Calculating Lyapunov exponents for short and/or noisy data sets

Reggie Brown
Institute for Nonlinear Science, University of California, San Diego, La Jolla, California 92093-0402
(Received 26 February 1993)

We present a technique for calculating the Lyapunov spectrum from a scalar time series. The technique is particularly useful when the data set is short and/or noisy. The method is based on an orthogonal polynomial expansion of the dynamics. For comparison purposes we test the new technique to two previous methods. We find that the global method performs as well as, or better than, either of the previous techniques.

PACS number(s): 05.45.+b

I. INTRODUCTION

Over the past few years there has been a great deal of research effort devoted to determining Lyapunov exponents from a scalar time series [1–7]. The spectrum of global Lyapunov exponents is important for a variety of reasons. Lyapunov exponents provide average limits on one's ability to predict the future evolution of phase-space locations. They are invariant under coordinate transformations. When coupled to the Kaplan-Yorke conjecture they provide a good estimate of the information dimension of the attractor, and when combined with the Pesin identity they provide a value for the Kolmogorov-Sinai entropy of the dynamics [8,9]. The Lyapunov exponents determine the average rate of convergence or divergence for nearby trajectories. If one or more of them is positive then trajectories that are initially nearby will diverge over time, and the system will have positive entropy. Finally, at least one positive Lyapunov exponent is the definition of a chaotic dynamical system.

A persistent problem facing many researchers occurs when one wants the Lyapunov exponents but the data set at hand is short and/or contains noise. This problem is constantly faced by experimental researchers. To the best of our knowledge, only one of the previous techniques specifically addressed this problem [7]. In this paper we address the problem of short and/or noisy data using a technique that is quite different from the ones previously used.

The technique we have developed for determining the global Lyapunov exponents involves globally fitting the dynamics as an expansion involving orthogonal polynomials. Assume that the scalar data set $x(n)$, $n=1,\ldots,N$ has been embedded into a d_E-dimensional phase space (time delay works well but is by no means required). The embedded vectors are given by $\mathbf{y}(n)$, $n=1,2,\ldots,N$, where

$$\mathbf{y}(n)=(x(n),x(n+T),\ldots,x([n+(d_E-1)T]))$$

and T as well as d_E are easily found [14,15]. Time evolution in the reconstructed phase space is given by $\mathbf{y}(n)\mapsto\mathbf{y}(n+1)$. We assume that the dynamics originates from or is best modeled by a mapping $\mathbf{y}(n+1)=\mathbf{F}(\mathbf{y}(n))$. The mapping $\mathbf{F}$ is determined as an expansion in terms of orthogonal polynomials $\pi^{(\mathbf{I})}$ [10,11]. These polynomials are constructed, via Gram-Schmidt, to be orthonormal on the attractor

$$\langle\pi^{(\mathbf{I})}|\pi^{(\mathbf{J})}\rangle=\int d\mathbf{z}\,\rho(\mathbf{z})\pi^{(\mathbf{I})}(\mathbf{z})\pi^{(\mathbf{J})}(\mathbf{z})=\delta_{\mathbf{I},\mathbf{J}}\ , \tag{1}$$

where the superscripts $\mathbf{I}$ and $\mathbf{J}$ are d_E-dimensional vectors that indicate the order of the polynomial. The density function $\rho(\mathbf{z})$ is the natural density on the attractor

$$\rho(\mathbf{z})=\frac{1}{N}\sum_{n=1}^{N}\delta(\mathbf{z}-\mathbf{y}(n))\ ,$$

and $\mathbf{z}$ is any point in R^{d_E}. The presence of $\rho(\mathbf{z})$ in Eq. (1) insures that the $\pi^{(\mathbf{I})}$'s are orthogonal on the portion of phase space occupied by the attractor.

The mapping $\mathbf{F}$ is given as an expansion in terms of the $\pi^{(\mathbf{I})}$'s via

$$\mathbf{F}=\sum_{\mathbf{I}}\mathbf{C}^{(\mathbf{I})}\pi^{(\mathbf{I})}\ . \tag{2}$$

The expansion coefficients $\mathbf{C}^{(\mathbf{I})}$ are determined by using the orthonormality of the $\pi^{(\mathbf{I})}$'s. Thus the $\mathbf{C}$'s are given by

$$\mathbf{C}^{(\mathbf{I})}=\langle\pi^{(\mathbf{I})}|\mathbf{F}\rangle=\frac{1}{N}\sum_{n=1}^{N}\mathbf{y}(n+1)\pi^{(\mathbf{I})}(\mathbf{y}(n))\ . \tag{3}$$

The details of the exact functional form of the $\pi^{(\mathbf{I})}$'s and how to explicitly calculate the $\mathbf{C}$'s are quite complicated. In order to preserve the continuity of this short paper we have chosen not to include these results here. For these details we direct the reader to the paper by Giona, Lentini, and Cimagalli as well as our larger paper [10,11].

It is worthwhile to note that the mapping that results from our procedure is a *global* map and will be used as such. Previous techniques for determining the spectrum of Lyapunov exponents from time series generated local mapping of "small" neighborhoods in order to model the dynamics (cf. Refs. [2–7]). Thus our use of a global map is quite different from the previous approaches. Furthermore, using orthonormal polynomials as a basis for expanding functions is numerically more stable than using the standard basis of $(1,x,x^2,\ldots,$ etc.) [12].

Having found $\mathbf{F}$ from Eqs. (2) and (3) it is straightfor-

ward to differentiate **F** to find its Jacobian. In fact, since **F** is a polynomial the Jacobian is just the linear part of the map. (An explicit expression for **DF** can be found in our larger paper [11].) Let **DF**(**z**) denote the Jacobian of the mapping **F** evaluated at the phase-space location **z**. To determine the Lyapunov exponents one forms the product of Jacobians

$$\mathbf{DF}^L(\mathbf{y}(n))=\sum_{i=0}^{L-1}\mathbf{DF}(\mathbf{y}(n+i))$$

and from this product the Oseledec matrix [13]

$$\mathbf{O}[L,\mathbf{z}]=\{[\mathbf{DF}^L(\mathbf{z})]^T\cdot\mathbf{DF}^L(\mathbf{z})\}^{1/2L} . \tag{4}$$

The Lyapunov exponents are determined from the eigenvalues of the Oseledec matrix in the limit $L\rightarrow\infty$ [5,13]. The usual QR decomposition technique is used to determine the Lyapunov exponents from Eq. (4) in the large L limit [5].

Equation (4) implies evaluating the Jacobians along an orbit of the dynamics. In our work we have evaluated the Jacobians along the dirty orbit given by the data. Since we don't have a clean orbit in our possession this is unavoidable. However, this does not seem to limit our ability to determine the Lyapunov exponents. (In our larger paper we investigated the calculation of Lyapunov exponents from true orbits of the fitted map **F** [11].) We have performed numerical experiments on a variety of dynamical systems in order to determine the usefulness of our method.

In each case the results from our technique are compared to the results one obtains from the techniques of Zeng, Eykholt, and Pielke [7] (ZEP) and Brown, Bryant, and Abarbanel [4] (BBA). The ZEP and BBA techniques are two of the standard methods for calculating Lyapunov exponents from time series data. Therefore, they represent reasonable choices for the purposes of comparisons. We are interested in time series data that has been contaminated by additive noise. The ZEP and BBA techniques differ in the manner used to form the neighborhoods employed by the local maps. The implications of these differences in the presence of noise will be discussed in Sec. II. (In passing, we remark that the BBA method was designed to work in the noise-free arena. Therefore, it should not be surprising that it does not perform well when noise exists in the signal.)

For the numerically generated test cases a clean scalar data set $x(n)$ was formed and then contaminated with additive noise η. Thus the data used in the embedding was $x(n)+\eta(n)$. The noise η was generated as $\eta=AN(0,1)$, where A is an amplitude, which we vary. $N(0,1)$ are random numbers whose distributions are either normal with mean 0 and standard deviation 1, or uniform between ±1. The dirty scalar data was then embedded, via time delays, into $d_E=2$ or $d_E=3$ dimensional phase spaces. (The correct value of the embedding dimension and time delay was found previously by the average-mutual-information and false-near-neighbor methods [14,15].) A prediction function **F** is then determined and the Lyapunov exponents are calculated.

All of the dynamical systems that were investigated in our numerical experiments are mappings of the plane into itself. The examples are the Hénon map, the Ikeda map, and a polynomial map we call the McDonald, Grebogi, Ott, and Yorke (MGOY) map [16–18]. The Hénon map is well known and given by

$$x(n+1)=1.0-1.4x^2(n)+y(n) ,$$
$$y(n+1)=0.3x(n) .$$

It has an exact time delay representation as a second-order polynomial. The Ikeda map is given by

$$\mathbf{z}(n+1)=1.0+0.76\mathbf{z}(n)\times\exp(i\{0.4-6.0/[1.0+|\mathbf{z}(n)|^2]\}) .$$

It contains an exponential term and cannot be represented as a finite order polynomial. The MGOY map is given by

$$x(n+1)=x^2(n)-y^2(n)+x(n)-0.295y(n)+0.048 ,$$
$$y(n+1)=2.0x(n)y(n)+x(n)+0.6y(n) .$$

This map differs from the previous cases in that it is not invertible, and, like the Ikeda map, cannot be represented as a finite order polynomial when embedded in $d_E=3$ dimensions. The behaviors of these maps are very different. Thus they, in some sense, are representative of the types of behaviors that occur in dynamical systems which are inherently maps. For the Hénon and MGOY maps the x coordinate was used as the scalar data set, while the imaginary part of **z** was used for the Ikeda data set.

In the next section we present the results of our numerical experiments on these systems. The final section of this paper contains our conclusions and a discussion of our future work in this area.

II. RESULTS AND NUMERICAL EXPERIMENTS

In Fig. 1 we present a $d_E=3$ dimensional embedding of data from the Ikeda map. In Fig. 2 we present the same data with an additive noise level of $A=0.1$ (20% noise). At this noise level all of the detailed structure that is evi-

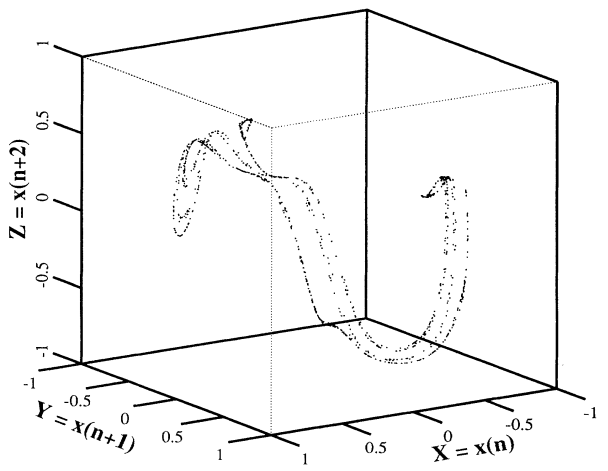

FIG. 1. $N=1000$ clean data vectors from the Ikeda map embedded into $d_E=3$ dimensions.

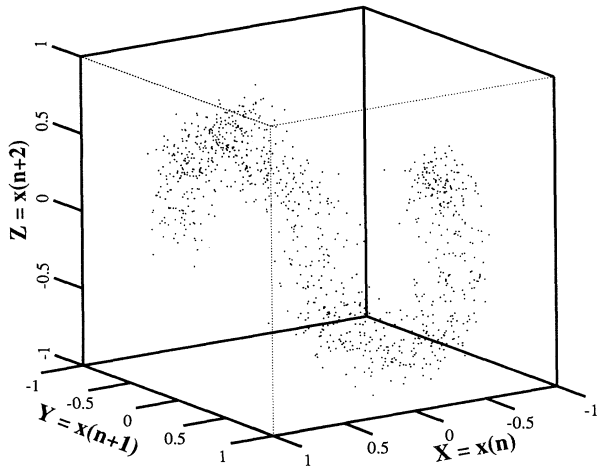

FIG. 2. $N=1000$ noisy data vectors from the Ikeda map embedded into $d_E=3$ dimensions. The noise level is 20%.

dent in the clean signal is lost. Figure 2 is representative of the quality of reconstructions that are available from many experimental systems. For each figure we used $N=1000$ data points. For the numerical experiments on our new method we restricted **F** to a maximum order of $N_p=10$. ($N_p=10$ was used whenever possible. If it was not possible to generate orthogonal polynomials of this order then we used the maximum order we could obtain. This was typically $N_p\sim 7$. In our larger paper we discuss the difficulties associated with obtaining polynomials for larger values of N_p [11].) For the ZEP and BBA methods we used fits of up to $N_p=5$ when modeling the dynamics of the neighborhoods. (Although the symbol N_p is used for all cases it is important to remember that for ZEP and BBA its value denotes the order of a *local* polynomial fit while for our new method its value denoted a *global* orthonormal polynomial fit.)

Examples of the type of raw output obtained from the ZEP, BBA, and our new method is given in Tables I–III. Respectively, they report results for Gaussian noise added to the Hénon map when $N=750$ and $A=0.056$ ($\sim 7\%$), the Ikeda map when $N=1100$ and $A=0.032$ ($\sim 7\%$), and the MGOY map when $N=2000$ and $A=0.0056$ ($\sim 2\%$). It is obvious that the noise is affecting the calculated values for λ_1 and λ_2 in each case. In order to obtain the single values of λ which are plotted in the figures we have often averaged over some of the orders of the fit. Averaging occurred when similar values of λ were produced for a range of N_p values. When a plateau was not present we observed that the $N_p=1$ values of λ_1 and λ_2 were the most accurate for the ZEP technique. Therefore, these values were used in the figures. (Exceptions occurred only when the noise levels were high enough that *all* of the calculated values of λ were grossly incorrect.) The BBA results typically displayed a wide range of values for the λ's, none of which were correct (cf. Tables I and II). To obtain the values plotted in the figures we averaged over any consistent a set of N_p values.

An example of a long plateau can be seen in Table I, where we have averaged over $N_p=2-9$ to obtain the plotted values of the Lyapunov exponents. The range of values produced by our technique for λ_2 in this case has a slight two-plateau structure. By averaging we are eliminating this behavior. We understand that this is an arbitrary procedure on our part and that other researchers may use different approaches. (One could, for example, only use values for $N_p>5$ or $N_p<5$.) But in the absence of knowledge as to the true value of the exponent we consider ours to be a reasonable course to pursue. For the ZEP results we do not observe a plateau. Therefore, we have plotted the $N_p=1$ data. In Table II we observe a short plateau for $N_p=1$ and 2 in the ZEP data. The values for λ_1 and λ_2 when $N_p\leq 2$ are very different from the values found for $N_p>2$. We have averaged these two values for our figures. Table III again indicates the complete absence of a plateau for the ZEP method. (This occurred in MGOY data for all values of $A>0.01$ and often for smaller values of A.)

The results of our calculations for the Hénon, Ikeda, and MGOY maps are presented in Figs. 3–5, respectively. In all of our figures the solid symbols indicate averages obtained by either the ZEP and BBA method for calculating Lyapunov exponents, while the empty symbols indicate the averages of our current method. The

TABLE I. Raw values of the Lyapunov exponents for Hénon data for $N=750$, and Gaussian noises with $A=0.056$. The accepted values are $\lambda_1=0.408$ and $\lambda=-1.62$.

	ZEP method		BBA method		Our method	
N_p	λ_1	λ_2	λ_1	λ_2	λ_1	λ_2
1	0.310 17	−0.732 67	0.823 64	−0.318 11	−0.530 56	−1.2253
2	0.484 40	−0.623 07	0.573 71	−0.474 90	0.416 37	−1.6842
3	0.730 63	−0.387 01	0.696 36	−0.411 89	0.443 99	−1.6923
4	0.817 12	−0.234 82	0.657 68	−0.404 60	0.435 43	−1.5629
5	1.062 3	−0.164 52	0.674 85	−0.344 91	0.455 32	−1.6255
6					0.425 21	−1.3114
7					0.407 17	−1.3287
8					0.408 13	−1.3460
9					0.409 61	−1.3285
10					0.423 29	−1.1828

TABLE II. Raw values of the Lyapunov exponents for Ikeda data for $N=1100$, and Gaussian noises with $A=0.032$. The accepted values are $\lambda_1=0.355$ and $\lambda=-0.904$.

	ZEP method		BBA method		Our method	
N_p	λ_1	λ_2	λ_1	λ_2	λ_1	λ_2
1	0.343 32	−0.800 65	0.435 62	−0.697 47	−0.441 55	−0.760 52
2	0.344 64	−0.841 55	0.366 70	−0.788 65	0.234 95	−0.779 89
3	0.472 96	−0.747 07	0.409 58	−0.730 21	0.443 67	−1.185 8
4	0.567 81	−0.671 77	0.473 58	−0.746 48	0.388 08	−1.078 7
5	0.620 32	−0.598 74	0.477 34	−0.684 42	0.392 77	−0.949 59
6					0.368 58	−1.070 8
7					0.364 13	−0.987 27
8					0.348 75	−0.929 57
9					0.364 30	−0.906 26
10					0.367 53	−0.771 70

solid line indicates the accepted values for the Lyapunov exponent λ_1 and λ_2.

In Figs. 3 we used a $d_E=2$ dimensional embedding with $N=750$ and/or $N=4000$ points. For this case the noise size A ranged between 0.001 and 0.1. As an example of the type of problems that occur with the BBA consider Fig. 3(c), where $N=750$ and Gaussian noise is used. In this figure $\lambda_1+\lambda_2>0$ for $A>0.03$. This is incorrect since it implies the absence of an attractor. The same thing happens for all other cases when the BBA technique is used. The only difference is the exact value of A where this failure occurs. The divergence of the calculated values of λ_1 and λ_2 gets worse when the amount of data N increases. This behavior is an easily explained property of the BBA method and will be discussed below.

The Figs. 3(a) and 3(b) indicate that the ZEP technique performs well when determining λ_1. This improvement over BBA will be observed in all of our test cases. However, the same figures show that ZEP, like BBA, is unable to determine the correct value of λ_2. In general we will find that ZEP often performs poorly when attempting to determine λ_2.

Figures 3 clearly indicate that our technique is more robust to noise than either the BBA or the ZEP method. Our method performs better than BBA for all cases, and performs as well as ZEP when determining the value of λ_1. The figures also show that our technique is capable of determining the correct value of λ_2. Neither of the other methods was able to accomplish this task. We also find that this method works with as few as 750 data vectors and noise levels as high as $A=0.1$ (~ 10–15% noise).

For the last two systems we used $d_E=3$ dimensional embeddings for the data. The true dynamical systems have only two Lyapunov exponents. The technique of Abarbanel and Sushchik [19] provides a way of determining which of the three exponents is spurious. Thus in a blind test one can identify which exponents are "true" and which are "artifacts" of the embedding. We direct the interested reader to Ref. [19], and references therein, for a complete discussion of this technique.

In Figs. 4 we show only the two calculated exponents that correspond to the true Ikeda map exponents. We have chosen not to show the spurious third exponent. In these figures we used either $N=1100$ or $N=20\,000$ data points while A ranged from 0.001 to 0.1. We see that once again BBA is unable to correctly estimate the value of λ_1. Although we will not show the data, we find that for BBA increasing N increases the divergence of λ_1 just as it did in the Hénon example. The ZEP method and our method are comparable in their performance on this data. Both methods are capable of determining accurate values for the positive Lyapunov exponents even for large noise levels. For $N=1100$ our method is marginally better than ZEP when determining λ_2, while for

TABLE III. Raw values of the Lyapunov exponents for MGOY data for $N=2000$, and Gaussian noises with $A=0.0056$. The accepted values are $\lambda_1=0.141$ and $\lambda=-0.405$.

	ZEP method			Our method		
N_p	λ_1	λ_2	λ_3	λ_1	λ_2	λ_3
1	0.128 48	−0.248 52	−0.679 03	−0.290 90	−0.290 90	−0.463 23
2	0.204 23	−0.154 23	−0.677 29	0.306 84	0.187 70	−0.364 21
3	0.299 89	−0.020 80	−0.679 60	0.335 89	0.169 86	−0.798 35
4	0.465 59	0.061 57	−0.571 07	0.184 91	−0.014 33	−0.623 47
5	0.667 05	0.173 11	−0.505 59	0.168 08	−0.373 47	−0.881 11
6				0.164 42	−0.386 53	−0.809 90
7				0.153 73	−0.417 09	−0.804 03
8				0.144 92	−0.453 05	−0.929 68

$N=20\,000$ ZEP is marginally better than our method. We conclude that for Ikeda data the two techniques obtain the correct answers and are equal in their performance.

In Figs. 5 we show all three calculated exponents for data from the MGOY map. Once again $N=1100$ and $N=20\,000$ for the figures, and the noise size ranged from $A=0.001$ to $A=0.1$. Figures 5(a) and 5(b) indicate that the ZEP method and our method are comparable in their abilities to determine the positive Lyapunov exponent. The figures also show that for all case the ZEP method is unable to determine the correct value for λ_2, whereas our method obtains the correct value. We found that the BBA method was unable to determine the correct value for either Lyapunov exponent for this map, and experienced the same type of divergence with increasing A and/or N as found for our test on the Hénon and Ikeda maps (cf. Figs. 3 and 4).

The figures show that for all of our test cases, the BBA method produces poorer results when larger data sets are used. The BBA method uses intimate details about the evolution of "small neighborhoods" to construct Jacobians. As N increases the size of the neighborhood used for the fit decreases [4]. When noise levels are high the evolution dynamics of points in small neighborhoods becomes dominated by noise. Thus it is natural that the BBA technique produces poorer results as N increases. The ZEP method circumvents this problem by using small shells instead of balls for its neighborhoods [7]. If the size of the shell is large compared to the noise then the dynamics of points within the shell will not be dominated by noise. For our numerical experiments we used minimum shell radii of $2A$, where A is the level of the noise. Consider Figs. 1 and 2. If the shell size is large compared to the noise then these figures indicate that the shell size could cover as much as 10–20 % of the attractor.

The original ZEP method used linear maps to evolve the data in the shells forward in time. The BBA method (and independently the method of Briggs [6]) demonstrat-

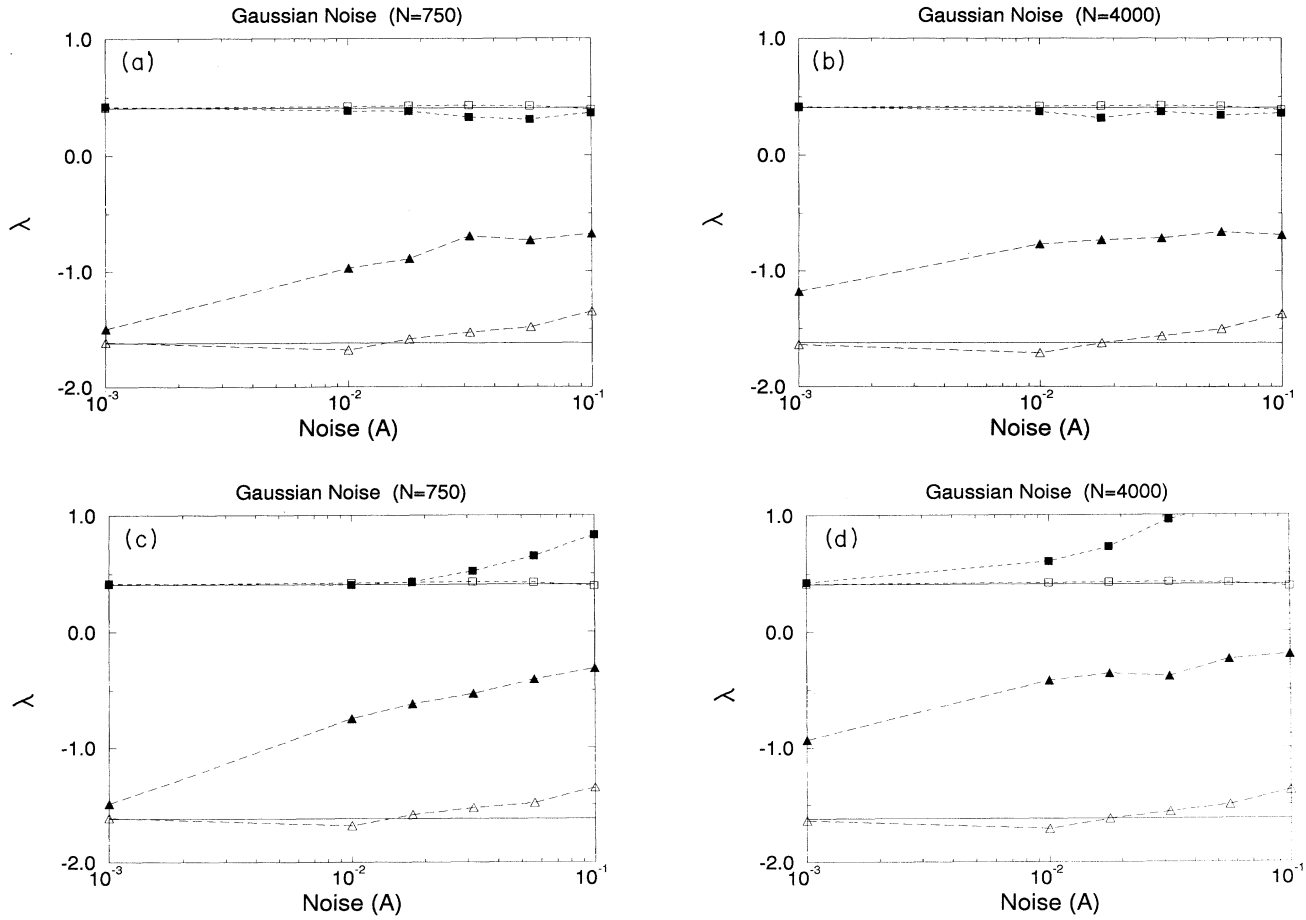

FIG. 3. Lyapunov exponents calculated from noisy Hénon map data. The solid lines indicate the accepted values of λ_1 and λ_2. (a) The solid symbols come from the ZEP method while the empty symbols come from our method. In this figure $N=750$ and the noise is Gaussian. (b) The solid symbols come from the ZEP method while the empty symbols come from our method. In this figure $N=4000$ and the noise is Gaussian. (c) The solid symbols come from the BBA method while the empty symbols come from our method. In this figure $N=750$ and the noise is Gaussian. (d) The solid symbols come from the BBA method while the empty symbols come from our method. In this figure $N=4000$ and the noise is Gaussian.

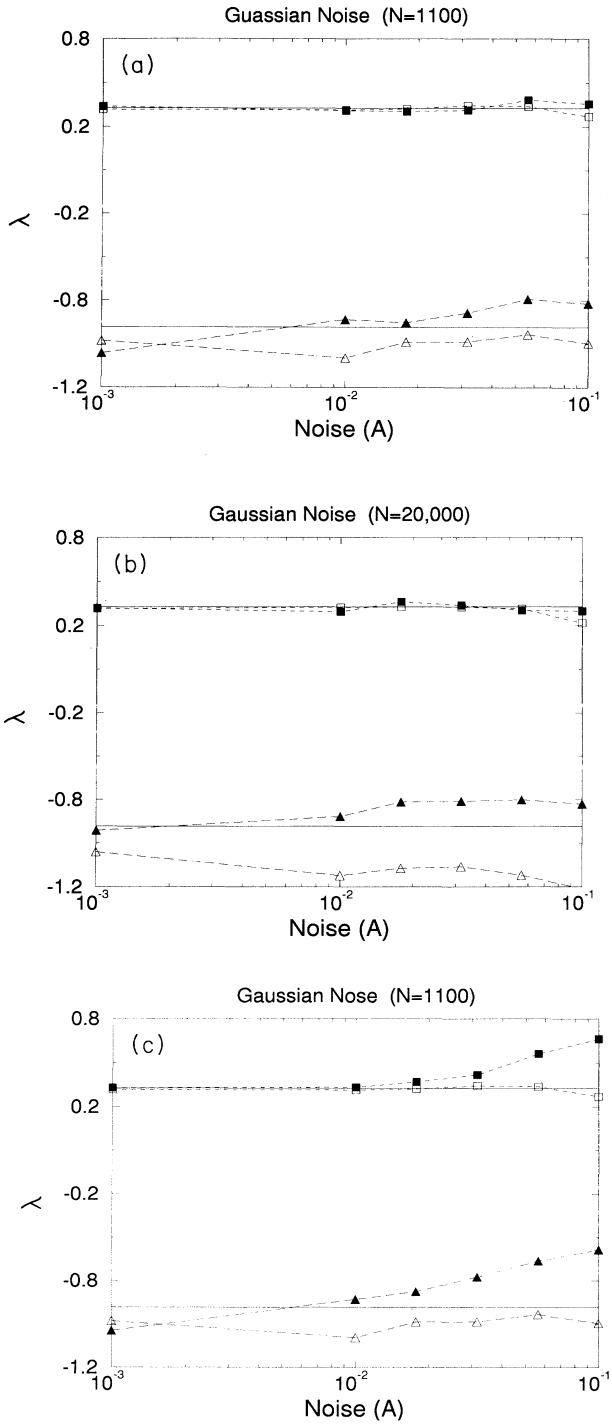

FIG. 4. Lyapunov exponents calculated from noisy Ikeda map data. Only the two Lyapunov exponents corresponding to the true values are shown. The solid lines indicate the accepted values of λ_1 and λ_2. (a) The solid symbols come from the ZEP method while the empty symbols come from our method. In this figure $N=1100$ and the noise is Gaussian. (b) The solid symbols come from the ZEP method while the empty symbols come from our method. In this figure $N=20\,000$ and the noise is Gaussian. (c) The solid symbols come from the BBA method while the empty symbols come from our method. In this figure $N=1100$ and the noise is Gaussian.

ed that if clean data was used, then employing higher-order fits can increase the accuracy of the calculated values of the negative Lyapunov exponents. In this paper we have included this adaptation to the original ZEP method. We see from the tables that in some instances the higher-order fits increase one's ability to accurately calculate the Lyapunov exponents. (For example the repeated values of $\lambda_1 \sim 0.344$ for $N_p \leq 2$ lends confidence to these values.) But this is not always the case, as is clear from Table III. For this case it is only our foreknowledge of the correct values of λ (or inconclusive arguments about the size of N versus the size of A) that allows us to know that $N_p = 1$ is the most correct answer. Our new method always has a clear plateau structure (cf. Table I for $N_p > 2$, Table II for $N_p > 5$, and Table III for $N_p > 4$).

We have repeated all of our calculations for data sets contaminated with uniform noise. We find that the qualitative behavior of the methods remains unchanged. Our method is better than ZEP since it is able to correctly determine the values of all of the Lyapunov exponents

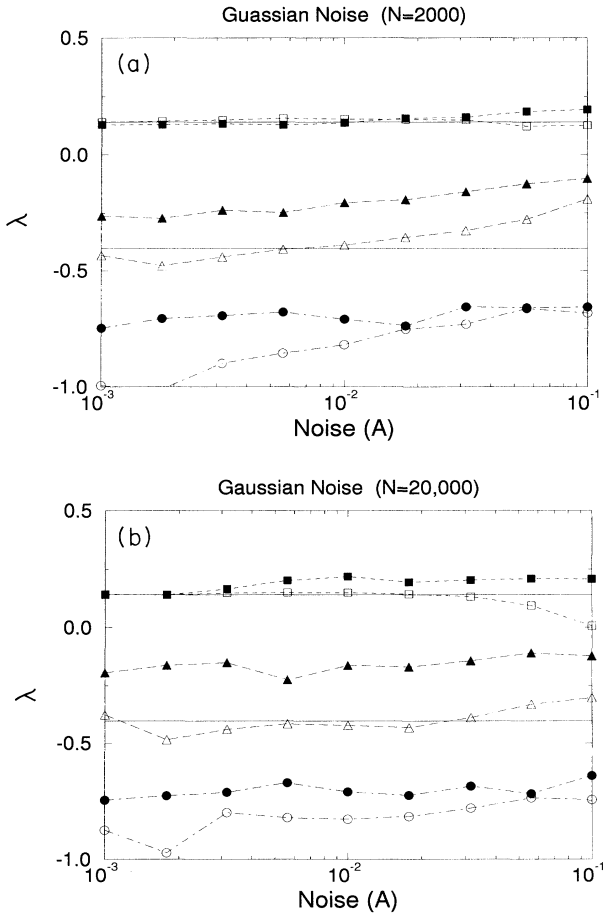

FIG. 5. Lyapunov exponents calculated from noisy MGOY map data. The solid lines indicate the accepted values of λ_1 and λ_2. (a) The solid symbols come from the ZEP method while the empty symbols come from our method. In this figure $N=2000$ and the noise is Gaussian. (b) The solid symbols come from the ZEP method while the empty symbols come from our method. In this figure $N=20\,000$ and the noise is Gaussian.

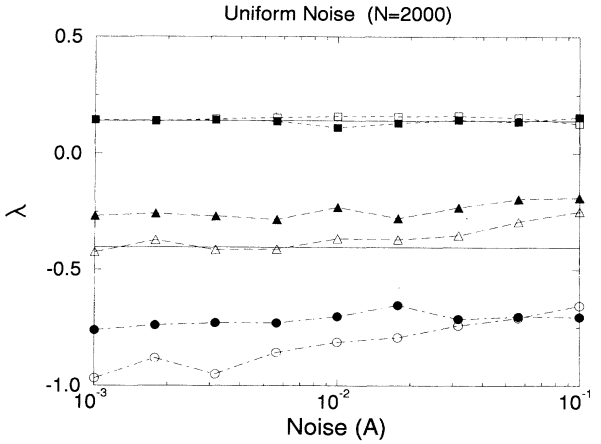

FIG. 6. Lyapunov exponents calculated from noisy data from the MGOY map. In this figure $N=2000$, and the noise is uniform. The solid lines indicate the accepted values of λ_1 and λ_2.

while the ZEP method is unable to determine the values of the negative Lyapunov exponents in the same two out of three cases. The amplitudes of the differences between the accepted and the calculated values of the Lyapunov exponents for fixed A decrease when uniform noise is used instead of Gaussian noise. [Compare the results of Fig. 5(a) and Fig. 6.] This is to be expected. Uniform noise is restricted to the range $[-A, A]$. The Gaussian noise from our tests exceeded this range 32% of the time. Thus for fixed A Gaussian noise can appear to be larger than uniform noise.

III. CONCLUSION

In conclusion we have presented a technique for determining a global map that can be used to calculate the spectrum of Lyapunov exponents from a scalar data set. Our approach is different from other approaches in two ways. (i) Our approach works well when the data has been contaminated by high values of additive noise. (ii) Our approach uses a *single global* function whose Jacobian is used to form the Oseledec matrix, Eq. (4). To substantiate the claims made above we have compared our method to the ZEP and BBA methods for calculating Lyapunov exponents. (The ZEP method is the only other technique we know of that specifically addresses the issue of extracting Lyapunov exponents from a noisy signal.)

The numerical experiments show that the ZEP method is capable of determining the positive Lyapunov exponent for all of our test cases. They also show that the ZEP method is *unable* to determine the correct value of the negative Lyapunov exponent in two of the three test cases. In contrast the method based on orthonormal polynomials was able to determine *both* the positive and negative Lyapunov exponent for *all* of the test cases. The BBA method was always incapable of determining *either* of the exponents. The conclusion we reach is that the results of our numerical experiments indicate that a map trained by the orthonormal polynomial technique is superior to current techniques for extracting Lyapunov exponents from scalar data sets. Furthermore, we have demonstrated that the training of the global polynomials requires a relatively small number of data vectors.

The existence of a clear plateau for a range of N_p values (cf. Tables I–III) is a significant result. One of the failings of all previous techniques for calculating Lyapunov exponents is the difficulty associated with determining the correct values for the λ's when different values of N_p are used. This "art"-like component of the calculation implied that one often has very little confidence in the values obtained by the calculations. The existence of relatively unambiguous plateaus in our method indicates that the calculated values of the Lyapunov exponents are consistent with improved modeling. Therefore, they warrant much more confidence than previous methods.

As always, when attempting to determine the Lyapunov spectrum from a data set one must be sure that the system under investigation actually has a low dimensional attractor. If the data that is used in our procedure does not come from a low dimensional attractor then the calculated spectrum will be of little, if any, value. For example, if stock-market data is fed into our technique one will get d_E Lyapunov exponents. However, it does not follow that the market must be a d_E-dimensional dynamical system living on a chaotic attractor.

In Fig. 7 we iterated the mapping that resulted from training **F** with $N=2000$ points contaminated with 20% noise. The first noisy data vector $\mathbf{y}(1)$ was then iterated using the trained map. As one can see by comparing Figs. 1 and 7 the fitting procedure we use produces a map in close approximation to the true dynamical mapping. This approximation is much closer than one might expect to find, given such large noise levels, and accounts for why one is able to calculate the Lyapunov exponents. Similar figures arise when other initial conditions are used.

The theorems of Mañé and Takens imply that if the true and embedded dynamical systems are

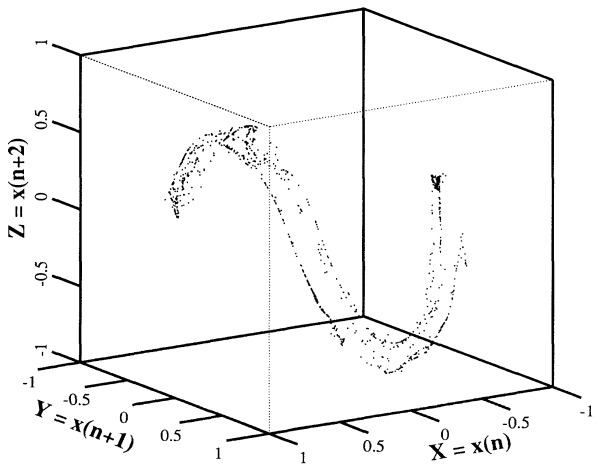

FIG. 7. Results of training a map with 20% noisy data and then iterating a single point 1000 times.

$\mathbf{z}(n+1)=\mathbf{G}(\mathbf{z}(n))$ and $\mathbf{y}(n+1)=\mathbf{F}(\mathbf{y}(n))$, respectively, then a diffeomorphism exists between the coordinates $\mathbf{z}$ and $\mathbf{y}$, $\mathbf{y}=\varphi(\mathbf{z})$. This result leads to $\mathbf{F}=\varphi\circ\mathbf{G}\circ\varphi^{-1}$. Figure 7 and the results of our Lyapunov exponent calculations suggest that the mapping determined by our training procedure is a close approximation to *the* map in the embedded phase space. For example, the $\mathbf{F}$ determined by our procedure is a close approximation to $\varphi\circ\mathbf{G}\circ\varphi^{-1}$. Since polynomials form a complete basis set we can, in principle, obtain an arbitrarily close approximation to the true $\mathbf{F}$. We will take up these speculations in more detail in our larger paper [11].

Further applications of this technique for finding $\mathbf{F}$ may involve noise reduction and the calculation of local Lyapunov exponents. We are currently investigating the possibility of using an $\mathbf{F}$ constructed by this method to determine the vector field when the scalar data set is from a differential equation instead of a map. It is our belief that this will allow one to calculate Lyapunov exponents for these systems in the presence of noise.

ACKNOWLEDGMENTS

This work was supported by the U.S. Department of Energy, Office of Basic Energy Sciences, Division of Engineering and Geoscience, under Contract No. DE-FG03-90ER14138. The theoretical underpinnings of this work were performed while the author was a visitor at the Institute for Applied Physics in Nizhny Novgorod, Russia. The author is grateful to his hosts and the Russian people for their kindness and support during that visit.

[1] J. Wright, Phys. Rev. A **29**, 2924 (1984).
[2] M. Sano and Y. Sawada, Phys. Rev. Lett. **55**, 1082 (1985).
[3] J.-P. Eckmann, S. O. Kamphorst, D. Ruelle, and S. Ciliberto, Phys. Rev. A **34**, 4971 (1986).
[4] R. Brown, P. Bryant, and H. D. I. Abarbanel, Phys. Rev. A **43**, 2787 (1991).
[5] H. D. I. Abarbanel, R. Brown, and M. B. Kennel, Int. J. Mod. Phys. B **5**, 1347 (1991).
[6] K. Briggs, Phys. Lett. A **151**, 27 (1990).
[7] X. Zeng, R. Eykholt, and R. A. Pielke, Phys. Rev. Lett. **66**, 3229 (1991).
[8] J. L. Kaplan and J. A. Yorke, in *Functional Differential Equations and Approximation of Fixed Points,* edited by H.-O. Pietgen and H.-O. Walther, Lecture Notes in Mathematics Vol. 730 (Springer, Berlin, 1979), p. 228.
[9] Y. B. Pesin, Usp. Mat. Nauk **32**, 55 (1977); Russ. Math. Survey **32**, 32 (1977).
[10] M. Giona, F. Lentini, and V. Cimagalli, Phys. Rev. A **44**, 3496 (1991).
[11] R. Brown, Institute for Nonlinear Science report, April, 1993.
[12] L. Fox and D. F. Mayers, *Computing Methods for Scientists and Engineers* (Clarendon, Oxford, 1968).
[13] V. I. Oseledec, Trudy Mosk. Mat. Obsc. **19**, 197 (1968).
[14] A. M. Fraser and H. L. Swinney, Phys. Rev. A **33**, 1134 (1986); A. M. Fraser, IEEE Trans. Inf. Theory **35**, 245 (1989).
[15] M. B. Kennel, R. Brown, and H. D. I. Abarbanel, Phys. Rev. A **43**, 3403 (1992).
[16] M. Hénon, Commun. Math. Phys. **50**, 69 (1976).
[17] S. M. Hammel, C. K. R. T. Jones, and J. V. Moloney, J. Opt. Soc. Am. B **2**, 552 (1985); K. Ikeda, Opt. Commun. **30**, 257 (1979).
[18] S. W. McDonald, C. Grebogi, E. Ott, and J. A. Yorke, Physica D **17**, 125 (1985).
[19] H. D. I. Abarbanel and M. Sushchik, Proceedings of the Second Workshop on Measures of Complexity and Chaos, June, 1993, Int. J. Bif. Chaos (to be published).