Detecting chaos in heavy-noise environments

Wen-wen Tung,[1] Jianbo Gao,[2,*] Jing Hu,[3] and Lei Yang[4]
[1]*Department of Earth & Atmospheric Sciences, Purdue University, West Lafayette, Indiana 47907, USA*
[2]*PMB Intelligence LLC, PO Box 2077, West Lafayette, Indiana 47996, USA*
[3]*Affymetrix, Inc., 3380 Central Expressway, Santa Clara, California 95051, USA*
[4]*Department of Electrical and Computer Engineering, University of Florida, Gainesville, Florida 32611, USA*
(Received 22 December 2010; revised manuscript received 21 February 2011; published 18 April 2011)

Detecting chaos and estimating the limit of prediction time in heavy-noise environments is an important and challenging task in many areas of science and engineering. An important first step toward this goal is to reduce noise in the signals. Two major types of methods for reducing noise in chaotic signals are chaos-based approaches and wavelet shrinkage. When noise is strong, chaos-based approaches are not very effective, due to failure to accurately approximate the local chaotic dynamics. Here, we propose a nonlinear adaptive algorithm to recover continuous-time chaotic signals in heavy-noise environments. We show that it is more effective than both chaos-based approaches and wavelet shrinkage. Furthermore, we apply our algorithm to study two important issues in geophysics. One is whether chaos exists in river flow dynamics. The other is the limit of prediction time for the Madden-Julian oscillation (MJO), which is one of the most dominant modes of low-frequency variability in the tropical troposphere and affects a wide range of weather and climate systems. Using the adaptive filter, we show that river flow dynamics can indeed be chaotic. We also show that the MJO is weakly chaotic with the prediction time around 50 days, which is considerably longer than the prediction times determined by other approaches.

DOI: 10.1103/PhysRevE.83.046210 PACS number(s): 05.45.Ac, 93.30.Vs, 05.40.Ca

I. INTRODUCTION

Detecting chaos and understanding the limit of prediction time based on experimental data analyses is an important task in many areas of science and engineering. It is often challenging, since experimental data can be very noisy. To make the task amenable, an important step is to first reduce noise in the signals. When the signals are linear, this is a simple task since we have a variety of linear filters to choose from to clean up the data. When the signals are nonlinear, and especially chaotic, then the problem becomes highly nontrivial since linear filters severely distort even clean chaotic signals [1], let alone effectively reduce noise from them. Thus, one must resort to either wavelet-based denoising techniques [2–4] or chaos-based approaches for such a purpose. Over the past two decades, there have been a number of important algorithms developed along the latter line [5–10]. While details of those methods vary, they share a common fundamental element: approximate the local chaotic dynamics in a neighborhood (of size ϵ) of a reference point. When noise is weak, ϵ can be chosen small, and the approximation to the local dynamics can be quite accurate. When noise is strong, however, ϵ has to be large and the local chaotic dynamics cannot be accurately estimated. Consequentially, noise cannot be effectively removed. To better detect chaos in heavy-noise environments, one has to go beyond chaos-based approaches for reducing noise. In this work, we propose a nonlinear adaptive denoising algorithm and show that it recovers chaotic signals in heavy-noise environments better than chaos-based and wavelet-based approaches.

In the remainder of the paper, we first describe our adaptive denoising algorithm, then compare its effectiveness in reducing noise with wavelet- and chaos-based approaches, using the chaotic Lorenz data superimposed by strong Gaussian noise as an example. We then apply our algorithm to detect chaos in river flow dynamics and the Madden-Julian oscillation (MJO), which is one of the most dominant modes of low-frequency variability in the tropical troposphere and affects a wide range of weather and climate systems.

II. ADAPTIVE DENOISING ALGORITHM

Our nonlinear adaptive denoising algorithm works as follows. First, it partitions a time series into segments (or windows) of length $w = 2n + 1$ points, where neighboring segments overlap by $n + 1$ points. Thus, the time scale introduced by the algorithm is $n + 1$ sample points. For each segment, we fit a best polynomial of order K. Note that $K = 0$ and 1 correspond to the piecewise constant and linear fitting, respectively. The fitted polynomials for the ith and $(i + 1)$th segments are denoted by $y^i(l_1)$, $y^{i+1}(l_2)$, respectively, where $l_1, l_2 = 1, \ldots, 2n + 1$. Note the length of the last segment may be smaller than $2n + 1$. We define the fitting for the overlapped region as

$$y^{(c)}(l) = w_1 y^i(l + n) + w_2 y^{i+1}(l), \quad l = 1, 2, \ldots, n + 1 \qquad (1)$$

where $w_1 = [1 - (l - 1)/n]$, and $w_2 = (l - 1)/n$ can be written as $(1 - d_j/n)$, $j = 1, 2$, where d_j denotes the distances between the point and the centers of y^i and y^{i+1}, respectively. This means the weights decrease linearly with the distance between the point and the center of the segment. Such a weighting ensures symmetry and effectively eliminates any jumps or discontinuities around the boundaries of neighboring segments. In fact, the scheme ensures that the fitting is continuous everywhere, is smooth at the nonboundary points, and has the right (left) derivatives at the boundary points.

*jbgao@pmbintelligence.com

To implement the algorithm, let us consider a curve defined by $(x_i, y_i), i = 1, \ldots, w = 2n + 1$. Let its least-squares polynomial fit be $(x_i, g_i), i = 1, \ldots, w$, where the polynomial is

$$g(x) = b_0 + b_1 x + b_2 x^2 + \cdots + b_K x^K, \tag{2}$$

where K is the polynomial order. Assuming a constant sampling interval δx, then we can write each x_i as

$$x_i = i\delta x + \alpha.$$

We can always choose α to make $i = -n, -n+1, \ldots, 0, 1, 2, \ldots, n$. Then, expanding x^j, $j = 1, \ldots, K$, and rearranging the terms, we can rewrite the polynomial as

$$g(i) = a_0 + a_1 i + a_2 i^2 + \cdots + a_K i^K. \tag{3}$$

So, the problem becomes to determine the vector $(a_0, \ldots, a_K)$ such that

$$\sum_{i=-n}^{n} [g(i) - y_i]^2$$

is minimized. The solution is given by [11]

$$(A^T A)a = A^T Y \quad \text{or} \quad a = (A^T A)^{-1}(A^T Y), \tag{4}$$

where A is the Vandermonde matrix with elements

$$\begin{gathered} A_{ij} = i^j, \quad i = -n, -n+1, \ldots, 0, 1, 2, \ldots, n, \\ j = 0, \ldots, K, \quad Y = (y_1, \ldots, y_w)^T. \end{gathered} \tag{5}$$

To summarize, we can first calculate $[(A^T A)^{-1} A^T]$ and save it. Then, for each Y, we can easily get a using Eq. (4), and get the desired polynomial $g(i)$ using Eq. (3). In turn, when we slide the window it gives y^{i+1} and y^i used in Eq. (1). Applying Eq. (1), we then obtain the overall trend.

Three comments are in order. (i) The method contains two free parameters, K, the order of the polynomial, and $w = 2n + 1$, the segment (or window) length. By properly choosing K and making w small enough, the fitting can be perfect, in the sense that the difference between the data and the fitting (which may be called the residual data) can be exactly zero. This property ensures that clean chaotic signals may not be distorted at all when the adaptive filter with appropriate parameters is applied. (ii) The smoothness of the method dictates that it may not be used to process data generated by discrete maps. On the other hand, it implies that it becomes more effective when the sampling rate increases. The latter property is quite advantageous, since the sampling rate is often chosen much higher than the Nyquist rate. In fact, after denoising one may downsample to save space, if file size is an issue. (iii) When the overall fitted curve is treated as the desired signal, then the filter is lowpass. When the residual data is treated as the desired signal while the overall fitted curve is the background or trend signal, then the filter is highpass. If one chooses two window sizes $w_1 < w_2$, and treats the difference between the two overall fitted curves as the desired signal, then the adaptive filter is bandpass.

How do we choose the parameters K and w when denoising an experimental time series, where a true clean signal is unknown? One key is to realize that w cannot be greater than $1/2$ of a local period of the variation of the signal, if one does want to trace out the detailed variations of the signal. Another important observation is that when the signal is highly nonlinear, then $K \geqslant 2$. However, a large K may not be recommended since polynomials may not be well defined when w is small. After realizing these observations, one can use the following rule to determine K and w. First, fix K to be 2, then check how the variance of the residual data varies with w. A generic pattern of the variation would be that variance (i) increases with w when w is small, (ii) flattens out when w keeps increasing, and (iii) increases sharply again when w is further increased. Behavior (i) corresponds to almost perfect fitting, and thus little denoising. Behavior (iii) indicates that the window size is too big such that local variations of the signal have been largely removed. Therefore, the desired w should be in the range of w corresponding to behavior (ii). Such an analysis can be repeated when K is gradually increased. Further increasing K becomes unnecessary when the variation of the variance of the residual data changes little. Our experience is that usually $K = 2$ or $K = 3$ is already sufficient. This procedure will be made more clear when we discuss how to remove noise from the chaotic Lorenz data.

To illustrate the use of the adaptive filter, we consider the Lorenz system

$$\begin{aligned} dx/dt &= -\sigma(x - y), \\ dy/dt &= -xz + \gamma x - y, \\ dz/dt &= xy - bz, \end{aligned} \tag{6}$$

with $\sigma = 10$, $\gamma = 28$, and $b = 8/3$. The system is solved using a fourth-order Runge-Kutta method. The x component is recorded with a sampling time of 0.01. To facilitate computation of invariant measures, including the Lyapunov exponent and fractal dimension, we have recorded a time series of length 6×10^4 points. Gaussian white noise of zero mean is added to the data such that the signal-to-noise ratio (SNR) of the resulting data is 13.89 dB, where SNR is defined as ten times the logarithm of the ratio of the signal and noise variance, $\text{SNR} = 10 \log_{10} \sigma_x^2/\sigma_n^2$. The phase diagrams of the clean and noisy data are plotted in Figs. 1(a) and 1(b), respectively. Obviously, noise is very strong.

We have denoised the data using the adaptive filter. Following the general rule for determining the parameters K and w, we have computed the variance of the residual data versus the window size w with $K = 2$, 3, and 4. The results are shown in Fig. 2 where, for convenience, the variance of the residual data is normalized by the variance of the original noisy data. For $K = 2$ and 3, we indeed observe the three behaviors of variation discussed earlier. In particular, we observe that behavior (ii) is better defined with $K = 3$. When K is increased to 4, behavior (ii) is no longer improved, while behavior (i) is degraded. Therefore, we have chosen $K = 3$. It turns out that a range of w, from 25 to 35, all yield good denoising results, in fact, this is the underlying reason that dyadic wavelet decomposition can effectively remove noise, as we will see shortly. The phase diagram with $K = 3$, $w = 33$ is shown in Fig. 1(e). Clearly, noise has been removed significantly.

To appreciate the effectiveness and limitations of the adaptive algorithm, we first compare its denoising performance

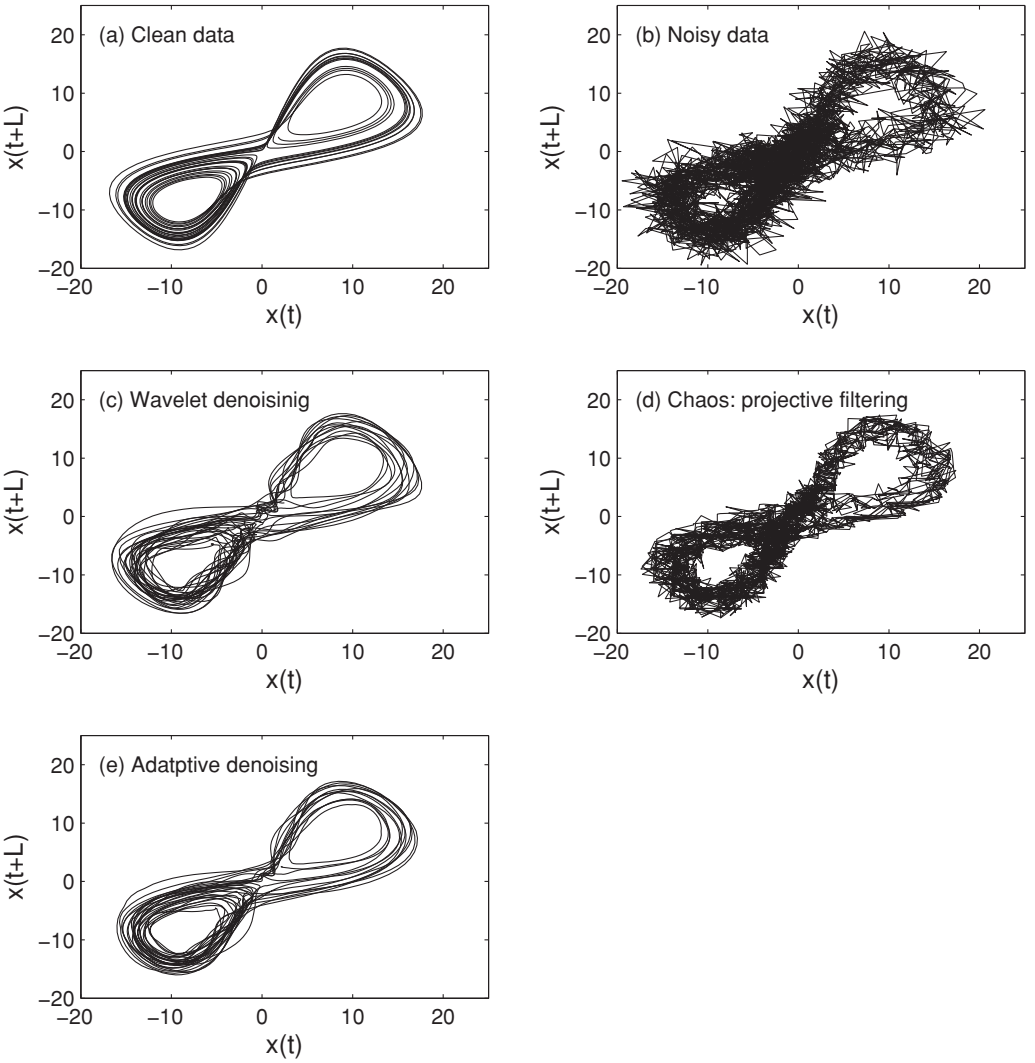

FIG. 1. Phase diagrams (with $L = 12$) for (a) the clean Lorenz signal, (b) the noisy Lorenz signal, (c) the signal filtered by wavelet denoising, (d) the signal processed by chaos-based projective filtering, and (e) the signal filtered by the proposed adaptive algorithm. The sampling time is 0.01.

with chaos-based approaches. It is generally thought that most chaos-based denoising algorithms perform similarly [5]. Here, we compare our adaptive denoising with the fast nonlinear projective filtering developed by Schreiber and Richter [12], as it has some additional features that are not shared by other chaos-based algorithms, such as real-time filtering of a data stream, causality, and capability of dealing with nonstationarity. It works as follows. Given a time series $x(1),x(2),\ldots$, one first constructs vectors

$$X_i = \{x(i),x(i+L),\ldots,x[i+(m-1)L]\}, \tag{7}$$

where m and L are the embedding dimension and the delay time, respectively. Then, for an arbitrary vector X_n in the reconstructed phase space, one finds all the past points $X_{n'}$ that are within a distance ϵ of X_n and are observed no longer than Δn time steps ago. These are denoted by

$$U^n_{\Delta n} = (X_{n'} : n - \Delta n \leqslant n' < n, \|X_{n'} - X_n\| < \epsilon). \tag{8}$$

The second major step of the algorithm is to estimate the local structure of the dynamics around X_n by local principal components. In the third step, one projects $X_n - \overline{X}^n$, where $\overline{X}^n$ is basically the center of mass of $U^n_{\Delta n}$ with some correction, onto the reconstructed Q-dimensional manifold and obtains the project vector $\hat{X}_n$, and then extracts the desired component.

Using the code in the Tisean software package [13], we have denoised the noisy Lorenz data by projective filtering. The resulting phase diagram is shown in Fig. 1(d). While some noise has indeed been filtered out, it is clear that most of the noise has remained. The reason is, as we have pointed out, the noise is so strong that the neighborhood size has to be chosen very big. Consequentially, most of noise remains.

Next, we compare our algorithm with wavelet shrinkage. The basic idea is to take the wavelet transform of the noisy data, threshold the resulting wavelet coefficients, and, finally, take the inverse wavelet transform to obtain an estimation of the

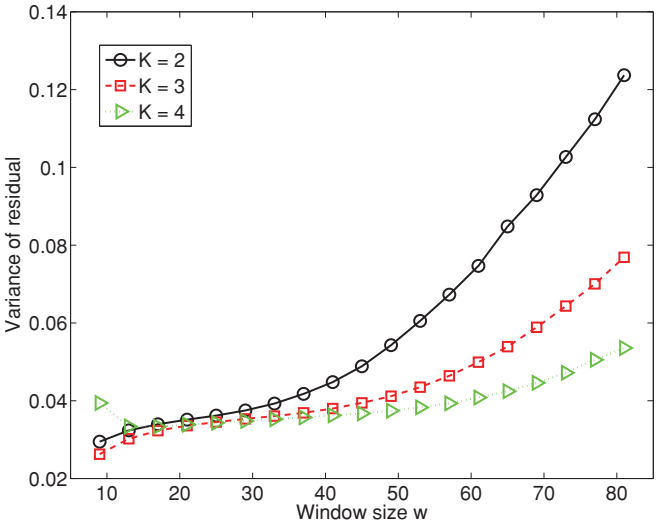

FIG. 2. (Color online) Normalized variance of residual data vs. window size w for different polynomial order K for the noisy Lorenz data.

signal. For details, we refer to [2–4,14]. In particular, three wavelet thresholding algorithms of [3] are examined here. For convenience, they are denoted wavelet-1, wavelet-2, and wavelet-3, respectively. We have found that the three wavelet thresholding algorithms barely differ when removing noise from the Lorenz data (illustrated in Fig. 3 below). The resulting phase diagram is shown in Fig. 1(c). We observe that it is much better than chaos-based projective filtering, as shown in Fig. 1(d). This may explain why wavelet denoising has been so widely used. Nevertheless, the phase diagram resulting from adaptive filtering shown in Fig. 1(e) is better than Fig. 1(c), indicating that the adaptive algorithm is even more effective than the wavelet-based approaches.

To better quantify the difference between the three types of filtering techniques, we have computed the root-mean-square error (RMSE) defined by

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}[s(i) - \hat{s}(i)]^2}, \tag{9}$$

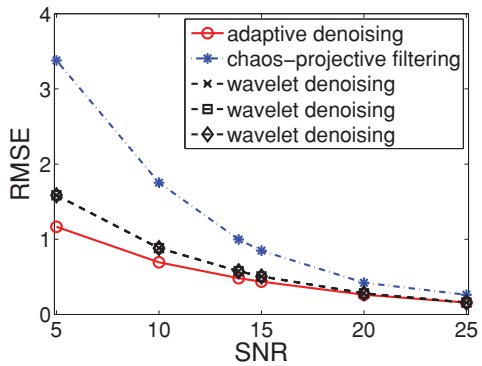

FIG. 3. (Color online) RMSE vs. SNR curves for the three types of filters examined here. In particular, three types of wavelet thresholding [3] have been examined.

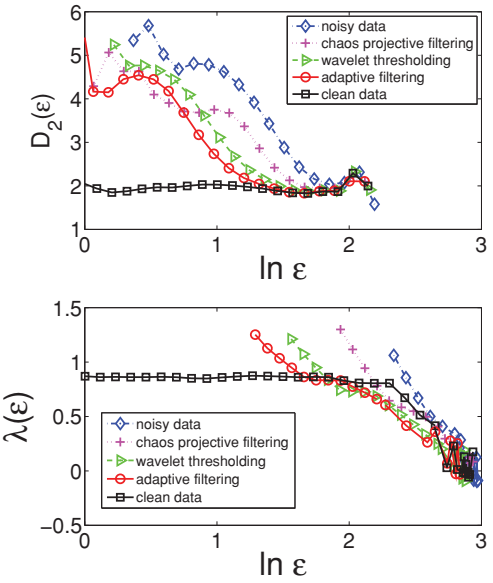

FIG. 4. (Color online) (a) $D_2(\epsilon)$ vs. $\ln\epsilon$ curves; (b) $\lambda(\epsilon)$ vs. $\ln\epsilon$ curves.

where $s(i)$ and $\hat{s}(i)$ are the clean and denoised data, respectively. The variations of the RMSE with SNR for these methods are shown in Fig. 3. Clearly, the adaptive algorithm is the best.

While the RMSE is a convenient metric to quantify the quality of denoising, for a chaotic signal it is important to examine how much chaotic signatures, including fractal dimension and Lyapunov exponents, are recovered after denoising. We first examine the fractal dimension of the Lorenz signal. It is a geometrical quantity characterizing the minimal number of variables that are needed to fully describe the dynamics of a motion. It is often estimated by the correlation dimension D_2 using the Grassberger-Procaccia's algorithm [15]

$$C(\epsilon) \sim \frac{1}{N'^2}\sum_{i,j=1}^{N'}\theta(\epsilon - \|X_i - X_j\|) \sim \epsilon^{D_2}, \tag{10}$$

where $C(\epsilon)$ is called the correlation integral, θ is the Heaviside step function, X_i and X_j are reconstructed vectors, $N' = N - (m-1)L$ is the number of points in the reconstructed phase space, and ϵ is a prescribed small distance. Figure 4(a) shows the variation of $D_2(\epsilon)$ $\ln\epsilon$ for the clean, noisy, and filtered Lorenz data. We observe that for the clean data there is a plateau with a value consistent with the correlation dimension of 2.02. For the noisy data, a plateau is hardly detectable. Projective filtering, while slightly recovering the plateau, is not very effective. The wavelet approach does better, however, it is still less effective than the adaptive algorithm.

Next, we consider the Lyapunov exponent (LE). It is often denoted by λ_1 and is a dynamical quantity. It characterizes the exponential growth of an infinitesimal line segment ϵ_0 (i.e., $\epsilon_t \sim \epsilon_0 e^{\lambda_1 t}, \epsilon_0 \to 0$). While the classical Wolf *et al.*'s algorithm [16] does a good job in estimating LE from clean, low-dimensional, chaotic signals, estimation of LE from noisy chaotic data is a different story: it even returns a positive value for random noise, thus interpreting random noise as chaos [17,18]. To better characterize dynamics of various

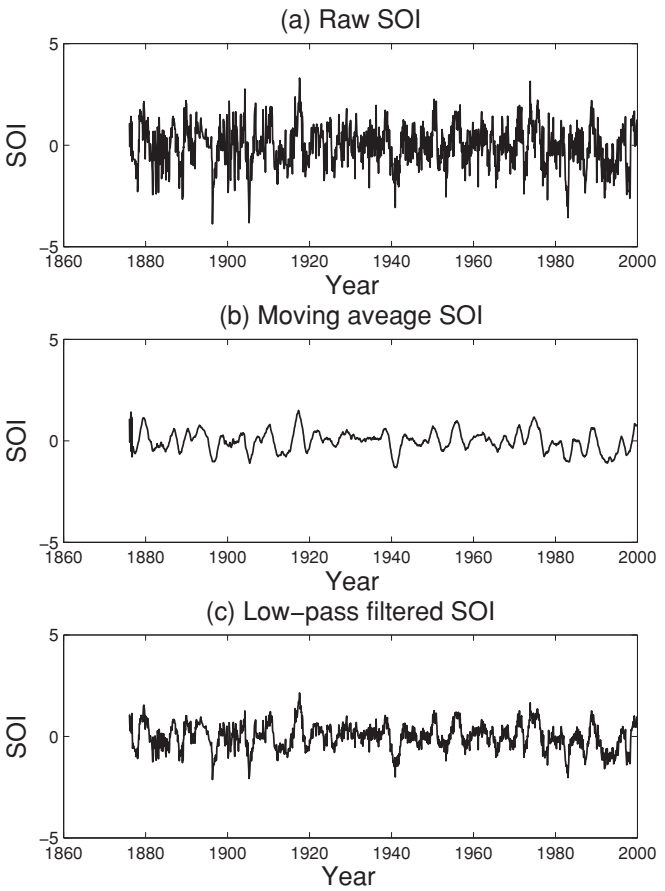

FIG. 5. Time series of (a) raw Southern Oscillation Index (SOI) and filtered data using (b) moving average, (c) low-pass filter. For the details of these linear filters, see Kawamura *et al.* [25].

kinds we have developed a multiscale complexity measure, the scale-dependent Lyapunov exponent (SDLE) [14,19–21]. It is defined in a phase space through consideration of an ensemble of trajectories. The initial distance between two nearby trajectories is denoted by ϵ_0, and their average distances at time t and $t + \Delta t$ are denoted, respectively, by ϵ_t and $\epsilon_{t+\Delta t}$, where Δt is small. The SDLE $\lambda(\epsilon_t)$ is defined by

$$\epsilon_{t+\Delta t} = \epsilon_t e^{\lambda(\epsilon_t)\Delta t} \text{ or } \lambda(\epsilon_t) = \frac{\ln \epsilon_{t+\Delta t} - \ln \epsilon_t}{\Delta t} \qquad (11)$$

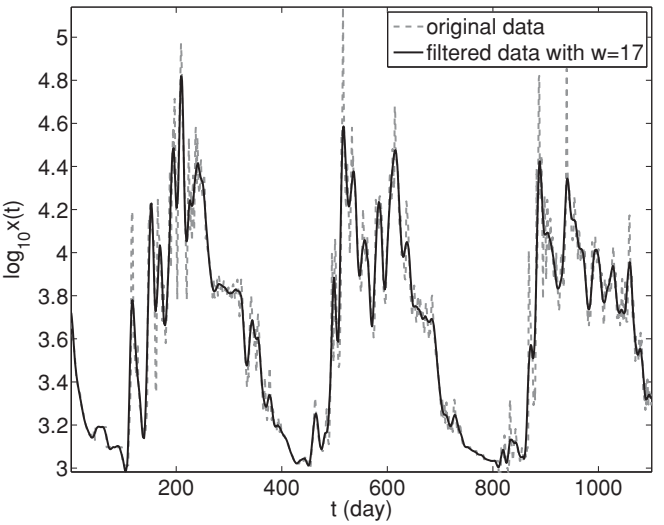

FIG. 6. A segment of the Umpqua River streamflow data.

or, equivalently, by

$$\frac{d\epsilon_t}{dt} = \lambda(\epsilon_t)\epsilon_t \text{ or } \frac{d \ln \epsilon_t}{dt} = \lambda(\epsilon_t). \qquad (12)$$

For computational details of the SDLE, we refer to Refs. [14, 19–21]. Note that for clean chaotic signals [14,19,20], $\lambda(\epsilon) \sim \lambda_1$ on small scales. Such plateaulike behavior is evident in Fig. 4(b). When there is noise, the plateau shrinks, and on the smallest scales $\lambda(\epsilon) \sim -\gamma \ln \epsilon$, where $\gamma > 0$ quantifies the speed of loss of information. From Fig. 4(b) we observe that the plateaus in Fig. 4(b) correspond to those in Fig. 4(a) very well, and the plateau is better recovered by the adaptive algorithm than by the chaos-based projective filtering or wavelet shrinkage.

To facilitate discussions below, we note that the SDLE can detect chaos in high-dimensional systems such as the Lorenz model with a dimension around 27 [22], and intermittent chaos [23]. Moreover, $1/\lambda(\epsilon)$ is closely related to the error doubling time [22,24], and

$$\epsilon_t = \epsilon_0 e^{\int_0^t \lambda(\epsilon_t)dt}$$

is the very error growth curve most commonly calculated in ensemble forecasting [22]. In particular, when ϵ_t reaches a steady state, we may say that the system is no longer

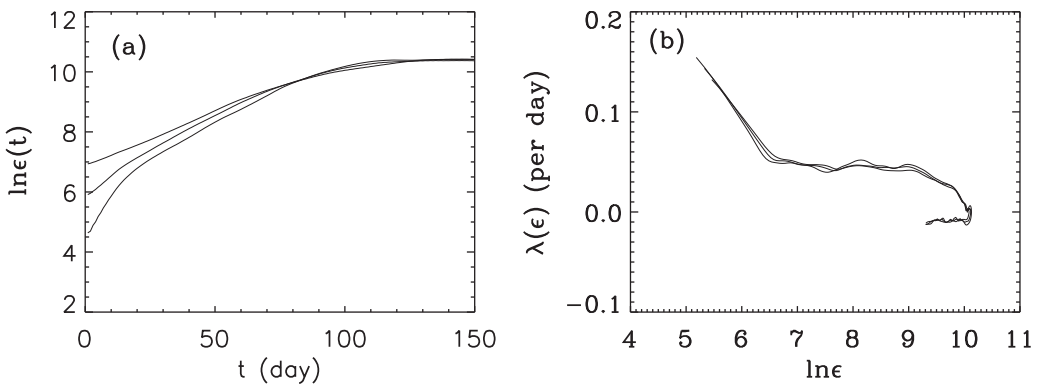

FIG. 7. (a) $\ln \epsilon$ vs. time and (b) $\lambda(\epsilon)$ vs. ϵ for the filtered data.

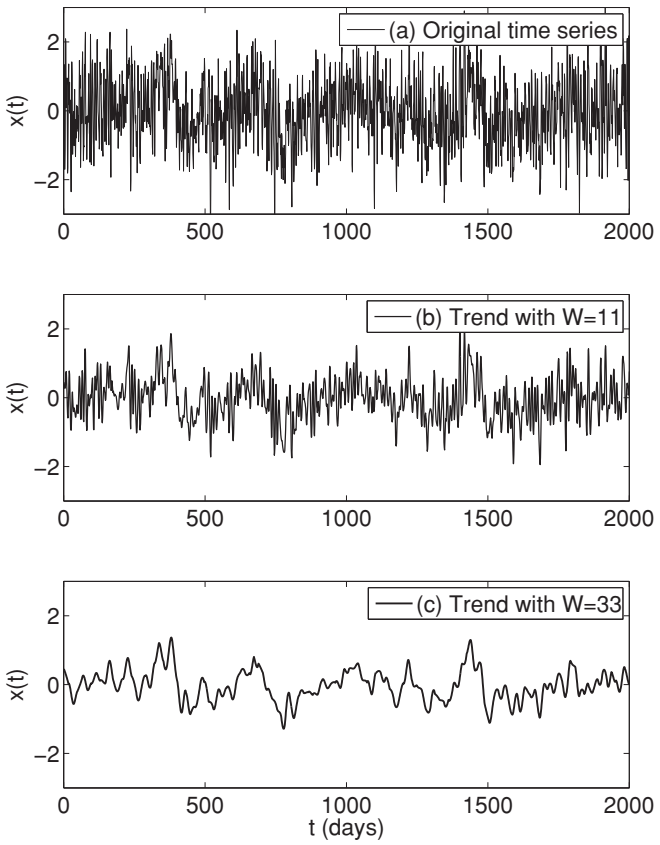

FIG. 8. Original and filtered MJO index data.

predictable. In other words, the time for ϵ_t to reach saturation is the prediction time.

III. GEOPHYSICAL APPLICATIONS

Geophysical data often are highly nonlinear and noisy. To determine whether geophysical processes are chaotic or not, considerable efforts have been made to reduce noise in observational geophysical data so that data are amenable for analysis using chaos theory. It is now generally thought (see, e.g., Refs. [25,26]) that linear filters remove significant parts of the nonlinear systems. Two examples are shown in Figs. 5(b) and 5(c). This, of course, is consistent with the fundamental observation that linear filters are not suitable for chaotic signals [1]. Chaos-based nonlinear filtering does not work either, as it generates many physically implausible straight-line segments [25,26]. In the following, we shall only focus on reducing noise from geophysical data using the proposed nonlinear adaptive filter and not take linear filters or other filters into consideration.

A. Detecting chaos in river flow dynamics

River flow dynamics are highly complicated, having heavy-tailed distributions [27], multifractal properties [28], and strong seasonal cycles. As an example, Fig. 6 (dashed line) shows the streamflow data of the Umpqua river, which is on the Pacific coast of Oregon in the United States. We observe that while the annual cycle is easily detectable, there are higher frequency variations.

So far, it has been debated whether river flow dynamics can be chaotic or not [29,30]. There have been efforts to reduce noise in hydrological time series using chaos-based noise reduction techniques [26]. However, the results are far from being acceptable [26]. We have tried our adaptive algorithm to reduce noise in the data shown in Fig. 6 using a second-order polynomial with a window size of 17 days. The filtered data is shown in the figure as the solid line. We then calculate error growth curves and the SDLE. The results are shown in Figs. 7(a) and 7(b), respectively. Clearly, we observe that for a considerable amount of time the error ϵ grows with time exponentially [Fig. 7(a)]. Consequently, a fairly large plateau region in the SDLE results, as is evident in Fig. 7(b). This suggests that river flow dynamics can indeed be chaotic.

B. Prediction time for the Madden-Julian Oscillation

As the second application, we consider the Madden-Julian oscillation [31], which is one of the dominant modes of low-frequency variability in the tropical troposphere [31,32]. The MJO affects a wide range of tropical weather and climate [33–37] as well as extratropical circulation [38–40]. Determining the limit of prediction of the MJO is not only important for

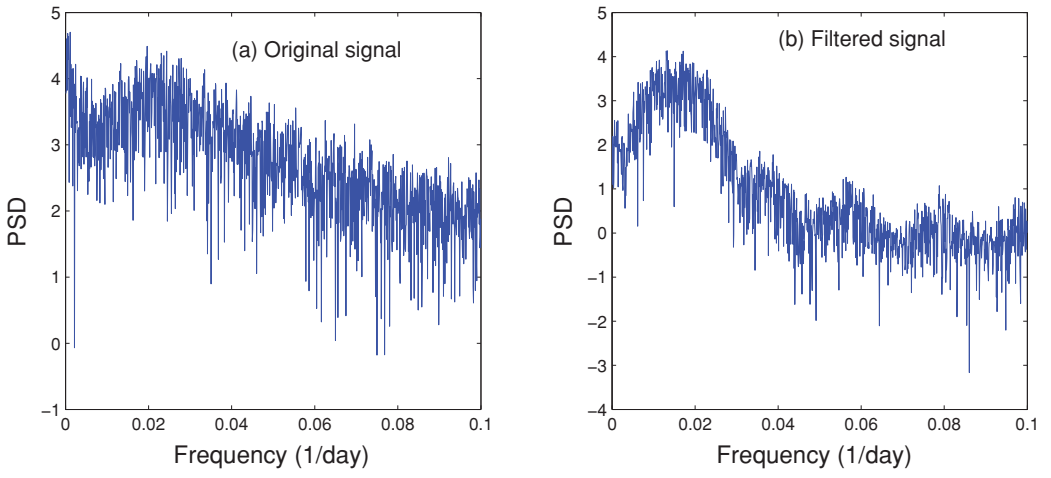

FIG. 9. (Color online) Power spectral density (PSD) for the original and filtered MJO index data.

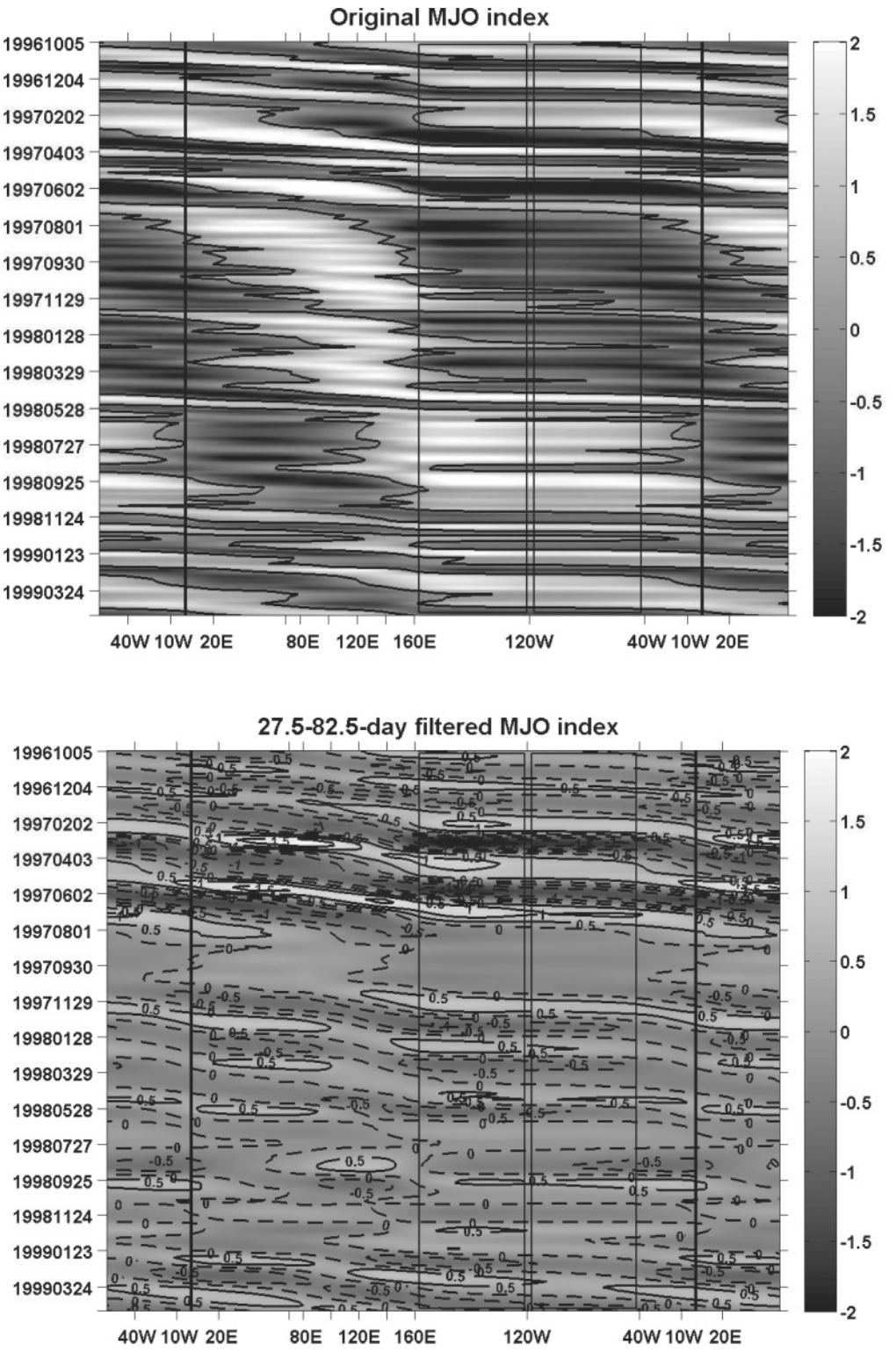

FIG. 10. Longitude-time plots for the original (top) and filtered (bottom) MJO index data showing propagation of wave motions.

the development of suitable models for the MJO but also may help understand many other weather and climate systems.

Recent estimates of the MJO predictability are $\sim 15-20$ and $\sim 25-30$ days, by statistical and dynamical models, respectively [41]. To better determine the limit of predictability of the MJO, there have been strong advocates for standardized observation-based diagnostics to evaluate objectively modeled simulations of the MJO [42]. Moreover, several empirical-orthogonal function-based (EOF-based) MJO indices have been developed for both operational and research purposes [43–46]. In particular, the NOAA CPC MJO indices [46], which are five-day mean indices based on the first extended

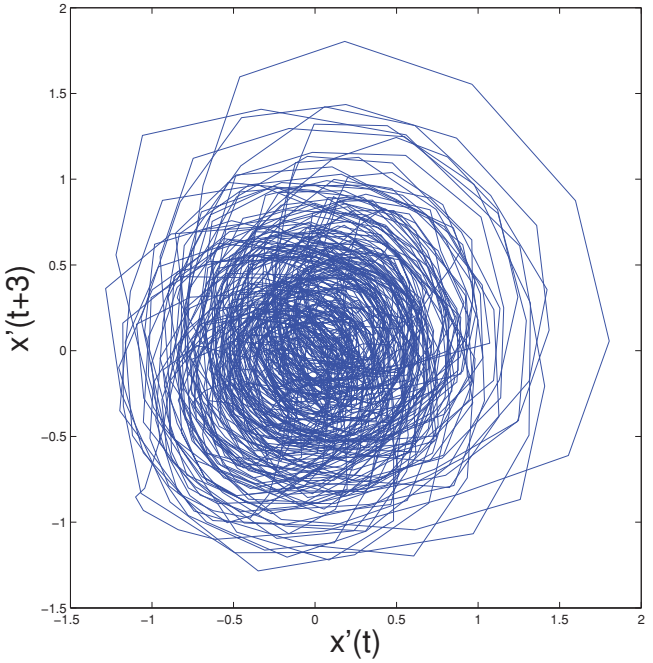

FIG. 11. (Color online) Phase diagram for the bandpassed MJO index data (denoted as x') at 80°E of the Equator.

EOF of 200-hPa velocity potential anomalies from equatorward of 30°N and have ten components around the tropical belt, have been used for real-time monitoring of the MJO. Those MJO index data are chosen for the present study for the purpose of determining the prediction time of the MJO.

For illustration purposes, Fig. 8(a) shows the MJO index data at longitude 80°E. Its power-spectral density (PSD) is shown in Fig. 9(a). We observe a broad frequency peak with its center slightly smaller than 0.02 (one per day). This translates to a period of about 60 days. This is the defining feature of the MJO. To further illustrate the propagation behavior of the MJO, Fig. 10(a) shows the time-longitude plot. We observe that the time interval between the major stripes is about 60 days, consistent with the frequency (or period) shown in Fig. 9(a). Figure 10(a) however, also shows many faster variations. They are caused by convective events in the tropical troposphere and are not part of the MJO. To determine the intrinsic predictability of the MJO, those features should be removed. Furthermore, seasonal effects may also incur bias in the estimation of the MJO prediction time, and thus should also be removed. These considerations motivate us to bandpass the index data with our adaptive filter parameters corresponding to 27.5–82.5 days and $K = 2$. The trend signals corresponding to 27.5 days and 82.5 days are shown in Figs. 8(b) and 8(c), respectively. The final bandpassed signal is thus the difference between them. Its PSD is shown in Fig. 9(b). We observe that now the spectral peaks become sharper. Moreover, we observe a number of harmonics of the basic MJO period of about 60 days. Such nice periodicities are also reflected as the circular structure in the phase diagram shown in Fig. 11. The time-longitude plot based on the bandpassed index data is shown in Fig. 10(b). Now we observe very well-behaved wave propagation phenomenon.

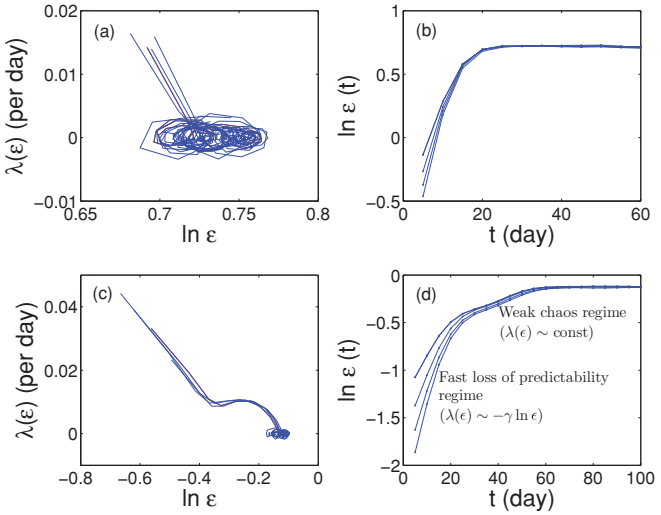

FIG. 12. (Color online) (a),(b) $\lambda(\epsilon)$ vs. ϵ and $\ln\epsilon$ vs. time curves for the MJO index data; (c), (d) $\lambda(\epsilon)$ vs. ϵ and $\ln\epsilon$ vs. time curves for the bandpass filtered MJO index data.

We are now ready to estimate the prediction time of the MJO. Figure 12(a) shows the SDLE for the raw MJO index data. Its integration, as pointed out earlier, is equivalent to the error growth curve [22] and is shown in Fig. 12(b). We only observe $\lambda(\epsilon) \sim -\gamma \ln\epsilon$ in Fig. 12(a). We also observe that the error growth curve saturates around $t \sim 20$ days. This is the limit of prediction time for the raw MJO index data, fully consistent with the 20-day prediction time found by model simulations [41]. After we bandpass the data, the SDLE curve [Fig. 12(c)] totally changes—the $\lambda(\epsilon) \sim -\gamma \ln\epsilon$ scaling has become clearer with a smaller γ, and, more importantly, a narrow plateau emerges with the value close to 0.01 per day (or, equivalently, 0.5 per cycle of the MJO). Correspondingly, we observe two types of behavior in the error growth curve in Fig. 12(d)—an initial rapid loss of predictability regime, with a time scale close to 20 days, followed by a weak chaos regime characterized by a well-defined exponential growth in the error curve with a time scale about 30 days. Note that Figs. 12(c) and 12(d) vary little along the Equator, indicating that the MJO is a very coherent event. We thus conclude that the MJO is weakly chaotic with a prediction time around 50 days.

Note that a similar effort of determining the predictability of the MJO has been made by Ding *et al.* [47], where the best linear filter, the 30–80-day bandpass Lanczos filter, is used to preprocess the data. The predictability reported there is about 35 days, considerably shorter than that found here.

IV. CONCLUDING DISCUSSIONS

To better detect chaos in heavy-noise environments and more accurately estimate the prediction time of dynamical systems, we have proposed a nonlinear adaptive denoising algorithm for preprocessing noisy data. Using heavily contaminated Lorenz data, we have shown that it is more effective in removing noise than chaos-based projective filtering and wavelet shrinkage. We have further applied the nonlinear

adaptive filter to study whether river flow dynamics can be chaotic, and the predictability of the MJO, one of the most important phenomena in atmospheric science. We have found that river flow dynamics can indeed be chaotic, and that the MJO is weakly chaotic with a prediction time close to 50 days, which is much longer than that found by other approaches.

Finally, we emphasize that although we considered the Lorenz data contaminated by Gaussian noise as an example, the adaptive filter does not need to assume noise to be Gaussian. This has been made evident by the effectiveness of our method in detecting chaos in river flow dynamics and determining the prediction time of the MJO—the noises there are obviously not simply Gaussian.

ACKNOWLEDGMENTS

This work was partially supported by National Science Foundation Division of Civil, Mechanical and Manufacturing Innovation Grants No. 1031958 and No. 0826119.

[1] R. Badii, G. Broggi, B. Derighetti, M. Ravani, S. Ciliberto, A. Politi, and M. A. Rubio, Phys. Rev. Lett. **60**, 979 (1988).
[2] D. L. Donoho and I. M. Johnstone, Biometrika **81**, 425 (1994).
[3] M. Han, Y. H. Liu, J. H. Xi, and W. Guo, IEEE Signal Proc. Lett. **14**, 62 (2007).
[4] J. B. Gao, H. Sultan, J. Hu, and W. W. Tung, IEEE Signal Proc. Lett. **17**, 237 (2010).
[5] H. Kantz and T. Schreiber, *Nonlinear Time Series Analysis*, 2nd ed. (Cambridge University Press, Cambridge, 2004).
[6] E. J. Kostelich and T. Schreiber, Phys. Rev. E **48**, 1752 (1993).
[7] P. Grassberger, R. Hegger, H. Kantz, C. Schaffrath, and T. Schreiber, Chaos **3**, 127 (1993).
[8] E. J. Kostelich and J. A. Yorke, Physica D **41**, 183 (1990).
[9] J. D. Farmer and J. J. Sidorowich, Physica D **47**, 373 (1991).
[10] S. M. Hammel, Phys. Lett. A **148**, 421 (1990).
[11] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in Fortran 77*, 2nd ed. (Cambridge University Press, Cambridge, 2006).
[12] T. Schreiber and M. Richter, Int. J. Bifurcat. Chaos **9**, 2039 (1999).
[13] [http://www.mpipks-dresden.mpg.de/tisean/]
[14] J. B. Gao, Y. H. Cao, W.-W. Tung, and J. Hu, *Multiscale Analysis of Complex Time Series: Integration of Chaos and Random Fractal Theory, and Beyond* (Wiley, New York, 2007).
[15] P. Grassberger and I. Procaccia, Phys. Rev. Lett. **50**, 346 (1983).
[16] A. Wolf, J. B. Swift, H. L. Swinney, and J. A. Vastano, Physica D **16**, 285 (1985).
[17] J. B. Gao, S. K. Hwang, and J. M. Liu, Phys. Rev. Lett. **82**, 1132 (1999); Int. J. Mod. Phys. B **13**, 3283 (1999).
[18] J. B. Gao, W. W. Tung, and N. Rao, Phys. Rev. Lett. **89**, 254101 (2002).
[19] J. B. Gao, J. Hu, W.-W. Tung, and Y. H. Cao, Phys. Rev. E **74**, 066204 (2006).
[20] J. Hu, J. B. Gao, and W. W. Tung, Chaos **19**, 028506 (2009).
[21] J. Hu, J. B. Gao, W. W. Tung, and Y. H. Cao, Ann. Biomed. Eng. **38**, 854 (2010).
[22] J. B. Gao, W. W. Tung, and J. Hu, Chinese Ann. Math B **30**, 569 (2009).
[23] J. B. Gao, J. Hu, W. W. Tung, and Y. Zheng, Quantitative Finance (in press).
[24] E. N. Lorenz, *Proceedings of Seminar on Predictability* (ECMWF, Reading, United Kingdom, 1996), Vol. 1, pp. 1–18.
[25] A. Kawamura, A. I. McKerchar, R. H. Spigel, and K. Jinno, J. Hydrol. **204**, 168 (1998).
[26] A. Elshorbagy, S. P. Simonovic, and U. S. Panu, J. Hydrol. **256**, 147 (2002).
[27] P. L. Anderson and M. M. Meerschaert, Water Resour. Res. **34**, 2271 (1998).
[28] Q. Zhang, C. Y. Xu, Y. Q. Chen, and Z. G. Yu, Hydrol. Process. **22**, 4997 (2008).
[29] B. Sivakumar, Chaos Soliton. Fract. **19**, 441 (2004).
[30] W. Wang, J. K. Vrijling, P. H. A. J. M. Van Gelder, and J. Ma, J. Hydrol. **322**, 247 (2006).
[31] R. A. Madden and P. R. Julian, Mon. Weather Rev. **122**, 814 (1994).
[32] B. Wang and H. Rui, Meteorol. Atmos. Phys. **44**, 43 (1990).
[33] T. Nakazawa, J. Meteorol. Soc. Jpn. **64**, 17 (1986).
[34] D. L. Hartmann, M. L. Michelsen, and S. A. Klein, J. Atmos. Sci. **49**, 1277 (1992).
[35] B. Liebmann, H. H. Hendon, and J. D. Glick, J. Meteorol. Soc. Jpn.**72**, 401 (1994).
[36] T. Yasunari, J. Meteorol. Soc. Jpn. **58**, 225 (1980).
[37] K. M. Lau and P. H. Chan, Mon. Weather Rev. **114**, 1354 (1986).
[38] L. Ferranti, T. N. Palmer, F. Molteni, and K. Klinker, J. Atmos. Sci. **47**, 2177 (1990).
[39] R. W. Higgins and K. C. Mo, J. Clim. **10**, 223 (1997).
[40] C. Jones, D. E. Waliser, K. M. Lau, and W. Stern, Mon. Weather Rev. **132**, 1462 (2004).
[41] D. Waliser, in *Intraseasonal Variability in the Atmosphere-Ocean Climate System*, edited by W. K. M. Lau and D. E. Waliser (Springer, New York, 2005), pp. 389–423.
[42] D. Waliser *et al.*, J. Clim. **22**, 3006 (2009).
[43] E. D. Maloney and D. L. Hartmann, J. Clim. **11**, 2387 (1998).
[44] M. C. Wheeler and H. H. Hendon, Mon. Weather Rev. **132**, 1917 (2004).
[45] G. J. Tian, D. E. Waliser, and E. J. Fetzer, Geophys. Res. Lett. **33**, L20704 (2006).
[46] [http://www.cpc.noaa.gov/products/precip/CWlink/daily_mjo_index/mjo_index.html]
[47] R. Ding, J. Li, and K. Seo, Mon. Weather Rev. **138**, 1004 (2010).