

## Predicting chaotic time series with a partial model

Franz Hamilton,<sup>1</sup> Tyrus Berry,<sup>2</sup> and Timothy Sauer<sup>1,\*</sup>

<sup>1</sup>*Department of Electrical and Computer Engineering and Department of Mathematical Sciences, George Mason University, Fairfax, Virginia 22030, USA*

<sup>2</sup>*Department of Mathematics, Pennsylvania State University, University Park, Pennsylvania 16802, USA*

(Received 25 February 2015; published 16 July 2015)

Methods for forecasting time series are a critical aspect of the understanding and control of complex networks. When the model of the network is unknown, nonparametric methods for prediction have been developed, based on concepts of attractor reconstruction pioneered by Takens and others. In this Rapid Communication we consider how to make use of a subset of the system equations, if they are known, to improve the predictive capability of forecasting methods. A counterintuitive implication of the results is that knowledge of the evolution equation of even one variable, if known, can improve forecasting of all variables. The method is illustrated on data from the Lorenz attractor and from a small network with chaotic dynamics.

DOI: [10.1103/PhysRevE.92.010902](https://doi.org/10.1103/PhysRevE.92.010902)

PACS number(s): 05.45.Tp, 05.45.Jn, 92.60.Aa

One of the hallmarks of chaotic systems is the breakdown of accurate prediction. Due to exponential divergence of trajectories, long-term forecasting is seldom possible. On the other hand, short-term prediction is often feasible and significant progress has been achieved. In particular, when the system is known through time series observations alone, nonparametric methods have been developed to forecast chaotic trajectories.

For relatively low-dimensional chaotic dynamics, Takens' method of attractor reconstruction [1–4] has long been the foundation of nonparametric time series prediction methods. Under suitable genericity hypotheses, the attractor may be represented by delay coordinate vectors built from the time series observations, and methods of prediction, control, and other applications from chaotic time series have been developed [5–7]. In particular, time series prediction algorithms locate the current position in the delay coordinate representation and use analogs from previously recorded data to establish a local, low-order predictive statistical model, which can be accomplished in several ways [8–21].

At the other end of the spectrum are parametric forecasting methods, applicable when a physically motivated, complete model for the network is available. Nonlinear approaches to filtering [22–25] allow forecasting models to use the model equations to develop close to optimal predictions. Even if some variables are not observable, they may be reconstructed, *provided that their model equations are known*. However, when the model is not completely known, there has been little progress, and few methods are known that are able to span the gap between nonparametric and data assimilation methods.

In this Rapid Communication we consider the problem where not only is time series data available from a dynamical network, but in addition we possess a subset of the dynamical equations for the network. This situation reflects the case of a “partial model,” in which evolution equations for some, but not all, of the measured variables are available. We address the question of how best to extend nonparametric methods with this added, partial information for system prediction purposes.

This problem is endemic throughout the study of physical processes, where the amount of accessible data can easily exceed the availability of physical parametric models. In geophysical processes, basic principles may constrain a variable in terms of other driving variables in a way that is well understood, but the driving variables may be unmodeled or modeled with large error [26–29]. In a numerical weather prediction code, the physics may be known on the small scale but the large scale might be poorly modeled [30,31]. In a fast-slow system governing excitable media, the slow variables are often driven in a known way by fast variables that are unmodeled [32,33].

A specific example serves to illustrate the problem. Assume we can observe the  $x$ ,  $y$ , and  $z$  variables of the Lorenz-63 system [34]:

$$\begin{aligned}\dot{x} &= \sigma(y - x), \\ \dot{y} &= x(\rho - z) - y, \\ \dot{z} &= xy - \beta z,\end{aligned}\tag{1}$$

where  $\sigma = 10$ ,  $\rho = 28$ , and  $\beta = 8/3$ , but that we have no knowledge of the generating equations. A reasonably successful nonparametric forecasting method with prediction horizon  $T$  can be derived from attractor reconstruction techniques, using delay-coordinate versions of the current  $x$ ,  $y$ , and  $z$  to predict future  $y$  values, for example. The *direct prediction* method begins by locating neighbors  $[y(t'), y(t' - h), \dots, y(t' - dh)]$  of the current delay coordinates  $[y(t), y(t - h), \dots, y(t - dh)]$  in the observed data, where  $h$  is the sampling step size. The neighbors can be found by minimizing the Euclidean distance norm or any other appropriate norm. Then, the known values  $y(t' + T)$  are used in a regression with a local model (typically locally constant or locally linear) to predict the future value  $y(t + T)$ . Since we know the  $x$  and  $z$  variables as well, we can include their delay coordinates to improve location of appropriate neighbors, which typically enhances the accuracy of the prediction of  $y(t + T)$ .

Now assume that in addition to the time series data, we have extra knowledge in the form of a differential equation for one of the variables, say the  $y$  variable of the Lorenz system. In the current example, this assumption may consist of knowing

\*tsauer@gmu.edu

the single equation  $\dot{y} = x(\rho - z) - y$ . Here we consider  $y$  to be the modeled variable, whose evolution is known in terms of the unmodeled variables  $x$  and  $z$ ; the evolution equations for  $x$  and  $z$  are considered to be unknown to us. The central question of this Rapid Communication is how to use the new information. Our proposed strategy is to use the differential equation to interpolate the known data, use the results to supplement the known data, and thereby substantially improve the accuracy of direct prediction methods.

This is achieved by solving the differential equation on a finer sampling scale. Assume the data is obtained by sampling the system with step size  $h$ . Subdivide the sampling interval into  $m$  smaller steps of length  $k$ , or  $h = mk$ . We will use the training data defined on the grid of step size  $h$ , and the single differential equation for  $y$ , to generate consistent  $x, y, z$  on the finer grid of step size  $k$ . The procedure works as follows: At a given sampling time  $t_i$ , apply a multistep quadrature method (see Supplemental Material [35]) that uses previous  $x, y$ , and  $z$  time series values on the step size  $h$  grid, and the differential equation for  $\dot{y}$ , to approximate  $y(t + k)$ , where  $k$  is the new, smaller step size. This will require a quadrature method for which the input and output step sizes can be arbitrarily adjusted, since the input data has step size  $h$  and the output has step size  $k$ . We used a modified “fractional-step” multistep method that uses arbitrary  $h, k$  and can be built with arbitrary order. As an example, if  $h = 5k$ , a useful fourth-order multistep method for  $\dot{y} = f(y)$  is

$$y(t_i + k) \approx y(t_i) + k \sum_{j=1}^4 b_j f(y(t_i - (j-1)h)), \quad (2)$$

where  $[b_1, b_2, b_3, b_4] = [3591, -1003, 533, -121]/3000$ . We use an explicit method like (2) as a predictor, and then use an analogous implicit method as corrector to improve fidelity. The derivation and directions for carrying out an explicit or implicit fractional-step multistep quadrature method for arbitrary order and step sizes  $h$  and  $k$  are given in the Supplemental Material.

Once  $y(t_i + k)$  has been calculated on the entire time grid, we use the delay coordinates of the  $y$  variable at the  $t_i + k$  times to find its nearest neighbors and the corresponding  $x$  and  $z$  values. This allows us to form a locally constant reconstruction of the  $x$  and  $z$  variables at time  $t_i + k$ , namely,  $x(t_i + k)$  and  $z(t_i + k)$ , by averaging their respective nearest neighbors in delay-coordinate space. Note the curious fact that although we only know the differential equation for  $y$ , the modeled variable, we have used the single equation to extrapolate our knowledge to the unmodeled variables  $x$  and  $z$  at the new fine-grid points. Now that we have computed values of  $x, y$ , and  $z$  at all  $t_i + k$  times, we are in position to approximate  $y(t_i + 2k)$  with the same quadrature formula.

The same idea is repeated to compute values at  $t_i + 2k$ ,  $t_i + 3k$ , and so forth until all variables  $x, y$ , and  $z$  are known on the finer  $k$  grid. The result of this procedure is essentially an interpolation of the training set data, where the accuracy is leveraged by knowledge of the partial model. In Fig. 1(a), we have available a training set of 2000 observations each of  $x, y$ , and  $z$  from the Lorenz system at  $h = 0.2$  time intervals. These intervals are divided into  $m = 10$  substeps, and a four-step, fourth-order multistep quadrature method is used to integrate the known  $y$  equation. As discussed above, we can infer the

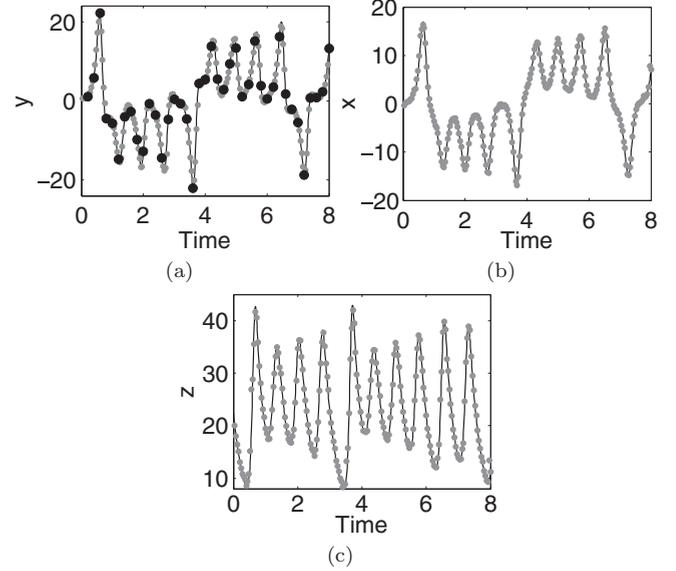


FIG. 1. Using one equation from the Lorenz system, the data set is interpolated to reconstruct all of the system variables at subsample intervals. (a) Measured data  $y$  values (black circles) are interpolated by the multistep equation solver (gray circles), approximately matching the exact solution of the Lorenz equations (gray line). At each interpolation step, we use delay coordinates of the  $y$  variable to reconstruct the corresponding (unmodeled) (b)  $x$  and (c)  $z$  variables. In (b) and (c), the reconstructed  $x$  and  $z$  variables (gray circles) are compared to the exact solution (gray line).

$x$  and  $z$  variables as well at the small step size, which are shown in Figs. 1(b) and 1(c). For this example, we used eight delays, with a time delay of 0.2 time units, and ten neighbors to reconstruct the unmodeled  $x$  and  $z$  variables at the finer grid points.

We have described the method as working forward in time in steps of size  $k$  from the known data point on the  $h$  grid. Since ordinary differential equations are time reversible, we can integrate backwards in time and average the two results to improve fidelity of the reconstruction.

After supplementing the training data set with the new interpolated values, direct nonparametric prediction algorithms can be applied with the augmented data set. As previously described, direct prediction begins by locating the neighbors  $[y(t'), y(t' - h), \dots, y(t' - dh), x(t'), x(t' - h), \dots, x(t' - dh), z(t'), z(t' - h), \dots, z(t' - dh)]$  of the current delay coordinates  $[y(t), y(t - h), \dots, y(t - dh), x(t), x(t - h), \dots, x(t - dh), z(t), z(t - h), \dots, z(t - dh)]$  where  $d$  is the number of delays and  $h$  is the sampling step size. Since the unmodeled  $x$  and  $z$  variables are observed, their delay coordinates can be included to enhance the identification of appropriate neighbors. The known  $y(t' + T)$ ,  $x(t' + T)$ , and  $z(t' + T)$  values are then used with a local model (for example locally constant, which is just an average of the neighbors) to predict  $y(t + T)$ ,  $x(t + T)$ , and  $z(t + T)$ . Note that both the standard nonparametric method and the proposed partial model method use direct prediction. The difference is that while the nonparametric forecast relies on the original training set at sample step size  $h$  to find the nearest neighbors, the partial model method has access to a finer

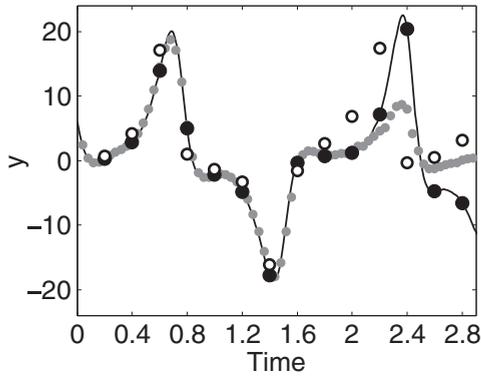


FIG. 2. Forecasting an individual trajectory of the Lorenz-63  $y$  variable. Using the equation to supplement the forecast (gray circles) is more accurate than the direct forecast method alone (open circles). The black curve is the true  $y$  trajectory and the black circles on the black curve denote the sampling rate ( $h = 0.2$ ).

resolution training set at sample rate  $h/m$ . Effectively, we have leveraged the partial knowledge of the system to build a more robust training set for finding better neighbors during direct prediction. Additionally, use of this partial knowledge allows us to predict at arbitrary step size  $h/m$  allowing for finer prediction resolution. This combination of parametric and nonparametric methods results in a hybrid method which improves the predictability of the system compared with nonparametric methods alone.

Figure 2 shows an example of this improvement. Time series data of length 2000 is known up to time  $t = 0$ ; the gray curve is the prediction going forward by the partial model method, compared with the exact trajectory (black curve). The black circles represent the time series sampling rate from the known data, and the open circles denote the nonparametric (direct forecasting) predictions without use of the partial model.

Figures 3(a)–3(c) compare mean forecasting error for the partial model method and the nonparametric method under noiseless conditions. Both methods used eight delays with a time delay of 0.2 units and ten nearest neighbors in constructing a locally constant forecast model. The partial model method consistently outperforms the nonparametric method at all short-term forecasting horizons. Note a side benefit of the method: Even though the sampling rate of the data is sparse, predictions can be made not only at the original sampling rate but at any desired intermediate time horizon. Even under a moderate amount of observational noise [Fig. 3(d)] the partial model method offers an improved short-term forecast of the  $y$  variable when compared to the nonparametric approach.

For very small sampling rates, the advantage of the partial model approach decreases. For example, when the sampling rate is reduced to  $h = 0.04$ , the partial model approach has prediction error RMSE equal to 1.00, compared with 1.34 for the nonparametric method, for prediction horizon  $T = 0.2$ . These errors are averaged over 1000 realizations.

Now that we have shown a simple example in detail, we describe the method in more general terms. In the general network setting, assume  $x_1, \dots, x_p$  are generic observable nodes

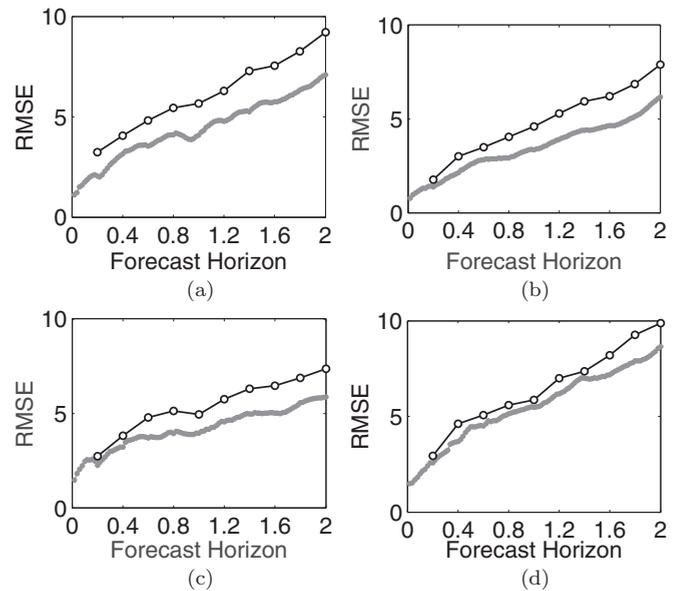


FIG. 3. Forecasting error statistics versus prediction horizon  $T$  for the Lorenz-63 (a)  $y$ , (b)  $x$ , and (c)  $z$  trajectories over 5000 realizations (standard error is less than 0.15 at all forecast horizons). The data set consists of 2000 points sampled at  $h = 0.2$ . Using our knowledge of the  $y$  equation, we can interpolate the training data at subsample step size  $k = 0.02$  and supplement the original training set. Direct forecasting using this interpolated data set (lower trace) outperforms direct forecasting with access to only the original data set (upper trace). Note the finer forecasting resolution of the lower trace. (d) Even under a moderate amount of observational noise (8% of the standard deviation), our hybrid forecasting offers better short-term prediction of the  $y$  variable than the nonparametric forecast.

in the sense of Takens [1,3], and nodes  $x_1, \dots, x_p, \dots, x_r$ , where  $r > p$ , form the complete input set to  $x_1, \dots, x_p$ . Assume further that  $x_1, \dots, x_r$  are observed in the training data at step size  $h$  and that we know the evolution equation of  $x_i$  for  $1 \leq i \leq p$ , say  $\dot{x}_i = f_i(x_1, \dots, x_r)$ . As above, we apply a multistep method to the known equations to upsample the values of  $x_1, \dots, x_p$ , and use a nonparametric method with the delay vectors of  $x_1, \dots, x_p$  to approximate the remaining unmodeled variables at the same time points, resulting in an interpolation of all  $r$  variables at the smaller time step  $k$ . This augmented training set is then used along with a nonparametric prediction algorithm to obtain forecasts for all  $r$  variables.

As an illustrative example we consider a network of  $p + 1$  nodes, comprised of a central Lorenz-63 attractor driving a coupled ring of  $p$  Lorenz-96 nodes [36]. We assume that we know the differential equation of the ring nodes

$$\dot{x}_i = (x_{i+1} - x_{i-2})x_{i-1} - c_i x_i + F_i + b_i x_{p+1}^1 \quad (3)$$

for  $i = 1, \dots, p$ , and lack knowledge of the evolution equations for all three Lorenz-63 variables  $x_{p+1}^{\{1,2,3\}}$ . In other words, we are driving the Lorenz-96 ring with the  $x$  coordinate of the classical Lorenz attractor, while assuming that we know the Lorenz-96 equations but not the classical Lorenz equations. We form the training set from observations of variables  $x_1, \dots, x_p, x_{p+1}^1$  [not including the  $y$  and  $z$  variables of the

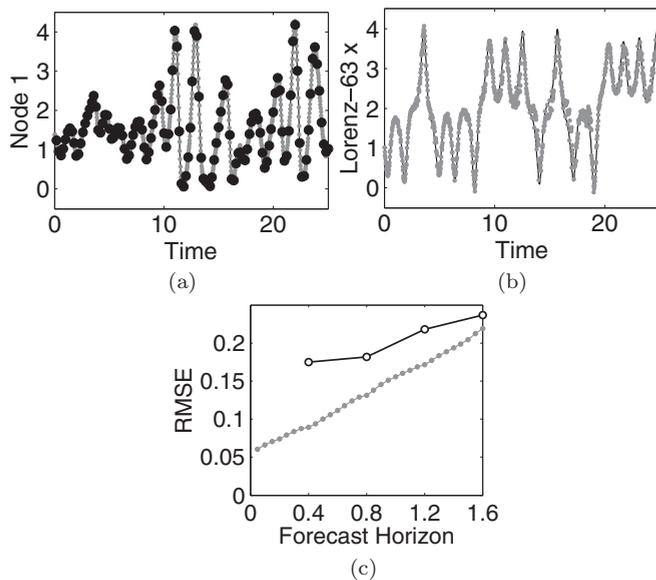


FIG. 4. Improved short-term forecasting of a small network. (a) Example interpolation of one node from the four node chaotic network driven by Lorenz-63. The data set (black circles) is used to interpolate Lorenz-96 (gray circles). (b) At each step of the interpolation, delay coordinates reconstruct the unmodeled Lorenz-63 forcing term (gray circles). (c) Mean forecasting error of the chaotic network over 300 realizations. Standard error is less than 0.01 at all forecast horizons. (Fewer realizations were necessary compared with Fig. 3 due to smaller variation in the dynamics.) Direct forecasting using this interpolated data set (lower trace) outperforms direct forecasting with access to only the original data set (upper trace).

classical Lorenz, since these variables do not occur in (3)] and attempt to forecast the Lorenz-96 system.

Figure 4 shows results from a Lorenz-96 ring consisting of  $p = 4$  nodes with  $F_i = 2$  and  $b_i = 1/8$  for  $i = 1, \dots, 4$  and  $[c_1, c_2, c_3, c_4] = [1, 1.2, 1.4, 1.6]$ . We collect 4000 data points

from each node of the network, sampled at  $h = 0.4$ . For this example, the prediction was done using six delays with a time delay of 0.4 units and 20 nearest neighbors to construct a locally constant forecast. Without knowledge of Eq. (3), we can use nonparametric methods to forecast [upper trace in Fig. 4(c)]. Assuming knowledge of Eq. (3), we can interpolate the training data from each node [Fig. 4(a)] and simultaneously reconstruct the Lorenz-63 forcing term [Fig. 4(b)] at step size  $k = 0.05$ . By augmenting the training data with these interpolated values, the hybrid prediction [lower trace in Fig. 4(c)] offers an improvement in short-term prediction accuracy compared to the nonparametric method.

As a hybrid method, this forecasting approach attempts to explore the middle ground between data-driven statistics, where no model information is known, and the parametric case with complete knowledge of the model. We have tried to place the idea in as general a setting as possible, but it is clear that many further adaptations can be formulated, depending on the different sampling rates of distinct variables, and varying levels of confidence in the separate model equations. The fractional-step integrators discussed here are the foundation of the idea, and generalizations to partial differential equation models may also be feasible and effective.

The limitations of this new hybrid prediction method are similar to those faced by nonparametric approaches: As the effective dimension of the dynamics increases, even short-term prediction becomes more difficult. The role of noise is also important. We have emphasized the noiseless case to more clearly explain the method. For data with observational or dynamical noise, the partial model assumption that a differential equation is satisfied no longer holds exactly. In such cases the method should be generalized to include noise in the underlying model. An optimal approach for this case is left for future work.

The research was partially supported by Grants No. CMMI-1300007, No. DMS-1216568, and No. DMS-1250936 from the National Science Foundation.

- [1] F. Takens, *Detecting Strange Attractors in Turbulence*, Lecture Notes in Mathematics Vol. 898 (Springer-Verlag, Berlin, 1981), p. 366.
- [2] N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw, *Phys. Rev. Lett.* **45**, 712 (1980).
- [3] T. Sauer, J. Yorke, and M. Casdagli, *J. Stat. Phys.* **65**, 579 (1991).
- [4] T. Sauer, *Phys. Rev. Lett.* **93**, 198701 (2004).
- [5] E. Ott, T. Sauer, and J. A. Yorke, *Coping with Chaos: Analysis of Chaotic Data and the Exploitation of Chaotic Systems* (Wiley, New York, 1994).
- [6] H. Abarbanel, *Analysis of Observed Chaotic Data* (Springer-Verlag, New York, 1996).
- [7] H. Kantz and T. Schreiber, *Nonlinear Time Series Analysis* (Cambridge University Press, Cambridge, 2004).
- [8] J. D. Farmer and J. J. Sidorowich, *Phys. Rev. Lett.* **59**, 845 (1987).
- [9] M. Casdagli, *Phys. D (Amsterdam, Neth.)* **35**, 335 (1989).
- [10] G. Sugihara and R. M. May, *Nature (London)* **344**, 734 (1990).
- [11] L. A. Smith, *Phys. D (Amsterdam, Neth.)* **58**, 50 (1992).
- [12] J. Jiménez, J. A. Moreno, and G. J. Ruggeri, *Phys. Rev. A* **45**, 3553 (1992).
- [13] T. Sauer, in *Time Series Prediction: Forecasting the Future and Understanding the Past* (Addison Wesley, Reading, MA, 1994), pp. 175–193.
- [14] G. Sugihara, *Philos. Trans. R. Soc., A* **348**, 477 (1994).
- [15] C. G. Schroer, T. Sauer, E. Ott, and J. A. Yorke, *Phys. Rev. Lett.* **80**, 1410 (1998).
- [16] D. Kugiumtzis, O. Lingjærde, and N. Christophersen, *Phys. D (Amsterdam, Neth.)* **112**, 344 (1998).
- [17] G. Yuan, M. Lozier, L. Pratt, C. Jones, and K. Helfrich, *J. Geophys. Res.* **109**, C08002 (2004).
- [18] C.-H. Hsieh, S. M. Glaser, A. J. Lucas, and G. Sugihara, *Nature (London)* **435**, 336 (2005).
- [19] C. C. Strelhoff and A. W. Hübler, *Phys. Rev. Lett.* **96**, 044101 (2006).
- [20] S. Regonda, B. Rajagopalan, U. Lall, M. Clark, and Y.-I. Moon, *Nonlin. Processes Geophys.* **12**, 397 (2005).

- [21] B. Schelter, M. Winterhalder, and J. Timmer, *Handbook of Time Series Analysis: Recent Theoretical Developments and Applications* (Wiley, New York, 2006).
- [22] E. Kalnay, *Atmospheric Modeling, Data Assimilation, and Predictability* (Cambridge University Press, Cambridge, 2003).
- [23] G. Evensen, *Data Assimilation: The Ensemble Kalman Filter* (Springer, Heidelberg, 2009).
- [24] S. Julier, J. Uhlmann, and H. Durrant-Whyte, *IEEE Trans. Autom. Control* **45**, 477 (2000).
- [25] S. Julier, J. Uhlmann, and H. Durrant-Whyte, *Proc. IEEE* **92**, 401 (2004).
- [26] R. H. Reichle and R. D. Koster, *Geophys. Res. Lett.* **32**, L02404 (2005).
- [27] H. Hersbach, A. Stoffelen, and S. De Haan, *J. Geophys. Res.: Oceans* (1978–2012) **112**, C03006 (2007).
- [28] H. Arnold, I. Moroz, and T. Palmer, *Philos. Trans. R. Soc. A* **371**, 20110479 (2013).
- [29] T. Berry and J. Harlim, *Proc. R. Soc. A* **470**, 20140168 (2014).
- [30] E. N. Lorenz and K. A. Emanuel, *J. Atmos. Sci.* **55**, 399 (1998).
- [31] T. N. Palmer, *Q. J. R. Meteorol. Soc.* **127**, 279 (2001).
- [32] Z. Qu, J. N. Weiss, and A. Garfinkel, *Phys. Rev. E* **61**, 727 (2000).
- [33] V. S. Zykov and K. Showalter, *Phys. Rev. Lett.* **94**, 068302 (2005).
- [34] E. Lorenz, *J. Atmos. Sci.* **20**, 130 (1963).
- [35] See Supplemental Material at <http://link.aps.org/supplemental/10.1103/PhysRevE.92.010902> for description of the fractional multistep methods used.
- [36] E. N. Lorenz, *Proceedings: Seminar on Predictability* (AMS, Reading, UK, 1996), Vol. 1, pp. 1–18.