldsymbol{\gamma}, \tag{23}$$

$$\dot{\boldsymbol{\gamma}} + \left(\frac{\partial \mathbf{W}}{\partial \boldsymbol{\lambda}}\right)^\top \boldsymbol{\gamma} + \frac{\partial}{\partial \boldsymbol{\lambda}}\left(\boldsymbol{\gamma}^\top \mathbf{S}\boldsymbol{\gamma}\right) = \mathbf{0} \tag{24}$$

with boundary values $\boldsymbol{\gamma}(t_i) = \mathbf{C}(\boldsymbol{\lambda}(t_i),t_i)\boldsymbol{\lambda}(t_i)$ and $\boldsymbol{\gamma}(t_f) = \mathbf{0}$. At measurement times there are jumps only in the backward equation. In the mean-field case the backward jump conditions are

$$\boldsymbol{\gamma}_m^+ = \boldsymbol{\gamma}_m^- + \mathbf{C}(\boldsymbol{\lambda}(t_m),t_m)\mathbf{R}_m^{-1}[\mathbf{m}(t_m) - \mathbf{y}_m], \quad m = 1,...,M. \tag{25}$$

In the case of KSP, we denote as $\boldsymbol{\gamma} = (\mathbf{g},\mathbf{G})$ the adjoint variable corresponding to $\boldsymbol{\lambda} = (\boldsymbol{\ell},\boldsymbol{\Lambda})$. The backward jump conditions are then

$$\begin{bmatrix} \mathbf{g}_m^- \\ \mathbf{G}_m^- \end{bmatrix} = \begin{bmatrix} \mathbf{g}_m^+ \\ \mathbf{G}_m^+ \end{bmatrix} + \begin{bmatrix} \mathbf{C}_{xx}(\boldsymbol{\lambda}(t_m),t_m) & \mathbf{C}_{xX}(\boldsymbol{\lambda}(t_m),t_m) \\ \mathbf{C}_{Xx}(\boldsymbol{\lambda}(t_m),t_m) & \mathbf{C}_{XX}(\boldsymbol{\lambda}(t_m),t_m) \end{bmatrix} \begin{bmatrix} \mathbf{R}_m^{-1}\mathbf{y}_m \\ -\frac{1}{2}\mathbf{R}_m^{-1} \end{bmatrix}, \tag{26}$$

$m = 1,\ldots,M$.

B. Cumulants and Generating Functions

The terms in the closure equation (10) involve various moments $\mu_n = \langle x^n \rangle$ of the dynamical variable x of the model problem II(33), for $n = 1,2,3....$ Likewise, the terms in the adjoint equation (13) require derivatives $\partial\mu_n/\partial\lambda_{n'}$ where $n' = 1$ for the first-order closure and $n' = 1,2$ for the second-order closure. The equation (20) requires also second-derivatives $\partial^2\mu_n/\partial\lambda_{n'}\partial\lambda_{n''}$, as does the solution of (23),(24) by a Newton method. We sketch here a method to obtain all of these moments and their derivatives to working precision, by means of cumulants. Although we might do so more generally, our remarks here shall be most specifically for the model equation II(33).

The single-time "free energy" $F_M(\boldsymbol{\lambda},t)$ defined in II,Section 2.1.1 is a *cumulant-generating function* for the moment-variables $\mathbf{M}(\mathbf{x},t)$ appearing in the closure. For the two exponential PDF closures discussed in II,Section 3, $M_n(x) = x^n$ with $n = 1,2$. Thus, nth-order cumulants C_n of the variable $M_1 = x$ can be obtained by derivatives

$$C_n = \frac{\partial^n F_M}{\partial \lambda_1^n} \tag{27}$$

with respect to the variable λ_1. It follows directly from this that

$$\frac{\partial C_n}{\partial \lambda_1} = C_{n+1}. \tag{28}$$

Furthermore,

$$C_{n;m} = \frac{\partial^{n+m} F_M}{\partial \lambda_1^n \partial \lambda_1^m} \tag{29}$$

is the joint cumulant of nth order in $M_1 = x$ and of mth order in $M_2 = x^2$. Therefore,

$$\frac{\partial C_n}{\partial \lambda_2} = C_{n;1}. \tag{30}$$

All of the cumulants C_n and $C_{n;m}$ may be readily calculated analytically by differentiating the closed-form expressions of F_M given for the first- and second-order closures in II,Section 3.

The moments μ_n required in the closure can be expressed by an expansion in cumulants.

$$\mu_n = \sum_{r_1 n_1 + \cdots + r_s n_s = n} \frac{n!}{(n_1!)^{r_1}\cdots(n_s!)^{r_s}\ r_1!\cdots r_s!}[C_{n_1}]^{r_1}\cdots[C_{n_s}]^{r_s} \tag{31}$$

See [14], Chapter 10. The sum is over all partitions of the integer n into the s summands n_i repeated r_i times, $i = 1, ..., s$, with $n_1 < n_2 < \cdots < n_s$:

$$n = \underbrace{n_1 + \cdots + n_1}_{r_1} + \underbrace{n_2 + \cdots + n_2}_{r_2} + \cdots + \underbrace{n_s + \cdots + n_s}_{r_s}. \tag{32}$$

The total number of such partitions for fixed n is the famous number-theoretic function $p(n)$. Explicitly for the first few values of $n = 1, 2, 3, 4$:

$$\mu_1 = C_1, \tag{33}$$

$$\mu_2 = C_2 + C_1^2, \tag{34}$$

$$\mu_3 = C_3 + 3C_1C_2 + C_1^3, \tag{35}$$

$$\mu_4 = C_4 + 4C_1C_3 + 3C_2^2 + 6C_1C_2 + C_1^4. \tag{36}$$

The derivatives of moments, $\partial\mu_n/\partial\lambda_{n'}$ can be obtained by differentiating (31), using the product rule and equations (28),(30).

C. Numerical Schemes for Variational Equations

The numerical integration of the Euler-Lagrange equations (12),(13) for the left-linear ansatz requires a suitable discretization. It is most practical to generate the required discrete form of the equations from a discretization of the action (11). For example, the discrete action

$$\Gamma[\boldsymbol{\gamma}, \boldsymbol{\lambda}] = \sum_{n=n_i}^{n_f} \boldsymbol{\gamma}_n^\top [\boldsymbol{\lambda}_n - \boldsymbol{\lambda}_{n-1} - \Delta t\, \mathbf{W}(\boldsymbol{\lambda}_{n-1}, t_{n-1})], \tag{37}$$

leads upon variation to an explicit Euler scheme for the forward equation

$$\boldsymbol{\lambda}_n = \boldsymbol{\lambda}_{n-1} + \Delta t\, \mathbf{W}(\boldsymbol{\lambda}_{n-1}, t_{n-1}) \tag{38}$$

and a backward Euler scheme for the adjoint equation

$$\boldsymbol{\gamma}_{n-1} = \boldsymbol{\gamma}_n + \Delta t \left(\frac{\partial \mathbf{W}}{\partial \boldsymbol{\lambda}}\right)^\top (\boldsymbol{\lambda}_{n-1}, t_{n-1})\boldsymbol{\gamma}_n. \tag{39}$$

This approach guarantees that structural properties of the continuous-time equations are preserved by the numerical discretization. For example, if jump conditions like (14),(15) are imposed at discrete measurement times $n_{(m)}$, $m = 1, ..., M$ and if one defines a "free-energy" $F_M(\boldsymbol{\lambda}_1, ..., \boldsymbol{\lambda}_M) = \sum_{m=1}^M (\Delta F_M)(\boldsymbol{\lambda}_1, ..., \boldsymbol{\lambda}_m)$ as in II,Section 2.1.1, then it can be shown that $\mathbf{m}_m = \partial F_M/\partial \boldsymbol{\lambda}_m$ is equal to

$$\mathbf{m}_m = \boldsymbol{\mu}(\boldsymbol{\lambda}^+_{n_{(m)}}, t_{n_{(m)}}) + \boldsymbol{\gamma}^+_{n_{(m)}} = \boldsymbol{\mu}(\boldsymbol{\lambda}^-_{n_{(m)}}, t_{n_{(m)}}) + \boldsymbol{\gamma}^-_{n_{(m)}}, \tag{40}$$

for $m = 1, ..., M$. Here $+$ indicates values just after the jump and $-$ values just prior to the jump at the measurement time $n_{(m)}$. This exact property of the continuum equations (see Eyink [13], Appendix 2) is therefore preserved by the discretized equations for every choice of Δt. Thus, there is no need to rely upon convergence as $\Delta t \to 0$ or high-order accuracy in Δt in order to ensure that the property (40) will hold. Since the minimax algorithm discussed in Section 2.1.1(a) depends upon this property, it is clearly desirable to retain it exactly in the numerical scheme. For this reason, we have been able to perform all of our calculations with the explicit Euler scheme in (37)-(39), which is only first-order accurate, and still obtain well-converged results

The same remarks apply to the closed-form cost function (21) from the double-exponential ansatz. A discrete-time form of the effective action can be written as

$$\Gamma_M[\boldsymbol{\gamma}, \boldsymbol{\lambda}] = \sum_{n=n_i}^{n_f} \left\{\boldsymbol{\gamma}_n^\top [\boldsymbol{\lambda}_{n+1} - \boldsymbol{\lambda}_n - \Delta t\, \mathbf{W}(\boldsymbol{\lambda}_n, t_n)] - \Delta t\, \boldsymbol{\gamma}_n^\top \mathbf{S}(\boldsymbol{\lambda}_n, t_n)\boldsymbol{\gamma}_n\right\} + H_M(\boldsymbol{\lambda}_{n_i}, t_{n_i}). \tag{41}$$

The Euler-Lagrange equations for this functional are

$$\boldsymbol{\lambda}_{n+1} = \boldsymbol{\lambda}_n + \Delta t\, \mathbf{W}(\boldsymbol{\lambda}_n, t_n) + 2\Delta t\, \mathbf{S}(\boldsymbol{\lambda}_n, t_n)\boldsymbol{\gamma}_n, \tag{42}$$

and

$$\boldsymbol{\gamma}_{n-1} = \boldsymbol{\gamma}_n + \Delta t \left(\frac{\partial \mathbf{W}}{\partial \boldsymbol{\lambda}}\right)^{\top} (\boldsymbol{\lambda}_n, t_n)\boldsymbol{\gamma}_n + \Delta t \frac{\partial}{\partial \boldsymbol{\lambda}} \left(\boldsymbol{\gamma}^{\top} \mathbf{S}(\boldsymbol{\lambda}, t_n)\boldsymbol{\gamma}\right)\big|_{\boldsymbol{\gamma}_n, \boldsymbol{\lambda}_n}, \tag{43}$$

for $n = n_i, ..., n_f$. The boundary values become $\boldsymbol{\gamma}_{n_i-1} = \mathbf{C}(\boldsymbol{\lambda}_{n_i}, t_{n_i})\boldsymbol{\lambda}_{n_i}$ and $\boldsymbol{\gamma}_{n_f} = \mathbf{0}$. This Euler discretization scheme is particularly convenient when solving the initial/final-value problem by a Newton relaxation method. In that case, the linear equations which appear at each iteration of the Newton algorithm are block-tridiagonal and can be easily solved by sparse matrix methods.

Acknowledgments

We wish to thank C. D. Levermore and M. Ghil in particular for much useful advice on the subject of this paper and, also, the anonymous reviewers of a previous version of the paper. This work, LAUR 02-3094, was carried out in part at Los Alamos National Laboratory under the auspices of the Department of Energy and supported by LDRD-ER 2000047. We also received support from NSF/ITR, Grant DMS-0113649 (GLE,JMR), as well as from NASA, Goddard Space Flight Center, Grant NAG5-11163 (JMR).

[1] G. L. Eyink, J. M. Restrepo, and F. J. Alexander, *A statistical-mechanical approach to data assimilation for nonlinear dynamics. i. analysis approximations, submitted* (2002).

[2] G. L. Eyink, J. M. Restrepo, and F. J. Alexander, *A statistical-mechanical approach to data assimilation for nonlinear dynamics. ii. analysis approximations, submitted* (2002).

[3] E. Tziperman and W. C. Thacker, J. Phys. Ocean. **19**, 1471 (1989).

[4] J. M. Restrepo, G. K. Leaf, and A. Griewank, SIAM J. Sci. Comput. **19**, 1586 (1998).

[5] R. N. Miller, M. Ghil, and P. Gauthiez, J. Atmos. Sci. **51**, 1037 (1994).

[6] R. N. Miller, J. E. F. Carter, and S. T. Blue, Tellus **51A**, 167 (1999).

[7] S. R. S. Varadhan, *Large Deviations and Applications* (SIAM, Philadelphia, 1984).

[8] G. Evensen, J. Geophys. Res. **99 (C5)**, 10 143 (1994).

[9] P. J. van Leeuwen and G. Evensen, Mon. Wea. Rev. **124**, 2898 (1996).

[10] G. Burgers, P. J. van Leeuwen, and G. Evensen, Mon. Wea. Rev. **126**, 1719 (1998).

[11] G. Evensen and P. J. van Leeuwen, Mon. Weather Rev. **128**, 1852 (2000).

[12] M. K. Tippett, J. L. Anderson, C. H. Bishop, T. H. Hamill, and J. S. Whitaker, Monthly Weather Review, to appear (2003).

[13] G. L. Eyink, *A variational formulation of optimal nonlinear estimation, submitted* (2002).

[14] K. Huang, *Statistical Mechanics, 2nd. Ed.* (J. Wiley, New York, 1987).

[15] A. B. Pippard, *Response and Stability* (Cambridge University Press, Cambridge, 1985).

[16] P. L. Houtekamer and H. L. Mitchell, Mon. Wea. Rev. **126**, 796 (1998).

[17] P. Kloeden and E. Platen, *Numerical Solution of Stochastic Differential Equations* (Springer-Verlag, Berlin, 1992).

[18] D. F. Shanno and K. H. Phua, ACM Trans. Math. Softw. **6**, 618 (1980).

[19] G. L. Eyink and F. J. Alexander, J. Stat. Phys. **91**, 221 (1998).

[20] R. Balgovind, A. Dalcher, M. Ghil, and E. Kalnay, Mon. Wea. Rev. **111**, 701 (1983).

[21] H. L. Mitchell and P. L. Houtekamer, Mon. Wea. Rev. **128**, 416 (2000).

[22] The $(\boldsymbol{\gamma}, \boldsymbol{\lambda})$-formulation has also an inequality constraint, $2\sigma^2\lambda_2 \leq 1$. However, it turns out that this inequality is always well-satisfied in most estimation problems, whereas the inequality $\mu_2 - \mu_1^2 \geq 0$ is only marginally satisfied as measurements become more accurate. If the boundary $\mu_2 - \mu_1^2 = 0$ is closely approached, then conjugate-gradient algorithms will easily cross it.