PHYSICAL REVIEW A VOLUME 46, NUMBER 6 15 SEPTEMBER 1992

Local-geometric-projection method for noise reduction in chaotic maps and flows

Robert Cawley
Mathematics and Information Sciences Branch, Naval Surface Warfare Center, Dalhgren Division, White Oak, Silver Spring, Maryland 20903-5000

Guan-Hsong Hsu*
Department of Mathematics, University of Missouri, Columbia, Missouri 65211

(Received 29 April 1991; revised manuscript received 21 April 1992)

We describe a method for noise reduction in chaotic systems that is based on projection of the set of points comprising an embedded noisy orbit in $\mathbb{R}^d$ toward a finite patchwork of best-fit local approximations to an m-dimensional surface $M' \subset \mathbb{R}^d$, $m \le d$. We generate the orbits by the delay coordinate construction of Ruelle and Takens [N. H. Packard *et al.*, Phys. Rev. Lett. **45**, 712 (1980); F. Takens, in *Dynamical Systems and Turbulence, Warwick, 1980*, edited by D. A. Rand and L.-S. Young (Springer, Berlin, 1981)] from time series $v(t)$, which in an experimental situation we would assume to have come, together with additional high-dimensional background noise, from an underlying dynamical system f^t: $M \to M$ existing on some low m-dimensional manifold M. The surface M' in $\mathbb{R}^d$ is the assumed embedded image of M. We give results of systematic studies of linear (tangent plane) projection schemes. We describe in detail the basic algorithm for implementing these schemes. We apply the algorithm iteratively to known map and flow time series to which white noise has been added. In controlled studies, we measure the signal-to-noise ratio improvements, iterating n_M times until a stable maximum δ_M is achieved. We present extensive results for δ_M and n_M for a wide range of values of embedding trial dimension d, projection dimension k, number of nearest-neighbor points for local approximation ν, embedding delay Δ, sampling interval ΔT, initial noise amplitude $\mathcal{N}$, and trajectory length N. We give results for very low and very high noise amplitudes $0\% \le \mathcal{N} \le 100\%$. We develop an empirical method for estimating the initial noise level for a given experimental time series, and for the optimal choice of algorithm parameters to achieve peak reduction. We present interesting results of application of the noise-reduction algorithm to a chaotic time series produced from a periodically driven magnetoelastic ribbon experiment on the control of chaos. Two noteworthy elements of the noise-reduction method we describe result in certain stabilizing and efficiency features. The first is our use of a physical replacement time series, which is a unique scalar time series with the property that its corresponding delay coordinate construction data state vector time series in $\mathbb{R}^d$ is optimally close to the noise-reduced replacement vector time series generated by the projection. The second is the introduction of a "measure-ordered" cover, which produces notable improvement in reliability, control, and computational efficiency of the whole algorithm.

PACS number(s): 05.45.+b, 02.60.+y, 02.50.+s, 02.70.+d

I. INTRODUCTION

In attempting to gain understanding, or control of a complex physical process, some kind of simple model, where possible, is always a valuable first step. The ubiquity of low-dimensional chaotic behavior in physical systems is likely a consequence of its genericity in the solution properties of nonlinear differential equations, which are again and again the basis of physical modeling. Experimentally, the underlying simplicity actually present can be masked by a high level of noisy background contamination. This can be especially serious in the presence of chaos, and especially when the simple low-dimensional system is coupled to a much larger reservoirlike system whose high-dimensional fluctuating input results in further complexity unimportant for the simple system model. On the other hand, the presence of low-dimensional chaos also presents an opportunity since chaotic phase portraits sample dimensionally large portions of the phase space.

There may be many physical systems where this kind of picture actually is possible, but where large-amplitude, high-dimensional background interference has obscured evidence of the low-dimensional behavior. Fluid systems, turbulent behavior, oceanographic and geophysical phenomena, and meteorological and biological phenomena are examples where data are typically very noisy and complex. The absence of adequate, experimentally useful and reliable chaotic noise-reduction techniques does not merely limit our ability to detect the presence of low-dimensional dynamics. In some cases, this lack may have led to premature judgments about the presence of chaos. In this paper, we present a robust noise-reduction procedure for chaotic systems that (i) does not require prior knowledge of an underlying dynamics, and (ii) has performed satisfactorily in very high noise environments [up to 100% noise, or 0 dB signal-to-noise ratio (SNR)] in the cases we have examined. The scheme possesses considerable structure and flexibility in how it can be applied. Also, through a systematic study of the output of our algorithm, we have been able to develop a method to estimate in some cases the noise level present in an experimental time series.

Generally speaking, a complicating feature to the chaotic noise-reduction problem is that the signal will have a broadband spectrum just like the background contaminant. Traditional bandpassing techniques might result in unacceptable signal distortions, or simply be ineffective. An interesting early exploration of such a Fourier based linear filtering approach in the context of chaotic dynamics is that of Badii and Politi [1]. Another approach based on linear signal processing ideas was Broomhead and King's singular value truncation [2] of the trajectory matrix for the whole orbit of a digitized, embedded time series. More recently, the problem of reduction of noise has been the subject of renewed attention. Several new approaches based on the theory of nonlinear dynamical systems have been developed. Kostelich and Yorke [3] combine a dynamical learning technique with a least-squares trajectory adjustment procedure to create a new trajectory. The procedure is then iterated several times. Farmer and Sidorowich [4] also employ a learning technique, and exploit expanding and contracting behavior to achieve a noise reduction. When the map is specified beforehand, the method of Hammel [5] has achieved more than ten orders of magnitude improvement using Anosov-Bowen shadowing ideas. Farmer and Sidorowich [6] have combined Hammel's shadowing approach with statistical methods to produce comparable results. The shadowing techniques have been implemented so only for two-dimensional maps. Also when a clean reference orbit is available for the noisy data, noise reduction can be achieved by a conditional probability technique (Marteau and Abarbanel [7]). These methods exploit dynamical properties and statistical properties of the dynamical system to effect noise reduction. In contrast, our method is motivated by the fundamental geometric picture of an underlying smooth manifold.

Our general approach is a natural development from *later* work of Broomhead, Jones, and King [8], who described a method for estimating topological dimension for the dynamics underlying a chaotic time series in the noise-free case. Their method involved another, *local* singular value analysis, *distinct* from that of Ref. [2]. In this later work [8] a procedure was developed to exploit the fact that points of a chaotic trajectory that lie in a small ball centered on any given trajectory point $\mathbf{p}_0$ generically will lie close to the tangent space at $\mathbf{p}_0$. The noise-reduction idea we explore here is that in the presence of noise these points typically lie off (the embedded image of) the manifold for the dynamics due to the fact that noise is generally very high dimensional. The basic idea of the method is to estimate where the manifold M' is and move the errant points closer to it to form a candidate replacement trajectory. In this work, we use the collection of local tangent spaces at points $\mathbf{p}_0$ as an approximation of the manifold M'. For points in a small neighborhood of a given $\mathbf{p}_0$, one simply projects them into the tangent spaces to remove the noise. We actually employ a refinement of this approach, designed to take M' curvature effects into account in a crude way, and to soften effects of rare statistical outliers, for high noise cases in particular, viz. we choose an average of original and projected points instead of just the projected point itself.

There is a possible difficulty with this kind of procedure, i.e., that of moving points in the phase portraits, but it can be remedied. Normally, the embedded trajectory is constructed from a *measured scalar* observable by the use of delay coordinates [9,10], but the candidate replacement trajectory described above is usually not itself realizable as a delay coordinate construction from a scalar time series. This alone likely implies some loss of information about an underlying dynamics. We can, however, identify a "physical" replacement trajectory that is so realizable (see Sec. IV), and thereby recover a cleaned-up scalar observable. In fact, this also leads to improvements in the noise-reduction performance.

A second element of the present work is our use of a systematic procedure for covering the embedded object. By ordering the elements $U^{(\alpha)}$ of the cover according to the magnitudes of r_α^{-1} for various α, where r_α denotes the radius of the ball enclosing a fixed number of points nearest neighbor to a given corresponding reference point $\mathbf{p}_0^\alpha$, we produce a rough separation of densely and sparsely populated regions of the noisy attractor. The resulting "measure-ordered" cover $\{U^{(\alpha)}|\alpha=1,\ldots,N_c\}$ allows us to design a natural scheme to prevent multiple projections. We first project points in cover elements having small radius r_α since the manifold should be better approximated by tangent spaces in these elements of the cover. Once a point has been projected, we prohibit it from being projected again. In contrast, the simple approach of covering the trajectory by nearest neighbors to, say, every fifth point in the orbit, misses some points and overcounts others. This leads to uncontrolled, uneven manipulations of trajectory points in the noise-reduction procedure. It is also inefficient. In one Hénon map study, this procedure involved 600 time-consuming nearest-neighbor searches, while for the same example the measure-ordered cover required only 400.

We begin by describing the theoretical foundation for the geometric method in Secs. II–IV, introducing in Sec. IV the physical replacement time series mentioned above. In Sec. V we describe the noise-reduction algorithm. In each of the following sections we examine a succession of separate noise-reduction issues. We present numerical results of a systematic study for the Hénon map in Sec. VI along with results from a few studies of the Ikeda map. We give our results for flows, Lorenz and Rössler systems, and perform an initial exploration of dependence on sampling rate and delay in Sec. VII. In Sec. VIII, we investigate the nature and limits of the method by examining performance against very high and very low noise for both the Hénon and Lorenz cases, sampling rate dependence for the Lorenz system, and some dynamical features of algorithm performance for a few specific examples. In Sec. IX we develop an empirical method for estimating in some cases the initial noise level in an experimental time series where a true underlying signal is unknown. We present results for application of the algorithm to a chaotic time series taken from a periodically driven magnetoelastic ribbon experiment on the control of chaos [11]. We conclude the paper with a discussion and summary in Sec. X. Two appendixes are attached at the end. In Appendix A we give a simple proof of Theorem 2 (from Sec. III); in Appendix B we describe the linear algebra of singular value theory and give a geometric interpretation of our method, sharply contrasting it with that of Ref. [2].

II. DELAY COORDINATE CONSTRUCTION

The mathematical model for a chaotic system is a nonlinear, smooth dynamical system defined on some differentiable manifold, the phase space. Let M be a differentiable manifold, and denote by

$$f^t : M \to M \tag{1}$$

a smooth dynamical system defined on M with $t \in \mathbb{R}$ or $\mathbb{Z}$. In the case of $t \in \mathbb{R}$, the dynamical system is a *flow*. When $t \in \mathbb{Z}$, the dynamical system is given by the iteration of a smooth *map* or diffeomorphism f. Both cases can arise as models of chaotic systems from physical, biological, and engineering sciences. We shall refer to these "real" systems as physical systems. Flows and maps model physical systems with continuous time and discrete time, respectively.

An experimental measurement on a physical system typically produces a time history of a scalar quantity that includes a noisy background component. Let

$$v(t) : t = t_1, \ldots, t_{N_D} \tag{2}$$

represent such a time series with $t_i \in \mathbb{Z}$ (we suppose for flows that $t \in \mathbb{R}$ is now replaced by $t_i \in \mathbb{Z}$ by means of a sampling procedure applied to the original physical data). Assume that there is a dynamical system of the form (1) underlying the evolution of the time series, and denote by

$$V(t) : t = t_1, \ldots, t_{N_D} , \tag{3}$$

the ideal measurement along a trajectory of (1) that would result in the absence of the noisy background. Let $S: M \to \mathbb{R}$ be a smooth scalar function, which is the quantity we seek to recover, i.e.,

$$V(t) = S(f^t\mathbf{x}) , \tag{4}$$

where $\mathbf{x} \in M$ is the initial point of the trajectory. We suppose we have sampled (2) at uniform time intervals τ_s so that $t_i = i\tau_s$, $i = 1, \ldots, N_D$ and we take $\tau_s = 1$ in the following discussion. The *delay coordinate construction* from $v(t)$ proceeds as follows [9,10]. One forms data state vectors $\mathbf{P} = \mathbf{P}(t)$ in a Euclidean space $\mathbb{R}^d$ from the time series $V(t)$ by

$$\mathbf{P}(t) = (V(t), V(t+\Delta), V(t+2\Delta), \ldots, V(t+(d-1)\Delta)) , \tag{5}$$

where Δ, normally a positive integer, is the time lag between coordinates and is called the *delay*. This gives a representation of the trajectory $f^t\mathbf{x}$, in a Euclidean space. With the help of this construction one can apply the theory of smooth dynamical systems to investigate the evolution of time series when these arise from a map or flow. When applied to the experimental time series $v(t)$ of Eq. (2), we assume that the data state vectors that result give a correspondingly noisy representation of $f^t\mathbf{x}$ in $\mathbb{R}^d$.

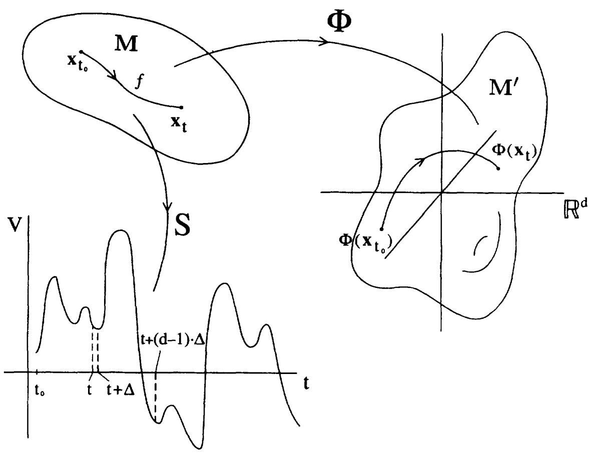

FIG. 1. Delay coordinate construction.

Combined with Eq. (3), Eq. (5) amounts to defining a map Φ from M to $\mathbb{R}^d$ where

$$\Phi(x) = (S(x), S(f^{\Delta}\mathbf{x}), \ldots, S(f^{(d-1)\Delta}\mathbf{x})) . \tag{6}$$

See Fig. 1.

The following theorem of Takens implies that, in general, one can expect complete information about the dynamical system f^t from just one time series.

Theorem 1 [10]: Suppose a smooth dynamical system f^t is defined on a manifold M, and a smooth scalar function S is also given as above. Define the map Φ: $M \to \mathbb{R}^d$ by Eq. (6). If $\dim M = m$ and $d \geq 2m+1$, then it is a generic property of f^t and S that the map Φ is a smooth embedding of M into $\mathbb{R}^d$.

A smooth embedding preserves geometric properties of the manifold and dynamical information about the system. One can thus perform computations on $M' = \Phi(M) \subset \mathbb{R}^d$ and expect the results to apply to M provided at least that the quantities one computes are invariants of the embedding. For Takens's proof, smoothness meant f and S should be at least C^2.

III. ERROR FUNCTION FOR ESTIMATION OF THE TANGENT SPACE

One of the simplest of embedding invariants is the *topological dimension* of M. A manifold is a space that is isomorphic to Euclidean space locally at every point. The topological dimension is the unique dimension m of these Euclidean spaces. Since Φ is an embedding, the topological dimension of M is also the dimension of the tangent space $T_{\mathbf{p}}M'$ at any point $\mathbf{p} \in M'$, as a linear subspace of $T_{\mathbf{p}}\mathbb{R}^d=\mathbb{R}^d$. Since we suppose m to be unknown, we form data state vectors as above from the observed time series $v(t)$ with trial dimension d chosen large enough that by some suitable test Φ may be assumed to be an embedding. We seek a suitable basis for estimating the tangent spaces $T_{\mathbf{p}}M'$.

We proceed by constructing a quantity that can represent the error of approximation of a finite cluster of points in $\mathbb{R}^d$ by a linear space. We will find the requirement that the error be minimized leads to an eigenvalue problem for a matrix X closely related to that employed in the local singular value analysis of [8].

Let H_k be a k-dimensional linear subspace of $\mathbb{R}^d$ and $\mathbf{v}$ be a vector in $\mathbb{R}^d$; we define the distance between $\mathbf{v}$ and H_k as

$$d(\mathbf{v},H_k)=\|(1-Q_k)\mathbf{v}\| , \tag{7}$$

where Q_k is the projection from $\mathbb{R}^d$ onto H_k and $\| \ \|$ is the Euclidean norm in $\mathbb{R}^d$.

The data state vector orbit for the pure time series $V(t)$ samples the image $\Phi(\Lambda)=\Lambda'\subset M'$ of the chaotic attractor Λ for the system. Owing to the presence of noise the points of the orbit in $\mathbb{R}^d$ for the noisy time series $v(t)$ fluctuate out of M'. Near any point $\mathbf{p}_0\in\mathbb{R}^d$ lying in the orbit for the noisy time series, let $\mathbf{p}_1,\ldots,\mathbf{p}_\nu$ be the ν nearest neighbors contained within a ball of radius r. We choose a reference point $\tilde{\mathbf{q}}\in M'$ that is close to all the points $\mathbf{p}_i$ so that $\|\mathbf{p}_i-\tilde{\mathbf{q}}\|<r$ holds for all $i=0,1,\ldots,\nu$ (Fig. 2). Denote by $\mathbf{x}_i$ the unit vector

$$\mathbf{x}_i=\frac{\mathbf{p}_i-\tilde{\mathbf{q}}}{\|\mathbf{p}_i-\tilde{\mathbf{q}}\|} , \quad i=0,1,\ldots,\nu . \tag{8}$$

Unless the noise is very strong, we expect these vectors to approximate tangent vectors to M' at $\tilde{\mathbf{q}}$ reasonably well if r is sufficiently small. For this sample we define the error of approximation by H_k as

$$D(r,H_k)=\frac{1}{\nu+1}\sum_{i=0}^{\nu} d(\mathbf{x}_i,H_k)^2 . \tag{9}$$

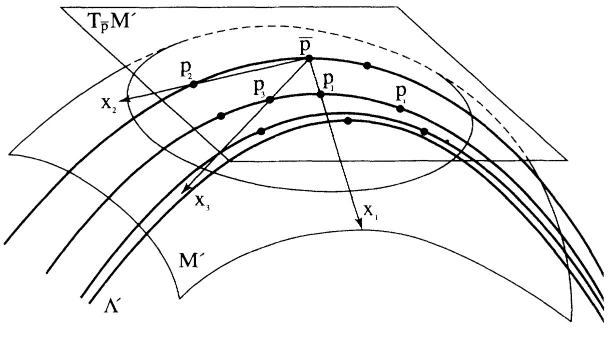

FIG. 2. Schematic picture of neighborhood of $\bar{\mathbf{p}}$ on reconstructed attractor Σ'. In the figure $k=m=2$ for the plane. We have taken $\tilde{\mathbf{q}}=\bar{\mathbf{p}}$ [see Eq. (15)]. The distance of the *unit* vector pointing at $\mathbf{p}_i$ to the tangent space $T_{\bar{\mathbf{p}}}M'$ is of order r, hence $D(r,k)$ is of order r^2. Any line through $\bar{\mathbf{p}}$ in the tangent plane is an example of an H_k with $k=1<2$. The distance of almost all unit vectors to this line is of the order of the length of the vector and hence of order 1.

For each k we seek to minimize D over all k-dimensional linear spaces H_k.

Let $\mathbf{u}_1,\ldots,\mathbf{u}_d$ be an orthonormal basis for $\mathbb{R}^d$ such that $\mathbf{u}_1,\ldots,\mathbf{u}_k$ spans H_k. Then for any vector $\mathbf{v}\in\mathbb{R}^d$

$$d(\mathbf{v},H_k)=\left[\sum_{j=k+1}^{d}(\mathbf{v}^T\mathbf{u}_j)^2\right]^{1/2} , \tag{10}$$

and we have

$$\begin{aligned} D(r,H_k)&=D(r;\mathbf{u}_1,\ldots,\mathbf{u}_k)\\ &=\frac{1}{\nu+1}\sum_{i=0}^{\nu}\sum_{j=k+1}^{d}(\mathbf{x}_i^T\mathbf{u}_j)^2=\sum_{j=k+1}^{d}\mathbf{u}_j^T X\mathbf{u}_j , \end{aligned} \tag{11}$$

where the "local excursion matrix"

$$X=\frac{1}{\nu+1}\sum_{i=0}^{\nu}\mathbf{x}_i\mathbf{x}_i^T \tag{12}$$

is a $d\times d$, symmetric positive definite matrix. Thus minimizing D is equivalent to minimizing the sum of the $d-k$ positive definite quadratic forms $\mathbf{u}^T X\mathbf{u}$ over orthonormal sets $\mathbf{u}_1,\ldots,\mathbf{u}_k$.

Theorem 2: Let $\lambda_1\geq\lambda_2\geq\cdots\geq\lambda_d$ be the eigenvalues of a $d\times d$ symmetric matrix X and let $\mathbf{w}_1,\ldots,\mathbf{w}_d$ be the corresponding orthonormal eigenvectors. Then $\mathbf{w}_{k+1},\ldots,\mathbf{w}_d$ minimize the sum of quadratic forms specified on the right-hand side of Eq. (11), and the minimum is

$$D=\sum_{j=k+1}^{d}\lambda_j . \tag{13}$$

The proof of the theorem is sketched in Appendix A. In particular, we have

$$\min D(r,H_k)\equiv D(r,k)=\sum_{j=k+1}^{d}\lambda_j(r) . \tag{14}$$

Let $\mathbf{w}_1,\ldots,\mathbf{w}_d$ be the eigenvectors of X. Then we estimate $\mathcal{H}_k=\langle\mathbf{w}_1,\ldots,\mathbf{w}_k\rangle$ as the best linear k-dimensional space for local representation of points near $\tilde{\mathbf{q}}$ on the orbit for $v(t)$. If $k\geq m$, our estimate $\mathcal{H}_k$ is expected to obey $T_{\mathbf{q}}M'\subset\mathcal{H}_k$ reasonably well.

We remark that X is the square of a $d\times(\nu+1)$ matrix R having columns given by the $\mathbf{x}_i, i=0,1,\ldots,\nu$, viz. $X=RR^T$. The eigenvalues of X are non-negative, they are squares of the corresponding singular values of R, and the $\mathbf{w}_i$ coincide with the corresponding singular vectors of R. A simple outline of matrix singular value theory is given in Appendix B, together with a geometric interpretation of the λ_i, $\mathcal{H}_k$, and the minimization procedure described above. We also provide in Appendix B a brief description of the so-called "singular spectrum analysis" of Ref. [2] in order to display clearly the important differences between their methods and the methods of the present paper.

For points selected from a ball of radius r about $\mathbf{p}_0$, we expect that generically the sample average will differ from a suitably unbiased choice for $\tilde{\mathbf{q}}$ by an amount $O(r^2)$, viz.

$$\bar{\mathbf{p}}=\frac{1}{\nu+1}\sum_{i=0}^{\nu}\mathbf{p}_i\cong\bar{\mathbf{q}}+r^2\mathbf{A}\ . \tag{15}$$

Accordingly, we choose $\bar{\mathbf{q}}=\bar{\mathbf{q}}(r)\cong\bar{\mathbf{p}}$.

We close this section with a few remarks regarding alternative formulations of the geometric method. In place of Eq. (9) for the error of approximation of the sample $\mathbf{p}_0,\ldots,\mathbf{p}_\nu$, by the linear space H_k, we might consider an alternative object,

$$\bar{D}(r,H_k)=\frac{1}{\nu+1}\sum_{i=0}^{\nu}d(\mathbf{r}_i,H_k)^2\ , \tag{16}$$

where $\mathbf{r}_i=\mathbf{p}_i-\bar{\mathbf{q}}=r_i\mathbf{x}_i$ in which $r_i=\|\mathbf{r}_i\|$. Mathematically, the foregoing development can be repeated with the result that $\bar{D}(r,H_k)$ attains its minimum value

$$\bar{D}(r,k)=\sum_{j=k+1}^{d}\bar{\lambda}_j(r) \tag{17}$$

for the last $d-k$ eigenvectors $\bar{\mathbf{w}}_{k+1},\ldots,\bar{\mathbf{w}}_d$ of the matrix

$$\bar{X}=\frac{1}{\nu+1}\sum_{i=0}^{\nu}\mathbf{r}_i\mathbf{r}_i^T=\frac{1}{\nu+1}\sum_{i=0}^{\nu}r_i^2\mathbf{x}_i\mathbf{x}_i^T\ , \tag{18}$$

and where $\bar{\lambda}_1(r)\geq\bar{\lambda}_2(r)\geq\cdots\geq\bar{\lambda}_d(r)$ are the corresponding eigenvalues. Except for our use of $\bar{\mathbf{q}}$ as reference point, $\bar{X}$ is a local excursion matrix for the $(\nu+1)\times d$ matrix $\bar{R}^T$ whose rows are the $\mathbf{r}_i^T$ introduced by Broomhead, Jones, and King [8].

The behaviors of $D(r,k)$ and $\bar{D}(r,k)$ as $r\rightarrow 0$ are different: generically, when $m=\dim M'$,

$$\bar{D}(r,k)\rightarrow\begin{cases}O(r)\ , & k<m\\ O(r^2)\ , & k\geq m\ ,\end{cases} \tag{19}$$

while

$$D(r,k)\rightarrow\begin{cases}O(1)\ , & k<m\\ O(r^2)\ , & k\geq m\ .\end{cases} \tag{20}$$

In other words $\bar{D}(r,k)\rightarrow 0$ for any k, while $D(r,k)\rightarrow 0$ only for $k\geq m$, a sharper measure of the error than $\bar{D}(r,H_k)$ (Fig. 2).

Nonetheless, our objective has been to identify a procedure to estimate the tangent spaces to M'. The treatment up to Eq. (14) takes the matrix X and hence $\mathbf{w}_1,\ldots\mathbf{w}_k$ as supplying an optimal basis of $\mathbb{R}^d$ from which to perform the projection into $T_{\mathbf{q}}M'$, while that based on $\bar{X}$ gives $\bar{\mathbf{w}}_1,\ldots,\bar{\mathbf{w}}_d$. The principle difference between X and $\bar{X}$ is the weighting effects provided by the coefficients of the term $\mathbf{x}_i\mathbf{x}_i^T$ in the respective sums in Eqs. (12) and (18). Each of the directions $\mathbf{x}_i$ is weighted equally in X, while directions to points $\mathbf{p}_i$ farthest away from $\bar{\mathbf{q}}$ receive much greater weight in $\bar{X}$ owing to the r_i^2 factor. The resulting eigenvectors will reflect this and the respective candidate tangent space will be tilted toward points receiving larger weights. For the case of $\bar{X}$ we expect the heavy weighting of the farthest points generally to be disadvantageous in that possibly unwanted bias is introduced by curvature effects of M' or random fluctuations when noise is present. It might be thought now that a better candidate matrix would have r_i^{-2} weighting factors since this would emphasize points closest to $\bar{\mathbf{q}}$. Here, however, one might as well choose a smaller value of ν and uniform weights, i.e., a corresponding X for a smaller value of ν. We conclude that the eigenvectors of X provide a reasonable and most natural basis for projection to $T_{\mathbf{q}}M'$.

IV. CONSTRUCTION OF THE PHYSICAL REPLACEMENT TIME SERIES

The noise-reduction procedure actually begins by averaging the data state vectors $\mathbf{p}(t)$ for the noisy time series $v(t)$ with their projections into the k-dimensional linear subspaces $\mathcal{H}_k$ of $T_{\bar{\mathbf{p}}}\mathbb{R}^d$ [Eq. (30)]. This results in a candidate replacement data state vector time series $\hat{\mathbf{p}}(t)$. In general, however, this data state vector time series is *unphysical* in the following sense: except for extremely rare accidents, $\hat{\mathbf{p}}(t)$ cannot be realized by a delay coordinate construction from a scalar replacement time series $\hat{v}(t)$. We recall Eq. (4) in Sec. II, which expresses the construction of the dynamical quantity which the noisy measurements $v(t)$ represent, namely, the coordinate function $S:M\rightarrow\mathbb{R}$ mediating the embeddings of Theorem 1. S represents the measurement process (see Fig. 1). This scalar function S is the only link between the observations and the dynamics of f on M, so we will impose a requirement that a replacement data state vector time series in $\mathbb{R}^d$ be realizable as a delay coordinate construction from some replacement scalar time series $\hat{v}(t)$. We refer to this as a *physical* requirement and such time series as physical replacement time series (PRTS).

Accordingly, the next step in the noise-reduction procedure is to construct a suitable physical replacement time series from a given candidate replacement data state vector time series, realized under some noise-reduction projection protocol. To determine the best $\hat{v}(t)$ we minimize the error committed relative to $\hat{\mathbf{p}}(t)$, viz.

$$\mathcal{E}=\sum_{t=1}^{N_D-(d-1)\Delta}\sum_{j=1}^{d}\{\hat{p}_j(t)-\hat{v}(t+(j-1)\Delta)\}^2\ , \tag{21}$$

where $\hat{p}_j(t)$ is the jth component of $\hat{\mathbf{p}}(t)$, and where we have chosen a definition for $\mathcal{E}$ appropriate to the Euclidean norm of $\mathbb{R}^d$ and root-mean-square (rms) l^2 norm for functions of t. For simplicity, we have again chosen $\tau_s=1$ in Eq. (21), but the algorithm does not have this restriction. We also assume N_D is greater than $(d-1)\Delta$. We regard $\mathcal{E}$ as a function of N_D independent variables $\hat{v}(1),\ldots,\hat{v}(N_D)$, whose minimizing values we seek. For the trivial case $d=1$, Eq. (21) reads

$$\mathcal{E}=\sum_{t=1}^{N_D}|\hat{p}_1(t)-\hat{v}(t)|^2\ , \tag{22}$$

with the obvious solution $\hat{v}(t)=\hat{p}_1(t)$, $1\leq t\leq N_D$. For $d=2$, we have

$$\begin{aligned}\mathcal{E} &= \sum_{t=1}^{N_D-\Delta} |\hat{p}_1(t)-\hat{v}(t)|^2 + \sum_{t=1}^{N_D-\Delta} |\hat{p}_2(t)-\hat{v}(t+\Delta)|^2 \\ &= \sum_{t=1}^{N_D-\Delta} |\hat{p}_1(t)-\hat{v}(t)|^2 + \sum_{t=1+\Delta}^{N_D} |\hat{p}_2(t-\Delta)-\hat{v}(t)|^2 \qquad (23) \\ &= \sum_{t=1}^{\Delta} |\hat{p}_1(t)-\hat{v}(t)|^2 \\ &\quad + \sum_{t=1+\Delta}^{N_D-\Delta} \sum_{j=1}^{2} |\hat{p}_j(t-(j-1)\Delta)-\hat{v}(t)|^2 \\ &\quad + \sum_{t=1+N_D-\Delta}^{N_D} |\hat{p}_2(t-\Delta)-\hat{v}(t)|^2 \ .\end{aligned}$$

One easily finds

$$\hat{v}(t) = \begin{cases} \hat{p}_1(t) \ , \quad 1 \le t \le \Delta \\ \frac{1}{2} \sum_{j=1}^{2} \hat{p}_j(t-(j-1)\Delta) \ , \quad 1+\Delta \le t \le N_D - \Delta \\ \hat{p}_2(t-\Delta) \ , \quad 1+N_D-\Delta \le t \le N_D \ . \end{cases} \tag{24}$$

The formulation of the general case can be constructed by inspired scrutiny from these examples. The analysis goes as follows. We write $\mathcal{E}$ as

$$\mathcal{E} = \sum_{k=1}^{d-1} E_k^{(i)} + E + \sum_{k=d-1}^{1} E_k^{(f)} \ , \tag{25}$$

where

$$E_k^{(i)} = \sum_{t=1+(k-1)\Delta}^{k\Delta} \sum_{j=1}^{k} |\hat{p}_j(t-(j-1)\Delta)-\hat{v}(t)|^2 \ , \qquad k=1,\ldots,d-1 \ ,$$

$$E = \sum_{t=1+(d-1)\Delta}^{N_D-(d-1)\Delta} \sum_{j=1}^{d} |\hat{p}_j(t-(j-1)\Delta)-\hat{v}(t)|^2 \ , \tag{26}$$

$$E_k^{(f)} = \sum_{t=1+N_D-k\Delta}^{N_D-(k-1)\Delta} \sum_{j=d-k+1}^{d} |\hat{p}_j(t-(j-1)\Delta)-\hat{v}(t)|^2 \ , \qquad k=d-1,\ldots,1 \ .$$

Let

$$\Sigma(t) = \begin{cases} \sigma_k^{(i)}(t) = \frac{1}{k} \sum_{j=1}^{k} \hat{p}_j(t-(j-1)\Delta) \ , \\ \qquad 1+(k-1)\Delta \le t \le k\Delta \ , \quad k=1,\ldots,d-1 \ ; \\ \sigma(t) = \frac{1}{d} \sum_{j=1}^{d} \hat{p}_j(t-(j-1)\Delta) \ , \\ \qquad 1+(d-1)\Delta \le t \le N_D-(d-1)\Delta \ ; \\ \sigma_k^{(f)}(t) = \frac{1}{k} \sum_{j=d-k+1}^{d} \hat{p}_j(t-(j-1)\Delta) \ , \\ 1+N_D-k\Delta \le t \le N_D-(k-1)\Delta \ , \\ \qquad k=d-1,\ldots,1 \ ; \end{cases} \tag{27}$$

then

$$\hat{v}(t) = \Sigma(t) \tag{28}$$

minimizes $\mathcal{E}$. $\hat{v}(t)$ is thus realized as the *unique* physical replacement scalar time series [12]. The delay coordinate construction from $\hat{v}(t)$ gives the corresponding physical replacement data state vector time series, which is the closest possible to that of $\hat{\mathbf{p}}(t)$ by the criterion of minimum $\mathcal{E}$.

V. ALGORITHM FOR NOISE REDUCTION BY PROJECTION

The method proceeds through three stages.

Stage 1. Implement a covering procedure

Step 1. Select a random set of N_c candidate reference points $\mathbf{p}_0^{(\alpha)}$ from among the $\mathbf{p}(t), t=1,\ldots,N$, $N=N_D-(d-1)\Delta$. For each $\mathbf{p}_0^{(\alpha)}$, $\alpha=1,\ldots,N_c$, in turn, find its ν nearest neighbors and form its neighborhood $U^{(\alpha)}=\{\mathbf{p}_0^{(\alpha)},\mathbf{p}_1^{(\alpha)},\ldots,\mathbf{p}_\nu^{(\alpha)}\}$. The $\mathbf{p}_0^{(\alpha)}$ are chosen randomly subject to the condition that $\mathbf{p}_0^{(\alpha)} \notin U^{(\beta)}$ when $\alpha \neq \beta$. We shall take N_c to be the smallest number such that the neighborhoods $U^{(\alpha)}$ cover the entire trajectory $\mathbf{p}(t)$.

Step 2. For each cover element $U^{(\alpha)}$, evaluate $\tilde{\mathbf{q}}=\bar{\mathbf{p}}^{(\alpha)}$, the sample average, and $r_\alpha = \max_{0\le i\le\nu}\|\tilde{\mathbf{q}}-\mathbf{p}_i^{(\alpha)}\|$, then order the cover elements according to the value of r_α^{-1}. This gives an ordered array

$$\mathcal{M}_c = \{(\mathbf{p}_0^{(1)},U^{(1)}),\ldots,(\mathbf{p}_0^{(N_c)},U^{(N_c)})\} \tag{29}$$

such that $r_1^{-1} \ge r_2^{-1} \ge \cdots \ge r_{N_c}^{-1}$. This order is chosen to correspond roughly, when the noise level is not too high, to that of the values of $\mu'(U^{(\alpha)} \cap \Lambda')$, where μ' is an assumed invariant measure for the dynamical system on the attractor Λ'. We call $\mathcal{M}_c$ the measure-ordered cover.

Stage 2. Implement a projection procedure

Step 3. Starting from the first neighborhood of the cover $(\mathbf{p}_0^{(1)},U^{(1)})$, estimate the corresponding $\tilde{\mathbf{q}} \cong \bar{\mathbf{p}}$ as in Eq. (15).

Step 4. Relative to the reference point $\tilde{\mathbf{q}}$ form the matrix X according to Eq. (12) from points $\mathbf{p}_0^{(1)},\mathbf{p}_1^{(1)},\ldots,\mathbf{p}_\nu^{(1)}$. Compute the eigenvectors $\mathbf{w}_1,\ldots,\mathbf{w}_d$ corresponding to the eigenvalues $\lambda_1 \ge \cdots \ge \lambda_d$ of X.

Step 5. Let $\mathcal{H}_k$ be the linear space $\langle \mathbf{w}_1,\ldots,\mathbf{w}_k \rangle$ spanned by $\mathbf{w}_1,\ldots,\mathbf{w}_k$, $k<d$. Denote by Q_k the projection operator from $\mathbb{R}^d$ onto $\mathcal{H}_k$. Replace the points $\mathbf{p}_j^{(1)} \in U^{(1)}$ by a new collection:

$$\mathbf{p}_j^{(1)} \to \hat{\mathbf{p}}_j^{(1)} = \tilde{\mathbf{q}} + f(\mathbf{p}_j^{(1)}-\tilde{\mathbf{q}}) + (1-f)Q_k(\mathbf{p}_j^{(1)}-\tilde{\mathbf{q}}) \ , \qquad 0 \le f < 1 \ , j=0,1,\ldots,\nu \ . \tag{30}$$

In our calculations, we take $f=0.5$. We are averaging the projection points with the original data state points by means of the factor f to take account in a rough way of the local curvature effects of M' [13].

Step 6. Repeat Steps 3–5 for each succeeding point $\mathbf{p}_0^{(\alpha)}$

in the order of the measure-ordered set $\mathcal{M}_c$ until $\mathcal{M}_c$ is exhausted, but with one difference. There will be almost surely a first value $\alpha=\alpha_1$ at which $U^{(\alpha_1)}$ has nonempty intersection with $\cup_{\alpha=1}^{\alpha_1-1}U^{(\alpha)}$. In Steps 3 and 4, the $\nu+1$ points from which $\bar{\mathbf{q}}$ is estimated and X is computed therefore will contain some points that already have been projected. Points previously projected are not projected again in Step 5.

Step 7. Denote the data state vector time series resulting from Steps 1–6 by $\hat{\mathbf{p}}(t), t=1,2,\ldots,N_D-(d-1)\Delta$. Compute the corresponding scalar PRTS $\hat{v}(t)$, $t=1,2,\ldots,N_D$ from Eqs. (27) and (28).

Stage 3. Carry out an iteration protocol

For a data state vector time series of given length [14] $N=N_D-(d-1)\Delta$ and a given noise amplitude $\mathcal{N}$ (see below), the basic parameters of the algorithm, from the standpoint of iterating Stage 1 and Stage 2, are the following: d, the embedding trial dimension; k, the projection dimension for the local regions of M'; Δ, the delay, which is the principle control on the embedding itself; τ_s, the algorithm sampling time which is now the constant interval with which the *given* [15] time series is sampled; and ν, the number of points nearest neighbor to $\mathbf{p}_0$ in each neighborhood.

Step 8. Replace the original times series $v(t)$ by $\hat{v}(t)$, re-embed and repeat Steps 1–7 for some choice of d, k, ν, Δ, and τ_s. Carry this out repeatedly for some choice or schedule of d, k, ν, Δ, and τ_s.

We give expressions for the noise remaining after n iterations of the algorithm. We write the scalar time series upon which the projection algorithm is applied as

$$v(t)=V(t)+\epsilon\eta(t)\ , \tag{31}$$

and the scalar PRTS after a single run as

$$\hat{v}(t)=V(t)+\epsilon\hat{\eta}(t)\ . \tag{32}$$

We assume that in the time series originally given $v(t)=v_1(t)$; the noise term $\eta(t)=\eta_1(t)$ has zero mean and unit variance; and ϵ is the strength of the initial noise. In practice, the initial relative noise amplitude $\mathcal{N}=\epsilon/||V||$, where $||V||$ denotes the (rms) l^2 norm of $V(t)$, $t=1,\ldots,N_D$, corresponds to the initial signal-to-noise ratio $\mathcal{S}_i$ directly. For example, a relative noise size of $\mathcal{N}=1\%$ corresponds to $\mathcal{S}_i=40$ dB and 100% corresponds to 0 dB.

Precisely, the change in the signal-to-noise ratio (SNR) produced by an iteration of the algorithm by Eqs. (31) and (32) is

$$\delta\mathcal{S}=\mathcal{S}_f-\mathcal{S}_i=10\log_{10}\left[\frac{\langle\eta^2(t)\rangle}{\langle\hat{\eta}^2(t)\rangle}\right]\equiv 10\log_{10}R\ , \tag{33}$$

where the angular brackets refer to time average and R designates an "improvement ratio." In the iteration procedure the PRTS for the nth iteration is the initial time series for the $(n+1)$th iteration. We write

$$\begin{aligned} v_n(t)&=V(t)+\epsilon\eta_n(t)\ ,\\ \hat{v}_n(t)&=V(t)+\epsilon\hat{\eta}_n(t)\ , \end{aligned} \tag{34}$$

so that $\hat{v}_n(t)=v_{n+1}(t)$; also $\hat{\eta}_n(t)=\eta_{n+1}(t)$. The SNR change from n iterations is

$$\delta_n=10\log_{10}\prod_{i=1}^{n}R_i\ , \tag{35}$$

where the improvement ratio from the ith iteration is

$$R_i=\langle\eta_i^2(t)\rangle/\langle\hat{\eta}_i^2(t)\rangle\ ,\quad i=1,\ldots,n \tag{36}$$

in which $\eta_1(t)$ is the initial noise time series. Since $\langle\eta_1^2(t)\rangle=1$ we also have

$$\delta_n=10\log_{10}\frac{1}{\langle\hat{\eta}_n^2(t)\rangle}\ ,\quad n\geq 1\ . \tag{37}$$

As long as $R_n>1$ continues to hold, $\hat{\eta}_n\to 0$ in the sense of an l^2 norm (rms). A change in sign of $\log_{10}R_n$ signals an onset of SNR performance degradation.

VI. SYSTEMATIC STUDY OF ALGORITHM PERFORMANCE FOR MAPS

We concentrate on the Hénon map [16],

$$\begin{aligned} x&\to x'=a-x^2+by\ ,\\ y&\to y'=x\ , \end{aligned} \tag{38}$$

with more or less standard parameter choice; $a=1.42$, $b=0.3$; at the end we state a couple of results for the Ikeda map. We assume $\eta_1(t)$ to be uncorrelated Gaussian white noise. We take $\Delta=1$ and $\tau_s=1$ in this section. For the studies we report in this section, we generated two long orbit time series, one on two Sun 4 computers and one on two Sparc stations. We selected an $N=3000$ or $N=10\,000$ portion of each of these for our computations. For a fixed $\mathcal{N}$, we generated a Gaussian white-noise time series [17] of appropriate variance to be added to the orbit time series. The initial noisy time series was then held fixed for each combination of N and $\mathcal{N}$. Thus for a given noisy time series, i.e., fixed N and $\mathcal{N}$, the parameters of the algorithm are (d,k,ν).

A. Initial logarithmic growth of SNR improvement

The behavior of the algorithm under iteration, for a range of d and fixed choice of k,ν, is systematic and stable [Fig. 3(a)]. The improvement in SNR typically first rises with each iteration. For a few typical examples we have examined, this initial rise is approximately logarithmic [Fig. 3(b)]. In all cases we have studied where the initial rise occurs, the graph goes through a simple maximum,

$$\delta_n=\delta_{n_M}=\delta_{\max}\equiv\delta_M\ , \tag{39}$$

after $n=n_M$ iterations (except for occasional very low amplitude oscillations near the peak), and then it falls. δ_M and n_M will be featured in most of the following studies. Figure 4 shows typical examples.

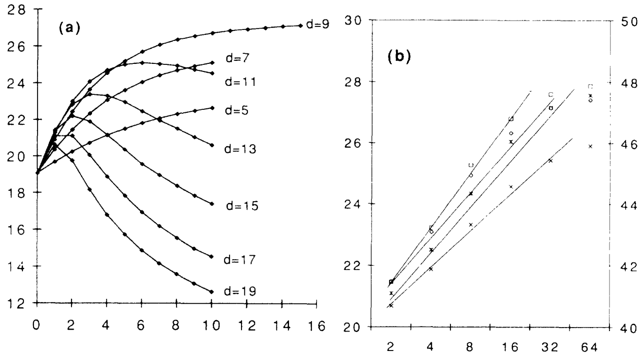

FIG. 3. (a) Typical variation of $\mathcal{S}$ (dB) with number of iterations n of the projection algorithm. The examples shown are Hénon runs for $N=3000$, $\mathcal{N}=10\%$ and $(k,v)=(3,20)$ with various d. (b) Examples of the initial logarithmic change of $\mathcal{S}$(dB) for several runs of Hénon data with varied conditions. ×: $N=3000$, $\mathcal{N}=1\%$, $(d,k,v)=(9,4,10)$; $*$: $N=3000$, $\mathcal{N}=10\%$, $(d,k,v)=(9,3,10)$; ◇: $N=10\,000$, $\mathcal{N}=1\%$, $(d,k,v)=(8,4,20)$; □: $N=10\,000$, $\mathcal{N}=10\%$, $(d,k,v)=(8,3,20)$.

B. Statistical fluctuations

Measured values of δ_M and n_M fluctuate slightly with noise and orbit realizations. There are also small variations with choice of $\{\mathbf{p}_0^\alpha\}$, the centers of the cover for the noisy attractor, which we varied randomly for every calculation. Here, in ten 3000, 1%, $(d,k,v)=(7,3,20)$ runs, keeping noise and orbit realizations fixed, we found the SNR improvement after 15 iterates to be (6.56±0.09) dB. Incidentally, we also computed the SNR for the time series found by averaging the cleaned-up orbits from the ten runs. The result was 45.99 dB, for a 6.80-dB improvement, only slightly better than the mean improvement of 6.56 dB. We conclude that variations in random selections of the $\mathbf{p}_0^\alpha$ do not affect algorithm performance significantly.

We estimated the statistical scatter for δ_M for fixed (d,k,v) by applying the algorithm to each of ten different noisy orbit realizations. The ten realizations were generated by ten different Hénon orbits with ten, also independently generated, white-noise time series added. The error flag in Fig. 4(a) was obtained from these ten different realizations. As stated earlier, the $\{\mathbf{p}_0^\alpha\}$ choice variations are smaller but always present as well. The initial SNR for the ten realizations was (40.02±0.06) dB. That for the computation results shown in the figure was 39.07 dB.

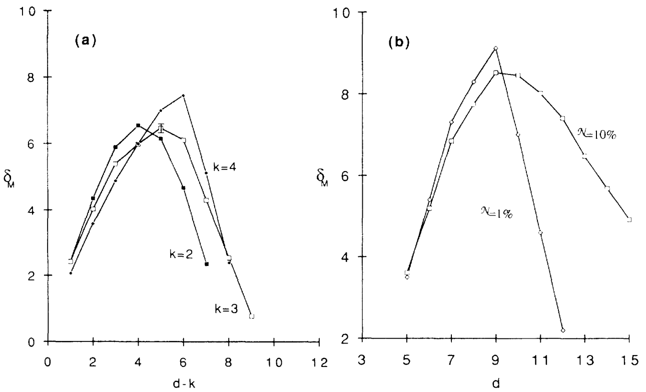

FIG. 4. (a) δ_M (dB) vs $d-k$ plots for Hénon runs with $N=3000$, $\mathcal{N}=1\%$ runs, and $v=10$. The error flag indicates a statistical fluctuation of ±0.3 dB due to different orbit and noise realizations. (b) Example of the widening of δ_M(dB) vs d plots as initial noise level $\mathcal{N}$ increases. The examples shown are Hénon for $N=10\,000$ and $(k,v)=(4,40)$.

TABLE I. Peak algorithm performance as a function of (k,v): $(d_{\rm pk})(\delta_M,n_M)$, with δ_M in dB. Hénon system, $N=3000$, $\mathcal{N}=1\%$. Largest peak δ_M is in boldface.

k	$v=10$	$v=20$	$v=30$	$v=40$
2	(6)(5.6,13)	(5)(6.1,7)	(5)(6.0,5)	(4)(5.5,6)
3	(8)(6.5,26)	(7)(7.1,12)	(7)(7.1,8)	(7)(6.4,7)
4	(10)(7.4,46)	(8)(**7.5**,46)	(7)(7.3,40)	(8)(6.7,25)

C. Performance dependence on *d*

For given N, $\mathcal{N}$, k, and v, the location of the peak value δ_M depends on d [Fig. 3(a)]. For low d, the curves for various N, $\mathcal{N}$, k, and v combinations go through very broad maxima after many iterations before turning around, while for higher d the peaks are sharper, lower, and occur much earlier. If d is sufficiently high, the first iteration yields a negative improvement, and with each successive iteration the resulting SNR continues to drop. We shall focus our study on d values low enough that this does not occur. Hence we may examine the dependence of δ_M vs d on the choice of (k,v) for each condition of the specified N and $\mathcal{N}$ values.

D. Systematic behavior

For $(N,\mathcal{N})=(3000,1\%)$ and $(N,\mathcal{N})=(3000,10\%)$, we performed systematic and complete series of computations of (δ_M,n_M) for all combinations of $v=10,20,30,40$ and $k=2,3,4$ (Tables I and II). We did the same for the more time-consuming $N=10\,000$ trajectory case, in the following combinations: $\mathcal{N}=1\%$ and 10% with $k=3$ and $v=20,40,60$, and with $v=40$ and $k=2,3,4$, and $\mathcal{N}=1\%$ with $v=20$ and $k=2,3,4,5$ (Tables III and IV). In all cases we began with $d=k+1$, continued until we reached a peak value, at $d=d_{\rm pk}$, of δ_M, and then continued beyond until performance degradation had clearly set in. We observed very systematic behavior of both δ_M and n_M with variation of N, $\mathcal{N}$, d, k, and v.

TABLE II. Peak algorithm performance as a function of (k,v): $(d_{\rm pk})(\delta_M,n_M)$, with δ_M in dB. Hénon system, $N=3000$, $\mathcal{N}=10\%$. Largest peak δ_M is in boldface.

k	$v=10$	$v=20$	$v=30$	$v=40$
2	(8)(8.9,29)	(6)(**9.4**,26)	(6)(9.1,13)	(5)(9.0,18)
3	(9)(8.5,84)	(8)(8.3,36)	(8)(7.8,23)	(7)(7.4,28)
4	(10)(7.7, > 100)	(10)(7.7,50)	(10)(7.0,16)	(10)(6.3,14)

TABLE III. Peak algorithm performance as a function of (k,ν): $(d_{\text{pk}})(\delta_M,n_M)$, with δ_M in dB. Hénon system, $N=10\,000$, $\mathcal{N}=1\%$. Largest peak δ_M is in boldface.

k	$\nu=20$	$\nu=40$	$\nu=60$
2	(7)(9.8,11)	(6)(8.5,8)	
3	(9)(**9.9**,18)	(8)(9.2,15)	(7)(8.6,14)
4	(10)(9.8,49)	(9)(9.1,40)	
5	(11)(9.1,116)		

E. δ_M vs d

For fixed k, δ_M first rises with increasing d then falls after reaching a peak as the examples in Fig. 5 show. For the higher noise level or longer trajectory, the plots are shifted upward and to the right. It takes larger d values to achieve peak performance but peaks are also higher. For the higher noise, not only are the plots generally higher, they are also broader [Fig. 4(b)]. We remark also that higher peaks resulted from lower ν values, especially for higher noise cases.

F. δ_M vs $d-k$

Since the main step of this algorithm is a projection from dimension d into k, one might expect that the performance depend uniformly on $d-k$. Plots of δ_M for fixed ν against $d-k$ do indicate a dependence of δ_M on $d-k$ whose form is somewhat independent of the values of k and ν [Figs. 4(a) and 6]. We remark that the plots of δ_M are higher for low k when $\mathcal{N}=10\%$, while the situation is reversed for $\mathcal{N}=1\%$. The effect is most pronounced for $\mathcal{N}=3000$, and somewhat less so for 10 000, 1% cases.

G. n_M vs d

The number of iterations needed to reach maximum SNR improvement, δ_M, was varied over a wide range with N, $\mathcal{N}$, d, k, and ν (Fig. 7). For fixed (d,k), n_M is nearly always larger for lower ν values for every combination of $(N,\mathcal{N})$. In addition, plots of n_M vs d lie higher for higher k and higher N and $\mathcal{N}$.

H. Ikeda map, $N=10\,000$

The Hénon map has high dissipation since the Jacobian determinant has magnitude $b=0.3$. Correspondingly, the dimension of the Hénon attractor is low. On the other hand, the Ikeda map [18]

TABLE IV. Peak algorithm performance as a function of (k,ν): $(d_{\text{pk}})(\delta_M,n_M)$, with δ_M in dB. Hénon system, $N=10\,000$, $\mathcal{N}=10\%$. Largest peak δ_M is in boldface.

k	$\nu=20$	$\nu=40$	$\nu=60$
2		(8)(**10.3**,13)	
3	(10)(10.0,42)	(8)(9.5,31)	(8)(8.8,30)
4		(9)(8.4,60)	

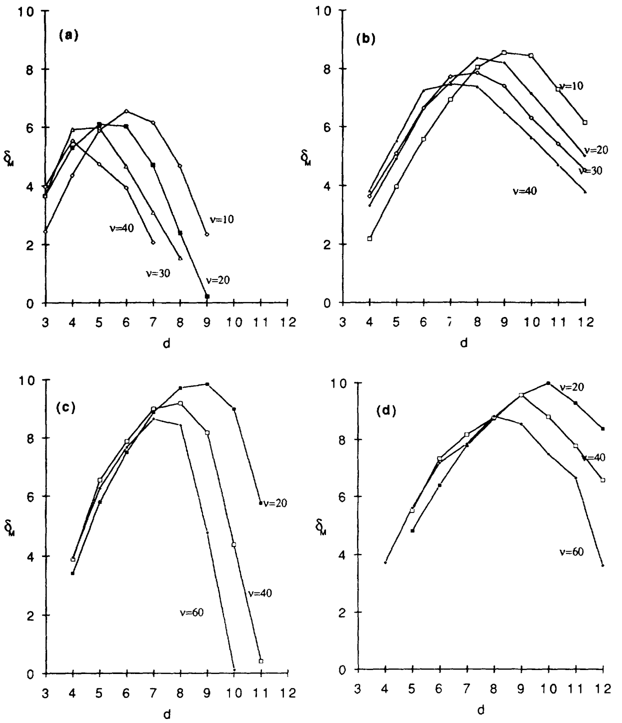

FIG. 5. δ_M(dB) vs d plots for Hénon runs: (a) $N=3000$, $\mathcal{N}=1\%$, $k=2$; (b) $N=3000$, $\mathcal{N}=10\%$, $k=3$; (c) $N=10\,000$, $\mathcal{N}=1\%$, $k=3$; (d) $N=10\,000$, $\mathcal{N}=10\%$, $k=3$. Each contains plots that correspond to different ν values.

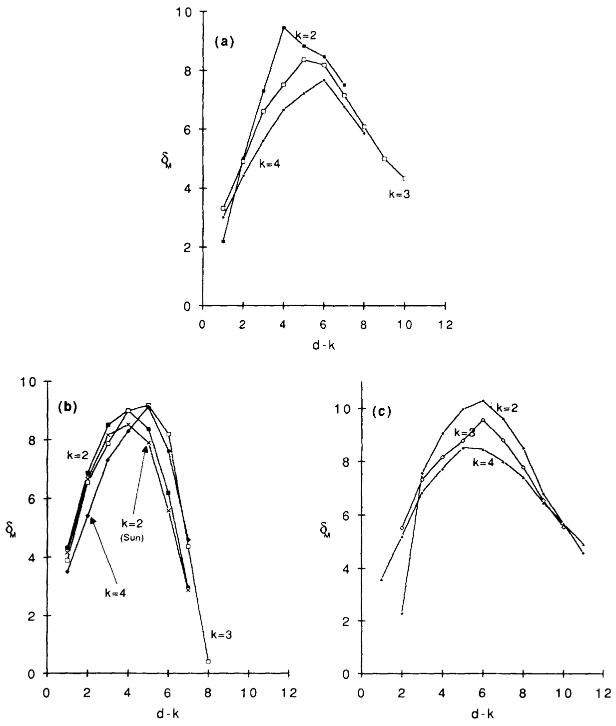

FIG. 6. δ_M(dB) vs $d-k$ plots for Hénon runs: (a) $N=3000$, $\mathcal{N}=10\%$, $\nu=20$; (b) $N=10\,000$, $\mathcal{N}=1\%$, $\nu=40$; (c) $N=10\,000$, $\mathcal{N}=10\%$, $\nu=40$. Each contains plots that correspond to different k values.

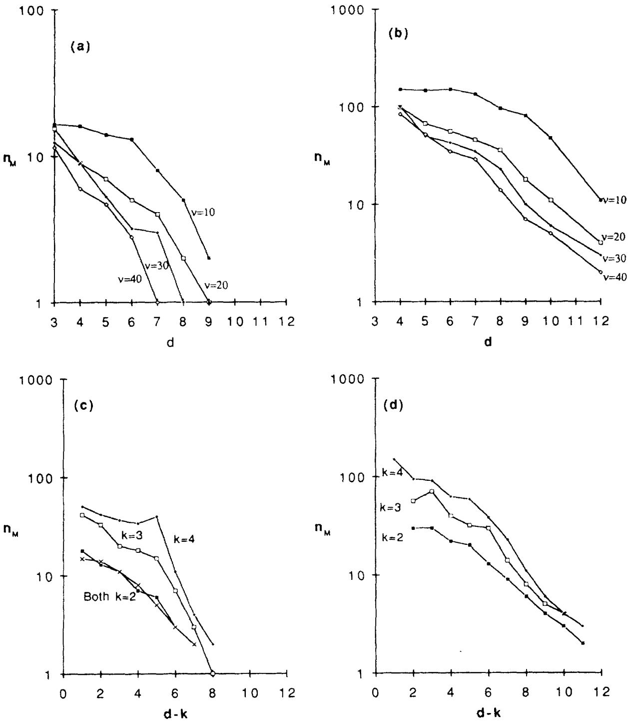

FIG. 7. n_M vs d plots for Hénon runs: (a) $N=3000$, $\mathcal{N}=1\%$, $k=2$; (b) $N=3000$, $\mathcal{N}=10\%$, $k=3$. n_M vs $d-k$ plots for Hénon runs: (c) $N=10\,000$, $\mathcal{N}=1\%$, $v=40$; (d) $N=10\,000$, $\mathcal{N}=10\%$, $v=40$.

$$z'=A+Bz\exp i[k-p(1+|z|^2)^{-1}]\ ,\quad z\in\mathbb{C} \tag{40}$$

for $A=0.85$, $B=0.9$, $k=0.4$, and $p=6.0$, is relatively weakly dissipative since its Jacobian determinant is $B^2=0.81$. Its attractor has somewhat higher dimension and has very different geometry. To see the effect of system dependence we ran a few calculations for this system. The plots in Fig. 8 show the kind of behavior we should now expect for δ_M, although the $\mathcal{N}=1\%$ cases give noticeably lower δ_M values than we found for the Hénon map. We discuss reasons for this in Sec. VIII when we consider low noise limits of useful applicability of the present method.

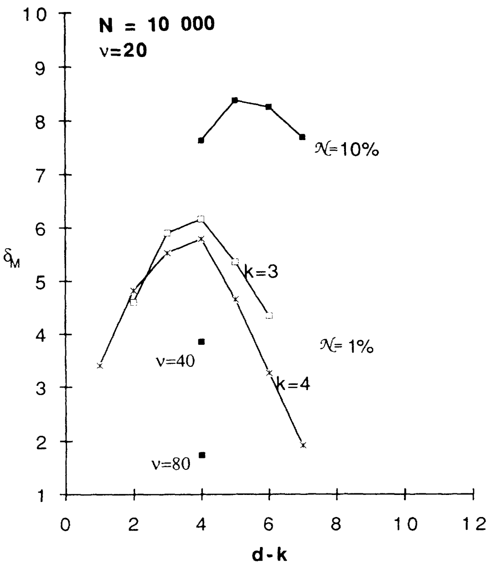

FIG. 8. δ_M (dB) vs d plots for the Ikeda map for $v=20$. The 10% noise case has $k=3$. Isolated runs with other v values for $k=3$ are also indicated in the plot.

VII. ALGORITHM PERFORMANCE FOR FLOWS: PRELIMINARIES

We present results of analysis of numerically generated flow data in this section, focusing principally on very finely sampled time series for benchmarking purposes. The noise time series again is an uncorrelated, unfiltered Gaussian white-noise process. Since the noise amplitude at time t_i is added to the clean time series value $V(t)$ for each $1\le i\le N_D$, there is a considerable spectral mismatch between the two, a setting in which any standard low-pass filter will give a good result. We give results of calculations for the more important, coarsely sampled case, in the next section, where also the context is that of high noise. We first fix attention on the Lorenz system [19],

$$\begin{aligned}&\dot{x}=\sigma(y-x)\ ,\\&\dot{y}=\rho x-y-xz\ ,\quad (x,y,z)\in\mathbb{R}^3\\&\dot{z}=-\beta z+xy\ ,\end{aligned} \tag{41}$$

taking the usual parameter values $\rho=28$, $\beta=\frac{8}{3}$, and $\sigma=10$. We give a few results at the end of the section for the Rössler system. We concentrate on $N=10\,000$, $\mathcal{N}=10\%$ studies, but we also include some $N=3000$ and $30\,000$ and $\mathcal{N}=1\%$ results.

The noise-free Lorenz time series that we work with represent the x coordinate of the solution of Eq. (41). The solution for the first was obtained from a fourth-order Runge-Kutta method with a fine integration time step $h_0=0.001$. The first 20 000 points were discarded as transient behavior from a randomly chosen initial condi-

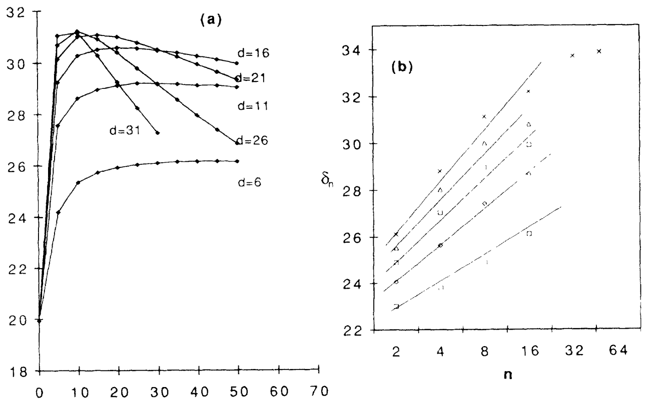

FIG. 9. (a) Typical variation of $\mathcal{S}$(dB) with number of iterations n of the projection algorithm. The examples shown here are Lorenz 3 ($\Delta T=0.05$) runs, $N=10\,000$, $\mathcal{N}=10\%$, and $(k,v)=(3,40)$ with various d. (b) Examples of initial logarithmic change of $\mathcal{S}$(dB) for Lorenz 0 runs, with again $N=10\,000$, $\mathcal{N}=10\%$, and $(k,v)=(3,40)$. The embedding dimensions are $d=5$ (□), 7 (◇), 9 (□), 11 (△), and 15 (×).

TABLE V. Properties of Lorenz time series.

Time series	τ_s	ΔT	Approx. no. of points per cycle
Lorenz 1	1	0.005	175
Lorenz 2	5	0.025	35
Lorenz 3	10	0.05	17.5
Lorenz 4	25	0.125	7
Lorenz 5	50	0.25	3.5
Lorenz 6	100	0.5	1.75

tion. The resulting very finely sampled time series of length $N=10\,000$ covers only about 11 oscillations, or approximately 900 points per cycle. We call this time series Lorenz 0. The other time series we use in this section and the next are produced by sampling from a very long second solution having integration time step $h_1=dt=0.005$. These time series, Lorenz 1 to Lorenz 6, have physical sampling intervals $\tau_s dt=\Delta T$, and τ_s and other parameters as shown in Table V.

The SNR improvement δ_n shows the same kind of behavior described in Sec. VI for the Hénon map cases [Fig. 9(a)]. We observe a roughly logarithmic increase prior to maximum for each fixed d [Fig. 9(b)]. For the rest of the section, we take $(d,k,v)=(15,3,40)$.

For the Lorenz 0 time series, the largest δ_M occurs for $\Delta=1$, and δ_M falls very slowly out to $\Delta=140$ (Table VI). On the other hand, the dependence on delay is much more pronounced for the less finely sampled Lorenz 2 and Lorenz 5 time series (Table VII). Thus, for the Lorenz 2 time series the time interval represented by delay 1 is 25 times that for Table VI and the number of points per typical oscillation is about 35. For the Lorenz 5 time series the latter number has fallen to 3.5.

Returning to the Lorenz 0 study, a few sample calculations for $N=3000$, 10 000, and 30 000 for $\mathcal{N}=10\%$ indicate that there is no significant difference in the values of δ_M (Table VIII).

TABLE VI. Algorithm performance as a function of delay Δ for the Lorenz 0 time series. $N=10\,000$, $\mathcal{N}=10\%$ (22 dB), $(d,k,v)=(15,3,40)$, $h_0=\Delta T=0.001$.

Δ	δ_M (dB)
1	10.85
10	10.47
20	9.82
30	10.41
40	9.76
50	9.73
60	9.69
70	9.65
100	9.61
110	9.38
120	9.20
130	9.11
140	9.19

TABLE VII. Algorithm performance as a function of delay Δ for the Lorenz 2 and Lorenz 5 time series. $N=10\,000$, $\mathcal{N}=10\%$ (22 dB), $(d,k,v)=(15,3,40)$, $dt=0.005$: first column, $\tau_s=5$, $\Delta T=0.025$; second column, $\tau_s=50$, $\Delta T=0.25$. $\delta_M<0$ means immediate ($n=1$) SNR degradation.

Δ	δ_M (dB)	δ_M (dB)
1	10.31	6.31
2	10.78	0.02
3	9.94	−2.3
4	9.33	−3.4
10	7.44	−5.0
20	2.16	−5.3
30	1.34	−5.6

Just as for the map studies of Sec. VI, the improvement in SNR saturates in all the calculations presented above for every delay chosen. Different delays lead to different embeddings, and it might be hoped that if we follow up a series of iterations using one delay with another series having a new delay, a jump in improvement would result. As shown by several calculations of this type in Table IX, improvement jumps are real if modest. As we shall see in Sec. VIII, however, this strategy of changing delays does not seem to work for maps and there is reason to expect it may fail for coarsely sampled flows.

For comparison, we also applied the algorithm to a finely sampled time series obtained from the Rössler system [20]:

$$\begin{aligned}\dot{x}&=-y-z\ ,\\ \dot{y}&=x+ay\ ,\quad (x,y,z)\in\mathbb{R}^3\ ,\\ \dot{z}&=bx-cz+xz\ ,\end{aligned}\tag{42}$$

where we chose Rössler's values for the spiral-type attractor, $a=0.36$, $b=0.4$, and $c=4.5$. A fourth-order Runge-Kutta method was used, with integration time step 0.005. Our results are similar to those for the finely sampled Lorenz case. For an $N=10\,000$, $\mathcal{N}=1\%$ example with $(k,v)=(3,40)$, we found $\delta_M=9.03$ dB for $d=9$, $\delta_M=9.97$ dB for $d=12$, and $\delta_M=10.58$ dB for $d=15$.

We conclude that for well-sampled flow time series, the algorithm performs reasonably well. The sampling interval $\Delta T=\tau_s dt$ is an important parameter for flows. We can understand the issues involved as follows. As ΔT grows, the number of sample points per typical oscillation falls. For ΔT sufficiently large, a flow time series will become sawtooth, resembling the time series for a map. For finely sampled time series, the projection scheme performs some averaging along the flow lines, which mimics low-pass filtering. Thus the effect of spectral mismatch between signal and noise gives us a few-decibel reduction for free. Unlike low-pass filtering, there is also some smoothing here across the flow lines, but probably not as

TABLE VIII. δ_M dependence on trajectory length for the Lorenz 0 time series. $N=10\,000$, $\mathcal{N}=10\%$, $(d,k,v)=(15,3,40)$, $h_0=\Delta T=0.001$.

Δ	$N=3000$	$N=10\,000$	$N=30\,000$
1	11.55	10.85	
30		10.41	10.56

TABLE IX. Algorithm performance under various embedding strategies for the Lorenz 0 time series. δ_M(dB) for $N=10\,000$, $(d,k,v)=(15,3,40)$, $h_0=\Delta T=0.001$. Final SNR are in boldface.

	$\Delta=1\rightarrow10$	$\Delta=30\rightarrow20$	$\Delta=10\rightarrow(5)\rightarrow1$
$\mathcal{N}=0.1\%$	10.30→**12.06**		8.72→(9.44)→**11.59**
$\mathcal{N}=1\%$	10.39→**14.38**	9.79→**11.46**	
$\mathcal{N}=10\%$	10.14→**13.61**	8.94→**10.09**	10.09→**14.55**

much for the finely sampled case as the coarsely sampled. As the sampling rate falls, the Nyquist frequency $1/2\Delta T$ approaches the frequency corresponding to a typical oscillation of the signal. When this happens, the number of points per typical oscillation approaches 2. For coarsely sampled time series, the spectra of signal and noise become more closely matched. Here conventional low-pass techniques become ineffective. The advantages of geometric projection, in which smoothing is performed in phase space, should come into play in these cases since low-pass filtering cannot perform smoothing across flow lines. Long time series, however, may be required to assure adequate recurrence properties. Accordingly, for any given choice of (d,k,v) we might expect lower values of SNR improvement. Therefore to optimize performance, systematic studies like those done for the Hénon map need also to be done for the coarsely sampled case.

VIII. NATURE AND LIMITS OF THE METHOD

In this section we investigate the nature and limits of the method—for both very high and very low noise cases. We examine variations in performance with the choice of delay for the Hénon map, and with physical sampling interval for the very high noise Lorenz case. For the latter case, we give results of comparison with a local averaging of the time series, which simulates a low-pass filter. We display results of effects of application of the algorithm on phase portraits for Hénon and Lorenz examples and for the case of a pure white-noise time series containing no underlying signal [i.e., $V(t)\equiv0$]. We also give a few comparisons of Lyapunov exponent calculations.

A. Very low noise limits of the method (the "20-dB rule")

We begin with a Hénon map study for very low noise amplitude (0.1%,0.01%). We choose $(a,b)=(1.42,0.3)$ as in Sec. VI. Comparing the δ_M vs d plots for noise amplitudes $\mathcal{N}=1\%$ (39 dB), 0.1% (60 dB), and 0.01% (80 dB) indicates that as the noise amplitude comes down, so do the plots in both δ_M values and in d [Fig. 10(a)]. For 80 dB, the method fails to produce significant improvement.

Generally speaking, any scheme of noise reduction will introduce some noise into the data. We refer to the amount of noise produced by application of the method to a clean time series as the *baseline noise*. In Table X, we show results of a single iteration of the algorithm on a noise-free Hénon time series for the same $(k,v)=(2,20)$ and for a range of d values. Thus -91.5 dB is the single-iterate baseline noise for projection from $d=3$ to $k=2$, while -70.5 dB is the single-iterate baseline noise for projection from $d=7$. In the second row of the table we have entered the change in SNR after one iterate of the 80-dB time series. Reading the table, we may characterize the significance of baseline noise by saying, for example, that projecting from $d=7$, the algorithm is essentially unable to distinguish an 80-dB noise time series from a perfectly clean one. Correspondingly, the -91.5 dB for $d=3$ represents an absolute lower limit of noise amplitude ($\sim0.003\%$) for which the algorithm might be able to produce an improvement, while this lower bound is -70.5 dB ($\sim0.03\%$) for $d=7$.

From a practical point of view, however, one would most likely decide from Fig. 10(a) that a noise-reduction effort for $\mathcal{N}=0.01\%$ ($\mathcal{S}_i=80$ dB) is hardly worthwhile. For $\mathcal{N}=0.1\%$ ($\mathcal{S}_i=60$ dB) the situation is better. As we have seen, plots of δ_M against d for fixed k attain peak

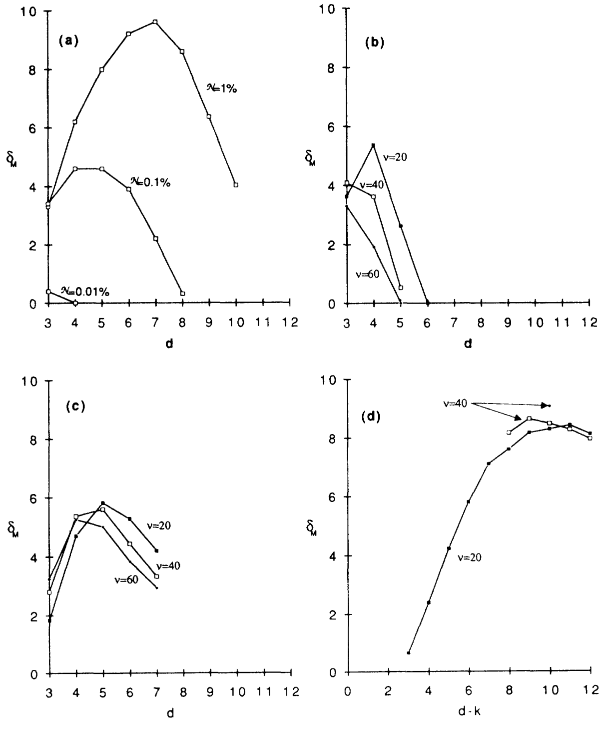

FIG. 10. (a) Comparison of δ_M(dB) vs d plots of Hénon runs for various initial noise levels $\mathcal{N}$. (b) Algorithm performance for delay 2 cases for Hénon: $N=10\,000$, $k=2$ and $\mathcal{N}=1\%$. (c) Algorithm performance for delay 2 cases for Hénon: $N=10\,000$, $k=2$, and $\mathcal{N}=10\%$. (d) Algorithm performance for high noise ($\mathcal{N}=30\%$) Hénon: $N=10\,000$ and $k=2$. The isolated point showing a large reduction was run on a Sun and the rest run on the Sparc stations, hence having different orbit and noise realizations.

TABLE X. Comparison of the effect of one iteration on very low noise (80 dB) time series with baseline noise for the Hénon system. Baseline noise $= -\mathcal{S}_1$ for 0% noise case. $N = 10\,000$, $(k,\nu)=(2,20)$.

d	3	4	5	6	7
$-\mathcal{S}_1$ (clean)	-91.5	-84.0	-81.7	-75.0	-70.5
δ_1 (80.0 dB)	0.4	0.0	-1.6	-5.4	-9.8
Sum	-91.1	-84.0	-83.3	-80.4	-80.3

values for d a little larger than k, how much larger depends on $\mathcal{N}$. This can be seen from Fig. 10(a), also. For $\mathcal{N}=0.1\%$, the peak region has $d \geq 3 = k+1$. From Table X, the baseline noise for projection from $d=4$ or 5 is about -85 to -80 dB, or 20 dB below $\mathcal{S}_i$ for 0.1% (60 dB). For $\mathcal{N}=1\%$, the peak region has $d \geq 5 = k+3$, and the baseline noise for projection from $d=6$ or 7 is -75 to -70 dB, which is 30 dB below $\mathcal{S}_i$ for 1% (40 dB). From these examples we might infer a general, if rather crude, rule of thumb for deciding ahead of time for an experimentally generated time series, say, whether a noise-reduction effort will be worthwhile, provided there is some idea about the noise level in the experimental data. For instance, if a model dynamical system is available for the experiment, then baseline noise data can be calculated for projection from d values a little larger than $k=m$, the topological dimension for the model. This baseline noise value should be, say, at least 20 dB below the lowest practical data noise level for use of the algorithm to be worthwhile. If no model is available, then a baseline noise calculation is not possible, and, of course, this "20-dB rule" cannot be used.

B. Dependence on delay

In Table XI we show baseline noise results for the Hénon map with $\Delta=1$, 2, and 3. For $d=3$, the baseline noise jumps a whopping 16 dB with each increase of delay by 1, and for $d=4$, the jump is 20 dB each time. We can expect performance for delay $\Delta=2$ to be degraded relative to the $\Delta=1$ results of Sec. VI. Thus the 20-dB rule applied to $d=4$ gives a practical lower limit noise level of -40 dB ($\sim 1\%$) while for $d=5$ it gives -30 dB ($\sim 3\%$). The results portrayed in Fig. 10(b) support these arguments. Going to $\mathcal{N}=10\%$ we (again) get better performance as expected [Fig. 10(c)]. The trend of the plots from Fig. 10(b) to Fig. 10(c) is the same as that for the $\mathcal{N}=0.01\%$ plot to the $\mathcal{N}=1\%$ plot in Fig. 10(a).

We have performed single iterate baseline noise calculations for the Ikeda map and Lorenz system also. Comparing Table X and the first line of Table XII, we see a striking difference between the corresponding Hénon and Ikeda map baseline noise levels. There is a typical 20 to 30 dB difference for every value of d from $d=3$ to 7. If we apply the 20-dB rule to the Ikeda map for $k=2$ we would expect a SNR of 40 dB or so to represent the lowest practical noise level with which the algorithm might successfully deal. The story for projecting into $k=3$ is similar, and the relatively, though not abysmally, poor showing for $\mathcal{N}=1\%$ (Fig. 8) is consistent with this expectation. The baseline noise levels for the Ikeda map for projection into $k=4$ and those for the Hénon map for $k=2$ are relatively close (Tables XI and XII). But this is not enough to conclude that δ_M values will be comparable, as may be seen by comparing the corresponding data depicted in Figs. 6(b) and 8 [21]. A lower baseline noise appears necessary, but not sufficient to assure good performance. Still, we might expect better Ikeda map results for $\mathcal{N}=10\%$, and our results support this (Fig. 8).

For the Lorenz system, baseline noise levels for the Lorenz 0 time series are uniformly very low for a wide range of delays (Table XIII). For Lorenz 1 to Lorenz 6 series, a significant dependence on sampling interval and delay appears (Table XIV). It is interesting to compare the declining performance of the algorithm with increasing sampling time and delay shown in Table VII with the rising levels of baseline noise in succeeding rows and columns of Table XIV. The 20-dB rule does reasonably well once again in signaling the onset of significant degradation of algorithm performance.

C. Very high noise cases for the Hénon map

It may be surprising at first sight to find ourselves predicting better improvement for higher noise levels. But a little reflection reveals the cause. The projection method is actually a noise-processing scheme; the more there is to process the more effective the method can be. Of course, there are limits to this. The SNR improvement saturates with n, and its maximum values go through peaks as functions of d. As long as $\mathcal{N}$ is well above the appropri-

TABLE XI. Baseline noise (dB) dependence on delay for the Hénon system. $N=10\,000$, $\mathcal{N}=0\%$, $(k,\nu)=(2,20)$.

d	3	4	5	6	7
$\Delta=1$	-92	-84	-82	-75	-71
$\Delta=2$	-76	-63	-51	-42	-37
$\Delta=3$	-60	-43	-36	-32	-29

TABLE XII. Baseline noise (dB) dependence on k for the Ikeda system. $N=10\,000$, $\mathcal{N}=0\%$ (baseline noise), $\nu=20$.

$d-k$	1	2	3	4	5
$k=2$	-61	-61	-59	-47	-45
$k=3$	-69	-74	-61	-56	
$k=4$	-90	-76	-70	-69	

TABLE XIII. Baseline noise (dB) for the Lorenz 0 time series. $N=10\,000$, $\mathcal{N}=0\%$, $(d,k,v)=(15,3,40)$, $h_0=\Delta T=0.001$. Two time series realizations.

Δ	1	10	20	30	40	50	60	70	80	90	100	110	120	130	140	150
$-\mathcal{S}_1$	−160	−108	−91	−91	−89	−84	−85	−84			−82	−85	−85	−88	−88	
$-\mathcal{S}_2$	−160	−104	−92	−91	−88			−86	−85	−85	−86	−87	−90	−93	−109	−112

ate baseline noise, these saturation limits do not seem to depend too much on $\mathcal{N}$. The obvious question is, how far can we go with this? How high an $\mathcal{N}$ can this noise-reduction scheme deal with?

Unfortunately run time grows significantly with $\mathcal{N}$, in particular, for the long trajectories (at least 10 000) that seem to be needed for the larger improvements. In addition, the large δ_M values occur for increasingly high d where nearest-neighbor searches are more time consuming. We computed a 30% noise (10 dB) case for the Hénon map for $(k,v)=(2,20)$ and $(2,40)$. The resulting peak δ_M values of 8–9 dB [Fig. 10(d)] are only slightly lower than the peak δ_M values of 9–10 dB for the 10% noise, $k=2$ case [Fig. 6(c)]. This is not enough to conclude a significant loss of performance for $\mathcal{N}=30\%$ relative to $\mathcal{N}=10\%$. We also ran three 100% noise (-1 dB) calculations for $(d,k,v)=(10,2,20)$, $(12,2,20)$, and $(11,2,40)$ and found 5–6-dB improvement. The dimension values chosen were below the peak, so the best δ_M might be larger.

D. Sampling interval dependence for Lorenz time series with very high noise

Again as in Sec. VII, we take $(d,k,v)=(15,3,40)$ except where explicitly otherwise stated. For the very finely sampled Lorenz 0 time series ($\Delta T=0.001$), we see 12–15-dB improvement for 30% (11.6 dB) noise cases (Table XV). For the Lorenz 1 time series ($\Delta T=0.005$) and 100% (0–2 dB) noise cases we see similar improvements (Table XVI). To achieve these results, we have employed a strategy of reembedding by changing delay after a succession of iterations using one embedding. Since baseline noise levels are very low for the delays chosen for the reembeddings (Table XIII and Table XIV), we could expect reasonably good performance for each embedding.

For comparison with standard linear techniques we performed a 100% noise study on Lorenz 1 ($\Delta T=0.005$) and Lorenz 3 ($\Delta T=0.05$) time series, for which the number of points per typical oscillation is 175 and 17.5, respectively. For a linear filter we just performed a very simple sliding local average of nearest neighbor points in the time series, 50 points for the Lorenz 1 time series and 5 points for the Lorenz 3 time series. For the geometric projection, we chose a reembedding strategy for the Lorenz 1 time series. We iterated a delay 1 run to δ_M, then followed up with delays 10, 20, and 30, iterating to a maximum improvement each time again. For the Lorenz 3 time series we took delay $\Delta=1$ and iterated to maximum improvement. Further application of the algorithm with $\Delta=2$ did not yield noticeable improvement. SNR improvements for the study show that the geometric projection algorithm has held up better than the local averaging in both cases (Table XVII).

We compare the effect of the two methods further. For the Lorenz 1 time series, the time series resulting from the geometric projection still retains some high-frequency oscillations (Fig. 11). The residual noise still riding the cleaned-up time series is no longer white. Instead, it is correlated. For comparison, we show in Fig. 12 what a corresponding 17-dB noisy Lorenz 1 time series looks like when the added noise is unfiltered white noise. The spectral mismatch consequent to the fine sampling is apparent as the fuzzy quality of the noise [Fig. 12(b)].

Local averaging did not perform as well for the more coarsely sampled time series, Lorenz 3, where the number of points per typical oscillation is 17.5 (Fig. 13). Since the time series is sampled ten times more coarsely than the Lorenz 1 time series, the high-frequency structure seen in Fig. 11(c) cannot survive here. Nevertheless the smoothness of the cleaned-up time series in Fig. 13(b) has been a surprise to us.

We note an additional effect of the geometric projection, associated with the reembedding strategy. In Fig. 14 we show a Lorenz 0 study for $\mathcal{N}=10\%$. Although the SNR improvement from following up a $\Delta=1$ iteration sequence with a $\Delta=10$ sequence is only 3.5 dB, the charac-

TABLE XIV. Baseline noise (dB) for the Lorenz 1 to Lorenz 6 time series. $N=10\,000$, $\mathcal{N}=0\%$, $(d,k,v)=(15,3,40)$; $dt=0.005$.

Δ	1	2	3	4	10	20	30
$\Delta T=0.005$	−118	−104	−95	−89	−71	−59	−56
$\Delta T=0.025$	−82	−68	−60	−54	−42	−28	−26
$\Delta T=0.05$	−68	−53	−47	−42	−26	−22	−21
$\Delta T=0.125$	−49	−39	−31	−23	−18	−17	−17
$\Delta T=0.25$	−39	−23	−19	−18	−16	−16	−15
$\Delta T=0.5$	−23	−18	−16	−16	−16	−16	−15

TABLE XV. High noise algorithm performance for the Lorenz 0 time series, $N=10\,000$, $\mathcal{N}=30\%$ (11.60 dB), $h_0=\Delta T=0.001$.

Conditions	$\Delta=1$	$\rightarrow$	10	$\rightarrow$	20		δ
(15,3,20)	8.03 dB	+	3.57 dB	+	0.85 dB	=	12.45 dB
n	15		15		15		
(15,3,40)	11.88 dB	+	2.69 dB	+	0.19 dB	=	14.76 dB
$n=n_M$	106		13		8		

ter of the residual noise is changed, in the direction of less high-frequency structure.

We close this subsection with some results from a 100% noise coarse sampling Lorenz study (Table XVIII). The results are fair and promising. The low values of n_M strongly suggest that (d,k,v) values have not been well chosen for these runs.

E. Some dynamical features of algorithm performance

Although SNR improvement δ depends only on the relative rms power in the residual noise $\|\tilde{\eta}\|$ by Eq. (37), nonetheless it still constrains the possibilities significantly. For both maps and not too highly sampled flows, the residual noise is significantly shifted towards the low-frequency end of the spectrum, corresponding to residual distortions riding on the signal [Fig. 13(b)]. These effects show up in the phase portraits as exactly the same thing, namely, attractor distortions. In fact, attractor recovery can be pretty good (Figs. 15 and 16).

In a certain sense also, these distortion effects are different for a map than for a flow. For a map, the attractor shape is set by the intricate winding of an unsta-

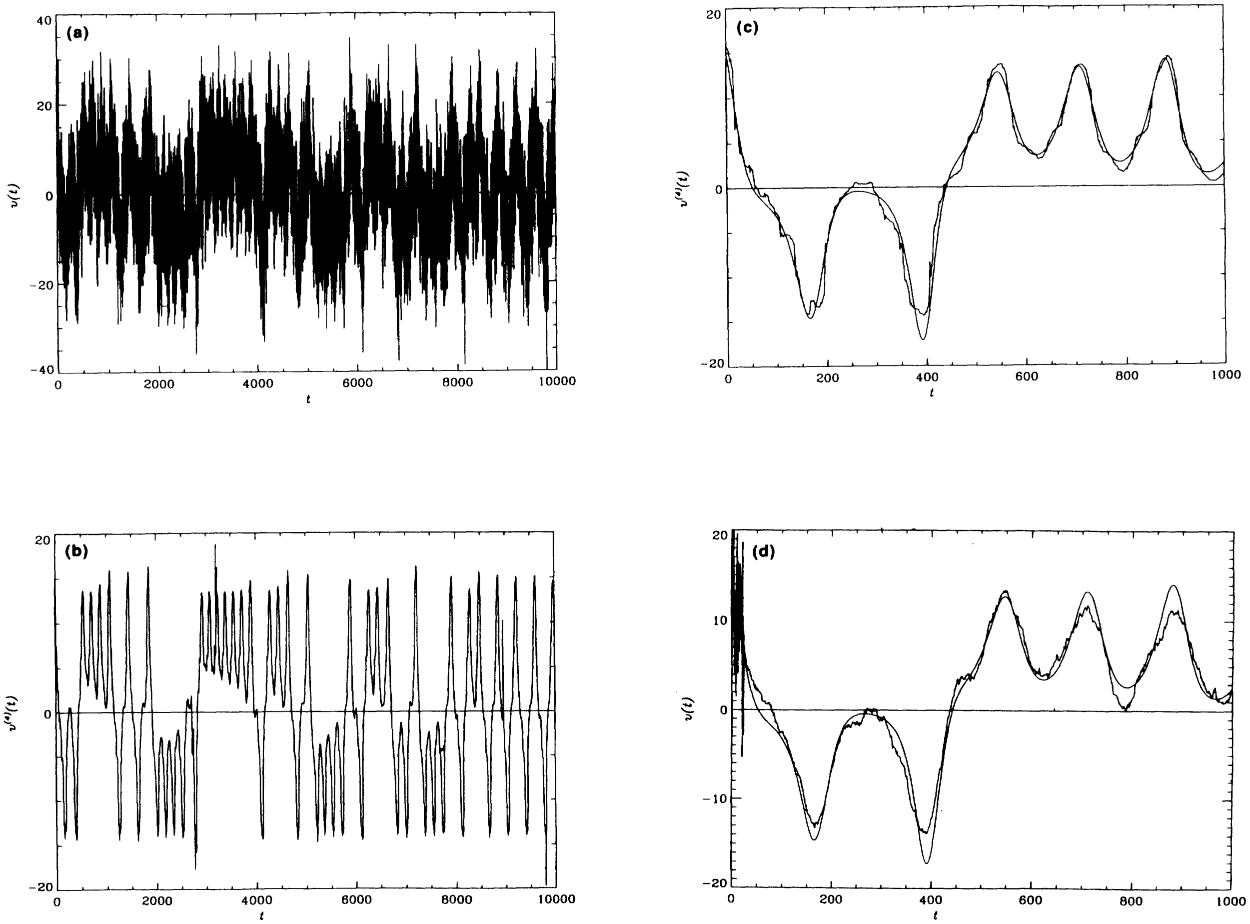

FIG. 11. Time series plots for (a) 100% (0.3 dB) noise Lorenz 1 data. (b) The noise-reduced time series after 4 passes [$\Delta=1$, 10, 20, 30 and $(d,k,v)=(15,3,40)$] of 15 iterations by geometric projection ($\mathcal{S}_f=17.3$ dB). (c) A blowup of the first 1000 points from (b). The resulting time series still tracks the original time series well but small-amplitude, high-frequency oscillations are still clearly visible. (d) The same short segment of the time series reduced by a 50-point local averaging ($\mathcal{S}_f=14.3$ dB). The clean Lorenz 1 time series is plotted in each figure for comparison.

TABLE XVI. High noise algorithm performance for the Lorenz 1 time series, $N=10\,000$, $\mathcal{N}=100\%$, $(d,k,v)=(15,3,40)$.

Conditions	$\Delta=1$	$\rightarrow$	10	$\rightarrow$	20		δ
$\mathcal{S}_i=1.95$ dB	12.14 dB						12.14 dB
$n=n_M$	98						
$\mathcal{S}_i=0.29$ dB	11.33 dB	+	4.71 dB	+	0.68 dB	=	16.72 dB
$n=\min(n_M,50)$	50		50		18		

ble manifold, which cannot be traced by connecting points time sequentially. But for a flow we can connect the dots, and this renders the attractor shape more visible. It is a fair question to ask whether the noise-reduction procedure will "invent" an attractor, a relatively simple-looking phase portrait shape. When the original time series is white noise, the answer, whether as a flow or map, is *no* (Fig. 17). For some more general kinds of random noise, this is unlikely to be the case, an issue we do not try to investigate here.

We discuss some of the unfinished business of our present studies later in Sec. X, such as what are the systematics of performance of the algorithm *qua* dynamics, that is, how well are dynamical properties of the original noise-free time series recovered? But we also venture a few guesses here based on the properties of the examples

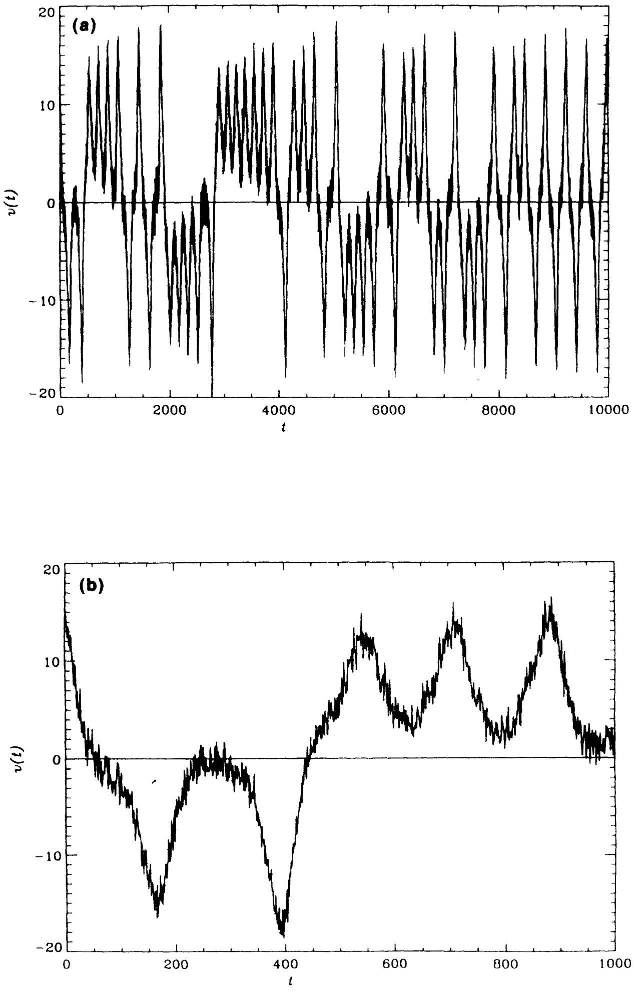

FIG. 12. Lorenz 1 time series with 13% white noise added (~ 17.5 dB) for comparison with the 17.3-dB time series plots in Fig. 11. (a) The complete 10 000-point time series. (b) The first 1000 points. The spectral mismatch of the noise to the (Lorenz 1 time series) signal is clearly evident in (b).

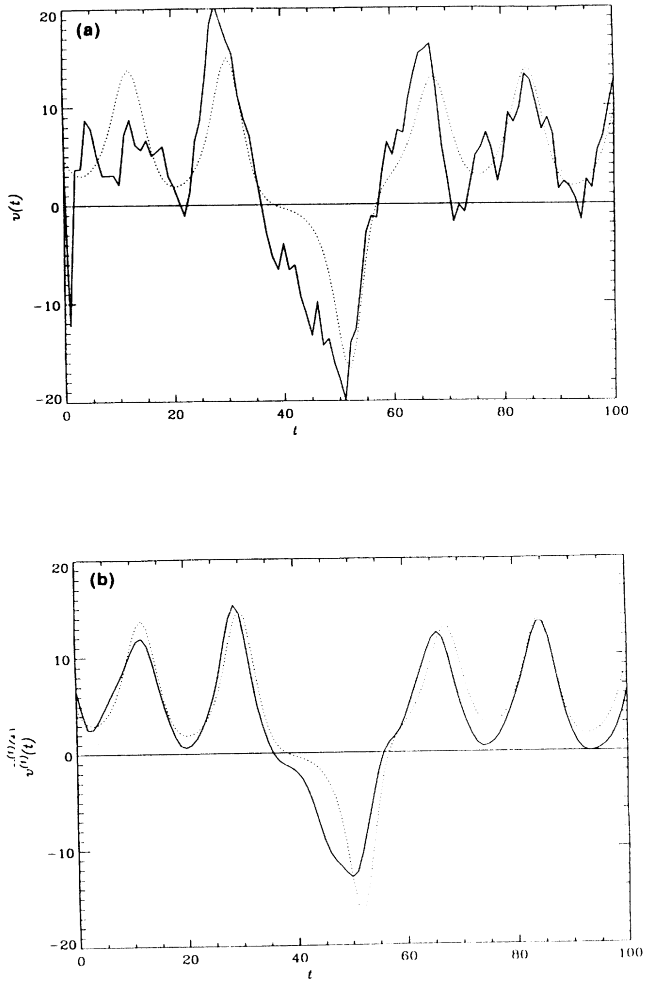

FIG. 13. Plots of the first 100 points for 100% noise Lorenz 3 time series resulting from (a) five points local averaging, $\mathcal{S}_f=6.7$ dB; and (b) geometric projection with $\Delta=1$ and $(d,k,v)=(15,3,40)$, $\mathcal{S}_f=10.7$ dB. The original Lorenz 3 time series is also plotted as the dotted curve for comparison.

TABLE XVII. SNR improvement $\delta\mathcal{S}$ (dB) for $\mathcal{N}=100\%$.

	Geometric projection	Local averaging
Lorenz 1 ($\Delta T=0.005$)	17.0	14.0 (50 pts)
Lorenz 3 ($\Delta T=0.05$)	10.3	6.3 (5 pts)

TABLE XVIII. Coarsely sampled Lorenz data, $\mathcal{N}=100\%$ with $(d,k,\nu)=(18,3,40)$.

	No. of points/cycle	δ_M (dB)	n_M
Lorenz 5	3.5	6.3	7
Lorenz 6	1.75	3.5	4

above. For all but the largest exponent, the Lyapunov spectrum is sensitive to high-frequency noise, and somewhat less so to low-frequency distortion since it is an average quantity computed from derivatives over the attractor. So this may bode well for algorithm performance in regard to metric quantities such as these. In addition, the time sequencing of the points controls the chaotic invariant measure and periodic orbit structures. As this time sequencing actually is very well recovered for the examples shown in Figs. 11 and 13(b), we think it likely that low-order periodic orbit structures and the information dimension will be also well recovered.

We have tested these speculations with a few Lyapunov exponent calculations. For the Hénon example of Fig. 15 the largest, λ_1, is closer to the correct value $\lambda_1=0.408$ for the cleaned-up data than for the noisy data formed by adding an equivalent amount of white noise (Table XIX). These results were gained by using a code that computes Lyapunov exponents described in Ref. [22]. The exponents are computed from orbits realized under delay coordinate construction from a given $N=10\,000$-point time series using local cubic fits in dimensions $d=2$ and 4. Ten separate 1000-point segments were used to compute the required Lyapunov-Oseledec average; each table entry is an average of the resulting ten λ's.

For the Lorenz 3 example of Fig. 16 we performed the

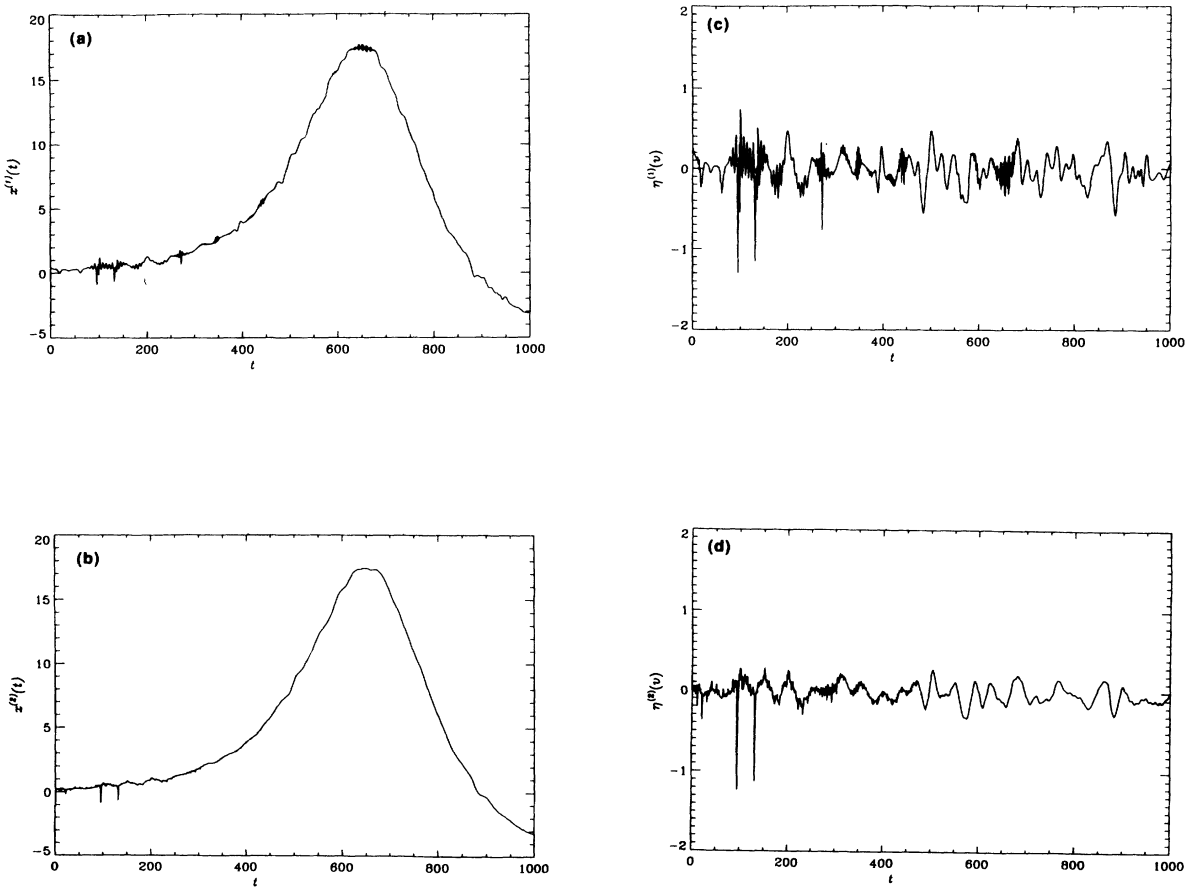

FIG. 14. The nature of the noise changes with each pass [i.e., each series of iterations to maximum δ with a given delay and with $(d,k,\nu)=(15,3,40)$] of the projection algorithm in the finely sampled case. Starting from a 10% noise (20 dB) Lorenz 0 time series, output time series: (a) after first pass with delay 1 ($\mathcal{S}_f=32.2$ dB) and (b) after following up with a second pass with delay 10 ($\mathcal{S}_f=35.5$ dB); (c) and (d), residual noise time series for (a) and (b), respectively. Note some small-amplitude, high-frequency oscillations are removed from the noise after the second pass of the algorithm.

TABLE XIX. Lyapunov exponent results for Hénon data. For $d=4$, the smallest of the four computed exponents was chosen as the λ_2 entry and the largest as λ_1.

Lyapunov exponents ($N=10\,000$)	λ_1 $d=2$	λ_1 $d=4$	λ_2 $d=2$	λ_2 $d=4$
Unsoiled data	0.417	0.437	-1.608	-1.618
10% white noise (20 dB)	1.635	0.555	-0.162	-0.675
Cleaned up (29.9-dB $\mathcal{S}_f$)	0.663	0.436	-0.455	-0.924
3.2% white noise (29.9 dB)	1.161	0.460	-0.242	-0.764

calculations the same way with $d=3$ and 4. Again λ_1 computed from the cleaned-up time series is closer to the value computed from the unsoiled time series than that for the data with an equivalent amount of white noise (Table XX).

IX. ANALYSIS OF EXPERIMENTAL DATA

We have applied the noise-reduction algorithm to time series produced by a magnetoelastic Metglas ribbon experiment that has been used for studies of the control of chaos [11]. The ribbon is vertical, free standing, and clamped at the base. The ribbon buckles gravitationally by application of a vertical oscillatory magnetic field $H=H_0+H_1\cos 2\pi ft$, whose effect is to cause a variation of Young's modulus [23]. The dynamics of the ribbon are extremely nonlinear under the driving conditions of the experiment. The data series we examine here are for return maps obtained from observation of a Fotonic sensor voltage output. The voltage reading is a monotone function of the horizontal displacement of a point on the ribbon near its base. The return map is obtained by strobing the sensor output at the frequency of the drive.

The observed low-dimensional behavior in the ribbon experiments is certainly perturbed by dynamical noise [24] as well as additive noise. Before presenting our analysis of the ribbon data, we make a few theoretical remarks concerning the application of noise-reduction procedures to experimental data. In the previous sections, we have benchmarked our studies only for additive noise, and have used our knowledge of $V(t)$ to perform our measurements of SNR improvement. In an experimental situation, in general, there will often be dynamical noise

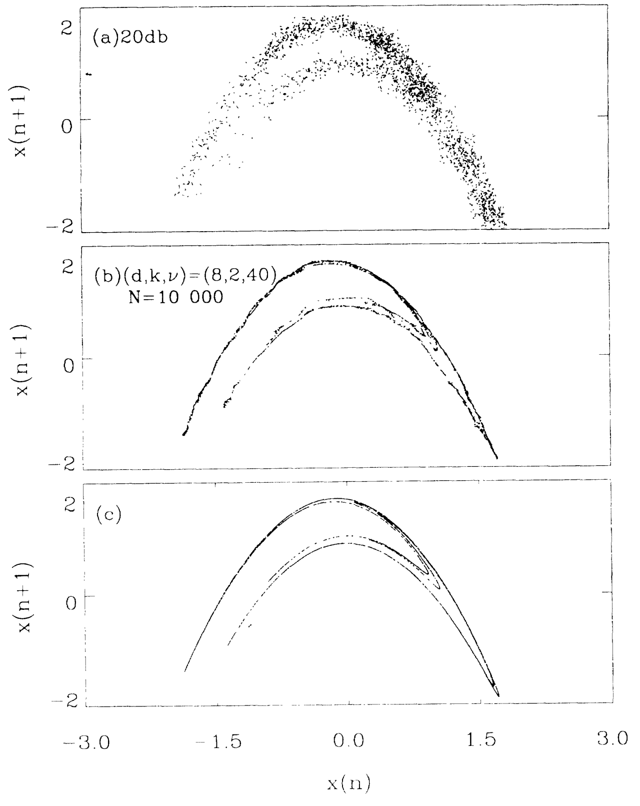

FIG. 15. Hénon attractor with $N=10\,000$: (a) with 10% (20.0 dB) added Gaussian white noise; (b) after $\delta_M=9.9$ (dB) noise reduction with $(d,k,\nu)=(8,2,40)$; and (c) original unsoiled case.

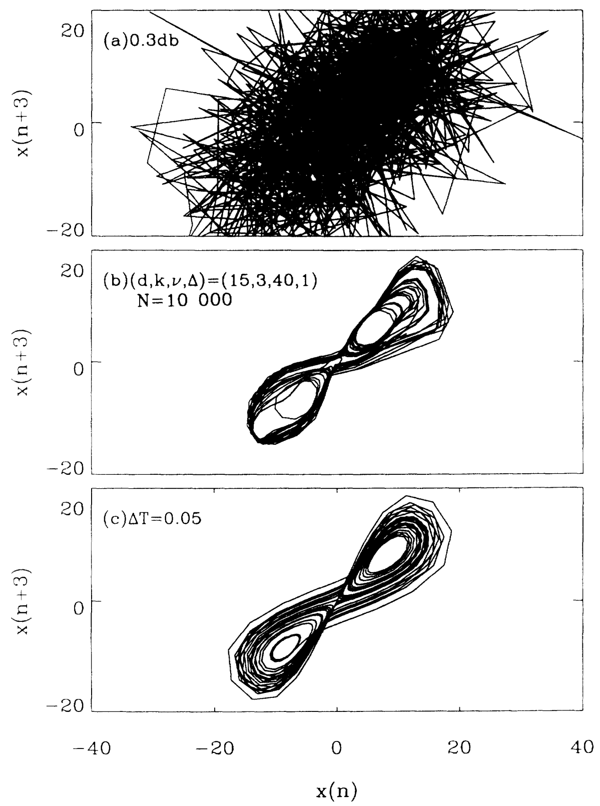

FIG. 16. Phase portraits for Lorenz 3 time series, $N=10\,000$: (a) with 100% added Gaussian white noise; (b) after noise reduction $\mathcal{S}_f=10.7$ dB ($\delta_M=10.3$ dB). First 100 points of the time series are shown in Fig. 13(b); and (c) the original time series.

TABLE XX. Lyapunov exponent results for Lorenz 3 data. In $d=4$ cases the largest exponent computed is spurious for the unsoiled data and the * entry is the second largest. For the noisy cases the spurious exponent is the third largest and the † entries are the smallest in each case. (See Ref. [22] for discussion of these effects.)

Lyapunov exponents ($N=10\,000$)	λ_1		λ_2		λ_3	
	$d=3$	$d=4$	$d=3$	$d=4$	$d=3$	$d=4$
Unsoiled data	0.758	0.750*	−0.032	−0.042	−12.1	−12.2
100% white noise (0.3 dB)	37.9	28.2	30.1	22.9	15.2	1.39†
Cleaned up (10.7-dB $\mathcal{S}_f$)	6.58	5.38	−0.213	1.24	−9.87	−13.9†
29.8% white noise (10.5 dB)	29.2	21.5	22.0	16.6	6.97	−5.57†

and an underlying true signal $V(t)$ will be unknown. Moreover, in many cases, $V(t)$ may not be uniquely defined. For example, in the case of a map $f: M \to M$ in the presence of dynamical noise, a random perturbation of size ϵ is added to each image point under f, $\mathbf{x}_{t+1}=f(\mathbf{x}_t)$, $\mathbf{x}_t \in M$, $t=\ldots,-1,0,1,2,\ldots$. The perturbed image point is then mapped forward by f, its image again being perturbed, and so on, thus creating typically an ϵ-noisy orbit. When shadowing occurs, there are likely to be (uncountably) many true orbits of f "δ close" to the ϵ-noisy orbit. Any of these δ-close true orbits can serve as generator of a corresponding $V(t)=S(f^t\mathbf{x})$, $\mathbf{x}\in M$. Similar remarks may be made for flows. In this setting, the proper object of measurement in the noise-reduction effort, though not unique, can still be identifiable to order δ [25].

We have found in Secs. VI–VIII that there is considerable variation of δ_M and n_M with d, k, ν, Δ, dt, and τ_s for given conditions of initial noise amplitude $\mathcal{N}$ and trajectory length N, and furthermore that specific quantitative results are system dependent as well. As we remarked earlier, it is easy to get less than optimal results. So for

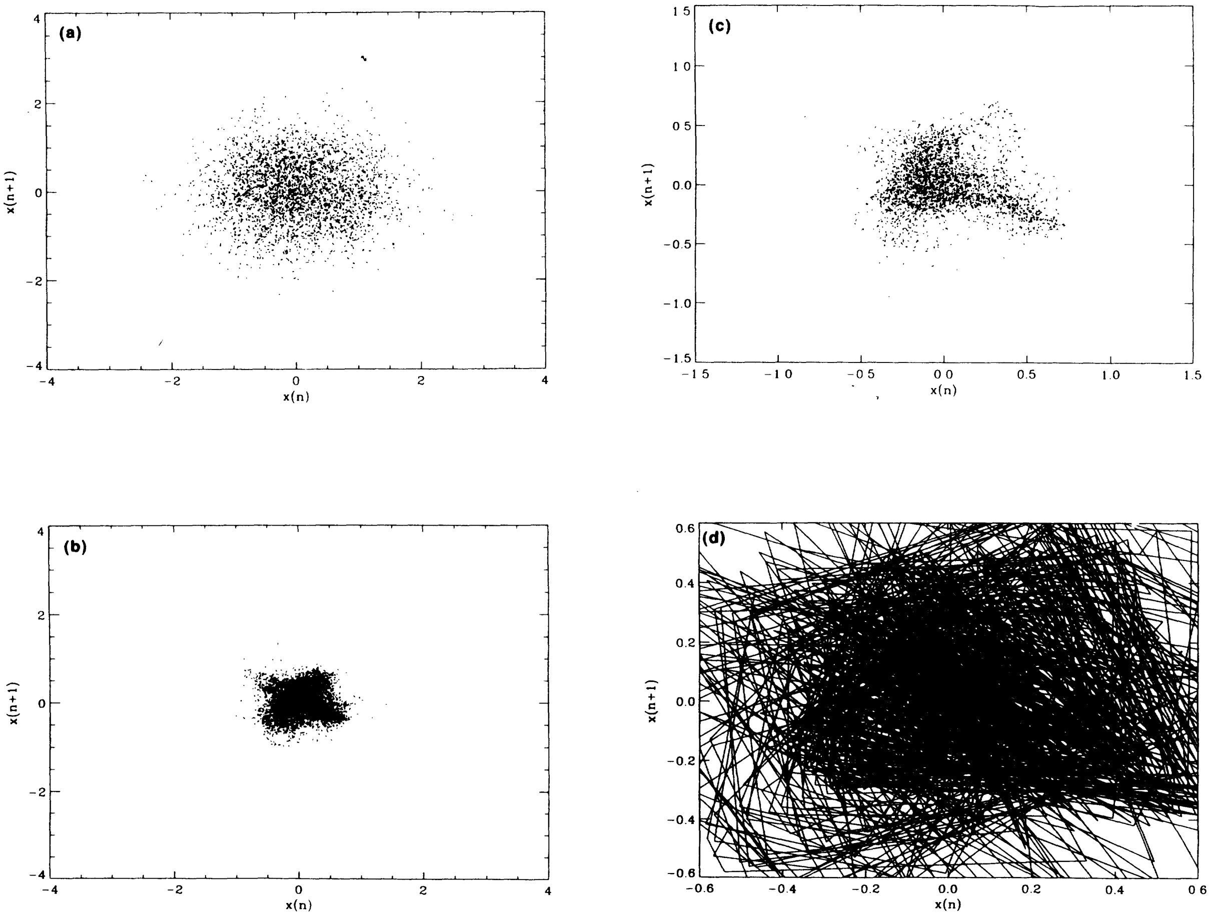

FIG. 17. Delay $\Delta=1$ phase portraits for white-noise time series with $N=10\,000$: (a) "pure" original; (b) after $n=15$ iterations ($\delta_{15}=9$ dB) for $(d,k,\nu)=(10,2,20)$; (c) blowup of (b); and (d) further blowup of (b) but portrayed as a flow (i.e., with time sequential points connected).

application to experimental data, we would like to have some idea of how we are doing.

There does not appear as yet to be a general recognition or solution for this problem. There are a few things we can do, however. We consider the following quantity, which *is* measurable from an experimental time series:

$$\Sigma_n = 10\log_{10}\frac{\|v_n\|^2}{\|v_1 - v_n\|^2} \ . \tag{43}$$

Assuming Eq. (31), we have also

$$\Sigma_n = 10\log_{10}\frac{\|v_n\|^2}{\epsilon^2\|\eta_1 - \eta_n\|^2} \ . \tag{44}$$

For $n < n_M$, $\|\eta_n\|^2$ is falling, by Eq. (37). We might expect that for noise levels not too high $\|v_n\|^2 \simeq \|V\|^2$, and that as n rises toward n_M, we would have

$$\Sigma_n \simeq 10\log_{10}\frac{\|V\|^2}{\epsilon^2\|\eta_1 - \eta_n\|^2} \simeq 10\log_{10}\frac{\|V\|^2}{\epsilon^2} \ , \tag{45}$$

where in the last step we have neglected η_n and used $\|\eta_1\| = 1$. The right-hand side is the initial SNR. Thus, we might hope to get an estimate of the initial noise level present in the data through computing the quantity Σ_n.

In fact, this expectation is pretty much met in the examples of Hénon and Ikeda map cases where we have computed Σ_n. In calculations we have done, Σ_n falls rapidly at first with increasing n, as v_n pulls away from v_1, but then quickly slows its rate of fall and becomes nearly constant. This behavior of Σ_n corresponds to the change in the growth of δ_n: initial logarithmic growth of δ_n and eventually slower than logarithmic growth as saturation sets in [Fig. 3(b)].

Let $d_{\rm pk} = d_{\rm pk}(k,\nu)$ be the value of the embedding trial dimension for which δ_M reaches its peak value for a given choice of (k,ν) and Δ. The results in Table XXI suggest that we may adopt $n = 15$ as a convenient fiducial value at which to perform the estimate in Eq. (45) without incurring more than +2-dB error, typically. The results in the table cover a wide range of (k,ν) and Δ values, two orders of magnitude of initial noise amplitude 0–40-dB $\mathcal{S}_i$ plus a 60-dB example, and include both Hénon and Ikeda map cases. There is also a very wide range of values of n_M. The mean difference $\Sigma_{15} - \mathcal{S}_i$ is only +1 dB with ±1dB spread (standard deviation). We note that Σ_{50} gives a little better estimate than Σ_{15} for $n_M \simeq 2^6$ or more; still Σ_{15} is better than Σ_{10} even for low n_M. The very weak dependence of Σ_n on n makes $n = 15$ a good choice: Σ_{10} often gives unnecessarily high estimates especially for large n_M; on the other hand, Σ_{50} can be too small in low-noise cases, especially where n_M is small, i.e., when the algorithm performance peaks quickly and the slow Σ_n falloff can fail at higher n.

The quantity Σ_{15} exhibits an interesting dependence on

TABLE XXI. Behavior of Σ_n for $d_{\rm pk}$ conditions for maps: $\Sigma_{15} \simeq \mathcal{S}_i$. $\langle(\Sigma_{15} - \mathcal{S}_i)\rangle = 1.0 \pm 1.1$ dB. Hénon, $d = d_{\rm pk}$ and $\Delta = 1$ except where otherwise stated. † denotes maybe not peak $\delta_M(d)$; * denotes an extrapolation of no more than 0.5 dB and accurate to 0.1 dB.

N		Conditions	$\mathcal{S}_i$ (dB)	n_M	Σ_{10} (dB)	Σ_{15} (dB)	Σ_{50} (dB)	$\Sigma_{15} - \mathcal{S}_i$ (dB)	δ_M (dB)
10 000		(4,2,20)	60.0	8	61.0	60.2		0.0	4.64
10 000	$\Delta=2$	(6,2,40)	39.2	7	39.7	39.2		0.0	9.01
10 000		(4,2,20)	39.2	9	40.5	39.6		0.4	5.34
3000		(5,2,30)	39.1	5	39.2	38.2		−0.9	5.98
10 000		(9,3,20)	39.2	19	40.3	39.8	39.1*	0.6	9.84
10 000		(8,3,40)	39.2	15	40.1	39.7	39.0	0.5	9.19
3000		(8,3,10)	39.1	26	41.2	40.5	39.3*	1.4	6.47
3000		(6,3,40)	39.1	14	40.4	39.8	38.8	0.7	6.46
10 000		(10,4,20)	39.2	49	41.0	40.4	39.7	1.2	9.87
10 000		(10,5,20)	39.2	121	42.3	41.4	40.2	2.2	8.90
10 000	$\Delta=2$	(5,4,30)	39.2	64	45.1	43.9	41.8	4.7	3.81
3000		(8,4,20)	39.1	43	41.4	40.7	39.8	1.6	7.54
3000	$\Delta=2$	(5,2,20)	39.1	15	21.2	20.5		1.3	5.86
3000		(6,2,30)	19.1	13	20.2	19.8		0.7	9.07
3000		(9,3,10)	19.1	81	22.1	21.1	19.8	2.0	8.54
10 000		(10,2,40)†	9.6	8	10.1	9.8		0.2	9.08
10 000		(12,2,20)†	−0.8	13	−0.3	−1.6		−0.8	5.37
10 000	Ikeda	(7,3,20)	40.0	10	41.3	40.6		0.6	5.79
10 000	Ikeda	(7,3,10)	40.0	24	42.6	41.8	40.5*	1.8	5.66
10 000	Ikeda	(8,3,20)	20.0	34	22.0	21.4	20.5*	1.4	8.37

d which is general in nature. We can exploit this dependence for the purpose of making educated choices of (d,k,ν). We assume we have found a choice for which Σ_n exhibits the behavior described above. We expect that to be the case as long as we did not choose so badly that Σ_n fails to have a very slowly falling region following its initial rapid fall from Σ_1. As d is increased, δ_M first rises towards its value at d_{pk} while n_M falls (see Figs. 5 and 7). For large enough d, beyond the peak of δ_M vs d, performance degrades drastically and the alogorithm produces negative improvement after at best only a few iterates. Correspondingly, for these high d values $n_M \to 0$. These effects show up in the behavior of Σ_n for any fixed n: in particular, Σ_{15} falls as we raise d, and just beyond the value d_{pk}, it plummets precipitously [Fig. 18(a)]. Graphs of Σ_{15} vs d such as the ones shown in Fig. 18(a) thus can be used to supply an estimate of the best d to choose for enhanced algorithm performance. At the same time, by Table XXI, the corresponding Σ_{15} value $\Sigma_{15}(d_{pk})$ gives us our estimate for $\mathcal{S}_i$.

This approach has produced clear results for the ribbon data [Fig. 18(b)]. We chose $k=3$ and 4 to implement the algorithm based on the following considerations. A reasonable low-dimensional phenomenological model for the ribbon dynamics is a parametrically driven Duffing equation, which is three dimensional. Since one of the degrees of freedom is time (mod$2\pi/f$), M should be something like $S^1\times\mathbb{R}^2$, or maybe $T^2\times\mathbb{R}$ which requires $\mathbb{R}^4$ for an embedding. In a previous experiment on the same system under physically similar driving conditions [23], the flow data analysis showed that all evidence of transverse self-intersections were absent by $d=5$, and very probably by $d=4$. Moreover, a Poincaré section was found distinct from the trivial one based on the periodicity of the drive, so $M=T^2\times\mathbb{R}$ could be inferred.

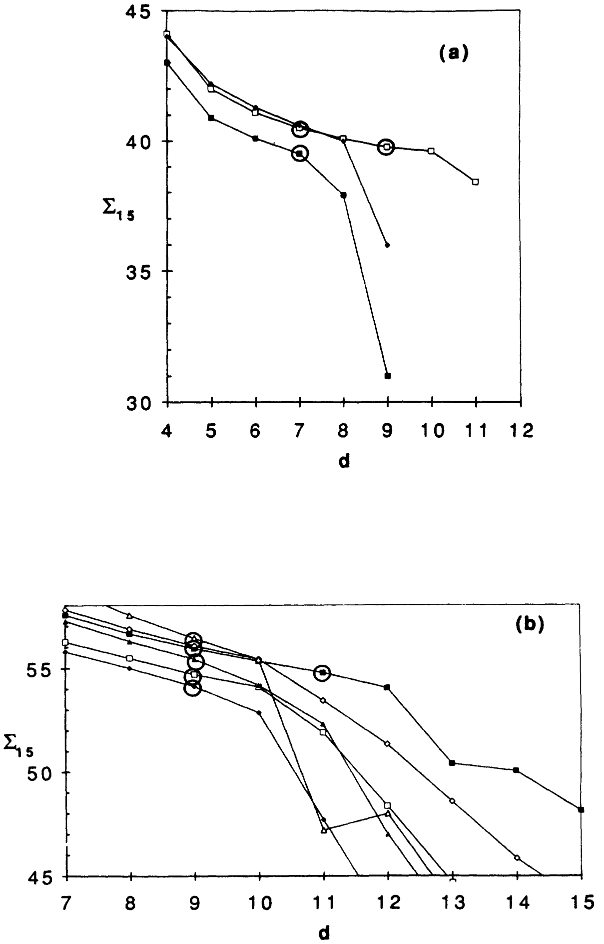

FIG. 18. (a) Σ_{15} (dB) vs d plots for a few cases of known maps. All cases have initial 1% noise (~40 dB). □: Hénon, $N=10\,000$, $(k,\nu)=(3,20)$; ◆: Ikeda, $N=10\,000$, $(k,\nu)=(3,20)$; ■: Hénon, $N=3000$, $(k,\nu)=(3,30)$. The points circled correspond to where peak δ_M were achieved in their individual cases. (b) Σ_{15}(dB) vs d plots for the ribbon data. Each line corresponds to a different choice of (k,ν) values: ■: (3.10); □: (3,20); ◆: (3,30); ◇: (4,15); ▲: (4,25); and △: (4,40). The points circled indicate the optimal choice of d values which is determined by one back from the sharp drop of Σ_{15}. These d values are the choice for best δ_M performance for their corresponding (k,ν) values.

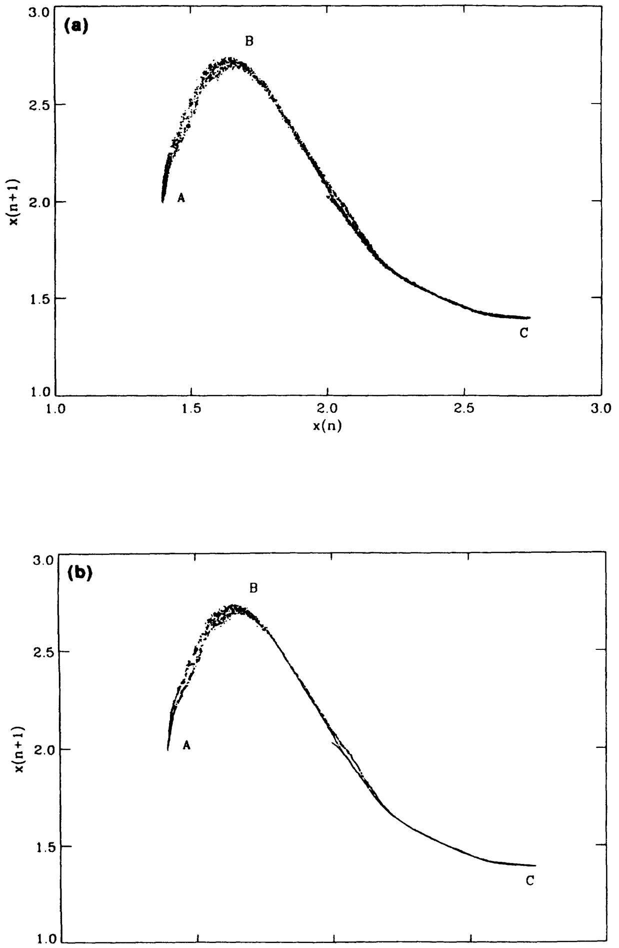

FIG. 19. The image of the embedded attractor (a) for the ribbon data, and (b) for a cleaned-up data set, both projected into the first two components. The reduction was done for 15 iterates with $(d,k,\nu)=(9,3,20)$. This choice of parameters is determined from Fig. 18(b).

Thus we expect that $m=2$ for the data depicted in Fig. 19(a), and they may require $d=3$ for an embedding. Our choice of k is to allow for the possibility of $m=3$ and to avoid doing damage to the data by projecting it into a space of lower dimension. Such caution may be unnecessary, but we have not studied effects of projecting into spaces having $k<m$.

To estimate $\mathcal{S}_i$ from the computations depicted in Fig. 18(b), we average the six Σ_{15} values for the circled points and deduct 1 dB. The result is $\mathcal{S}_i=54.2$ dB ($\mathcal{N}\simeq 0.20\%$). The authors of [11] have estimated a dynamical noise amplitude of ±0.005 V since any structure below this scale on the attractor [Fig. 19(a)] was blurred out. For our ribbon time series $\|V\|=0.42V$, which gives a noise size $\mathcal{N}=1.25\%$ ($\mathcal{S}_i=38$ dB). This is well outside the 1-dB standard deviation from our results (0.9 dB for Σ_{15} fluctuations from the six values above, and 1.1 dB overall fluctuation from Table XXI).

Our $\mathcal{S}_i$ estimation procedure has been reasonable, but it is still largely empirical in nature. We can develop an independent check of our result as follows. Let $D(t)=x(t)$ denote the experimental ribbon data time series, and $D_r(t)$ the time series resulting from fifteen iterations of the algorithm for some choice of (d,k,ν) likely to give reasonable improvement, e.g., (9,3,20) [Fig. 19(b)].

Numerical experiment A. Suppose we now generate a new "noisy" time series $v(t)$ by adding 1% white noise to $D(t)$. We can evaluate the SNR of $v(t)$ two ways: (1) with $D(t)$, regarded as a reference, i.e., as the "true" time series, and (2) with $D_r(t)$ as a reference. Apply the algorithm to $v(t)$ and iterate to maximum δ_M according to each measure. For example, with respect to $D(t)$, the initial SNR is precisely 40 dB since the noise added to $D(t)$ is 1%. Fixing (k,ν), we can generate δ_M vs d plots for each case. Suppose our high 54-dB estimate for the true SNR of the original data $D(t)$ is correct. Then we expect $D(t)$ and $D_r(t)$ to look almost noise-free to the relatively much noisier time series $v(t)$, and most, if not all, of its cleaned-up versions $v_n(t)$ as well. The two δ_M vs d plots should then be close together, and the one with $D_r(t)$ as a reference should lie higher since the cleaner signal should be farther away from the noisy one ($\|v_n-D_r\|\leq\|v_n-D\|$).

Numerical experiment B. Let us imagine a second numerical experiment just like this one but with the "noisy" time series now formed by adding a *low*-amplitude white noise to $D(t)$, $\mathcal{N}=0.07\%$. This is somewhat less than the 0.20% implied by our assumed 54 dB, and gives a combined noise amplitude $\mathcal{N}=0.21\%$. The δ_n values should be different from those of experiment A since the "noisy" time series $v(t)$ *now lies close* to $D(t)$. As the iterations proceed, $v_n(t)$ should pull away from $D(t)$ and approach the true signal $V(t)$. Since the SNR measures the ratio of the l^2 norm of the reference time series to the l^2 distance of $v_n(t)$ from the reference time series, we expect δ_M computed with $D(t)$ as a reference to be small and the maximum to be reached quickly, i.e., the corresponding n_M also should be small. In contrast, when $D_r(t)$ is used as a reference we should see a strong peak in the δ_M vs d plot.

On the other hand, if the 38-dB estimate is right, the result of experiment A should look more or less like the result we have just described for experiment B, and the result of experiment B should remain the same [26].

We first test these ideas on a Hénon time series having 57-dB $\mathcal{S}_i$ as our "toy" $D(t)$. For our "noisy" time series $v(t)$ we add to $D(t)$ white noise with $\mathcal{N}=0.07\%$. We used three reference time series: (1) $D(t)$, (2) $D_r(t)$ obtained from $D(t)$ after 15 iterations of the algorithm, and (3) $V(t)$, the original unsoiled Hénon time series that we used to produce $D(t)$. The results are precisely those described above for experiment A and experiment B [Fig. 20(a)]. The "noisy" time series shows little or no improvement relative to $D(t)$ but large and nearly equal improvements relative to $D_r(t)$ and $V(t)$. These results confirm all the theoretical expectations described in experiments A and B. Results for the ribbon data [Fig. 20(b)] show clear consistency with the 54-dB $\mathcal{S}_i$ estimate and are inconsistent with the 38-dB estimate.

There is an obvious possible explanation for the discrepancy between our calculation and the noise assumed in [11]: projection effects can give misleading impressions regarding the presence of attractor structure. The strandlike appearance of the attractor in a delay one, $\mathbb{R}^3$ construction [Fig. 21(a)], shows features not present in Fig. 19. The thin region of the attractor marked C in Fig. 19 is resolved into clearly separated filaments in this $\mathbb{R}^3$ picture, while region A is seen edge on. In addition, there is a good deal of twisting in the strand, so that C, for instance, instead of curving towards the viewer as A does, curves away. Let us now look down the $D(t+1)$

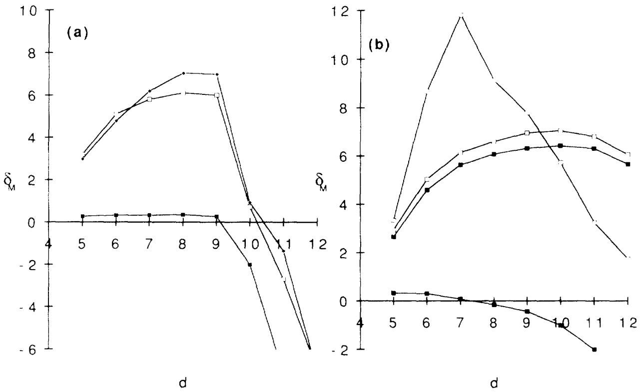

FIG. 20. (a) δ_M (dB) vs d plots for Hénon data. Three reference time series were used. (1) ■: Hénon data with 0.14% noise (57 dB); (2) □: cleaned-up time series after 15 iterates with $(d,k,\nu)=(5,2,20)$ from the 57-dB time series (the conditions for this reduction were not chosen to be optimal); (3) ♦: the original unsoiled Hénon time series. (b) δ_M(dB) vs d plots resulting from experiments A and B for the ribbon data. Two reference time series were used for both an additive 0.07% noise case and a 1% noise case. The following is the notation for reference time series. □: the cleaned-up ribbon data [15 iterates $(d,k,\nu)=(9,3,20)$]; ■: original ribbon data. The two plots that are close together are 1% cases. There the cleaned-up and original ribbon data $D_r(t)$ and $D(t)$ both look squeaky clean to the "1%-noisy" time series.

axis (y axis) [Fig. 21(b)], which is exactly the same as the projection into $\mathbb{R}^2$ of a delay 2 embedding. The flat band B in Fig. 19 is seen edge on, while the previous edge-on view of C is now showing a band.

To get upper and lower bounds initial noise in the ribbon data we focus on region B, shown enlarged in Fig. 22. To get an upper bound, we assume the region of the box in Fig. 22(a) is an edge-on view exactly. The vertical spread is then an upper bound on the noise. For the 179 points in the box, the standard deviation is 24 mV, so $\mathcal{N} \leq 0.6\%$ and $\mathcal{S}_i \geq 45$ dB.

A blowup of the region in the box [Fig. 22(b)] after application of the algorithm reveals fairly flat upper and lower boundaries, with only a few outliers. This suggests that the view is not quite edge on. For the 178 points involved, the standard deviation is now 18 mV. We can get a lower limit on the initial noise by assuming the error in the cleaned-up case is zero. Then 3 mV $=\frac{1}{2}(24-18)$ mV would be a minimum value for the initial noise, i.e., $\mathcal{N} \geq 0.07\%$ (63 dB). We conclude that our estimate for the initial noise $\mathcal{N}=0.20\%$ (54 dB) is in reasonable agreement with the properties of the actual data, and is essentially correct.

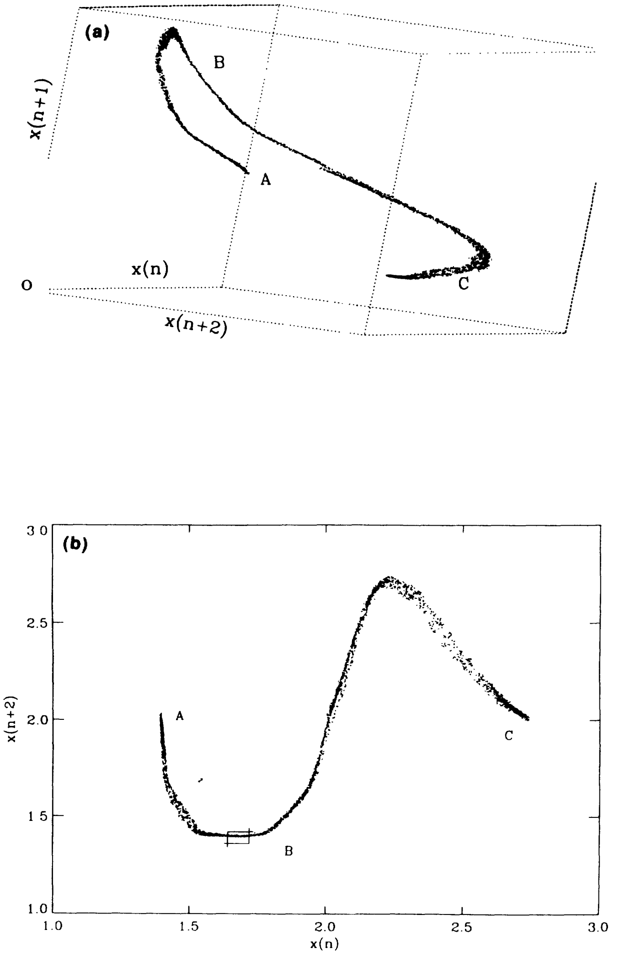

FIG. 21. (a) A three-dimensional view of the ribbon attractor embedded in $\mathbb{R}^3$. (b) Projection image of the ribbon attractor similar to that of Fig. 19 except now we have delay $\Delta=2$. Another way to get this picture is to take a three-dimensional image starting from the position of Fig. 19 and rotate $-90°$ around the horizontal axis.

X. SUMMARY AND DISCUSSION

We have presented a local geometric method for noise reduction that is applicable to time series produced by chaotic systems. The basic idea of the method is related to the local singular value analysis technique for estimating topological dimension due to Broomhead, Jones, and King [8]. Using the delay coordinate construction, we (typically) obtain an embedded orbit from a scalar time series which lies on the embedded image M' of the underlying manifold M. We identify the best local linear approximation H of the embedded manifold M' near a point $\bar{\mathbf{p}}$ lying in (or close to) M' by solving an eigenvalue problem of a matrix X formed of unit vectors from $\bar{\mathbf{p}}$ to nearby points of the embedded orbit. Noise reduction near $\bar{\mathbf{p}}$ is achieved by averaging these points with their projections into H. We have introduced the measure-ordered cover to organize the noise reduction globally. The phys-

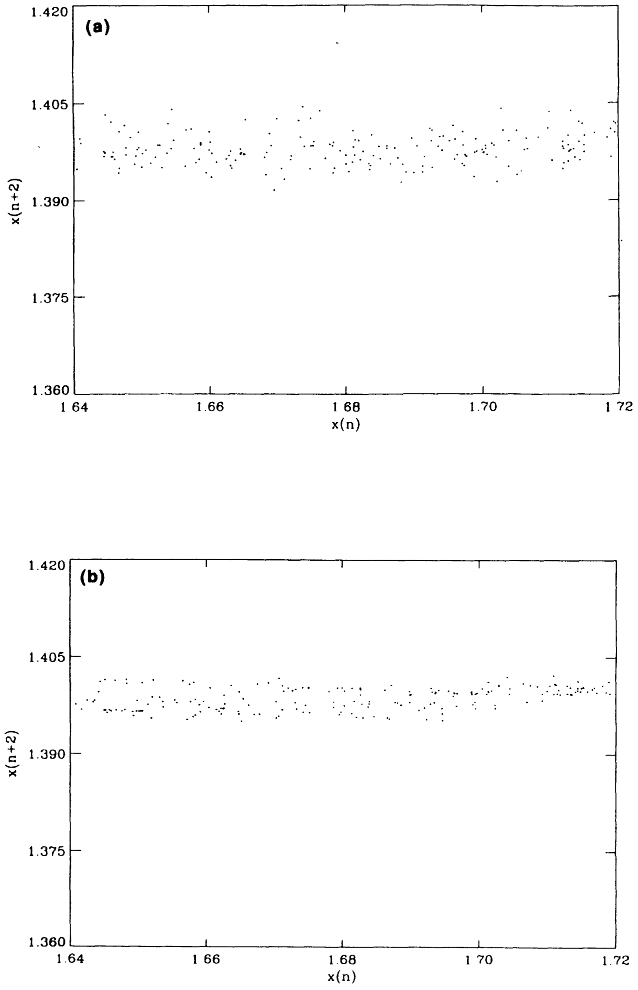

FIG. 22. The blowup of the boxed region B in Fig. 21(b) for (a) the ribbon data and (b) the cleaned-up time series shown in Fig. 19(b).

ical replacement time series produces an optimal data state vector time series that is realizable as a delay coordinate construction. This PRTS is the input for the next iteration. We have presented results of extensive systematic studies of algorithm performance for the Hénon map with additive white noise. These studies consisted of iterating the algorithm until the SNR improvement achieved a stable maximum δ_M after n_M iterations. δ_M and n_M data were collected for a wide range values of the parameters of the method d, k, and ν, and for different trajectory lengths, different initial noise levels, and different noise and orbit realizations. We have explored algorithm performance for a wide range of physical sampling intervals ΔT for the Lorenz system, several orders of magnitude of additive white noise and under a range of embeddings produced by different delays. We have investigated comparison of performance of time series behavior under the algorithm with those resulting from a linear local averaging filter. A few calculations for the Ikeda map and Rössler system were done to test system dependence. We have also explored limits of the method for very low initial noise, including the theoretically important baseline noise (0%) case. We have achieved good improvements for very high levels of initial noise for the Hénon map (9-dB gains for 10-dB $\mathcal{S}_i$ and a nonoptimal 6-dB gain for -1-dB $\mathcal{S}_i$). For Lorenz 1 ($\Delta T = 0.005$) cases with $\mathcal{N} = 100\%$, under a variety of conditions where (d,k,ν) choice was not optimized, good results were also achieved, 17 dB in one instance. For the coarsely sampled Lorenz 5 time series, $\mathcal{N} = 100\%$, we found $\delta_M = 6$ dB where again the (d,k,ν) choice was not optimized. Finally, we have analyzed data produced by a Metglas ribbon experiment and found a method to allow a quantitative estimate of the noise present in the data. Our analysis has resulted in a large revision downward of the estimated noise of [11].

Two aspects of the present work appear noteworthy to us. First, the local projection technique for noise reduction requires no prior dynamical information for its implementation, and no general dynamical properties of maps and flows, such as hyperbolicity, are used. The method is purely geometric. The combined point of view of signal processing theory and dynamical systems theory has provided insight into the operation of our method.

Since we do not use dynamical information to perform noise reduction, an important indicator of the performance of our algorithm will be how well dynamical quantities, such as the Lyapunov spectrum, attractor dimensions, etc., are correctly reproduced from the cleaned-up data. We have explored this issue briefly, in Sec. VIII, with a couple of sample calculations, and found that recovery of dynamical information was good. But more systematic studies still need to be done.

Another interesting direction for future work is to pursue the optimization of (d,k,ν) choices for coarsely sampled flows. Baseline noise calculations can provide a guide here and a relatively fast way to assess time series length requirements for various ΔT. A related problem is performance against correlated noise, e.g., colored or $1/f$-like scale invariant noise.

The second point we wish to note is the remarkable circumstance laid out in Sec. IX allowing us to estimate the initial noise level for a given time series, and even to estimate the best $d \simeq d_{pk}(k,\nu)$ value to implement the algorithm. The systematics of Σ_{15} provide a good guidance for an educated guess for values of (d,k,ν) for good noise reduction. Thus in analyzing data with only limited information about its origin, we do not have to proceed blindly.

The algorithm has done surprisingly well considering the crudeness of linear local modeling by the H_k. An obvious extension of the method is to include manifold curvature effects by means of local polynomial fits. This improvement may be most useful for low noise, high delays, and coarsely sampled flow cases since it should result in reduced levels of baseline noise. On the other hand, this also could have far-reaching effects in the case of coarsely sampled flows even for very high noise. Short of local polynomial modeling, there may be significantly better choices of f than the 0.5 value chosen in this paper.

Although we have described a basic theoretical framework for the scheme, we have not produced a theoretical formulation for the algorithm performance, e.g., an expression for δ_n, which might enable us to provide explanation for the logarithmic growth at low n. An outstanding issue for the method is the saturation of δ_n. We believe that significant increases in δ_M will be possible with future developments of the local projection method. We think this is especially true for the high-noise case.

Note added in proof. Since this work was completed we have investigated several issues not addressed or adequately treated here. We have performed systematic studies of effects of sampling time and trajectory length on algorithm performance for the Lorenz system [27]. In Ref. [28] we performed multiple iterate baseline noise (attractor stability) studies. In Ref. [29] we investigated algorithm performance against dynamical noise for the Hénon map. And in Refs. [28,29], we examined $k < m$ effects for the Lorenz system.

ACKNOWLEDGMENTS

R.C. wishes to thank Steve Martin for a useful remark. G.-H.H. wishes to thank Carmen Chicone, Jan Segert, and S. Notaris for extensive use of their workstations for the computations of this work. Both authors thank Liming W. Salvino for performing the Lyapunov exponent calculations, Reggie Brown for giving us the Lyapunov exponent code, and W. L. Ditto for providing the experimental data. This work was supported by the NSWC Navy dynamics institute Program, the Office of Naval Research (Physics Division), and the Naval Surface Warfare Center Independent Research Program.

APPENDIX A

Consider $Q(\mathbf{u}) = \mathbf{u}^T X \mathbf{u}$, X symmetric and $\|\mathbf{u}\| = 1$. Let the eigenvalues of X be ordered $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$. Writing $\mathbf{u}$ in terms of the eigenvectors $\mathbf{w}_i$, viz. $\mathbf{u} = \sum_{i=1}^{d} u^i \mathbf{w}_i$, we have

$$Q(\mathbf{u})=\sum_{i=1}^{d}\lambda_i(u^i)^2 . \tag{A1}$$

Since $\sum_{i=1}^{d}(u^i)^2=1$,

$$Q(\mathbf{u})=\lambda_d+\sum_{i=1}^{d-1}(\lambda_i-\lambda_d)(u^i)^2\ge\lambda_d=Q(\mathbf{w}_d) . \tag{A2}$$

Thus

$$\begin{aligned} D(r,H_k)&=\sum_{j=k+1}^{d}\mathbf{u}_j^TX\mathbf{u}_j \\ &=\sum_{j=k+1}^{d}Q(\mathbf{u}_j) \\ &=\sum_{j=k+1}^{d-1}Q(\mathbf{u}_j)+Q(\mathbf{u}_d) \\ &\ge\sum_{j=k+1}^{d-1}Q(\mathbf{u}_j)+\lambda_d(r) , \end{aligned} \tag{A3}$$

where in the last sum $\mathbf{u}_j\in\langle\mathbf{w}_d\rangle^{\perp}$, the orthogonal complement to $\langle\mathbf{w}_d\rangle$. Repeating the procedure,

$$\begin{aligned} D(r,H_k)&\ge\sum_{j=k+1}^{d-1}Q(\mathbf{u}_j)+\lambda_d(r) \\ &\ge\cdots\ge\sum_{j=k+1}^{d}\lambda_j(r) . \end{aligned} \tag{A4}$$

APPENDIX B

The *singular value decomposition* for an arbitrary $m\times n$ matrix is given by the following theorem of linear algebra.

Theorem B1: Let A be an arbitrary (real) $m\times n$ matrix. Then there exist an orthogonal $m\times m$ matrix U and an orthogonal $n\times n$ matrix W such that $U^TAW=\Sigma$ is an $m\times n$ "diagonal matrix" of the following form:

$$\Sigma=\begin{bmatrix} D & 0 \\ 0 & 0 \end{bmatrix} ,$$

$$D\equiv\mathrm{diag}(\sigma_1,\ldots,\sigma_r),\sigma_1\ge\sigma_2\ge\cdots\ge\sigma_r>0 .$$

Here, $\sigma_1,\ldots,\sigma_r$ are the *singular values* of A, and r is the rank of A.

The decomposition $A=U\Sigma W^T$ is called the singular value decomposition of A. The corresponding row vectors of U and column vectors of W are called the singular vectors. Notice that these singular vectors form bases of their corresponding Euclidean spaces. A simple consequence is the following.

Corollary B2: An $m\times n$ matrix A has rank r if and only if A has r positive singular values.

Let $\mathbf{v}_1,\ldots,\mathbf{v}_l$ be arbitrary vectors in $\mathbb{R}^n$. We denote by $\langle\mathbf{v}_1,\ldots,\mathbf{v}_l\rangle$ the linear subspace spanned by vectors $\mathbf{v}_1,\ldots,\mathbf{v}_l$. Also denote by $\|\;\|$ the l^2 norm in $\mathbb{R}^n$. Then the singular values can be viewed as the measurement of the excursion of the column (or row) vectors of A into the directions of the corresponding singular vectors.

Proposition B3: Let A be an arbitrary $m\times n$ matrix. The $n\times n$ matrix $W=(\mathbf{w}_1,\ldots,\mathbf{w}_n)$ has all its columns the singular vectors of A as given in Theorem B1. Denote by $E_j, j=1,\ldots,n$ the sequence of linear spaces of dimension $n-j+1$ given by

$$\begin{aligned} E_1&=\langle\mathbf{w}_1,\ldots,\mathbf{w}_n\rangle=\mathbb{R}^n , \\ E_2&=\langle\mathbf{w}_2,\ldots,\mathbf{w}_n\rangle , \\ &\vdots \\ E_n&=\langle\mathbf{w}_n\rangle . \end{aligned}$$

Then the singular values σ_j of A are the positive square roots of the extrema of the Rayleigh quotients

$$\sigma_j^2=\max_{\mathbf{v}\in E_j,\mathbf{v}\ne0}\frac{\|A\mathbf{v}\|^2}{\|\mathbf{v}\|^2} , \quad j=1,\ldots,n$$

and the maximum is achieved at $\mathbf{v}=\mathbf{w}_j$.

Denote by $\mathbf{w}_i$ and $\mathbf{u}_i$ the singular vectors of A. Notice that for $i\le n$

$$A\mathbf{w}_i=\begin{cases}\sigma_i\mathbf{u}_i & \text{for } i\le r\\ 0 & \text{for } i>r .\end{cases}$$

Thus the singular values can be viewed as the measurement of the excursion of columns of A into the direction of the corresponding singular vector.

The λ_j of Theorem 2 in Sec. III are the squares of the singular values of the $d\times(\nu+1)$ rectangular matrix R since $X=RR^T$. The λ_j measure the excursions of the $\nu+1$ column vectors of R, viz., $\mathbf{x}_i$, $i=0,1,\ldots,\nu$, into the directions of the respective $\mathbf{w}_j$, while D in Eq. (13) measures their "total" excursion into the space $\langle\mathbf{w}_{k+1},\ldots,\mathbf{w}_d\rangle$. Minimizing D amounts to estimating as best H_k, the local linear space containing most of the corresponding local $\mathbf{x}_i$ collection.

In a single iteration of the algorithm described in Sec. V on a time series of 10 000 points, for example, there will be typically 500 such matrices X when $\nu=40$, one for each element of the cover for the set Λ'. Correspondingly there will be 500 such locally determined $\mathcal{H}_k$ each "pointing" in its own locally determined "direction," viz. that of $\mathbf{w}_1,\ldots,\mathbf{w}_k$.

This is in *sharp contrast* to the analysis of Ref. [2], which is based on properties of a *single* $d\times N$ trajectory matrix S formed of the entire time series. The N columns of S are the data state vectors $\mathbf{p}(t)$, $t=1,\ldots,N$, from the time series $v(t)$. The collection of N vectors whose excursion along a given singular vector direction in $\mathbb{R}^d$ is measured by the corresponding singular values is now formed of the entire time series $\mathbf{p}(t)$. That excursion is actually just the (square root of the) variance of the trajectory matrix in the corresponding singular direction. The square of S, $\Theta=SS^T$, is accordingly called the covariance matrix for the orbit. For further contrast we note the indexing of the columns of S is precisely the time itself while for the $\nu+1$ columns of R the indexing is arbitrary. Were these to be time ordered, for some given neighborhood, say, the corresponding times would be scattered throughout the collection of N available values. To emphasize the geometrical interpretation given above for the matrices X (and their eigenvalues) we have called them *local excursion matrices* (Sec. III).

*Present address: Mathematics and Information Sciences Branch, Naval Surface Warfare Center, Dahlgren Division, White Oak, Silver Spring, MD 20903-5060.

[1] R. Badii and A. Politi, in *Dimensions and Entropies in Chaotic Systems,* edited by G. Mayer-Kress (Springer-Verlag, Berlin, 1986), pp. 67–73.

[2] D. S. Broomhead and G. King, Physica D **20**, 217 (1986).

[3] Eric Kostelich and Jim Yorke, Phys. Rev. A **38**, 1649 (1988).

[4] J. D. Farmer and J. Sidorowich, in *Evolution, Learning, and Cognition,* edited by Y. C. Lee (World Scientific, Singapore, 1988), pp. 277–325.

[5] Stephen M. Hammel, Phys. Lett. A **148**, 421 (1990).

[6] J. D. Farmer and J. Sidorowich, Physica D **47**, 372 (1991).

[7] P.-F. Marteau and Henry D. I. Abarbanel, J. Nonlin. Sci. **1**, 313 (1991).

[8] D. S. Broomhead, R. Jones, and G. King, J. Phys. A **20**, L563 (1987).

[9] N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw, Phys. Rev. Lett. **45**, 712 (1980).

[10] F. Takens, in *Dynamical Systems and Turbulence, Warwick 1980,* edited by D. A. Rand and L.-S. Young, Lecture Notes in Mathematics Vol. 898 (Springer-Verlag, Berlin, 1981), pp. 366–381.

[11] W. L. Ditto, Steven N. Rauseo, and M. Spano, Phys. Rev. Lett. **65**, 3211 (1990).

[12] We note that the uniqueness property must be understood as holding in the context of the Euclidean and l^2 norms adopted in defining $\mathcal{E}$ in Eq. (21).

[13] The principal purpose in allowing $f>0$ in the algorithm is to give us a small edge of performance improvement in the locally flat model setting without going through the extra computational effort of local curvilinear modeling. The introduction of f additionally serves the purpose of proportionately limiting the distance a given point is moved at each stage. As we shall see in subsequent sections, the algorithm is normally iterated many times for a given choice of input parameters before maximum improvement is reached. In each iteration a completely new set of $\mathbf{p}_0^{(\alpha)}$ is chosen, resulting each time in a new set of local facets for the polytope model of M'. Thus the true effective model of M' that results actually stems from an extremely complicated procedure, and is some sort of average of these polytopes. Note that this average already also includes global M' curvature effects. Thus the effect of $f>0$ should be to soften the noise-reduction process, the case $f=1$ being that of doing nothing at all. In particular, if f is too close to 1, we would expect the algorithm to be ineffectively slowed. In a few preliminary runs we found that $f=0.5$ did give a little better performance then $f=0$, which is the reason for our choice.

[14] For the studies of Secs. VI–VIII we have chosen for definiteness to fix N and allow N_D to vary with d and Δ.

[15] For convenience, in our discussion of flows (Sec. VII) we also employ an actually more important physical sampling interval $\Delta T=\tau_s dt$. Where $dt\in\mathbb{R}$ will be the step size used in the numerical solution of the differential equations that generated the time series $V(t)$, $t\in\mathbb{Z}$.

[16] Michel Hénon, Commun. Math. Phys. **50**, 69 (1976).

[17] The noise was generated numerically by the random number generator routine GASDEV in W. Press, B. P. Flannery, S. Teukolsky, and W. T. Vetterling, *Numerical Recipes* (Cambridge University Press, Cambridge, 1986) with corresponding uniform random number generated by the routine RAN1 described in the same book.

[18] S. M. Hammel, C. K. R. T. Jones, and J. V. Moloney, J. Opt. Soc. Am B **2**, 552 (1985).

[19] Edward Lorenz, J. Atmos. Sci. **20**, 130 (1963).

[20] O. Rössler, Ann. N.Y. Acad. Sci. **316**, 376 (1979).

[21] Although Fig. 6(b) shows results for $\nu=40$, a similar figure (not shown) for $\nu=20$ reveals only slightly higher δ_M vs d-k plots.

[22] Reggie Brown, P. Bryant, and Henry D. I. Abarbanel, Phys. Rev. A **43**, 2787 (1991); P. Bryant, Reggie Brown, and Henry D. I. Abarbanel, Phys. Rev. Lett. **65**, 1523 (1991).

[23] W. L. Ditto, Steven N. Rauseo, R. Cawley, C. Grebogi, G.-H. Hsu, E. Kostelich, E. Ott, H. T. Savage, R. Segnan, M. L. Spano, and J. A. Yorke, Phys. Rev. Lett. **63**, 923 (1989).

[24] The physical system is a distributed mass and so is actually infinite dimensional.

[25] Underlying mathematical ideas for these remarks include the shadowing property of pseudo-orbits for uniformly hyperbolic maps and flows, and the stochastic properties of hyperbolic attractors. For the shadowing lemma see Rufus Bowen, J. Diff. Equations **18**, 333 (1975) for maps, and J. E. Franke and J. F. Selgrade, *ibid.* **26**, 27 (1977) for flows. The problem of identification of the "deterministic component" of a noisy orbit for a map is discussed in M. L. Blank, Russ. Math. Surveys **46:6**, 1 (1989).

[26] The reason is that now $v(t)\sim D(t)$ holds for the added 1% noise case as well as for the added 0.07% noise case. Adding 1% noise to a data string having 1.25% noise (38 dB) gives $\mathcal{N}\sim 1.6\%$ (36 dB), which is not too different from 1.25%.

[27] Robert Cawley and Guan-Hsong Hsu, Phys. Lett. A **166**, 188 (1992).

[28] Robert Cawley and Guan-Hsong Hsu, in *Proceedings of the 1st Experimental Chaos Conference,* edited by S. Vohra, M. Spano, M. Shlesinger, L. Pecora, and W. Ditto (World Scientific, Singapore, 1992), pp. 38–46.

[29] Robert Cawley and Guan-Hsong Hsu, *Proc. SPIE 1771,* edited by Helena S. Wisniewski (SPIE, Bellingham, WA, 1992).