Physica D 47 (1991) 373–392
North-Holland

OPTIMAL SHADOWING AND NOISE REDUCTION

J. Doyne FARMER and John J. SIDOROWICH[1]
Complex Systems Group, Theoretical Division and Center for Nonlinear Studies, Los Alamos National Laboratory, Los Alamos, NM 87545, USA

Received 23 February 1990
Accepted 29 June 1990
Communicated by R.H.G. Helleman

The *shadowing* problem is that of finding a deterministic orbit as close as possible to a given noisy orbit. We present an optimal solution to this problem in the sense of least-mean-squares, which also provides an effective and convenient numerical method for noise reduction for data generated by a dynamical system. Given a noisy orbit y and a dynamical system f, we derive a set of nonlinear equations whose solution x is the deterministic orbit with the smallest possible Euclidean distance to y. We present a numerical method for solving these equations. The quality of the solution depends on the initial noise level. When f is known exactly, the noise can be reduced to machine precision over long trajectory segments; with higher noise levels there are regions where the algorithm has difficulty, but significant overall noise reductions are still achieved. If f must be learned from the data the noise reduction is limited by the accuracy of the learning algorithm and the number of available data points, but large reductions are still possible in some cases.

Contents

1. Introduction

1.1. Noise reduction

Noise reduction is a vague term for a class of problems in which we wish to decompose a signal into a part that we want and a part that we do not want. To be a little more precise, suppose we observe a noisy time series $\{y_t\}$, where $t = 1, 2, \ldots, N$. Assume that it can be decomposed as

$$y_t = x_t + n_t, \tag{1}$$

where x_t is the "desired" part of the *signal* and n_t is the "noise". We want to get rid of the noise. We can only observe the noisy signal y_t, but we wish to recover the pure signal x_t.

What we call "noise" is usually subjective, and in order to reduce it we must have a criterion for

[1]Current address: Institute for Nonlinear Science, University of California at San Diego, La Jolla, CA 92093, USA.

distinguishing n_t from x_t. For example, if the noise is predominantly at high frequencies, and the signal predominantly at low frequencies, we can reduce the noise by simply applying a low pass filter. However, if the noise and the signal have similar frequencies spectral methods are ineffective, and more sophisticated criteria are required.

In this paper we assume that the distinguishing feature of the signal $\{x_t\}$ is that it is a trajectory of an invertible dynamical system f, i.e.,

$$x_{t+1} = f(x_t). \tag{2}$$

The noise $\{n_t\}$ might also be a trajectory of a dynamical system; if so, we will assume that it is substantially different from f. Often in cases of interest this means that the dimension of the noise is much greater than that of the signal. When noise is described as "random", this typically could be stated more precisely by saying that it is high dimensional. Noise due to measurement errors that are independent of the dynamics is *observational noise*.

The noise reduction problem can be broken into two parts, learning the dynamical system f, and reducing the noise once it is known. There are three cases:

(1) We are given f.

(2) We must learn an approximation to f, but we have available some "clean" data with known values of x_t.

(3) We must learn f directly from the noisy data y_t.

In all three of these cases we are ultimately faced with problem (1), that of reducing the noise once we know f or have an approximation to it. We will thus begin by assuming that f is already known, and then return in section 4.2 to discuss the case when f has to be learned or when f is uncertain.

The best approach to noise reduction depends on the goal. For example, we may not care about recovering each value of the signal x_t exactly – we may be interested only in recovering a statistical property, such as the power spectrum. Statistical noise reduction is intrinsically simpler than detailed noise reduction. We return to discuss this issue in section 5.

1.2. Shadowing

Suppose the noisy orbit y evolves according to a mixture of known deterministic dynamics and noise, with an equation of motion of the form

$$y_{t+1} = f(y_t) + \sigma n_t. \tag{3}$$

We will call this *dynamical noise*, since it is coupled to the dynamics. We wish to find a purely deterministic orbit $x_{t+1} = f(x_t)$ that stays close to y_t, or in other words, a deterministic orbit that *shadows* the noisy orbit.

The problems of shadowing and noise reduction are closely related. On one hand, shadowing is not a noise reduction problem per se, since the shadowing orbit x is an artificial construction. On the other hand, if we pretend that the shadowing orbit x is a signal, then once we have found it we can trivially write

$$y_t = x_t + \tilde{n}_t, \tag{4}$$

where $\tilde{n}_t \equiv y_t - x_t$ is the "effective" noise. Conversely, if we pretend that observational noise is dynamical noise, then the true orbit x is automatically a shadowing orbit. Although it may not be the *best possible* shadowing orbit, it should be at least close. This suggests that we can seek an approximation to the true orbit by finding an optimal shadowing orbit. Under certain conditions, which remain to be made more precise, these two should asymptotically coincide.

In the dynamical systems literature the shadowing problem was originally motivated by concerns about chaos in noisy environments. Chaotic dynamics strongly amplify the smallest fluctuations, so that a noisy orbit quickly diverges from a purely deterministic orbit with the same initial condition. Thus in physical experiments, where

noise is always present, the detailed behavior of a trajectory is quite different from that of a purely deterministic trajectory with the same starting value. This raises the question: In the presence of chaos, can purely deterministic dynamical equations be valid models of physical systems? Similarly, in computer experiments where there is chaos, because of roundoff error simulated trajectories may deviate significantly from those of the equations they are supposed to simulate. Do they nonetheless correctly reproduce their statistical properties?

Even though noise radically alters the details of individual orbits when the dynamics are chaotic, for small noise levels we can still hope that the *qualitative* properties of noisy orbits and purely deterministic orbits are similar. For instance, suppose there is *some* purely deterministic orbit, albeit with a different initial condition, that stays close to the noisy orbit. Providing this orbit is a *typical* orbit of the dynamical system, then we are assured that noisy orbits will have the same statistical properties for length scales larger than that of the noise[#1].

Anosov and Bowen [2, 3] presented a solution to this problem for the special case of everywhere hyperbolic (Anosov) dynamical systems. Loosely speaking, *everywhere hyperbolic* means that at each point the dynamics can be factored into directions where the motion is either exponentially expanding or exponentially contracting. The *stable manifold* for a given point is the set of points that exponentially converge as time goes to infinity; the *unstable manifold* is the set of points that exponentially converge at infinity when time is reversed. A point where the stable and unstable manifolds intersect is called a *homoclinic intersection*; if the manifolds are parallel, this is called a *homoclinic tangency*. Everywhere hyperbolic dynamical systems do not have homoclinic tangencies, i.e., the angle between stable and unstable manifolds is bounded away from zero.

Assume dynamical noise of the form of eq. (3). Assume that f is everywhere hyperbolic, and the noise is bounded, i.e., $\|n_t\| < \epsilon$, where $\epsilon > 0$. Anosov and Bowen proved that for every $\delta > 0$ there is an ϵ such that every noisy orbit y_t of f is shadowed by an orbit x_t with $\|y_t - x_t\| < \delta$ for all t.

The Anosov–Bowen construction depends on the fact that the stable and unstable manifolds are never parallel where they intersect. Although it was originally believed that Anosov systems were generic, we now know that quite the opposite is true; most dynamical systems of interest have homoclinic tangencies. Furthermore, in many cases they display *sensitive dependence on parameters* [9, 15], so that arbitrarily near any parameter value with a chaotic attractor there is a parameter value with a stable periodic attractor. In this case the noise may induce bifurcations, so that even though an orbit is periodic in the deterministic limit, with the addition of some noise it becomes chaotic [6]. In this case there is no shadowing orbit, and thus the shadowing lemma is not true in general[#2].

Hammel et al. [12] applied the Anosov–Bowen construction to two-dimensional maps that were not hyperbolic and showed that at low-noise levels they can find good deterministic shadowing orbits for finite segments of noisy trajectories. The length of these segments is limited by the presence of homoclinic tangencies. As the length of the trajectory segment goes up, so does the probability of finding a homoclinic tangency of a given severity, i.e., where the angle is smaller

[#1] The presence of a shadowing orbit per se does not imply that the statistical properties are the same; the shadowing orbit might be atypical. For example, consider the binary shift map $x_{t+1} = 2x_t \bmod 1$. Iteration on a computer asymptotically results in $x_t = 0$ as $t \to \infty$ under most common roundoff algorithms. $x_t = 0$ is a valid deterministic orbit, but it is highly atypical, with misleading statistical properties. The requirement that the shadowing orbit be typical is often forgotten in the literature.

[#2] The Anosov–Bowen shadowing lemma is not true in general, but if we modify the original problem to allow small changes of parameters in f, then there may still exist a shadowing orbit, for a "nearby" f. Thus there may be a modification of the theorem along these lines that applies to the general case of non-hyperbolic flow.

than a given value. The problem becomes more severe as the noise level increases.

The Anosov–Bowen construction is cumbersome and difficult to apply in more than two dimensions, and does not provide a practical tool for solving experimental noise reduction problems. Providing such a tool is our main interest, although the method we are about to present may also be of purely mathematical interest for the shadowing problem.

2. Optimal solution of shadowing problem with Euclidean norm

The original Anosov–Bowen construction is concerned with the worst-case distance between points on the noisy orbit y and points on the shadowing orbit x. This amounts to using the L_1 or sup norm to measure the distance between two trajectories. Instead, we use a root-mean-square or Euclidean norm. We define the distance between two trajectory segments x_t and y_t as

$$D(x,y) = \sqrt{\frac{1}{N}\sum_{t=1}^{N} \|y_t - x_t\|^2}\,, \tag{5}$$

where $\|y_t - x_t\|$ is the Euclidean distance between the vectors x_t and y_t. We are thus using the Euclidean norm in two senses, both to measure the distance between the two points x_t and y_t, and to measure the distance between two trajectories x and y.

Minimizing D is equivalent to minimizing D^2. For a given noisy trajectory segment y, we wish to find a trajectory segment x that minimizes $D^2(x,y)$, subject to the constraint that x is *deterministic with respect to* f, i.e.,

$$x_{t+1} = f(x_t) \tag{6}$$

for $t = 1,2,\ldots,N-1$. This problem is straightforward to solve with the method of Lagrange multipliers. It is equivalent to minimizing

$$S = \sum_{t=1}^{N} \|y_t - x_t\|^2 + 2\sum_{t=1}^{N-1} [f(x_t) - x_{t+1}]^{\dagger}\lambda_t\,, \tag{7}$$

where † denotes the transpose and $\{\lambda_t\}$ are the Lagrange multipliers. The factor of two in front of the sum is arbitrary, a convenience that simplifies subsequent expressions.

Differentiating with respect to the unknowns and searching for an extremum, we get the following system of equations:

$$\begin{aligned} \frac{\partial S}{\partial x_t} &= 0 = -(y_t - x_t) + f'^{\dagger}(x_t)\lambda_t - \lambda_{t-1}\,, \\ \frac{\partial S}{\partial \lambda_t} &= 0 = f(x_t) - x_{t+1}\,, \end{aligned} \tag{8}$$

where $\lambda_t = 0$ if $t < 1$ or $t > N-1$. $f'(x_t) = \mathrm{d}f(x_t)/\mathrm{d}x$ is the Jacobian matrix of f at x_t. In the first equation t runs from 1 to N, whereas in the second t runs from 1 to $N-1$. Typically f is nonlinear and this system of equations has no closed form solution.

Since eq. (8) is nonlinear, we must generally use numerical approximation to solve it. We can use any of a large variety of search algorithms. We simply use Newton's method, expanding f in a Taylor's series about a trial solution $\hat{x}$, setting $f(x_t) \approx f(\hat{x}_t) + f'(\hat{x}_t)(x_t - \hat{x}_t)$ and $f'(x_t) \approx f'(\hat{x}_t)$. To simplify subsequent expressions, define

$$\Delta_t \equiv \hat{x}_t - x_t = \text{deviation from true solution},$$

$$\gamma_t \equiv y_t - \hat{x}_t = \text{deviation from noisy orbit},$$

$$\epsilon_t \equiv \hat{x}_{t+1} - f(\hat{x}_t) = \text{deviation from determinism},$$

$$J_t \equiv f'(\hat{x}_t) = \text{Jacobian at } \hat{x}_t. \tag{9}$$

We are thus seeking a minimum of $\sum_{t=1}^{N} \|\gamma_t\|^2$, subject to the condition that $\epsilon_t = 0$ for $t = 1, 2, \ldots, N-1$. The linearized equations become

$$\gamma_t = J_t^{\dagger}\lambda_t - \lambda_{t-1} - \Delta_t, \tag{10}$$

$$\Delta_{t+1} = J_t \Delta_t + \epsilon_t, \tag{11}$$

or in matrix form,

$$\begin{bmatrix}
-1 & J_1^{\dagger} & & & & & & & \\
J_1 & 0 & -1 & & & & & & \\
 & -1 & -1 & J_2^{\dagger} & & & & & \\
 & & J_2 & 0 & -1 & & & & \\
 & & & -1 & -1 & J_3^{\dagger} & & & \\
 & & & & & \ddots & \ddots & & \\
 & & & & & & J_{N-1} & 0 & -1 \\
 & & & & & & & -1 & -1
\end{bmatrix}
\begin{bmatrix}
\Delta_1 \\ \lambda_1 \\ \Delta_2 \\ \lambda_2 \\ \Delta_3 \\ \vdots \\ \lambda_{N-1} \\ \Delta_N
\end{bmatrix}
=
\begin{bmatrix}
\gamma_1 \\ -\epsilon_1 \\ \gamma_2 \\ -\epsilon_2 \\ \gamma_3 \\ \vdots \\ -\epsilon_{N-1} \\ \gamma_N
\end{bmatrix}. \tag{12}$$

We will abbreviate this as

$$\mathbf{M}v = w. \tag{13}$$

$\mathbf{M}$ is a $(2N-1) \times (2N-1)$-dimensional matrix. When f is a d-dimensional dynamical system, the entries of $\mathbf{M}$ are themselves $d \times d$ matrices. Inverting $\mathbf{M}$ to solve for v gives an approximation to the true shadowing trajectory $\{x_t\}$ that is (hopefully) a better solution of eq. (8) than the trial solution $\{\hat{x}_t\}$. In the usual way, Newton's method can be iterated by replacing $\hat{x}_t$ by x_t and repeating this process, searching for a fixed point.

3. Numerical approximation

The structure of **M** complicates the solution of eq. (12). As we shall see, to get the largest possible noise reduction we want to make N as large as possible. However, when the underlying dynamics are chaotic **M** becomes ill-conditioned. Furthermore, approximate homoclinic tangencies cause it to be nearly rank deficient. Both of these factors make **M** difficult to invert when N is large.

3.1. Chaos implies ill-conditioning of **M**

The ill-conditioning of **M** becomes apparent if we attempt to take advantage of the banded structure of **M** and invert it using a simple algebraic technique. Starting at the upper left and working toward the right, the eigenvalues accumulate and diverge exponentially when the underlying dynamics are chaotic. This is illustrated by iterating eq. (11) and writing down the resulting series, which gives

$$\Delta_t = \left(\prod_{i=1}^{t-1} J_i\right)\Delta_1 + \epsilon_{t-1} + \sum_{i=1}^{t-2}\left(\prod_{k=i+1}^{t-1} J_k\right)\epsilon_i. \quad (14)$$

The Lyapunov numbers Λ_i, $i = 1, 2, \ldots, d$ are defined as the magnitudes of the eigenvalues of $\lim_{t\to\infty}(\prod_{i=1}^{t} J_i)^{1/t}$. Thus by definition, on average the magnitude of the largest eigenvalue of the product matrix $\prod_{i=1}^{t-1} J_i$ grows as $(\Lambda_1)^{t-1}$. If the underlying dynamical system is chaotic, then $\Lambda_1 > 1$, and $\prod_{i=1}^{t-1} J_i$ becomes exponentially more ill-conditioned as t grows large. Thus direct algebraic solution is numerically unstable for large t when f is chaotic.

3.2. Solution by manifold decomposition

Since **M** comes from a dynamical system, we know a great deal about its structure and can exploit this to overcome the problem of ill-conditioning. We do this by decomposing the solution of the problem along the stable and unstable manifolds, a technique which we call *manifold decomposition*. This technique was originally introduced as a heuristic by Hammel [11]. The following discussion provides a justification of manifold decomposition and makes precise the sense in which it is an approximation for the shadowing problem. Manifold decomposition is also closely related to the maximum likelihood method for noise reduction that we introduced earlier, as discussed in section 6.

The determinism constraint of eq. (2) means that with perfect numerical precision the initial condition is the only degree of freedom in the problem – with no noise the entire trajectory is determined by the initial condition. This is manifest from the fact that Δ_1 is the only unknown on the right-hand side of eq. (14). Suppose we choose an initial displacement vector Δ_1 which lies precisely along the stable manifold of f. Suppose for the moment that the noise level is zero so that under iteration the stable manifold at $\hat{x}_1$ goes into the stable manifold at $\hat{x}_2$. Because we are on the stable manifold, for large t, $\|(\prod_{i=1}^{t-1} J_i)\Delta_1\| \approx (\Lambda_s)^t\|\Delta_1\|$, where Λ_s is the largest Lyapunov number less than one. This implies that along the stable manifold for large t the solution for Δ_t in eq. (14) becomes independent of Δ_1, and can be stated directly in terms of ϵ_i and J_i, which are known quantities. The only significant contributions come from terms in the sum with $j \approx t$, which implies that points at times much earlier than t have little effect on the noise reduction process. Similar considerations apply going backward in time along the unstable manifold.

Since f is invertible we can solve eq. (11) for Δ_t in terms of Δ_{t+1} and iterate backward. This gives the following equation for Δ_t in terms of future values:

$$\Delta_t = \left(\prod_{k=t}^{N-1} J_k^{-1}\right)\Delta_N - \sum_{i=t}^{N-1}\left(\prod_{k=t}^{i} J_k^{-1}\right)\epsilon_i, \quad (15)$$

where J_k^{-1} is the inverse of J_k. If we pick Δ_N along the unstable manifold, for large t, $\|(\prod_{k=t}^{N-1} J_k^{-1})\Delta_N\| \approx (\Lambda_u)^{t-N}\|\Delta_N\|$, where Λ_u is the

smallest Lyapunov number greater than one. Just as before, Δ_t is dominated by the small t terms on the right-hand side, and so for large $(N-t)$ is independent of the choice of Δ_N.

Thus, for a point in the middle of the time series with t sufficiently greater than one so that $(\Lambda_s)^t \approx 0$ and sufficiently less than N so that $(\Lambda_u)^{t-N} \approx 0$, we have

$$[\Delta_t]_{\text{stable}} \approx \left[\epsilon_{t-1} + \sum_{i=1}^{t-2}\left(\prod_{k=i+1}^{t-1} J_k\right)\epsilon_i\right]_{\text{stable}}, \quad (16)$$

$$[\Delta_t]_{\text{unstable}} \approx \left[-\sum_{i=t}^{N-1}\left(\prod_{k=t}^{i} J_k^{-1}\right)\epsilon_i\right]_{\text{unstable}}. \quad (17)$$

$[\ ]_{\text{(un)stable}}$ denotes the projection along the (un)stable manifold. As long as the stable and unstable manifolds are transverse these two equations can be combined to solve for Δ_t, producing an approximate noise-reduced orbit. This orbit is accurate to $(\Lambda_s)^t$ or $(\Lambda_u)^{t-N}$, whichever is greater.

This suggests an efficient numerical method for inverting the matrix. We pick a vector Δ_1 along the stable manifold, arbitrarily setting $[\Delta_1]_{\text{unstable}} = 0$. We iterate this forward using eq. (11). For sufficiently large times the initial (generally incorrect) choice of $\|\Delta_1\|$ will become irrelevant, and in principle we should have the proper correction to the noisy orbit along the stable manifold. We then repeat this by iterating backward along the unstable manifold, and we have our answer.

Of course, in practice there is a problem: If Δ_1 is on the stable manifold of $\hat{x}_1$, then $J_1\Delta_1$ will generally be on the stable manifold of $f(\hat{x}_1)$, but it will *not* be on the stable manifold of $\hat{x}_2$, since $\hat{x}$ is a noisy orbit. Furthermore, this deviation gets exponentially worse as we iterate forward, causing Δ_i to leave the stable manifold and relax onto the unstable manifold. To solve this problem at each step we *project* Δ_2 onto the stable manifold of $\hat{x}_2$. This stabilizes the procedure, but it has the disadvantage that we are now making errors, which can be substantial when the noise is large. As a result, when we are done we may have

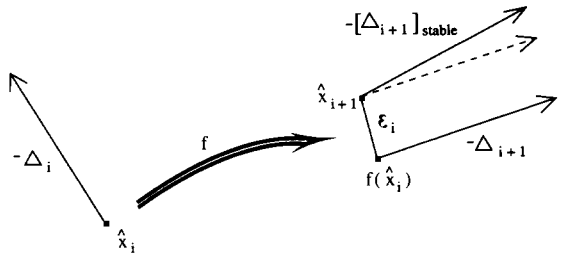

Fig. 1. An illustration of the approximation needed to iterate eq. (11) stably. $\Delta_i = \hat{x}_i - x_i$ represents the correction at $\hat{x}_i$. Assume Δ_i is on the stable manifold of $\hat{x}_i$. Then $J_i\Delta_i$ is also on the stable manifold of $f(\hat{x}_i)$. However, since the deviation from determinism, ϵ_i, is not, $\Delta_{i+1} = J_i\Delta_i + \epsilon_i$ is not on the stable manifold. The dashed vector is Δ_{i+1} transported so that its base is at $\hat{x}_{i+1}$, and the solid vector is Δ_{i+1} projected onto the stable manifold of $\hat{x}_{i+1}$. The difference between these two vectors generates an error term in the inversion of **M**.

significant errors in computing the inverse of the matrix **M**.

In stretches that are free of homoclinic tangencies manifold decomposition works well, but when the angle between the stable and unstable manifolds becomes too small, it blows up. There are two reasons for this:

(1) Since the stable and unstable manifolds are almost parallel, Δ_i is poorly resolved along the perpendicular direction.

(2) Homoclinic tangencies typically occur at places where the stable or unstable manifolds make sharp turns. Thus the angle of either the stable or unstable manifold is usually sensitively dependent on position in the neighborhood of a homoclinic tangency. This means that projection can cause a very large error for even very small noise levels. This problem is illustrated in fig. 1.

3.3. *Solution by singular value decomposition*

The net effect of homoclinic tangencies is to cause the matrix to become nearly rank deficient. A natural way to solve this is singular value decomposition, which is a numerical algorithm designed to invert nearly rank deficient matrices. This works quite well for short trajectory segments. Our experience is that when we have

sufficient computer time and space the singular value decomposition does a good job of inverting **M** and the resulting noise reduction is stable even with fairly large initial noise levels. However, inverting an $n \times n$ matrix with singular value decomposition requires the order of n^3 operations and n^2 memory locations. When n is large this is slow and requires a large computer memory. In contrast, the manifold decomposition technique described above requires only order n operations and order n memory locations, but is not as stable. Manifold decomposition is much faster than singular value decomposition, but is less accurate in the presence of homoclinic tangencies.

As a compromise, we use a mixture of manifold decomposition and singular value decomposition. We begin by computing the invariant manifolds at each point, searching for points with small angles between the stable and unstable manifolds. We perform singular value decomposition to invert **M** for short trajectory segments (typically ≤ 50 points) centered on these points. We then apply manifold decomposition to the entire time series. We check to see whether the resulting trajectory is deterministic; if not, we perform singular value decomposition again on the places where it is not. For particularly bad homoclinic tangencies it can be useful to simply add noise to $\hat{x}$ for points in a short trajectory segment centered on the tangency. Another useful manoeuver is to monitor how far each point wanders from its original noisy value. If a point strays substantially further than the original noise level, then something has obviously gone wrong, most likely due to a tangency. In this case the point can be reset to its original value and given another chance to settle into a copacetic position.

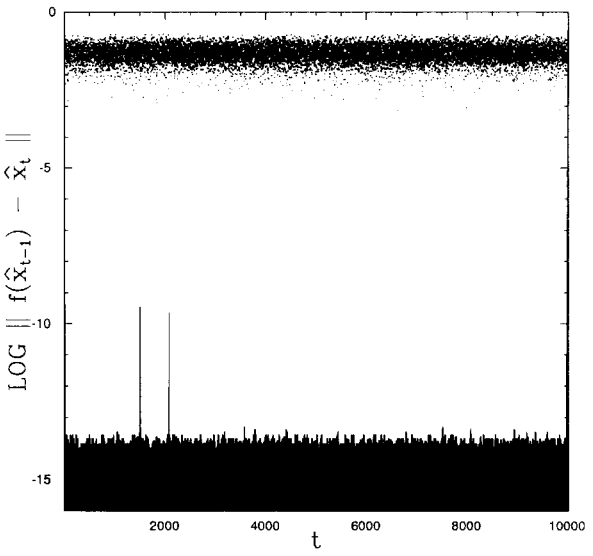

Fig. 2. Deterministic error versus time for a trajectory of the Hénon map. The points are the original noisy time series. (Uniformly distributed noise of magnitude 0.04 was added to both components of the deterministic time series.) After two iterates of the mixed manifold decomposition and singular value decomposition technique described in the text, the noise has been reduced to the level indicated by the solid curves at the bottom. Logarithms are taken to base 10.

4. Numerical results

4.1. When the dynamical system is known

To test the performance of our method we have performed many experiments that proceed along the following lines: We generate a "noiseless" trajectory segment $\{x_t\}$ by iterating a dynamical system at double precision. A random number generator adds noise to the orbit. Then we perform noise reduction, pretending that we did not know the original orbit. Two measures test the quality of our results: The *absolute error* $= \|\hat{x}_t - x_t\|$ is the deviation between a smoothed point and a true point, and the *deterministic error* $= \|\hat{x}_{t+1} - f(\hat{x}_t)\|$ is the discrepancy between the iterate of a smoothed point and the next smoothed point.

Fig. 2 shows the deterministic error obtained from an experiment using the Hénon map, $x_{t+1} = 1 + y_t - 1.4x_t^2$, $y_{t+1} = 0.3x_t$, with a trajectory segment of length 10000, and a 5% initial noise level. The deterministic error is reduced by roughly 13 orders of magnitude, or 130 dB.

Fig. 3 shows the absolute error obtained in an experiment with a trajectory segment of length

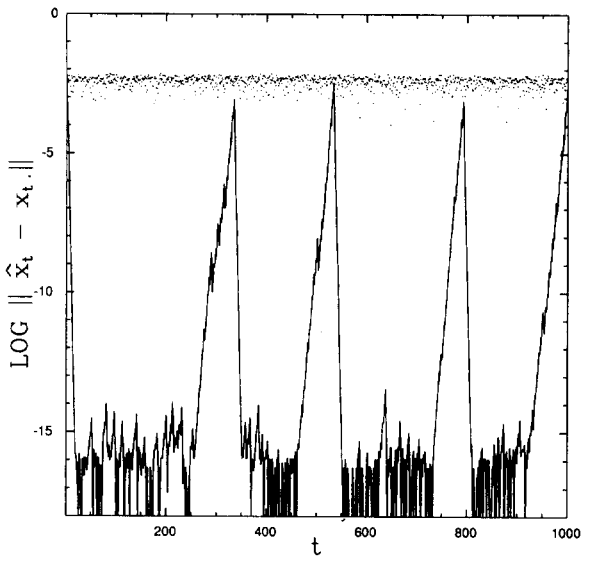

Fig. 3. Absolute error versus time for a trajectory of 1000 points from the Hénon map. Initial noise level indicated by scattered points = 0.005. Solid curve is the noise level after two iterates of our technique.

1000 and a 0.5% noise level. The average of the logarithm of the absolute error is -11.4, indicating an average noise reduction of more than 100 dB. There are regions of the smoothed trajectory that approach the true trajectory to within machine precision. However, in other regions that are close to homoclinic tangencies, the smoothed trajectory makes deviations from the true orbit that are on the order of the original noise level. Each of these generates a characteristic peak; the slope on the left is roughly the positive Lyapunov exponent, and the slope on the right is roughly the negative Lyapunov exponent[#3]. The deterministic error for the same experiment (shown later in fig. 8) indicates that the smoothed trajectory is deterministically clean to the level of machine precision over the entire segment. The initial noise level was large enough to cause our algorithm to locate another strictly deterministic trajectory segment closer to the noisy data than the original true trajectory. It should be noted that not all homoclinic tangencies cause peaks in the absolute error.

In the case of dynamical noise there is no true trajectory, so the absolute error cannot be defined. A plot of the deterministic error displays the same type of peaks that are present in the absolute error for the case of observational noise. Again there are regions where the noise is reduced to the level of machine precision, but the effects of homoclinic tangencies are more disruptive for dynamically noisy trajectories. Because the noise perturbs the evolution of the system, it gets magnified by the dynamics and accumulates. In some regions the linear approximations used in the formulation of our method begin to break down, and consequently our algorithm fails to substantially reduce the deterministic error there.

4.2. *Learning the dynamics*

Up until now we have assumed that we knew the dynamics f of the noiseless signal, and could use this knowledge to remove the noise. However, in most practical applications f is unknown and must be learned directly from the data. In fact, our interest in the noise reduction problem grew out of our work on forecasting chaotic time series. There are a variety of methods for learning the dynamics directly from data [1, 4, 5, 7, 8, 10, 14]. All of these methods employ similar principles. They begin by embedding the data in a state space [16, 18] and then attempting to approximate the graph of a function that maps present states into future states. The principal difference between the techniques discussed in refs. [1, 4, 5, 7, 8, 10, 14] lies in the choice of nonlinear function representations. Function representations can be divided into two categories: *Global representations*, such as polynomials, which consist of terms that apply uniformly throughout the domain, and *local representations*, in which the domain is broken up into local neighbor-

[#3]The Lyapunov exponents are the logarithms of the Lyapunov numbers.

hoods. We refer the reader to ref. [10] for a review.

The ability to reduce noise is ultimately limited by the accuracy of the approximation to the true dynamics f. Unless the approximation to f is an improvement over a (global) linear model, little improvement in noise reduction is possible over the usual moving average smoothing technique. The ability to learn the dynamics is usually the limiting factor for achieving noise reduction in practical applications.

As stated in section 1, when we must learn f, there are two basic cases to consider: The first case is where we are lucky, and have some noiseless data with the same dynamics. For example, we might have musical recordings made in a studio under ideal conditions, and other similar recordings made in the field under noisy conditions. We can then use the clean data to learn f, and apply this f to the noisy data. We examine this case in section 4.2.1. The second case is where we are unlucky, and have only noisy data so that we must learn f from the noisy data. We examine this case in section 4.2.2.

4.2.1. Learning from clean data

Fig. 4 illustrates the use of local function approximation to learn f from clean data. As a sample dynamical system we use the Ikeda map

$$(x_{t+1}, y_{t+1}) = (1 + \mu(x_t \cos s - y_t \sin s), \mu(x_t \sin s + y_t \cos s)), \qquad (18)$$

where $s = 0.4 - 6.0/(1 + x^2 + y^2)$ and $\mu = 0.9$. Fig. 4c shows 5000 successive iterates plotted in the (x, y) plane. Fig. 4a displays the same 5000 points after adding noise. The noise was generated with a random number generator, uniformly distributed with maximum magnitude of 0.10 for each component; this is roughly 5% of the signal's dynamic range or 20% of its standard deviation. We iterated the Ikeda map at double precision to generate 50000 clean data points, which were used as a learning set to approximate f. To calculate a local approximation of the Ikeda map near a given point q we searched through the learning set to find the 50 nearest neighbors of q. These 50 points define the local domain of f; their values one time step later define its range. To approximate f we fit a quadratic mapping $\hat{f}$ from the domain to the range using least-squares approximation over the neighboring points. The learned maps were then used as approximations to the system dynamics f, applying the optimal shadowing noise reduction algorithm. The result is shown in fig. 4b.

4.2.2. Learning from noisy data

Learning the dynamics directly from noisy data is a more challenging problem, and the strategy is somewhat different. The presence of noise favors more global approximation schemes. For local representations the neighborhood size must be increased. This increases the approximation error, but also averages over the noise. The best neighborhood size makes a compromise between these two competing effects. This depends on both the noise level and the complexity of the dynamics. The best way to find the optimal neighborhood size is to use a search algorithm to optimize the average noise reduction, but for the purposes of this paper we may do with trial and error.

We use an iterative procedure. We approximate f, apply the noise reduction algorithm, and then approximate f again using the smoothed data, repeating this until the results cease to improve. The rationale is as follows. Providing we use a sufficiently large approximation neighborhood, the learned representation of f averages over the behavior of neighboring points, providing an initial noise reduction. Application of the noise reduction algorithm averages together points from other times as well, including those from distant parts of the state space, thus amplifying the initial noise reduction.

Fig. 5 demonstrates the result of noise reduction with no prior knowledge, learning f directly from the noisy time series. Fig. 5a shows

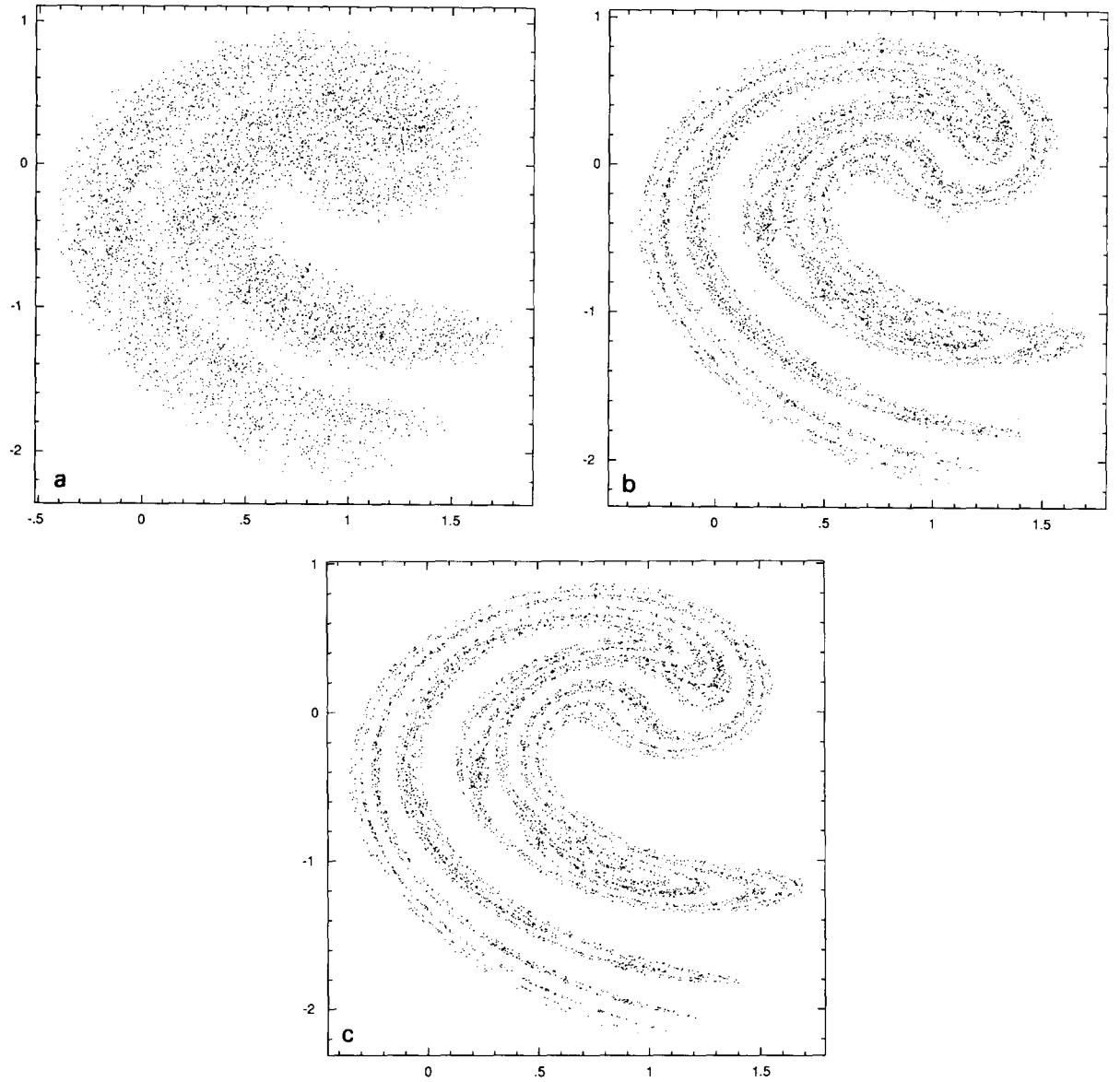

Fig. 4. (a) 5000 points of a noisy time series, made by adding uniformly distributed noise of magnitude 0.10 to the data of (c). (b) The attractor reconstructed by the optimal shadowing technique described in the text. f was learned from a data set of 50 000 clean points (see (c)), using local quadratic fits over 50 point neighborhoods. The stray points are caused by approximate homoclinic tangencies. Compare to the peaks shown in fig. 3. (c) A plot of 5000 successive iterates of the Ikeda map with $\mu = 0.9$.

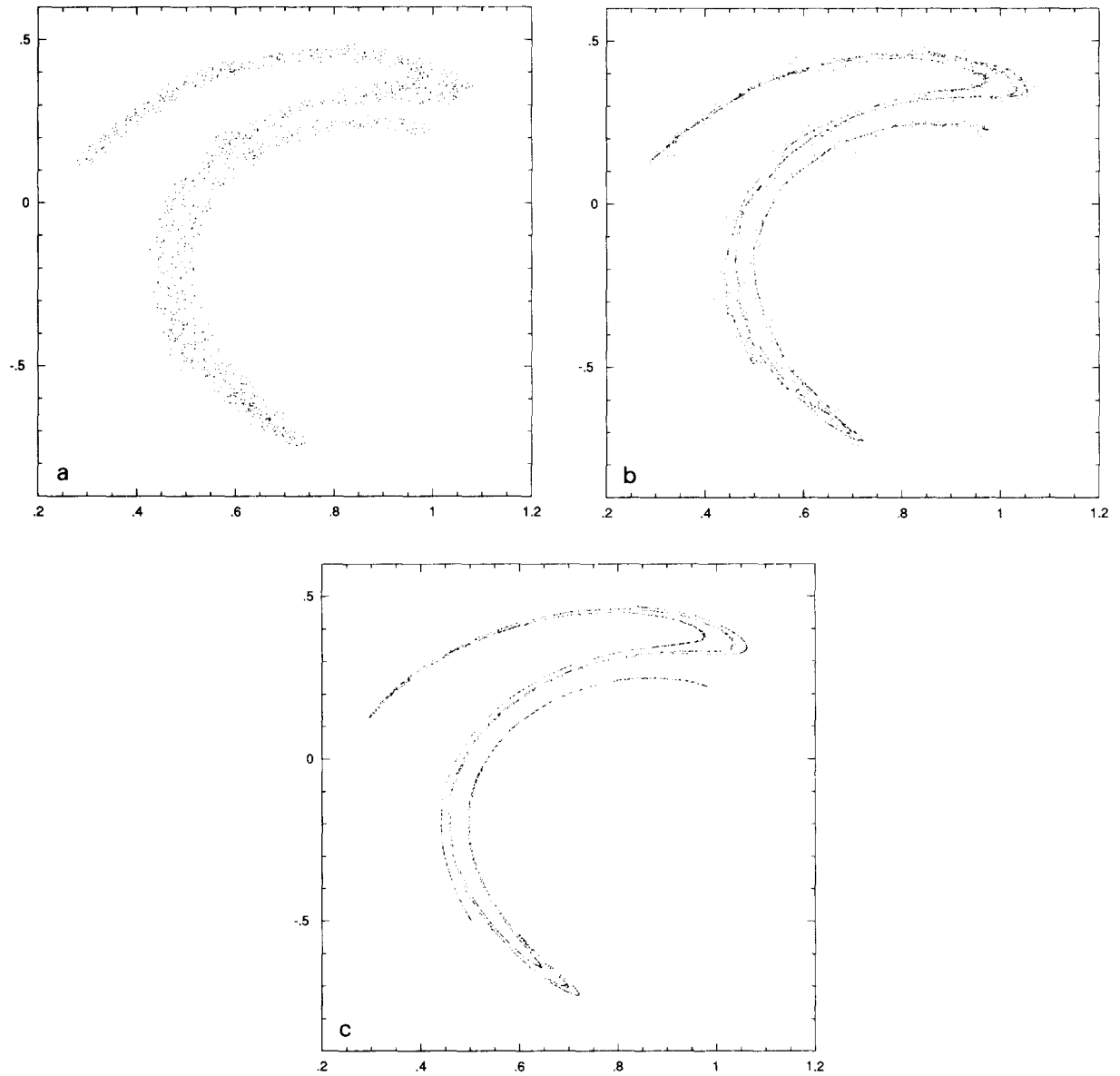

Fig. 5. (a) A phase plot showing 1500 points of a noisy time series, obtained by adding uniformly distributed noise of magnitude 0.02 to each component of the data shown in (c). (b) The result of applying our noise reduction method to the data in (a). See text for details of the technique. (c) A plot of 1500 successive iterates of the Ikeda map with $\mu = 0.7$. These are the "true" data for (a), before adding noise.

a phase plot of a portion of the 5000 points in the noisy time series; from the point of view of the computer algorithm its origin is unknown. Local quadratic fits were made over neighborhoods of 50 points to approximate the dynamics. We iterated the procedure of learning the dynamics and reducing the noise three times to produce the noise reduced phase plot shown in fig. 5b. For comparison, the original uncontaminated phase plot is shown in fig. 5c. Comparing (b) and (c), there are significant discrepancies; as usual, there are some stray points caused by homoclinic tangencies, and there is also some segmenting caused by inaccuracies in the approximation algorithm. However, the majority of the points correspond in detail.

Reducing noise when learning from noisy data is considerably harder than learning from clean data. Typically the initial noise levels must be lower in order to get the method to converge, although as shown in fig. 5 it can be successful with initial signal to noise ratios as low as 50.

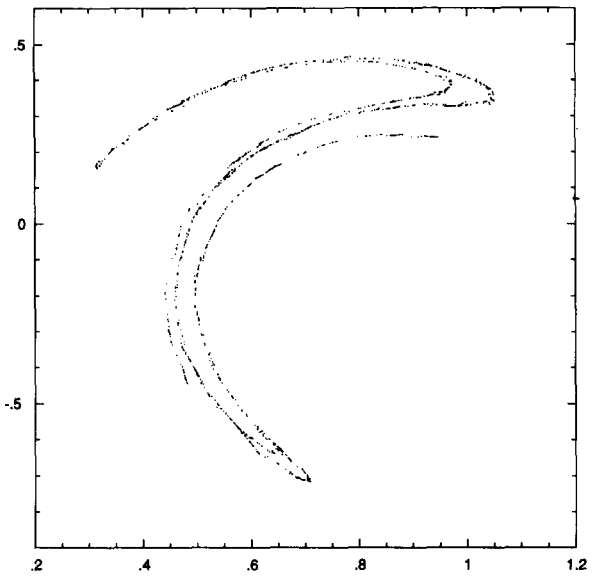

Fig. 6. The result of applying the simple statistical noise reduction technique described in section 5 to the data of fig. 5a. The overall features are quite similar, but in contrast to the reconstruction of fig. 5b, there is no detailed correspondence in the points of the time series.

5. Statistical noise reduction

Up until now we have addressed the problem of removing noise from each point in a time series. Given a noisy time series $\{y_t\}$, we want to find the corresponding deterministic time series $\{x_t\}$. This might be called *detailed noise reduction*. In some cases, however, we may not care whether we have a point by point correspondence. Instead, we may only be interested in statistical properties, such as the power spectrum, the dimension, or the shape of the attractor. For this purpose any clean time series with the same statistical properties as the "true" time series is good enough, even if there is no point by point correspondence between the two time series. We call this *statistical noise reduction*.

A very simple technique for statistical noise reduction is to fit a model to the noisy data, pick an arbitrary initial condition, and iterate it to create a new time series. In many cases the statistical properties of the surrogate time series are much closer to those of the true time series than those of the original noisy data.

We applied this procedure to the noisy data set of fig. 5a, using local quadratic fits for our model. The results are shown in fig. 6. Although there are noticeable discrepancies, the attractor is reproduced fairly well. In some respects the reproduction is superior to that of fig. 5b; however, there is no longer a detailed point by point correspondence to the set of "true" points shown in fig. 5c.

This approach to statistical noise reduction can be dangerous when the dynamics displays sensitive dependence on parameters [9, 15]. For example, in some cases the set of parameters that have stable periodic attractors are dense within those with chaotic attractors, so that a small change in f can cause a change from chaotic behavior to periodic behavior. In this case the errors accumulated from the noise may cause an approximation to the dynamics whose statistical properties are

radically different from those of the true time series. Detailed noise reduction avoids this problem by insuring that the shadowing orbit stays close to the noisy orbit. Thus, application of detailed noise reduction is more reliable than the simple method above, even when the final goal is only statistical noise reduction. But when statistical noise reduction works, it often works very well.

The statistical noise reduction procedure described above is closely related to the "bootstrapping" approach to extending time series suggested in refs. [4, 10]. This approach is based on the fact that good function approximation schemes may give information about the underlying functional form that is accurate on length scales smaller than the typical separation between data points. Fitting a model and iterating can be used to artificially extend a data set, a procedure that can be very useful, for example, in computing fractal dimension. Casdagli [4] has demonstrated this method by approximating the invariant measures of several attractors based on only a small amount of data.

6. Comparison with previous methods

In 1988 we proposed a method for reducing observational noise based on maximum likelihood, and Kostelich and Yorke [13] independently proposed a method based on reducing least-mean-squares error (which in this sense bears some similarities to the optimal shadowing technique discussed here). Hammel [11] has recently proposed a method for noise reduction based on his earlier work on the shadowing problem. In this section we compare these methods.

6.1. Maximum likelihood

In an earlier paper we introduced a method for solving observational noise problems based on the principle of maximum likelihood [10]. This method may be viewed as a nonlinear generalization of the usual moving average smoothing technique: In a simple moving average the points of a time series around a given point are simply averaged together. Averaging acts as a low-pass filter, which removes the high-frequency components. The noise reduction obtainable from this technique is severely limited, however, since averaging over too large an interval causes distortion of the signal. The problem is that simple averaging fails to take the dynamics of the signal into account. Our method improves on the usual moving average noise filter by using a model of the dynamics to transport points from different times to a common time, and then averaging them together. As a result many more points can be averaged together without distortion of the signal, improving the noise reduction.

When points are transported from the past to the present, errors may be amplified. This amplification is usually direction dependent. In a chaotic dynamical system, on average the errors grow exponentially along the unstable manifold, and contract exponentially along the stable manifold. The opposite is true of points transported from the future to the present. Thus points from the past are accurate along the stable manifold, while those from the future are accurate along the unstable manifold, as illustrated in fig. 7. For good noise reduction it is essential to take this into account.

Our method does this by assigning weighting matrices to the contribution of each point in each direction according to the derivatives of the dynamical system. To derive the proper weighting matrices, we assume that the noise fluctuations are independently Gaussian distributed. We then do a local linearization of the dynamics, and write down the rules for pulling back and pushing forward probability distributions to different points in time. Consider a segment of the noisy time series $\{y_t\}$ running from time $t-\alpha$ to time $t+\beta$. If we transport all of these points to time t, then the joint probability density function is a

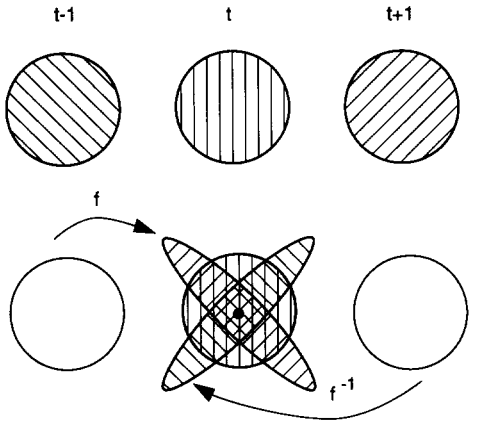

Fig. 7. *Nonlinear smoothing*. The circles represent noisy measurements of a deterministic trajectory at three different times. As successive measurements are transported to the same point in time, the associated noise probability distributions distort according to the local derivatives of the dynamical system. The true state should lie somewhere in the intersection of the three regions. Averaging the transported measurements at time t makes it possible to produce a better estimate of the true state at time t. However, to get a good estimate it is important to introduce the proper weighting factors to take the distortion into account.

product of Gaussians of the form

$$\bar{P}\left(f^{\alpha}(y_{t-\alpha}),\ldots,f^{-\beta}(y_{t+\beta})\right) \propto \prod_{j=\alpha}^{j=-\beta} \exp\left(-\frac{1}{2\sigma^2}\left\|\frac{f^j(y_{t-j})-x_t}{\mathrm{d}f^j(y_{t-j})}\right\|^2\right). \tag{19}$$

The factor in the numerator in the exponent is the difference between the transported value of y and the true value x_t. $\mathrm{d}f$ is the derivative, which takes into account the local linear distortion of the probability. We refer the reader to ref. [10] for the details.

To find an estimate of the true solution x_t we assume that the particular sequence of observations $\{y_i\}$ lies at the maximum of $\bar{P}$. (This is called the method of maximum likelihood.) Differentiating with respect to x_t in eq. (19), and setting this equal to zero gives an estimate of x_t:

$$x_t = \left(\sum_{j=\alpha}^{j=-\beta} \Theta_j\right)^{-1} \sum_{j=\alpha}^{j=-\beta} \Theta_j f^j(y_{t-j}), \tag{20}$$

where

$$\Theta_j = \left(\left[\mathrm{d}f^j(y_{t-j})\right]^{\mathrm{T}} \mathrm{d}f^j(y_{t-j})\right)^{-1}.$$

Θ_j is the Jacobian matrix of the jth iterate of f times its transpose, and is a $d \times d$ symmetric matrix that depends on y_{t-j}. It contains weighting factors that depend on local expansion and contraction rates, and take into account the distortion of the noise as it is transported to different times, as shown in fig. 7. The directions in which the noise is compressed contain more useful information, and receive higher weight, while those in which it is expanded contain less useful information and receive less weight.

Thus, the maximum likelihood analysis yields a nonrecursive, nonlinear moving average filter. Note that in practice if α and β are too large the assumption of local linearity may break down. This can often be solved by making α and β smaller (e.g. 1 or 2). The resulting estimate can be improved by making successive passes through the entire time series, setting $y_t = x_t$ after each pass.

This can be compared to the least-squares technique we have introduced here by making an approximation. The summation in eq. (20) uses all the available information. However, due to the exponential expansion and contraction, when multiplied by the weighting matrices the end points make the largest contributions. Suppose we neglect all but the endpoints, considering only the two terms with $j = \alpha = t - 1$ and $j = \beta = t - N$. Eq. (20) gives

$$x_t = (\Theta_{t-1} + \Theta_{t-N})^{-1} \times \left[\Theta_{t-1} f^{t-1}(y_1) + \Theta_{t-N} f^{t-N}(y_N)\right]. \tag{21}$$

In the limit that $1 \ll t \ll N$ the matrix $(\Theta_{t-1} +$

$\Theta_{t-N})^{-1}\Theta_{t-1}$ becomes an operator that projects its input onto the stable manifold[#4]. Similarly, the matrix $(\Theta_{t-1}+\Theta_{t-N})^{-1}\Theta_{t-N}$ projects its input onto the unstable manifold.

To make the correspondence with the least-squares method, we must change notation and assume that the noise is small so that we can expand in a Taylor's series. First, assume that we are making successive passes through the time series so that y on the right-hand side of eq. (21) can be treated as a trial solution $\hat{x}$. (Set $y=\hat{x}$.) Referring to eq. (9), by definition we have $f(\hat{x}_1)=\hat{x}_2-\epsilon_1$. Expanding this recursively we get $f^2(\hat{x}_1)=f(f(\hat{x}_1))=f(\hat{x}_2-\epsilon_1)$. Applying Taylor's theorem gives $f^2(\hat{x}_1)\approx f(\hat{x}_2)-J_2\epsilon_1=\hat{x}_3-\epsilon_2-J_2\epsilon_1$. Further recursive application leads to

$$f^{t-1}(\hat{x}_1)=\hat{x}_t-\epsilon_{t-1}-\sum_{i=1}^{t-2}\left(\prod_{k=i+1}^{t-1}J_k\right)\epsilon_i. \qquad (22)$$

Substitute this into eq. (21), remembering that $\Delta_t=\hat{x}_t-x_t$. Applying $(\Theta_{t-1}+\Theta_{t-N})^{-1}\Theta_{t-1}$ projects this onto the stable manifold. This is the same as eq. (16). Similarly, since $t-N$ is negative, examination of $f^{t-N}(\hat{x}_N)$ leads to eq. (17), which gives the correction along the unstable manifold. By making some approximations and rearranging terms, we have converted the nonrecursive noise filter of eq. (20) to a recursive filter of the form of eq. (11).

Thus, with these approximations the least-squares and the maximum likelihood estimates lead to the same results. This is perhaps surprising, in view of the fact that the maximum likelihood procedure is derived for observational noise, whereas the least-mean-square shadowing procedure is derived for the more general case of dynamical noise. It is interesting that we can compute a shadowing orbit in the presence of dynamical noise by simply taking successive weighted averages. According to the lore of statistical estimation agreement between maximum likelihood methods and least-squares methods is a common occurrence.

Note added: A technique that bears some relationship to our maximum likelihood procedure was suggested by A.S. Pikovsky in 1986 [17]. This technique uses kernel density estimation to approximate the dynamics, and averaging forward in time to reduce noise. It is similar to our method in one dimension, but since it only uses forward averaging, without error weighting matrices, it is quite different in more than one dimension.

6.2. *The method of Hammel*

The shadowing problem for Axiom-A dynamical systems was originally studied by Anosov and Bowen [2, 3]. This work was amplified by Hammel et al., who extended it to general non-hyperbolic dynamical systems [12]. Recently Hammel [11] demonstrated how the techniques used in the proof of the shadowing lemma can be used to reduce noise in chaotic data. Using a linear approximation to the deterministic constraint that defines a shadowing orbit, he noted that eq. (11) could be used to reduce noise, and then developed the first implementation of what we call the manifold decomposition method. For the noise reduction problem he sets $[\Delta_0]_{\text{stable}}=0$ and iterates forward, and sets $[\Delta_N]_{\text{unstable}}=0$ and iterates backward, and then combines the two vectors to reduce the noise. Our arguments give this procedure a rigorous justification, and make precise the extent to which it deviates from optimality.

6.3. *The method of Kostelich and Yorke*

Another least-squares approach to the noise reduction problem is due to Kostelich and Yorke (KY) [13]. Based on apparently ad hoc considera-

[#4] When the system is chaotic, for large positive t, Θ_t amplifies directions along the stable manifold, and damps (effectively to zero) directions along the unstable manifold. The normalization factor in front makes this a projection operator.

tions, they suggest finding x to minimize the sum

$$S_{\mathrm{KY}} = \sum_{t=1}^{N} \left[\|x_t - y_t\|^2 + \|x_t - f(x_{t-1})\|^2 + \|x_{t+1} - f(x_t)\|^2 \right], \tag{23}$$

where terms which include subscripts outside the range $1 \le t \le N$ are set to zero. They also assume that f is a local linear function.

This can be compared to the other methods discussed here in two ways. First, it is easily compared to our least-squares approach by examining the differences between S_{KY} and our minimization function, eq. (7). The first term in S_{KY} is the deviation from the noisy orbit, while the second two are deviations from determinism. The x_t that minimizes the sum S_{KY} is an ad hoc compromise between one that is deterministic and one that stays close to the noisy orbit. As a result, the orbit is neither very deterministic nor very close to the true orbit. In contrast, we propose finding an orbit x_t that is simultaneously exactly deterministic, and as close as possible to the noisy orbit.

Another way to gain insight into the difference between these approaches is to compare this with the maximum likelihood method discussed above. As a point of comparison, we construct an effective likelihood function for this method, by noting that finding the x_t that minimizes the sum S_{KY} is equivalent to finding the x_t that maximizes the likelihood of

$$\bar{P}\left(f(x_{t-1}), x_t, f^{-1}(x_{t+1})\right) \propto \exp\left[-\|f(x_{t-1}) - x_t\|^2 - \|x_t - y_t\|^2 - \|f(x_t) - x_{t+1}\|^2 \right]. \tag{24}$$

Compare this with eq. (19). Apart from an arbitrary constant, if we set $\alpha = \beta = 1$, there are two significant differences between eq. (24) and eq. (19). First, in eq. (24) only the middle term is referenced to the noisy value y_t, while in eq. (19) all the terms are referenced to the noisy value.

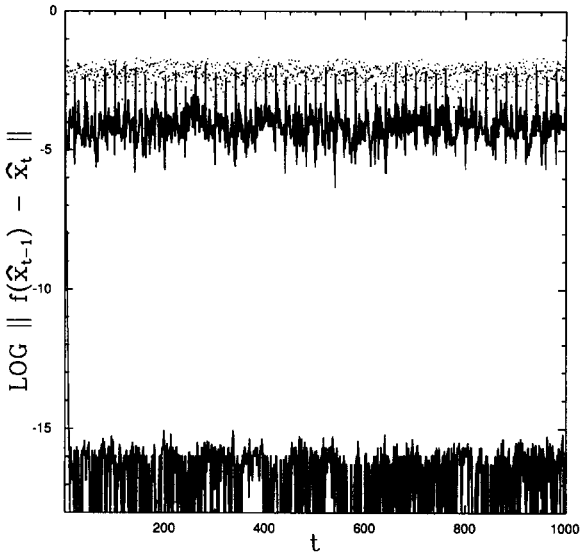

Fig. 8. Deterministic errors for the same data in fig. 3. The solid curve at the bottom is the noise level after two iterates of our technique. The solid curve at the top indicates the performance of the KY algorithm on the same data set.

Second, and more important, eq. (24) does not contain the weighting factors based on derivatives. Without them the probabilities do not transform correctly. As a result, the contributions from the past, present, and future are weighted uniformly in each direction. This is dangerous, since errors are amplified along directions where the derivative is large. Including the derivative weighting factor is particularly important in chaotic systems, where errors are exponentially expanding along the unstable manifold and exponentially contracting along the stable manifold. The derivative factor not only makes the expression technically correct, but from a practical point of view ensures that each term is given its proper weighting according to its accuracy. By dropping the derivative factor, their method fails to exploit the geometry of chaotic systems.

A quantitative comparison of the optimal shadowing technique of section 2 and that of Kostelich and Yorke (KY) is given in fig. 8. For this comparison we have generalized their technique by assuming an exact knowledge of the system dynamics, i.e. in eq. (23) we let f be the true

Table 1
A comparison of errors for the initial noisy data, the average results using the KY noise reduction method, and the optimal shadowing approach described here. The deterministic error is the discrepancy between successive iterates, and the absolute error is the difference compared with the "true" data. The error measures are averages of the logarithms over the entire trajectory segment of 1000 points. The logarithms are in base 10; reducing the absolute error from −2.4 to −13.8 is greater than a 100 dB noise reduction.

	Initial noise level	Kostelich–Yorke	Optimal shadowing
log absolute error	−2.4	−3.4	−13.8
log deterministic error	−2.2	−4.1	−16.7

function describing the dynamics rather than a learned local linear approximation. The overall performance of the KY algorithm and the optimal shadowing algorithm is summarized in table 1. The noise reduction obtained through the optimal shadowing technique is more than ten orders of magnitude greater.

In all fairness, it should be noted that the KY algorithm was developed as a tool for the analysis of experimental time series, where the dynamics is unknown. When the dynamics must be learned from the noisy data, the difference between the two techniques is not as dramatic. In this case, both techniques are limited by the accuracy of the learned model equations for the dynamics. Our imposition of determinism by using constrained least-squares seems to make our method converge better under iteration, as discussed in section 4.2.

6.4. *Summary*

We have experimented with several different two-dimensional mappings under a variety of conditions. Our numerical experience can be summarized roughly as follows: Performance of the pure manifold decomposition technique and the full maximum likelihood technique of eq. (19) are roughly equivalent in accuracy, although manifold decomposition is considerably more efficient in terms of computer time. Improvements in accuracy can be made by using the full optimal shadowing equations with singular value decomposition to resolve approximate homoclinic tangencies, but with some cost in computer time. Performance of the KY method is poorer than any of the other three methods, and it is also expensive in terms of computer time.

7. Limits to noise reduction

The arguments at the beginning of section 3.2 imply that the maximum obtainable noise reduction depends on the position in the time series. Near the beginning of the time series the obtainable noise reduction is proportional to $(\Lambda_s)^t$, where Λ_s is the largest Lyapunov number less than one. This is because for errors pushed forward from the past this is the slowest rate of contraction along the stable manifold. Similarly, for points pulled back from the future the errors shrink along the unstable manifold proportional to $(\Lambda_u)^{(N-t)}$, where Λ_u is the smallest Lyapunov number greater than one.

For a chaotic system the maximum obtainable noise reduction grows *exponentially* with the length of the time series. Ironically, this is much better than for nonchaotic systems, where the obtainable noise reduction typically grows according to the central limit theorem as the square root of the length of the time series.

The bad aspect for chaotic systems is that it is not possible to reduce noise significantly at the ends of the time series. This implies that there is a strong limit to how much noise reduction can be used to improve prediction. For the purposes of prediction, the future of the present state is

unknown. Noise reduction can be used to reduce errors along the stable manifold, but not along the unstable manifold. While noise reduction can be used to get a better approximation for f, it cannot reduce all the error on the present state.

For regular systems, in contrast, the obtainable noise reduction is more or less independent of the position in the time series. For regular systems noise reduction *can* be used to give significant improvements in predictions, but for chaotic systems it cannot. Also, regular systems do not suffer from the problems caused by homoclinic tangencies.

Continuous flows always have at least one Lyapunov number equal to one. This implies that we cannot reduce noise significantly in directions along trajectories. This is not usually a problem, however, since we are typically concerned with reproducing the form of a trajectory, and not with the detailed position of the sampling points on the trajectory. In high-dimensional systems, Lyapunov numbers close to one could cause a problem; in the directions where this occurs the obtainable noise reduction is similar to that of regular dynamical systems.

In most practical applications where we must learn f the real limit to noise reduction comes from our ability to approximate f. This is limited by statistics. As demonstrated in ref. [10], it becomes difficult to approximate f accurately when the dimension of the attractor or limit set is large. Thus these noise reduction techniques are not likely to be effective for high-dimensional systems.

8. Conclusions

In this paper we present a method for solving the shadowing problem: Given a noisy trajectory $\{y_t\}$, what is the deterministic trajectory $\{x_t\}$ that is as close as possible to $\{y_t\}$? Our approach differs from previous formulations of this problem [2, 3, 12] in that we define "close" in the sense of least-mean-squares. To solve the shadowing problem we do not have to specify whether the noise is dynamical noise or observational noise; observational noise may be viewed as a special case. Thus, a solution to the shadowing problem also provides a method for noise reduction.

This allows us to solve the problem using the method of Lagrange multipliers. The exact nonlinear equations are given by eq. (8). A numerical method for solving them is described in the remainder of section 2 and in section 3.

The geometry of chaotic dynamical systems causes difficulties that are reflected in the structure of the matrix that must be inverted. These difficulties can be resolved somewhat by singular value decomposition, although it is expensive in terms of computer time. An alternative is manifold decomposition, which takes advantage of the structure of the matrix caused by the chaotic geometry. Manifold decomposition is not as accurate as singular value decomposition, but it is much faster. We suggest a procedure that alternates between the two methods, using manifold decomposition on most of the time series, and singular value decomposition on the difficult segments caused by homoclinic tangencies.

If we know the dynamics a priori, we can reduce the deviations from determinism to the limit of machine precision. Similar reductions in the absolute errors are also possible, except for segments of the time series that are near some of the homoclinic tangencies. The result is that, even with initial noise levels exceeding 5%, over most of the time series we can often reduce the noise by more than 10^{10} or 100 dB. Dynamical noise is more difficult than observational noise, but solutions are still possible.

When we do not have exact knowledge of the system dynamics we must first learn it from the data. The approximation involved in doing this naturally limits the power of the method. It also makes it difficult to get the method to converge at high initial noise levels. However, under favorable circumstances we can still obtain noise reductions of several orders of magnitude. Solu-

tions can be improved by iterating the procedures of learning the dynamics, reducing noise, then learning from the cleaned data and reducing noise with the updated approximation to the dynamics.

We have discussed the relationship of our new technique to other methods which have been proposed for noise reduction in the context of chaotic time series. We demonstrate that the manifold decomposition method recently proposed by Hammel provides an approximation to the optimal shadowing solution, as does our previously proposed maximum likelihood solution. This analysis also points out the similarities and differences to the method of Kostelich and Yorke, and indicates some of its shortcomings. We also compare these approaches to a simple statistical noise reduction technique.

All the examples presented here are two-dimensional dynamical systems. Our method applies in more than two dimensions, but so far we have not implemented it in higher dimensions, nor have we tested it on any practical applications. For low-dimensional chaotic dynamics where it is possible to obtain good approximations to the true nonlinear dynamics, however, the methods discussed here clearly offer significant improvements over standard methods of noise reduction.

Acknowledgements

We would like to thank Steve Hammel for valuable conversations. We are grateful for support from the Department of Energy and the Air Force Office of Scientific Research under grant AFSOR-ISSA-87-0095.

We urge the reader to use these results for peaceful purposes.

References

[1] H.D.I. Abarbanel, R. Brown and J.B. Kadtke, Prediction in chaotic nonlinear systems: Methods for time series with broadband Fourier spectra, Phys. Rev. A 41 (1990) 1782–1807.

[2] D.V. Anosov, Geodesic flows and closed Riemannian manifolds with negative curvature, Proc. Steklov Inst. Math. 90 (1967).

[3] R. Bowen, ω-limit sets for axiom A diffeomorphisms, J. Diff. Eqs. 18 (1975) 333–339.

[4] M. Casdagli, Nonlinear prediction of chaotic time series, Physica D 35 (1989) 335–356.

[5] J. Cremers and A. Hübler, Construction of differential equations from experimental data, Z. Naturforsch. 42a (1987) 797–802.

[6] J.P. Crutchfield, J.D. Farmer and B.A. Huberman, Fluctuations and simple chaotic dynamics, Phys. Rep. 92 (1982) 45–82.

[7] J.P. Crutchfield and B.S. McNamara, Equations of motion from a data series, Complex Systems 1 (1987) 417–452.

[8] J.D. Farmer and J.J. Sidorowich, Predicting chaotic time series, Phys. Rev. Lett. 59 (1987) 845–848.

[9] J.D. Farmer, Sensitive dependence on parameters, Phys. Rev. Lett. 55 (1985) 351–354.

[10] J.D. Farmer and J.J. Sidorowich, Exploiting chaos to predict the future and reduce noise, in: Evolution, Learning and Cognition, ed. Y.C. Lee (World Scientific, Singapore, 1988).

[11] S.M. Hammel, Noise Reduction for Chaotic Systems (Naval Surface Warfare Center, Silver Spring, MD 1989).

[12] S.M. Hammel, J.A. Yorke and C. Grebogi, Do numerical orbits of chaotic processes represent true orbits? J. Complexity 3 (1987) 136–145.

[13] E.J. Kostelich and J.A. Yorke, Noise reduction in dynamical systems, Phys. Rev. A 38 (1988) 1649–1652.

[14] A.S. Lapedes and R. Farber, Nonlinear signal processing using neural networks: Prediction and system modeling, Technical Report LA-UR-87-2662, Los Alamos National Laboratory (1987).

[15] E.N. Lorenz, The problem of deducing the climate from the governing equation, Tellus 16 (1964) 1–11.

[16] N.H. Packard, J.P. Crutchfield, J.D. Farmer and R.S. Shaw, Geometry from a time series, Phys. Rev. Lett. 45 (1980) 712–716.

[17] A.S. Pikovsky, Discrete-time dynamic noise filtering, Sov. J. Commun. Technol. Electron. 31 (1986) 81–85 [originally in Russian: Radiotekhnika i Elektronika 5 (1986) 911–914].

[18] F. Takens, Detecting strange attractors in fluid turbulence, in: Dynamical Systems and Turbulence, eds. D. Rand and L.-S. Young (Springer, Berlin, 1981).