UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION
INTERNATIONAL ATOMIC ENERGY AGENCY
INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS

SMR: 960/5

WORKSHOP ON NONLINEAR CONTROL AND CONTROL OF CHAOS

(17 - 28 June 1996)

"Nonlinear Data Analysis and Targeting"

presented by:

Eric Kostelich
Arizona State University
Department of Mathematics
Tempe, AZ 85287-1804
U.S.A.

Time Series Analysis of Chaotic Processes

Eric Kostelich

Arizona State University
Dept. of Mathematics
Tempe, AZ 85287–1804

Collaborators

Jim Yorke, Univ. of Maryland
Harry Swinney, Univ. of Texas
Anke Brandstater, Randy Tagg, Don Hirst

fluidized bed

C.S. Daw, K. Nguyen, U. of Tennessee

Grebogi, Ott, Yorke, U. Maryland

Examples of dynamical systems

$$\dot{x} = f(x,t)$$

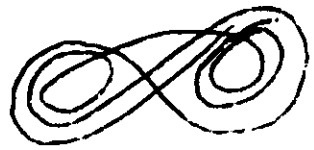
As $t \to \infty$, initial conditions in some region of phase space tend to a limit set or attractor.

Poincaré-Bendixson theorem:
These are the only possibilities for autonomous ordinary differential equations in the plane

fixed point (equilibrium)

limit cycle (periodic orbit)

"strange attractor"



- complicated geometric structure ("fractal")
- many unstable periodic orbits (saddles)
- sensitive dependence on initial conditions

Dynamical Systems

$\dot{x} = f(x,t)$ (ordinary differential equations)

$x_{n+1} = f(x_n)$ (maps - difference equations)

Linear ODE's

$\dot{x} = Ax$ (A an $n \times n$ matrix)

Solution is $\varphi_t(x_0) = e^{tA}x_0$

where $e^{tA} \equiv (I + tA + \frac{1}{2}t^2A^2 + \frac{1}{6}t^3A^3 + \cdots)$

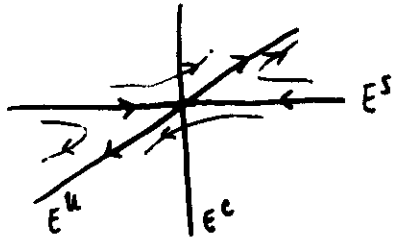
The eigenvalues and eigenvectors of A determine the dynamics

stable subspace $\{s_1, \ldots, s_k\}$ eigenvalues with negative real part - exponential decay

unstable subspace $\{u_1, \ldots, u_m\}$ eigenvalues with positive real part - exponential growth

center subspace $\{c_1, \ldots, c_\ell\}$ eigenvalues with zero real part - constant solutions or oscillations with constant amplitude

$k+m+\ell = n$



Eigenspaces give global information about the dynamics

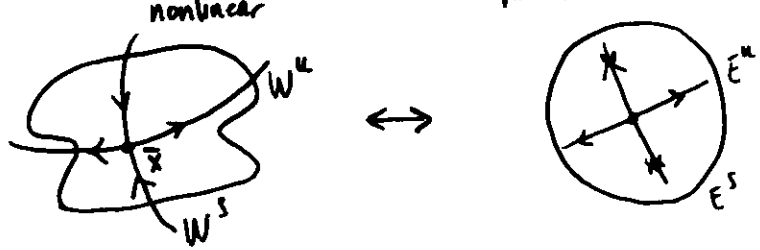
Nonlinear ODE's

Linear analysis can be done locally to determine dynamics near fixed points. $f(\bar{x}) = 0$

$Df(\bar{x})$ = Jacobian (matrix of partial derivatives at $\bar{x}$)

Definition. $\bar{x}$ is hyperbolic if $Df(\bar{x})$ has no eigenvalues with zero real part.

Fact. If $\bar{x}$ is hyperbolic, then in a neighborhood of $\bar{x}$, the solutions of $\dot{x} = f(x)$ (nonlinear) and of $\dot{x} = (Df(\bar{x}))x$ (linear) are equivalent



$W^s(\bar{x})$ is the stable manifold of $\bar{x}$.

$$W^s(\bar{x}) = \{x : \lim_{t \to \infty} \varphi_t(x) = \bar{x}\}$$

$W^u(\bar{x})$ is the unstable manifold of $\bar{x}$.

$$W^u(\bar{x}) = \{x : \lim_{t \to -\infty} \varphi_t(x) = \bar{x}\}$$

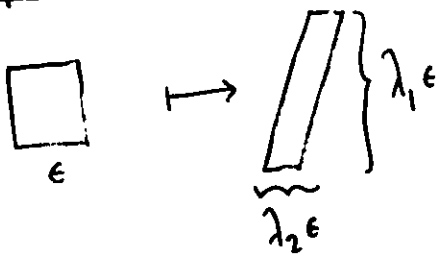
$W^s(\bar{x})$ and $W^u(\bar{x})$ are well-defined when $\bar{x}$ is hyperbolic.

Definition of chaos

stretching leads to sensitive dependence on initial conditions

"average" rate of stretching and folding is given by the

Lyapunov numbers.



λ_1 = average expension factor

λ_2 = average contraction factor

For a map with n variables, there are n Lyapunov numbers

in general $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$.

If $\lambda_1 \leq 1$, there is no expansion. Initial conditions that start close remain close.

If $\lambda_1 > 1$ initial conditions tend to separate exponentially fast. The attractor is chaotic.

Note Chaos is a property of sets (e.g. attractors), not systems of equations per se. A map can have multiple attractors, not all of which are chaotic.

Basic question:

What can we say about the dynamical behavior when we know the time series but not the governing equations?

Some possible approaches

1. Try to determine the governing equations directly from the data (difficult or impossible in general)

2. Build a statistical description

$$x_{n+1} = f(x_n, x_{n-1}, \ldots, x_{n-k+1}) + \epsilon_{n+1}$$

Linear autoregressive models:

$$x_{n+1} = a_n x_n + a_{n-1} x_{n-1} + \cdots + a_{n-k+1} x_{n-k+1} + \epsilon_{n+1}$$

(little utility for chaotic processes. Certain nonlinear models have been useful in some cases — see Howell Tong's book on nonlinear statistical models)

3. Try to find a description of the dynamics that is consistent with portions of the time series

(often doable) fixed point dynamics

4. Try to characterize some basic properties of the attractor

static analysis: fractal dimension

5. Look for evidence of determinism in the data

surrogate data

perform static analyses on the original time series and selected surrogates.

A surrogate time series is a random time series that has the same "statistics" as the original, e.g., the same power spectral density, the same autocorrelation function, etc.

Fundamental question from a physicist's point of view:

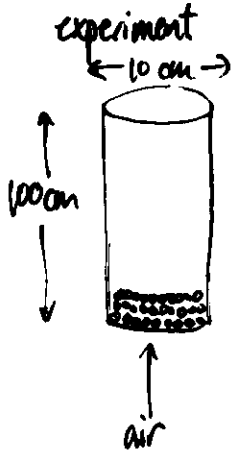
Is there evidence to suggest that the behavior of the process is consistent with that of a low-dimensional dynamical system?

Such evidence can take many forms:

- a priori modeling or other analysis
- data visualization
- power spectral analysis
- attractor reconstruction and dimension estimates
- dynamical models for estimating Lyapunov exponents

It is not always easy to determine whether low-dimensional dynamics might be present.

C. S. Daw et al. (1995) - fluidized bed



- $\sim 10^5$ particles (4 mm BB's)
- wide variety of dynamical behavior as flow rate varies
- length scales over which "high dimensional" particle collisions occur is relatively large compared to size of container
  - "slugging" and other coherent (weakly chaotic?) behavior at intermediate flow rates
  - turbulent mixing of particles and fluidizing air at high flow rates.

Two basic classes of analysis

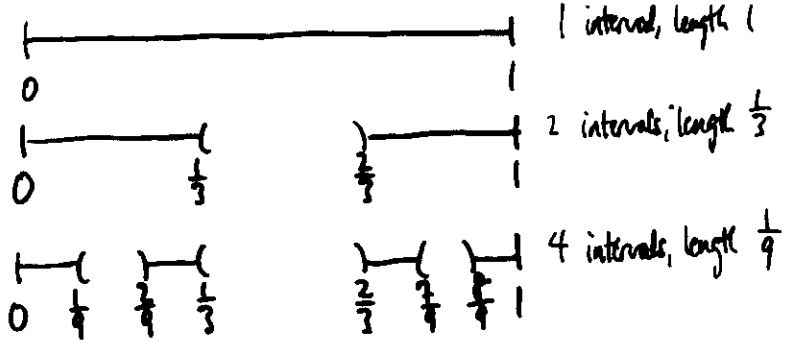
- "Static" analysis
  - Geometric properties of the attractor, such as dimension
- "Dynamic" analysis

  One goal: to find the simplest dynamical system consistent with the data.
  - prediction
  - control
  - targeting

Difficulty: Noise impairs one's ability to extract information from time-varying signals.

"Static" analysis

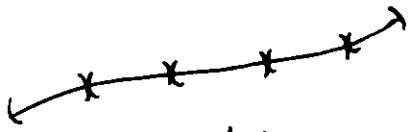
- Geometric structure is "fractal"
- Dimension quantifies the number of degrees of freedom relevant to the dynamics
- Limit cycle has dimension 1, torus dimension 2
- Several definitions of dimension (Farmer, Ott, Yorke, 1983)
- Cantor middle third set



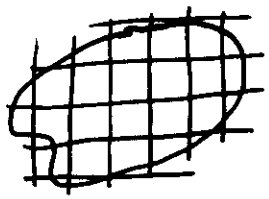
$N(\epsilon) \sim \left(\frac{1}{\epsilon}\right)^{D}$

Box counting (fractal) dimension

The number of boxes of size ϵ needed to cover the set is proportional to $\left(\frac{1}{\epsilon}\right)^{D_0}$ where D_0 = fractal dimension



$N(\epsilon) \sim \frac{1}{\epsilon}$, $D_0 = 1$



$N(\epsilon) \sim \left(\frac{1}{\epsilon}\right)^2$ $D_0 = 2$

Cantor middle third set: 2^k intervals of length $\left(\frac{1}{3}\right)^k$.

$N\left(\frac{1}{3^k}\right) = 2^k$

$\log N = D_0 \log \epsilon$

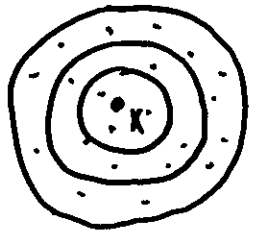
$k \log 2 = D(k \log 3)$

$D = \frac{\log 2}{\log 3} \approx .69$

Note: the limit is taken as $\epsilon \to 0$.

$N(\epsilon) \sim \left(\frac{1}{\epsilon}\right)^{D}$

Pointwise dimension



- select reference point x
- Let $p_x(\epsilon)$ be the proportion

  of the total number of total points within ϵ of x
- $p_x(\epsilon) \sim \epsilon^{d_1}$ $\quad d_1$ = pointwise dimension at x
- Basic idea: d_1 is the same for "typical" points on the attractor

Grassberger-Procaccia correlation dimension (1983)

- Let $p(\epsilon) = \langle p_x(\epsilon) \rangle$ (over all reference points x)
- $p(\epsilon) \sim \epsilon^{d_2}$

Algorithm:

for each $\epsilon \downarrow 0$
  for each reference point
    calculate $p_x(\epsilon)$
  print $\log \epsilon$, $\log \langle p_x(\epsilon) \rangle$

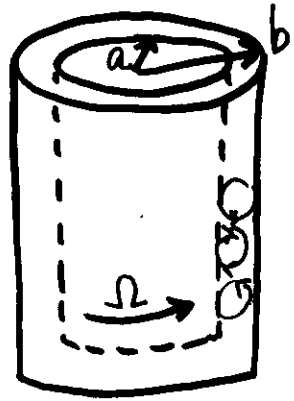
→ find slope using linear regression

Couette-Taylor experiment

Reynolds number

$$R = \frac{a\Omega(b-a)}{\nu}$$

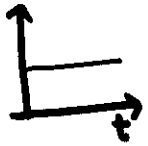
Dynamical behavior as R increases:



laminar (Couette) flow

↓ ← R_c

Taylor vortex flow
(time independent)



dynamics: • fixed point — dimension: $d=0$

↓

wavy vortex flow
(time dependent - 1 azimuthal wave)

limit cycle — $d=1$

↓

modulated waves
(quasiperiodic)

torus — $d=2$

↓

weak turbulence
(chaos)

strange attractor — $d>2$

Ruelle & Takens (1971); Newhouse, Ruelle & Takens (1978)

Goals of "dynamic" analysis

Exploit time-delay embeddings to analyze the dynamics of experiments

- nonlinear noise filter
- simplicial approximations

These ideas will give experimentalists access to many of the same tools long available to numerical theoreticians, including:

- phase portraits
- attractor dimension (Farmer, Ott, Yorke (1983); Grassberger and Procaccia (1983))
-
- Lyapunov exponents (Wolf et al (1985); Eckmann-Ruelle (1985))
- periodic orbits
- stable and unstable manifolds
- prediction
- control (stabilization) of saddle periodic orbits
- targeting

Three strategies in time series analysis of chaotic data

① Strategy for reconstructing attractor from data
- Takens embedding theorem (1980)
- Sauer, Yorke, Casdagli improved embedding theorem (1991)

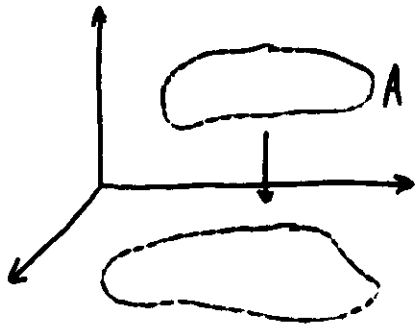
② Strategy for inferring dynamics from data
- choose a class of models
- fit parameters from data
- Eckmann-Ruelle linearization (1985)

③ Strategy for adjusting observations
- how can the data be adjusted slightly to be "more consistent" with the imputed dynamics?
- many criteria: local projective maps least-squares minimization techniques

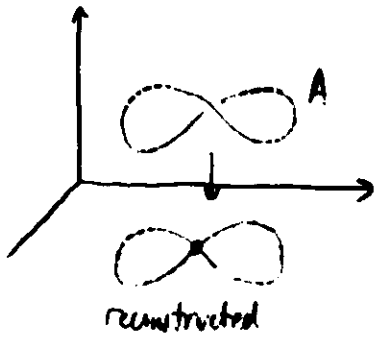
Requirements for embeddings

Reference: "Embedology," T. Sauer, J. Yorke, M. Casdagli (1991)

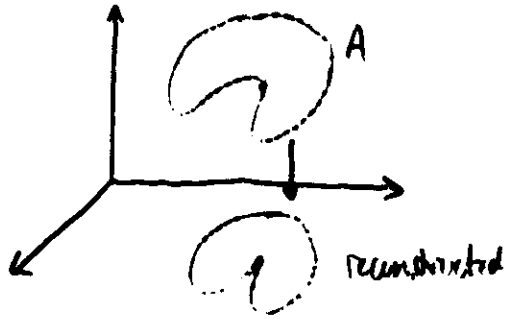
- one-to-one (points are not collapsed together)
- preserves differentiable structure of the attractor



Examples of things to avoid:



not one-to-one)



(not always differentiable)

Basic result (Sauer, Yorke, Casdagli)

If $d_A = \text{boxdim}(A)$ and $n > 2d_A$ then "almost every" smooth function yields a good reconstruction of A in $\mathbb{R}^n$.

Typical procedure (F. Takens, 1981)

Time-delay embedding method

experiment: $\varphi_{t_i}(x) \subset A$ (flow)

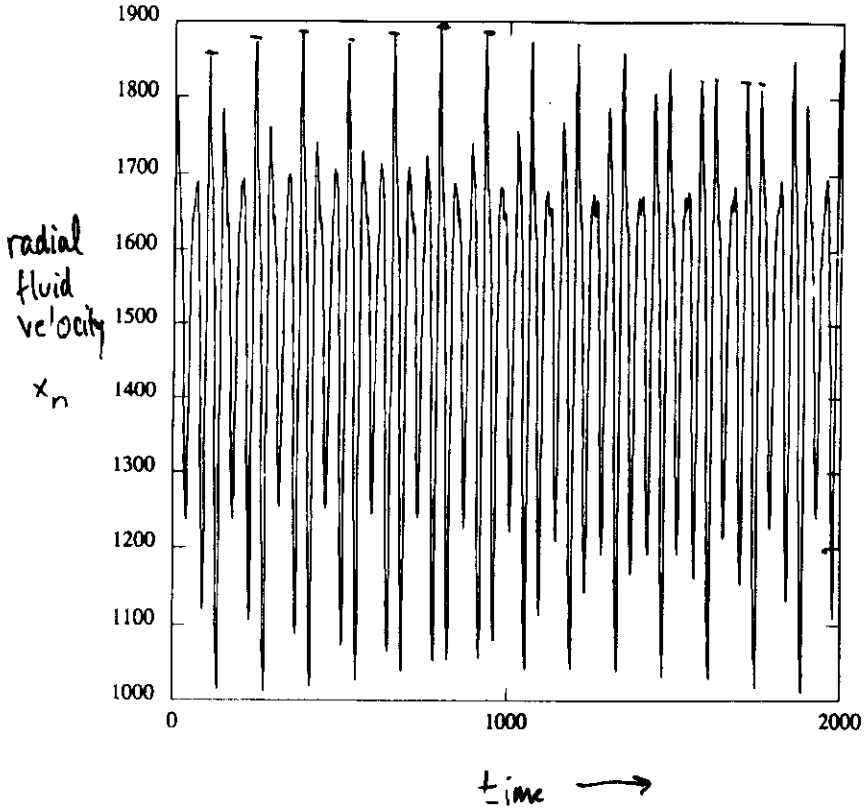
measurement: $\{s_i\}$ (time series)

$$\underline{x}_1 = (s_1, s_{1+\tau}, s_{1+2\tau}, \cdots, s_{1+(m-1)\tau})$$

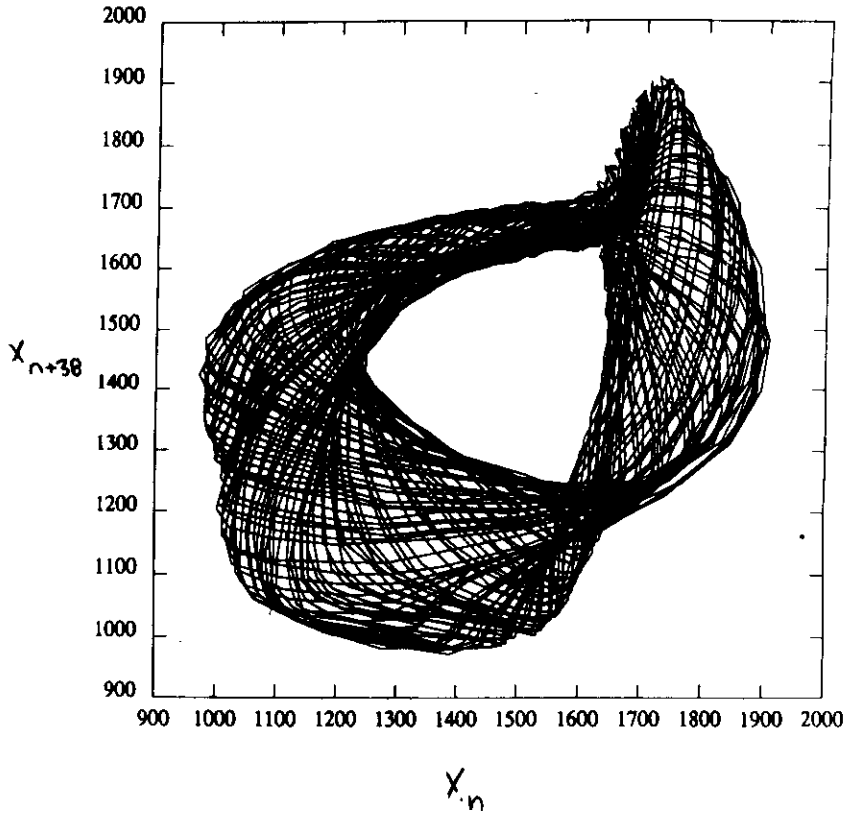
$$\underline{x}_2 = (s_2, s_{2+\tau}, s_{2+2\tau}, \cdots, s_{2+(m-1)\tau})$$

$$\vdots$$

Note: Reasonable filtering will not cause problems.



Reconstructed Attractor



Choosing embedding dimension and time delay

- too short a delay means $x(t+\tau) \simeq x(t)$
  so observations lie near 45° line

- too long a delay means observations become decorrelated due to sensitive dependence on initial conditions

- autocorrelation criterion (Broomhead & King, 1986)

- information theoretic criterion (Fraser and Swinney, 1986)
  - find smallest τ that minimizes $P(x(t+\tau)|x(t))$

- some advice: make pictures
  keep delay short but "spread points out"
  rerun analysis for several reasonable values of m and τ

Roux, Simoyi, Swinney

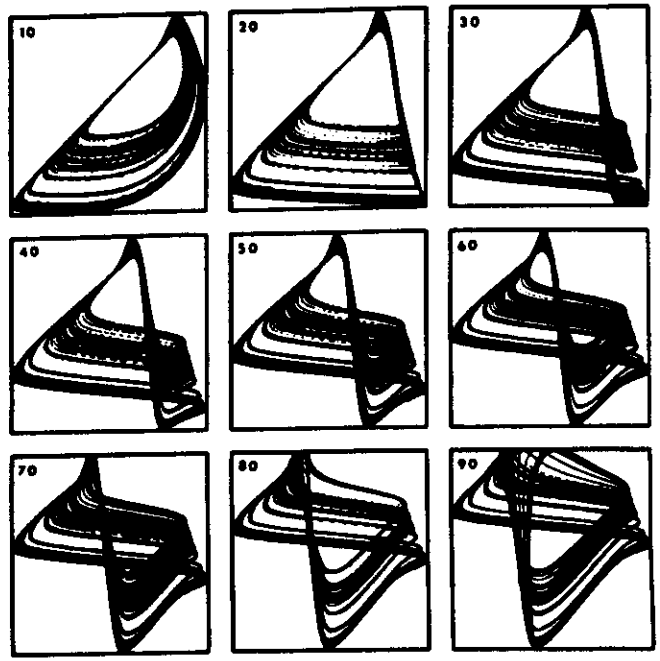


Fig. 3. Two-dimensional phase portraits, $B(t_i + T)$ vs $B(t_i)$, constructed from post-transient data for the nonperiodic state 1. The different portraits correspond to different time delays T. The values of T, expressed as multiples of the time between succ measurements (0.88 s), are indicated on each portrait; there are 130–140 measurements per orbit.

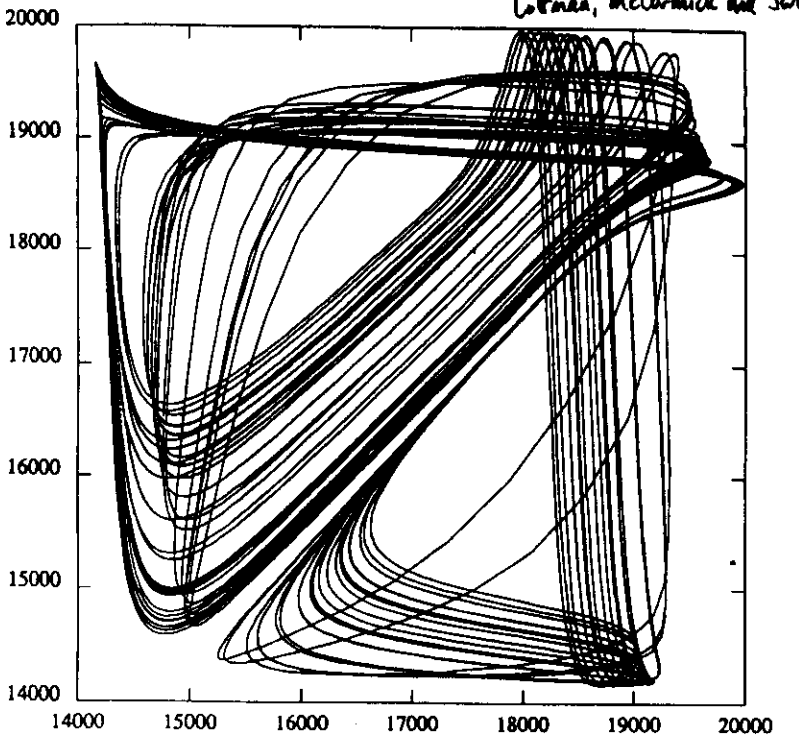
have been found to return to the attractor for all perturbations we have used, so this attractor could be globally attracting. Perturbations have included injections of bubbles into one of the feed lines, injections of bromide ions into the reactor, and turning off the stirrer or one of the chemical feed lines for a few seconds. For other values of the control parameters multiple stable states have been observed [13], each with its own basin of attraction – the trajectories will then still return to the original attractor for perturbations that are not too large, but a sufficiently large perturbation can send the system trajectory from one basin of attra into another basin.

5. A Lyapunov exponent

A quantitative measure of nonperiodic (cha behavior is provided by the value of the la Lyapunov exponent, which characterizes the age rate of separation of nearby trajectories This exponent is positive for a chaotic state, for a periodic state it is zero. An attractor w

Belousov-Zhabotinskii Chemical Data

Coffman, McCormick and Swinney



Locating Periodic Saddles

- Lyapunov exponents are relatively small
- Look for recurrent points

  If m is the smallest integer such that

  $$\| x - f^m(x) \| < \epsilon$$

  then x is an (m, ϵ) recurrent point.

- Simple-minded algorithm

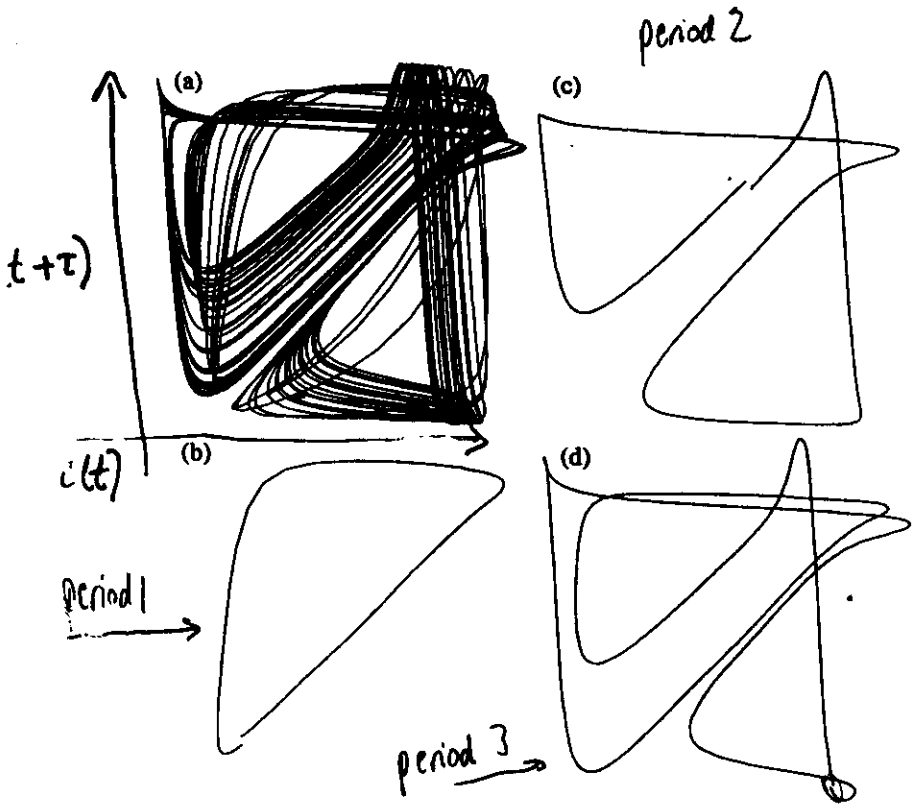
  1. Fix ϵ
  2. reconstruct attractor from time series
  3. for each point X_i

     find smallest positive m such that

     $$\| X_{i+m} - X_i \| < \epsilon$$

  4. record recurrent points

     make histogram of m values



Strange attractor reconstructed from laboratory data (Belousov-Zhabotinskii)

Simoyi, Roux, McCormick, Swinney (Univ. of Texas)

Lathrop and Kostelich (1989)

Note: time-delay embedding is not the only, nor the best, attractor reconstruction method.

Other strategies are possible and often are preferable.

Sauer (1992): "low pass" embedding

- choose "window" of time series values
- calculate FFT of the window but keep only the lowest frequencies
- invert transform
- the embedding consists of some number of evenly spaced samples in a "smoothed" window of time series values.

Albano et al (1988): } Singular value decomposition
Broomhead & King (1986): }

Both methods preprocess the data to reduce noise.

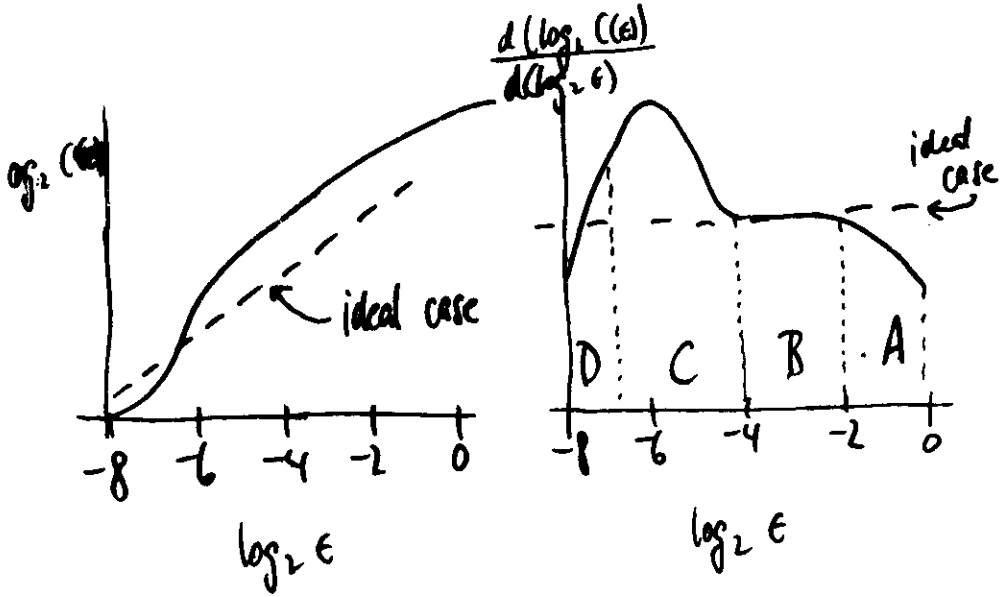
Grassberger-Procaccia correlation dimension

1. Choose reference point x.
2. Compute $C(x,\epsilon)$ = fraction of points on attractor that fall within ϵ of x.
3. Find $\langle C(\epsilon) \rangle$ = average over all reference points.

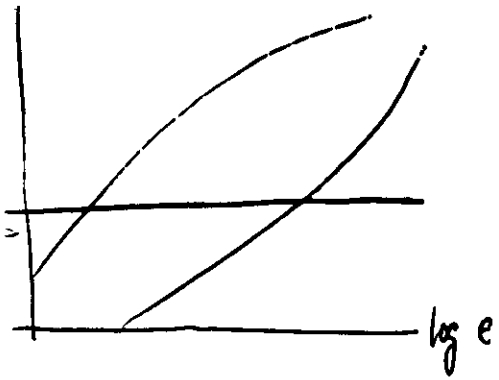
$$C(\epsilon) \sim \epsilon^{D}$$

4. Plot $\dfrac{d(\log C(\epsilon))}{d(\log \epsilon)}$

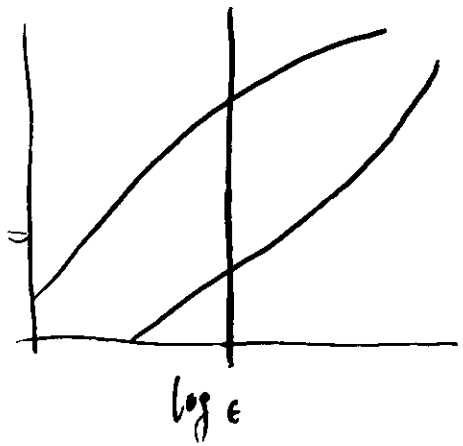
Brandstater and Swinney (1987, 1987)



A: global structure (folds) of attractor
B: scaling region ("plateau")
C: noise
D: lack of data



lack of data increases the minimum observed value of $C(\epsilon)$

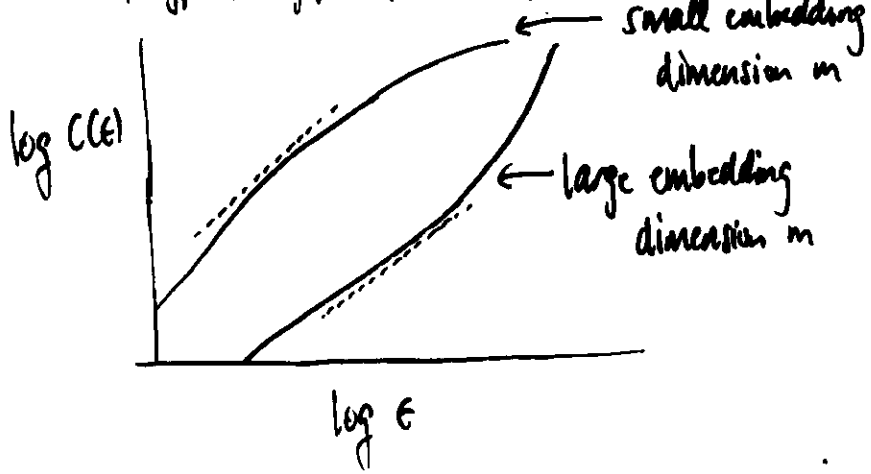


noise increases the smallest length scales over which scaling can be seen

Noise reduction methods can be used to try to push the scaling region to smaller values of ϵ.

"Plateau dimension" analysis

Ding, Grebogi, Ott, Sauer, Yorke (1993)



- Shape of small-m curve can be attributed to boundary effects.

  $C(\epsilon) \sim (2\epsilon - \epsilon^2)^m$ for uniformly distributed random data

  $\Rightarrow \frac{d \log C(\epsilon)}{d \log \epsilon} \sim m - \frac{m\epsilon}{2-\epsilon}$

- Shape of large-m curve can be attributed to the presence of folds on the attractor

Basic questions in time series analysis

- where is the relevant information contained in the signal?
- how can the data be preprocessed to make "patterns" more obvious?

Conventional approaches

- Fourier analysis / power spectra
  assumption: relevant information is contained in the frequency domain
- Kalman filters
  time domain approach
- attractor embedding and reconstruction
  examine trajectories of points that are nearby in an abstract "phase space"

Power spe[redacted]

Fourier transform

$$U(f) = \int_{-\infty}^{\infty} u(t)\, e^{2\pi i f t}\, dt \qquad \text{(time} \to \text{frequency)}$$

inverse:

$$u(t) = \int_{-\infty}^{\infty} U(f)\, e^{-2\pi i f t}\, df \qquad \text{(frequency} \to \text{time)}$$

This is a linear functional of u and U.

Let $u(t)$ = "true" signal

$s(t)$ = signal output by experiment
(convolved with a response function $r(t)$)

$e(t)$ = "corrupted" signal

$= s(t) + \underbrace{n(t)}_{\text{noise}}$

Wiener filters

Construct a function $\Phi(f)$ such that

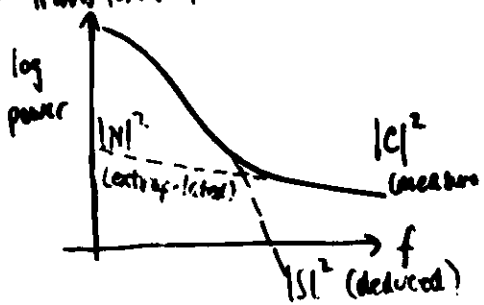
$$\tilde{U}(f) = \frac{C(f)\,\Phi(f)}{P(f)}$$

where $\int_{-\infty}^{\infty} |\tilde{U}(f) - U(f)|^2 \, df$ is a minimum.

Fact. $$\Phi(f) = \frac{|S(f)|^2}{|S(f)|^2 + |N(f)|^2}$$

Usually one has some model or assumption about the noise in order to construct Φ.

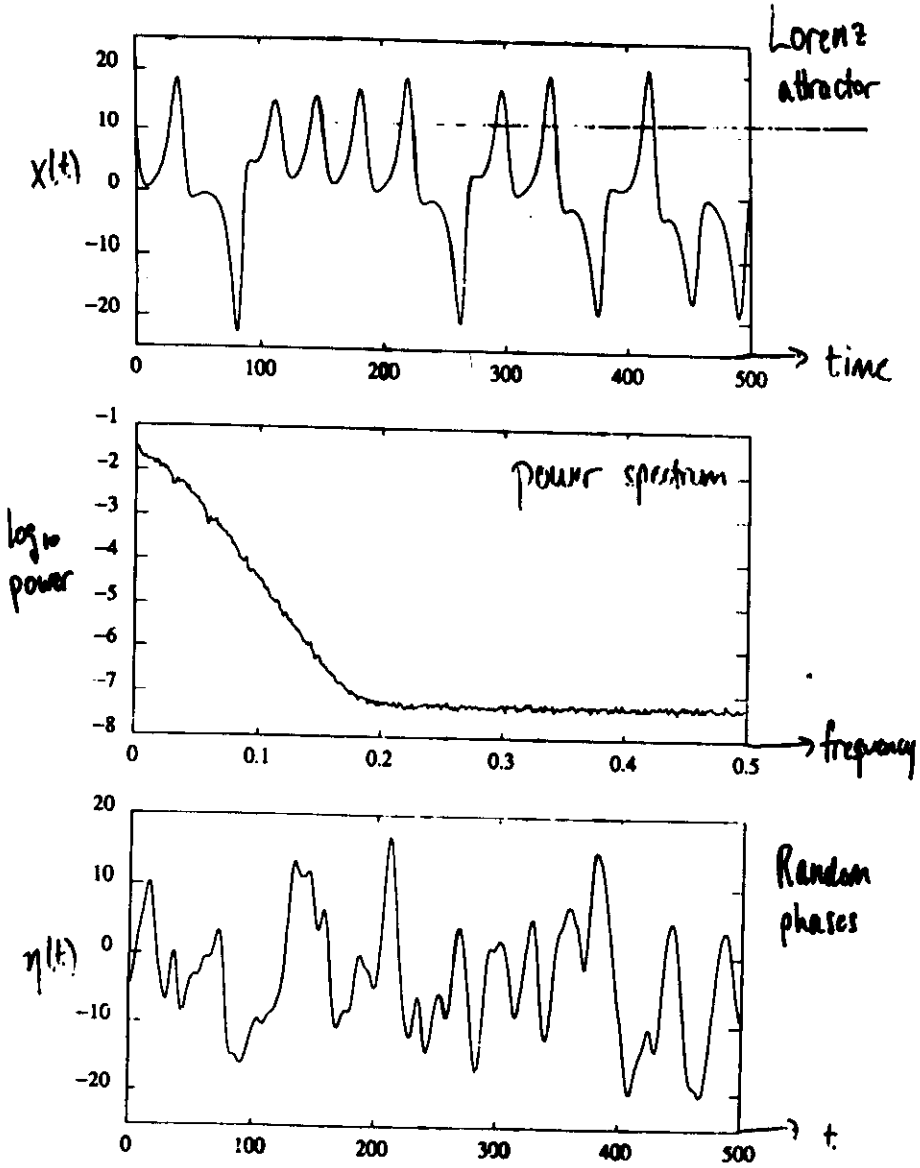
In this procedure, one adjusts the power spectrum then applies the inverse Fourier transform to recover a "better" signal.



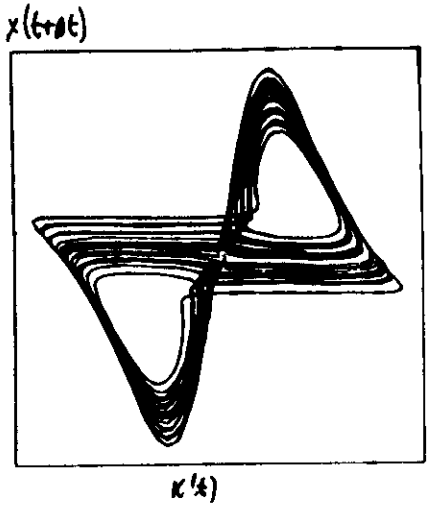
In general, power spectra cannot distinguish between deterministic, low-dimensional chaos and noise.

Numerical experiment (A. Fraser)

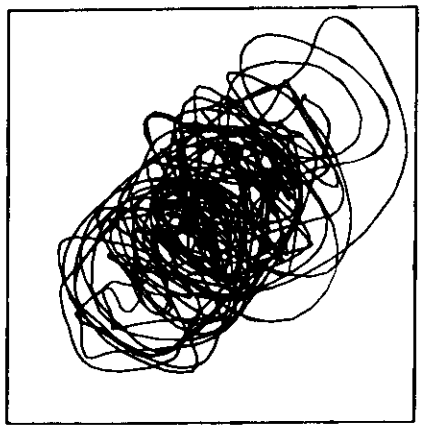
- Fourier coefficients are complex numbers
  write in polar coordinates $C = (r\cos\theta, r\sin\theta)$
- Taking modulus means phase information is lost
  add random phase $C' = (r\cos(\theta+\eta), r\sin(\theta+\eta))$
- Apply inverse transform



(A. Fraser)



Lorenz attractor reconstructed from x-coordinate time series



Phase portrait of random phase time series with same autocorrelation

(A. Fraser)

input time series

s_i
$s_{i+\tau}$
s_{i+2}
s_{i+3}

→ reconstructed attractor

$\vec{x}_i = (s_i, s_{i+\tau}, s_{i+2\tau})$

$\vec{x}_{i+1} = (s_{i+1}, s_{i+1+\tau}, s_{i+1+2\tau})$

$\vec{x}_{i+2} = (s_{i+2}, s_{i+2+\tau}, s_{i+2+2\tau})$

↓

linear approximation of dynamics at $\vec{x}_i$

adjust trajectories to satisfy them better

↓

fitted (less noisy) time series

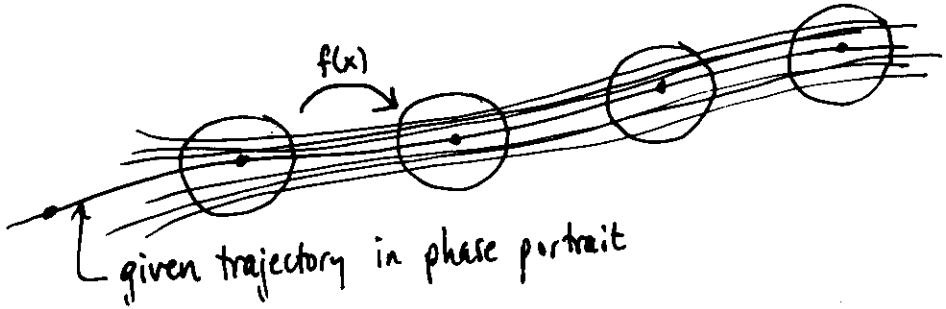
$\hat{x}_i = (\hat{s}_i, \hat{s}_{i+\tau}, \hat{s}_{i+2\tau})$

$\hat{x}_{i+1} = (\hat{s}_{i+1}, \hat{s}_{i+1+\tau}, \hat{s}_{i+1+2\tau})$

$\hat{x}_{i+2} = (\hat{s}_{i+2}, \hat{s}_{i+2+\tau}, \hat{s}_{i+2+2\tau})$

Noise Reduction in Time Series Data

E. Kostelich and J. Yorke, Physica D (1990)



Assumptions

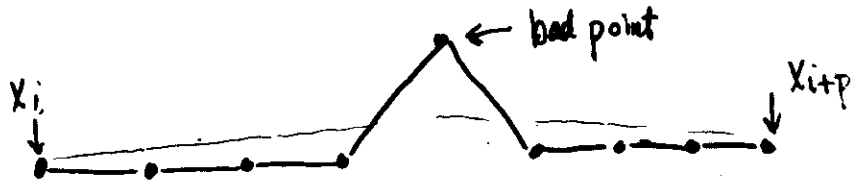
- The observed trajectory is a noisy representation of the "true" trajectory.
- The dynamics in small balls is nearly linear:
  $f(x) \approx Ax + b$

Procedure

- Use a <u>ensemble</u> of points to compute a linear approximation to the dynamics at each point on the attractor.
- Replace the observed (<u>individual</u>) trajectories with nearby ones that best satisfy the linear approximations (in a least squares sense).

Step 2. Trajectory adjustment

Suppose mapping is a horizontal translation



Objectives – identify bad points
– correct trajectories

Procedure Choose $(p-1)$ intermediate points s.t.

(1) distance between linear image of previous point and chosen point is small

(2) distance between chosen point and original point is small

(3) distance between linear image of chosen point and the next (original) point is small

How is this noise reduction method different from conventional ones?

- Information about dynamics is taken from clusters of nearby points on a phase space attractor
- Nearby points in phase space may be widely separated in a time or frequency domain
- The goal is to get a more self-consistent set of data

Measures of effectiveness

① Visual inspection

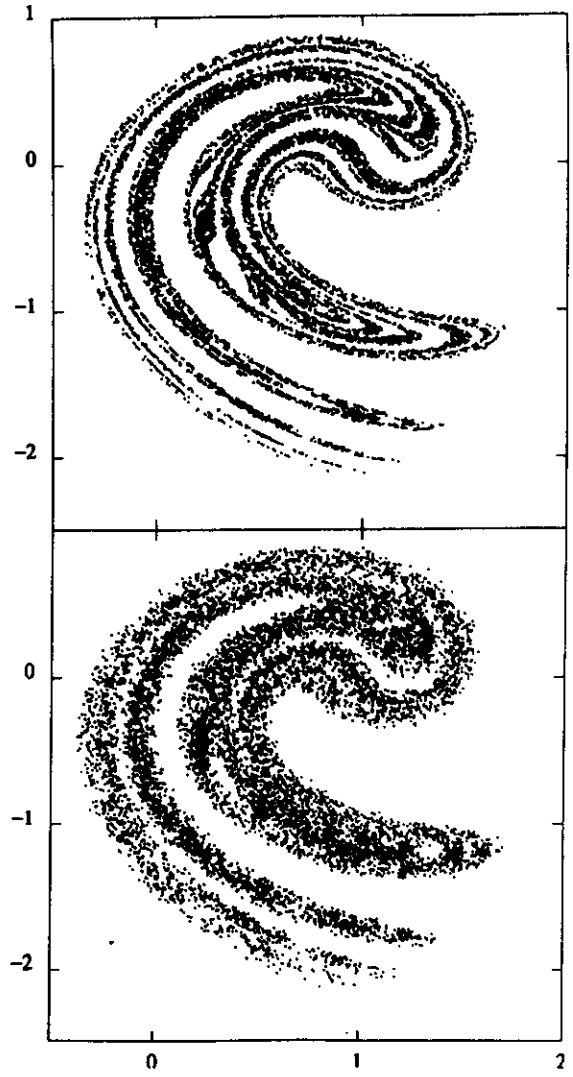
② Self-consistency measure:

pointwise error: $e_n = \| \hat{f}(x_n) - x_{n+1} \|$

noise reduction: $1 - \dfrac{\langle e_n \rangle \text{ fitted}}{\langle e_n \rangle \text{ original}}$

③ Check dimension calculations

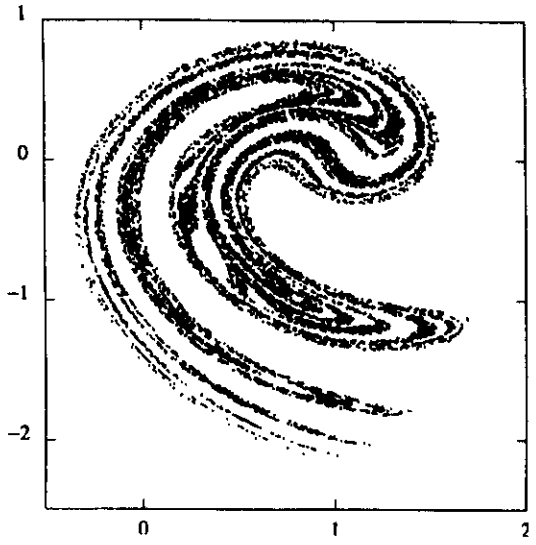
Note: power spectral analysis gives ambiguous results



Ikeda attractor

Original data

With 1% additive noise



After noise reduction

Numerical experiments for Ikeda attractor

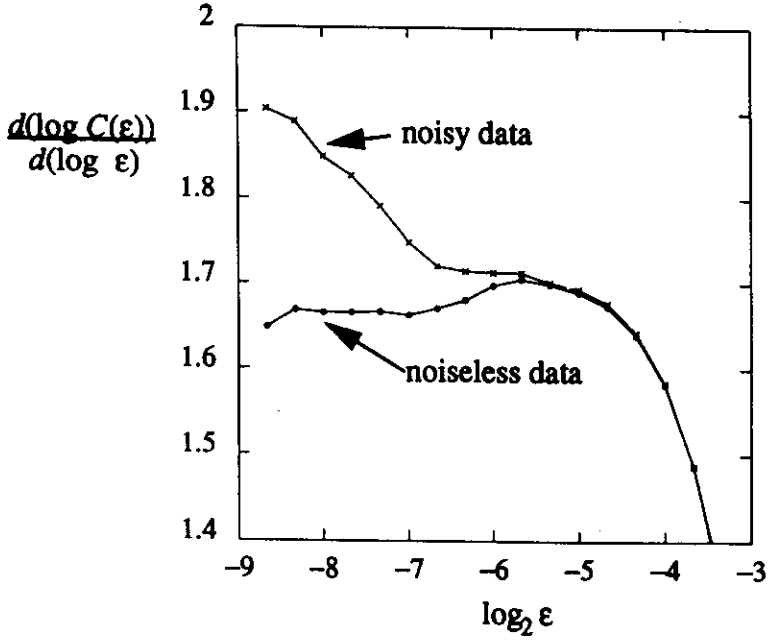
$2^{17} = 131,072$ points

4 iterative improvements

24 points per window

Noise	Reduction
0.1%	92%
1.0%	91%

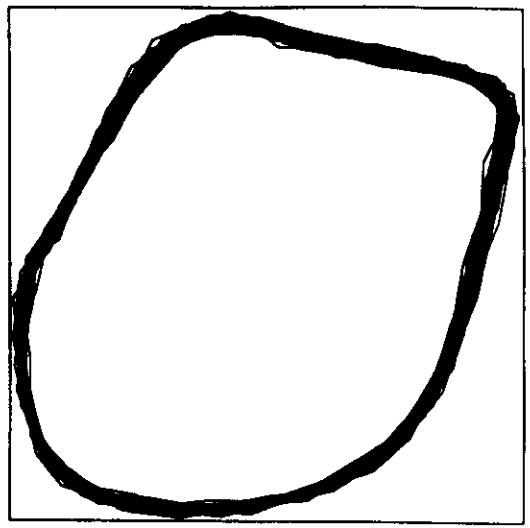
Correlation dimension for Ikeda attractor data



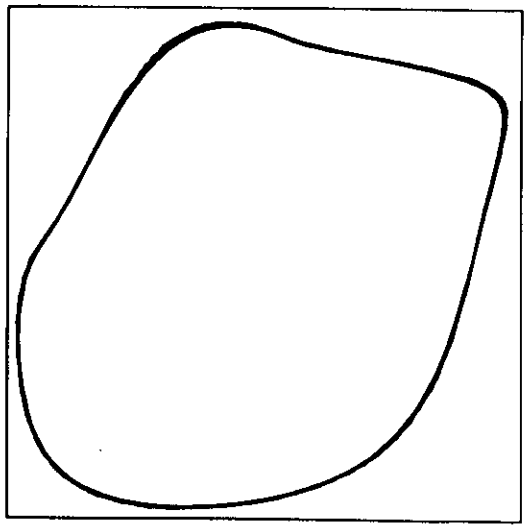
loise reduction
ith Couette-Taylo-
data
(wavy flow)

D. Hirst, R. Tagg.
H. (winner)

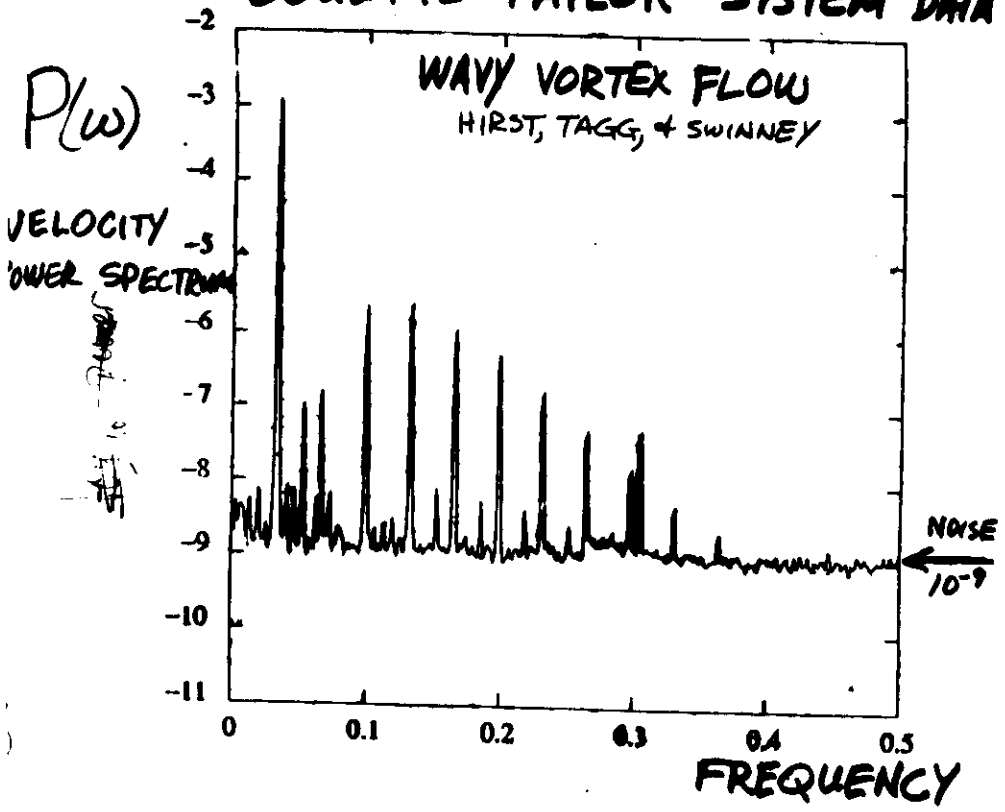
before



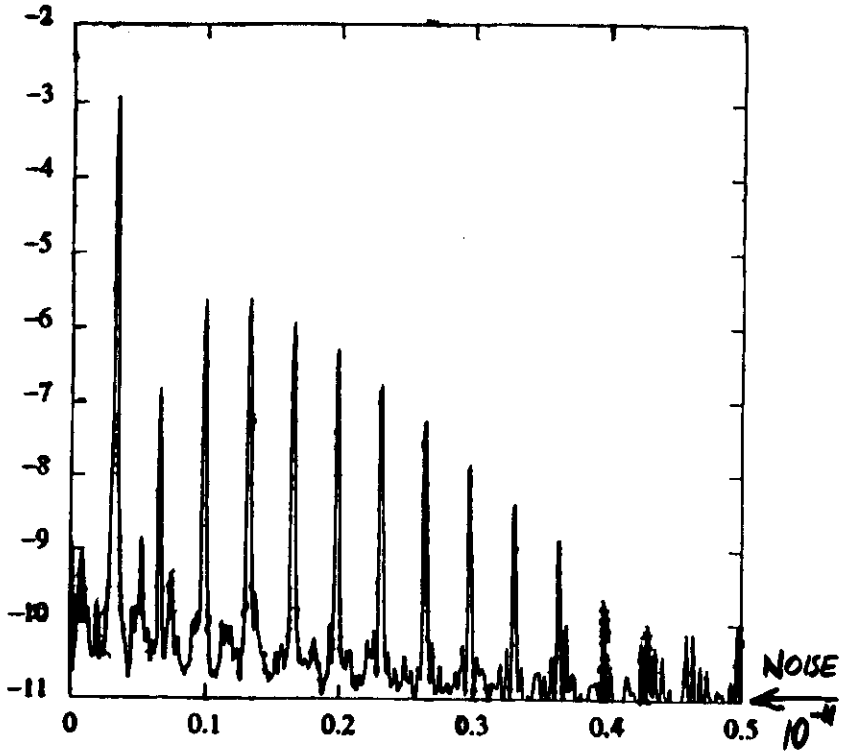
after

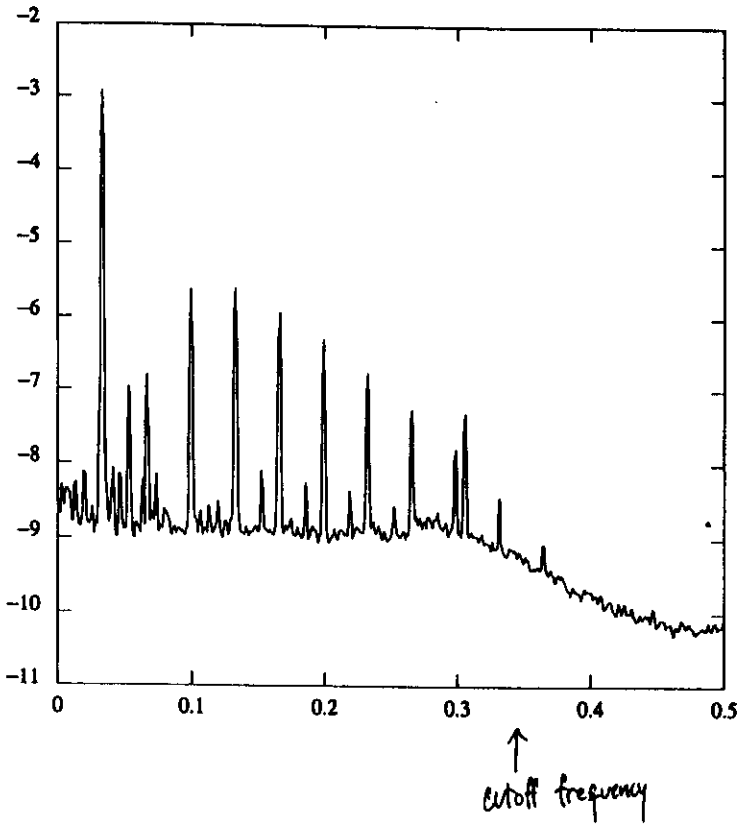


COUETTE-TAYLOR SYSTEM DATA



DYNAMICALLY FILTERED DATA





Low pass filtering

CHAOTIC DATA

Application to chaotic Couette-Taylor flow

(Brandstater et al., Phys. Rev. Lett. 51, August)

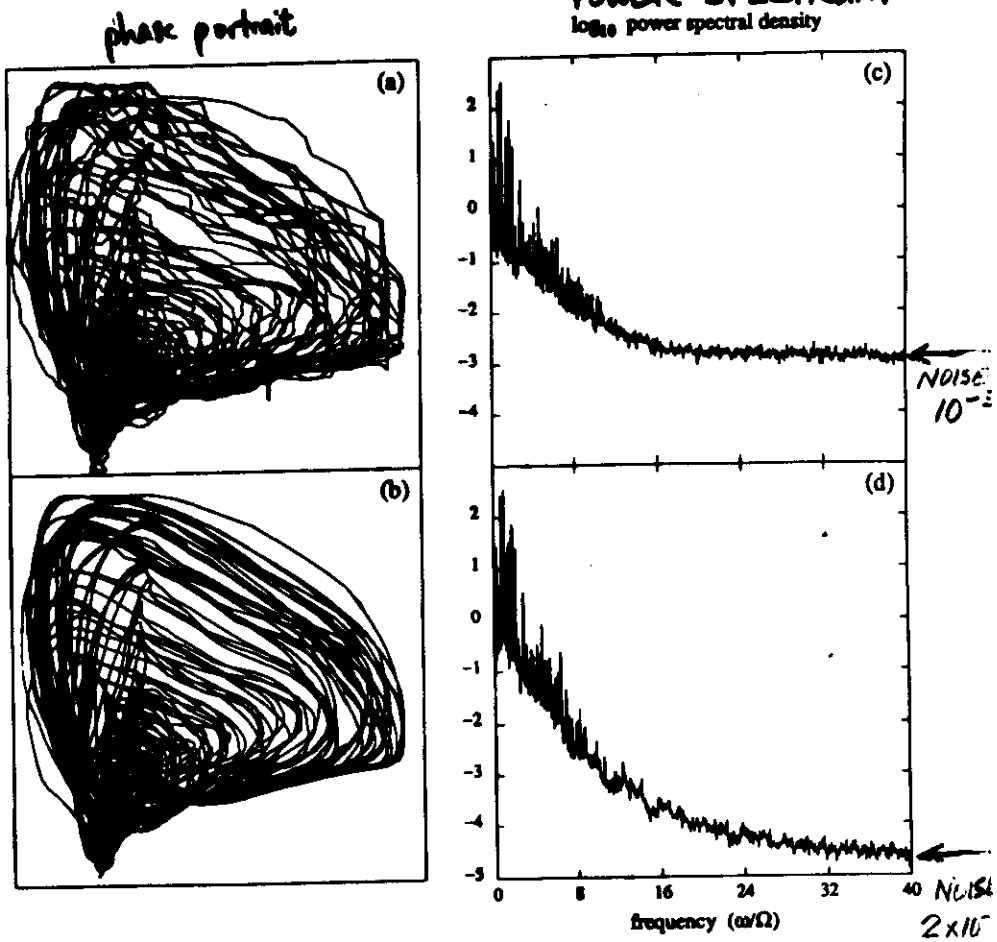
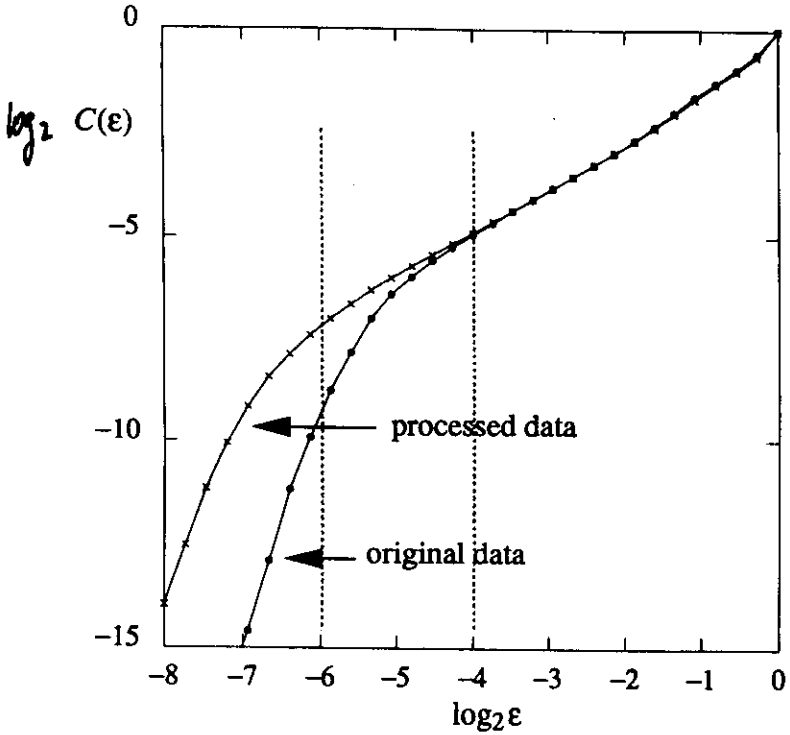


Figure 2:

experiment by A. Brandstater and H. Swinney
Phys. Rev. A 35, 2207 (1987)

Dimension results for chaotic Couette data



Conclusions

① Dimension calculations require low-noise data. Noise reduction may be essential in some cases.

② Dimension calculations can be used to assess the efficacy of noise reduction methods in some cases.

③ "Dynamic" analysis of chaotic data makes use of similar patterns in the time series, regardless of their distribution in time or frequency.

ftp://saddle.la.asu.edu/pub

Targeting and Control of Chaos

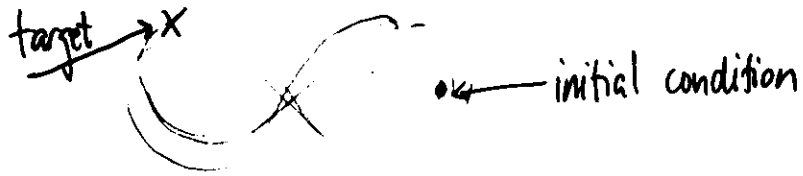
Eric J. Kostelich
Arizona State University

Collaborators

C. Grebogi
E. Ott
J. Yorke
E. Barreto
B. Hunt
G. Yuan

Basic idea:

Use the sensitive dependence on initial conditions to steer a given point on a chaotic attractor to a neighborhood of some prespecified point (target) as quickly as possible using only small perturbations to an available control parameter.



Sensitive Dependence and Spacecraft

- ISEE-3 satellite launched 1978 — made measurements of solar wind between Earth and Sun
- In 1982, the satellite was pushed from its old orbit to intercept the comet Giacobini-Zinner by September 1985.
- Limited fuel was available for the mission
- Precise targeting brought the spacecraft within 120 km of the moon's surface

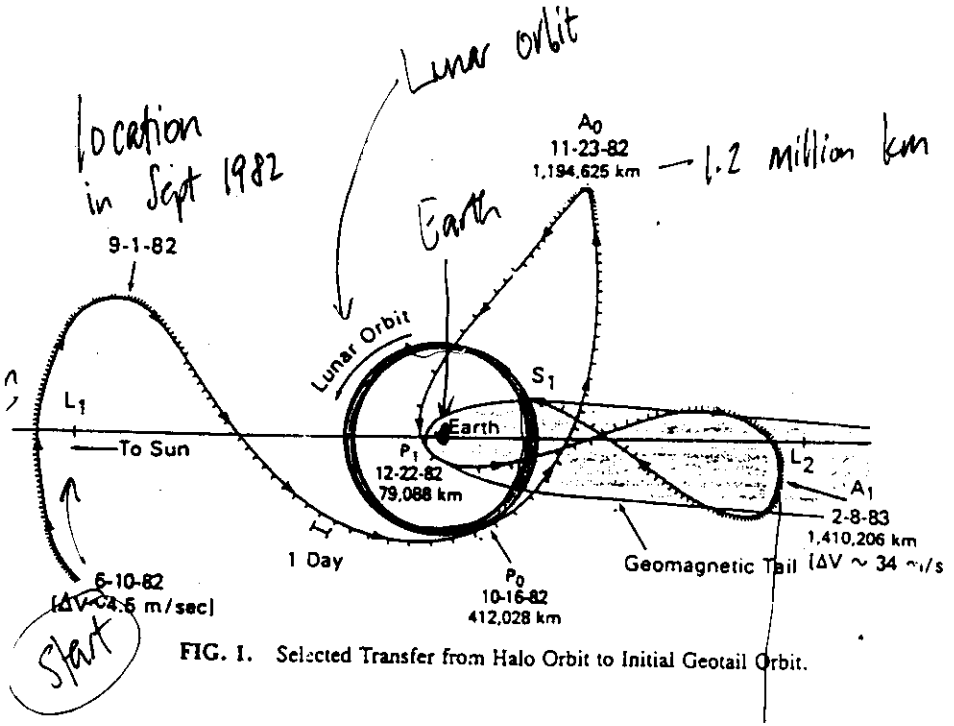


FIG. 1. Selected Transfer from Halo Orbit to Initial Geotail Orbit.

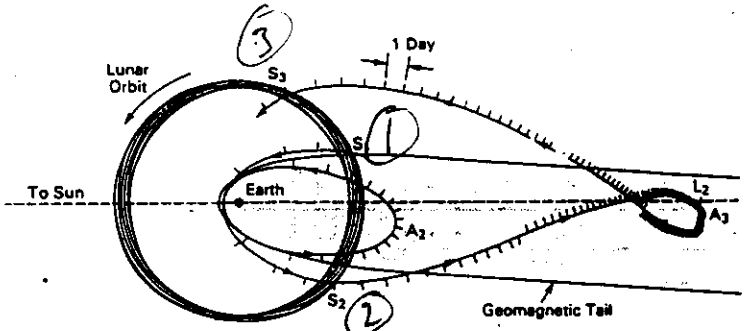


FIG. 5(a). First Portion of Sequence (1): $3 \rightarrow LO \rightarrow 5 \rightarrow LO \rightarrow 2$ (Selected Sequence.) Ecliptic Plane Projection.

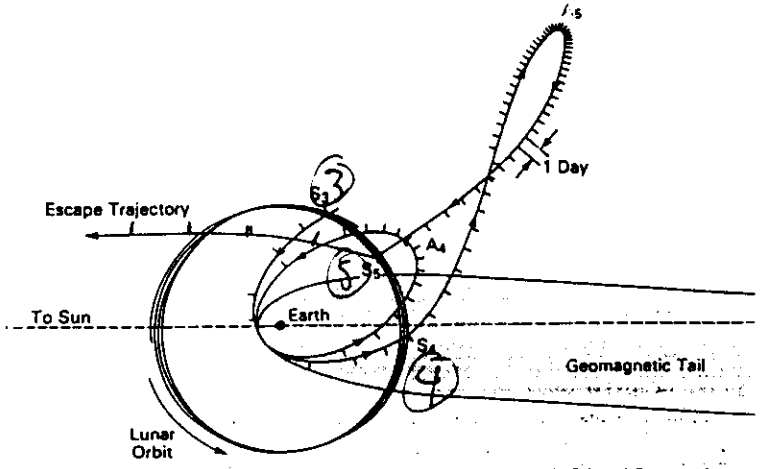


FIG. 5(b). Final Portion of Sequence (1): $3 \rightarrow LO \rightarrow 5 \rightarrow LO \rightarrow 2$ (Selected Sequence.) Ecliptic Plane Projection.

Hallmarks of chaos

(1) Sensitive dependence on initial conditions means small changes now cause large changes later

- small errors in measurement now foil long-term predictions

(2) Control - periodic orbits are dense in chaotic attractors

- use small perturbations to confine an orbit in a small neighborhood of the desired periodic point

One dimensional targeting (Shinbrot et al) - PRL (1990)
(Shinbrot, Ditto et al, PRL 1992)

$$x_{n+1} = f(p, x_n)$$

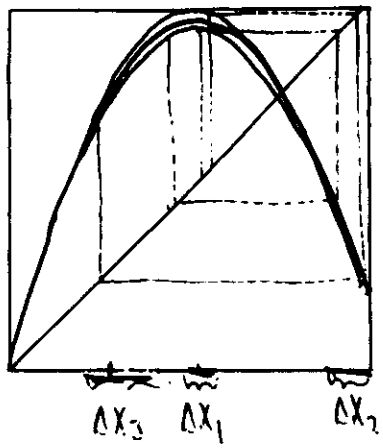
- Perturb the parameter by $\delta p \in [-\Delta p, \Delta p]$
- The next iterate is altered by an amount

$$\delta x_1 \doteq \delta p \cdot \frac{\partial f}{\partial p}(\bar{p}, x_1)$$

- Since $|\delta p| \leq \Delta p$, we know $|\delta x_1| \leq \Delta x_1$.
- Repeat the argument for successive iterates.
- The intervals $\Delta x_1, \Delta x_2, \Delta x_3, \ldots$ form an expanding sequence that eventually encompasses the target point.
- Use bisection to determine precise value of p needed to hit the target.

Basic picture:

Illistair Mees
Erwin Judd
Kathryn Glass
proprietary



- method is robust to moderate amounts of noise.
- the map can have a few discontinuities
- the map can be a Poincaré surface of section (e.g. Lorenz attractor)

Buckled ribbon experiment

W. Ditto et al

NSWC

strobe the system once each period

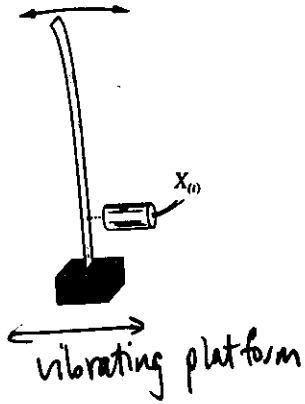
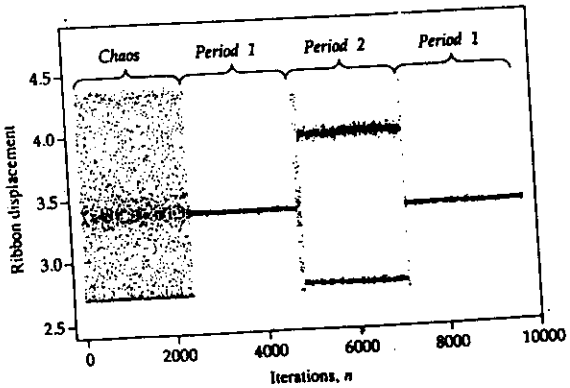


FIG. B1 Chaotic ribbon.



position

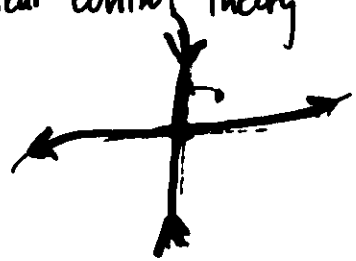
iterations (time)

Why targeting?

- chaotic attractors have many saddle periodic points that correspond to regular (perhaps "desirable") behavior.
- "control of chaos" algorithms require the starting point to be in a fairly small neighborhood of the saddle point
- the uncontrolled process may take a long time to reach the neighborhood
- targeting can reduce the waiting time by orders of magnitude

Grebogi, Ott, Yorke control of chaos

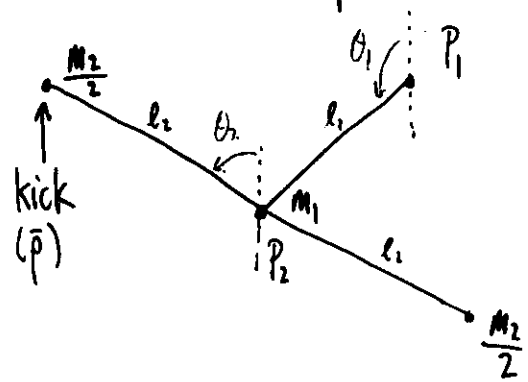
- stabilize saddle periodic points using linear control theory



- construct a gain matrix (which determines the change in parameter) to move the iterate onto the stable manifold
- apply the method iteratively to stabilize points
- the method works only if it is applied to a suitably small neighborhood of the fixed point
  - perhaps $\underline{10^{-4}}$ of the attractor size

Higher dimensional problem

kicked double rotor map



$$\Theta_{n+1} = \left(M\dot{\Theta}_n + \Theta_n\right) \bmod 2\pi$$

$$\dot{\Theta}_{n+1} = L\dot{\Theta}_n + G(\Theta_{n+1})$$

where

$$G(\Theta_{n+1}) = \begin{pmatrix} c_1 \sin\theta_1 \\ c_2 \sin\theta_2 \end{pmatrix}$$

M, L are constant 2×2 matrices

- Take

$l_1 = \frac{1}{\sqrt{2}}$, $l_2 = 1$, $m_1 = m_2 = 1$, $v = 1$

$\bar{p} = 9$ (strength of kick)

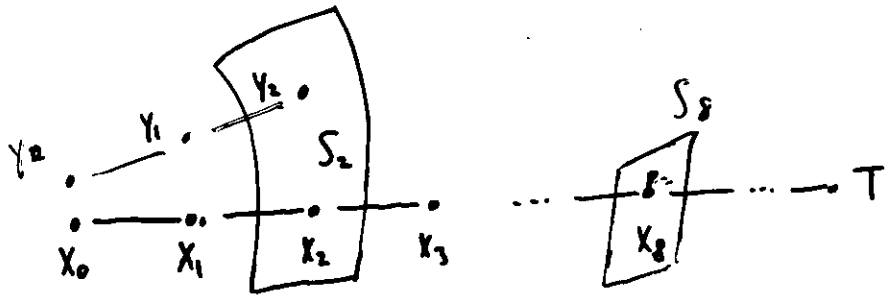
- Lyapunov exponents:

≈ 3 $\lambda_1 = 1.21$ $\lambda_2 = 0.26$

$\lambda_3 = -1.74$ $\lambda_4 = -2.72$

- Lyapunov dimension: $d_L = 2.8$
- Typical points on the attractor have 2-dimensional stable and 2-dimensional unstable manifolds

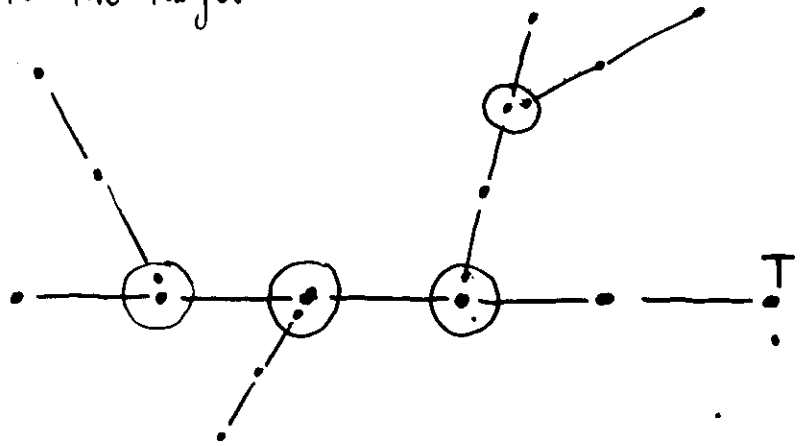
Basic targeting algorithm



- Goal: find two successive changes of one parameter (or one change to two parameters) to move Y_2 onto the stable manifold of X_2.
- Approximate S_2 by looking at preimages of points in the contracting eigendirection
- It is possible to set up a Newton iteration to solve the problem. When the method converges, we have the first part of the targeting strategy.

Targeting strategy - Refinements

Step 1: Build a collection of paths to the target.



- The "path tree" is used to find sequences of iterates that lead to the target or to preimages of the target
- Apply the targeting procedure to each path
- The tree holds typically 10,000 points on 3 levels

"tree targeting" method

Step 2: Build a "forest" of trees

- Construct one tree for each target
- When we want to switch control from one saddle periodic point to another, we target using the appropriate tree.

Tree optimization

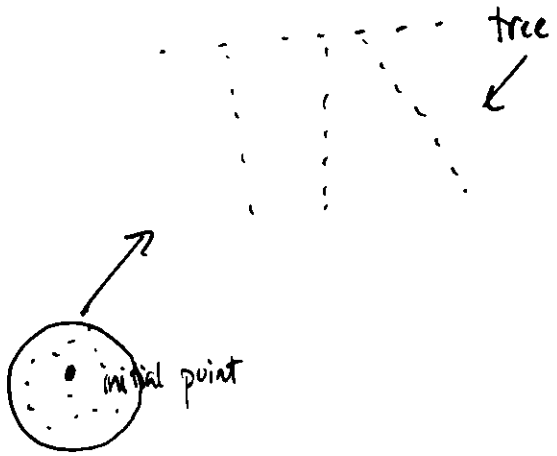
- Determine how long a "typical" initial condition takes to land on the tree.

The distribution of landing times is exponential.

The mean landing time saturates for paths of length $\gtrsim 16$.

$\langle t \rangle \doteq 70$ iterations

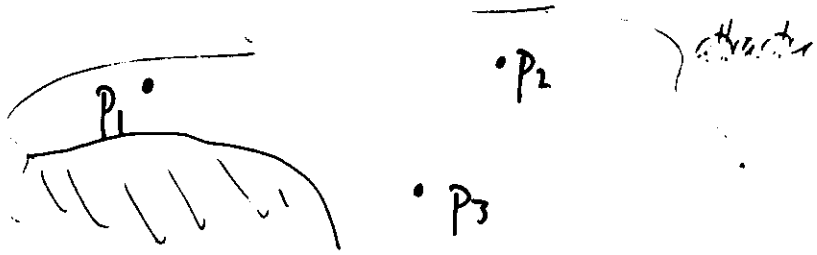
- Hasten the arrival of points on the tree by making random perturbations and following a cloud of orbits



- A cloud of 1000 points can reach a tree of 3 levels of paths of length 8 in 4-7 iterations.

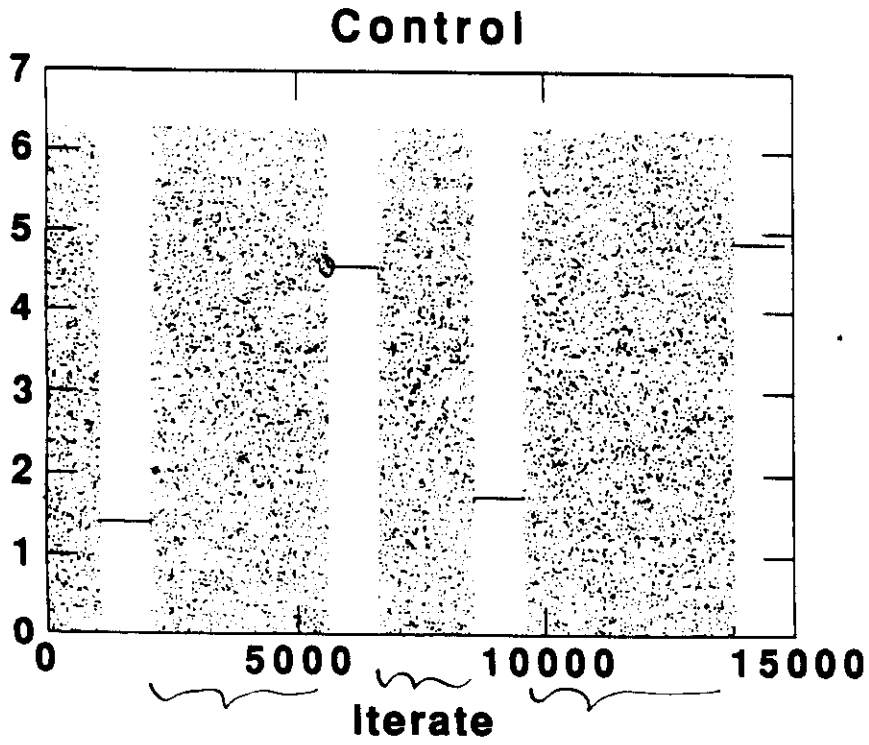
Application:

Use targeting to steer trajectories from one periodic orbit to another.



Ott, Grebogi, Yorke multi-parameter control

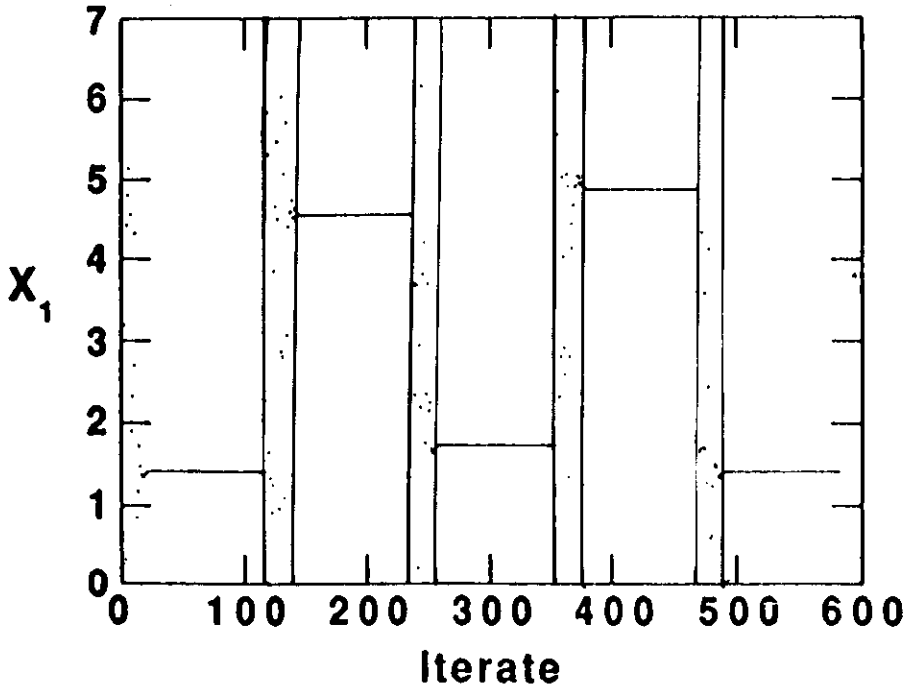
(Ernie Barreto)



There are long transients between controls

average: ~3000 iterates between controls

Eric and Ernie multiparameter targeting and control



average: 12-16 iterates

Barreto, Kostelich et al, Phys. Rev. E, Sept 1995.

Results and conclusions

- can reach 10^{-4} neighborhood of target in an average of 12-16 iterates, depending on the target
- without control, the time required to hit a 10^{-4} neighborhood of the attractor is

$$N \sim \left(\frac{1}{10^{-4}}\right)^D \approx (10^4)^{2.8} \approx 10^{11} \text{ iterates}$$

- application: choose target near a saddle periodic point. Once close, try stabilization (Romeiras et al)
- procedure can be extended in principle to any number of positive and negative Lyapunov exponents

Trees provide considerable flexibility

- It may be possible to avoid certain regions of the attractor when selecting paths for use in targeting
- Cost functions can be associated with paths in the tree
- Trees can be turned into "webs" to identify p-cycles

- In most cases, the target can be reached in 12 - 16 iterates
- The method is sensitive to noise: relative errors larger than 10^{-3} in position or 10^{-2} in control lead to significant degradation in performance (control succeeds only 50% of the time and average control time rises to ~20 iterates)
- The map must be known and be relatively easy to iterate
- Building the path tree requires a moderate amount of computation (10^4 points ⇒ 10^7 iterations)

Selected Bibliography on Time Series Analysis

to topics addressed during the lecture

See the references in each article for a more extensive bibliography.

Selected reprint collections

- P. Cvitanović, ed. *Universality in Chaos*. Bristol: Adam Hilger Ltd., 1984.
- B.-L. Hao, ed. *Chaos II*. Singapore: World Scientific, 1990.
- S. Vohra, M. Spano, M. Shlesinger, L. Pecora, and W. Ditto, eds. *Proceedings of the First Experimental Chaos Conference*. Singapore: World Scientific, 1992.

Embedding methods

- F. Takens, "Detecting Strange Attractors in Turbulence," in *Dynamical Systems and Turbulence*, ed. by D. A. Rand and L.-S. Young. Lecture Notes in Mathematics, Vol. 898 (Berlin: Springer-Verlag, 1981), pp. 366 381.
- T. Sauer, J. A. Yorke and M. Casdagli, *J. Stat. Phys.* **65**, 579 (1991).
- T. Sauer, "Time Series Prediction by Using Delay Coordinate Embedding," in *Time Series Prediction*, ed. by A. S. Weigend and N. A. Gershenfeld (Reading, MA: Addison-Wesley Publishing Company, 1994), pp. 175–194.

Selected experiments

- A. Brandstater and H. L. Swinney, *Phys. Rev. A* **35**, 2207 (1987).
- J. C. Roux, R. H. Simoyi, and H. L. Swinney, *Physica D* **8**, 257 (1983).

Dynamics from data and noise reduction

- J.-P. Eckmann and D. Ruelle, *Rev. Mod. Phys.* **57**, 617 (1985).
- P. Grassberger, R. Hegger and H. Kantz, *Chaos* **3**, 127 (1993).
- E. J. Kostelich and J. A. Yorke, *Physica D* **41**, 183 (1990).
- E. J. Kostelich and T. Schreiber, *Phys. Rev. E* **48**, 1752 (1993).
- E. J. Kostelich, *Physica D* **58**, 138 (1992).

Statistical methods and surrogate data

- F. Takens, *Inter. J. Bifurcation and Chaos* **3**, 241 (1993).
- J. Theiler and S. Eubank, *Chaos* **3**, 771 (1993).

Targeting and control

- T. Shinbrot, C. Grebogi, E. Ott, and J. A. Yorke, *Phys. Rev. A* **45**, 4165 (1992).
- F. J. Romeiras, C. Grebogi, E. Ott, and W. P. Dayawansa, *Physica D* **58**, 165 (1992).
- E. J. Kostelich, C. Grebogi, E. Ott and J. A. Yorke, *Phys. Rev. E* **47** 305, (1993).
- E. Barreto, E. J. Kostelich C. Grebogi, E. Ott, and J. A. Yorke, *Phys. Rev. E* **51**, 4169 (1995).

Physica D 58 (1992) 138–152
North-Holland

PHYSICA D

Problems in estimating dynamics from data

Eric J. Kostelich
Department of Mathematics, Arizona State University, Tempe, AZ 85287, USA

Received 8 October 1991
Revised 25 February 1992
Accepted 2 March 1992

This paper discusses some difficulties in estimating dynamics from time-delay embeddings of experimental data that can be characterized as low-dimensional. A new procedure is described to reduce noise by exploiting the properties of saddle periodic orbits on the reconstructed attractor.

1. Introduction

There are several different approaches to noise reduction in time series data whose underlying dynamics can be described as low dimensional. Kostelich and Yorke [1,2] outlined a procedure that uses an approach originally suggested by Eckmann and Ruelle [3] for computing Lyapunov exponents. Farmer and Sidorowich [4] describe another method for use when the dynamics are known. Schreiber and Grassberger [5] describe a simple time-series based approach. Cawley and Hsu [6] have suggested an approach based on projecting trajectories onto planes that locally approximate the manifold containing the attractor[#1].

Most of these methods involve the estimation of a derivative from the data or in some way require a least squares estimate of the location of some portion of the attractor. This paper describes some of the problems inherent in the estimation of dynamics from data, regardless of the type of model used to approximate the dynamics. These difficulties may arise from the fractal structure of the attractor and errors in all the observations. The problems persist regardless of the amount of available data and affect one's ability to determine an accurate local model of the dynamics, even when an accurate model should be obtainable in principle. These issues are discussed in section 2.

Many of these problems can be circumvented by using as much dynamical information as possible in the formulation of the statistical relationship between the observations. One attempt to do this involves the use of recurrent orbits to derive an accurate linear model of the dynamics in the vicinity of saddle periodic orbits on the attractor. Section 3 describes the method and its application to two experimental data sets previously analyzed in [2].

2. Statistical estimation of dynamical information

A standard procedure in the analysis of chaotic experimental data is to reconstruct the attractor using the method of time delays [8]. Let $\{s_i\}_{i=1}^{N}$ be a time series of N scalar observations, sampled at equal intervals. The reconstructed attrac-

[#1] The reprint collection by Hao [7] contains several articles on the analysis of low dimensional chaotic experimental data.

tor consists of points of the form

$$x_i = (s_i, s_{i+\tau}, \ldots, s_{i+(m-1)\tau}) \tag{1}$$

where m is the *embedding dimension* m and τ is the *time delay*. (The discrete sampling means that we may still treat the observed dynamics as a map, and x_i refers to the point on the reconstructed attractor whose first coordinate is the ith observation in the time series.) Takens [8] shows that under suitable hypotheses, this reconstruction is equivalent in some sense to the original attractor if m is large enough. (Sauer et al. [9] consider the embedding problem in greater generality. Theiler [10] and Fraser [13] discuss strategies for choosing m and τ.)

The simplest model for estimating the dynamics is a local linear one suggested by Eckmann and Ruelle [3]. Let x_{ref} be a reference point on the reconstructed attractor, and let $\{x_i\}_{i=1}^n$ be a collection of points in a small neighborhood containing x_{ref}. Let $\{y_i\}$ be the corresponding images; i.e., y_i is the point to which x_i maps at some later time. The "true" dynamics are $y_i = f(x_i)$ for some unknown function f. In the Eckmann–Ruelle approach, f is approximated as

$$f(x) \approx Ax + b, \tag{2}$$

where A is an $m \times m$ matrix and b is an m-vector. Here A is an estimate of the Jacobian matrix Df evaluated at x_{ref}. This is an accurate model of the dynamics if all the observations and their images are contained in sufficiently small neighborhoods of the reference point and its image.

Estimates of A and b can be found by ordinary linear regression. Although the basic approach is straightforward, there are many factors that affect one's ability to estimate the map A accurately.

Some limiting factors are the result of the dynamics. For instance, the dimension d of the attractor cannot be too large, since the number of data points needed to occupy a ball of a fixed size ϵ around a typical point on the attractor is proportional to $1/\epsilon^d$ [14]. The ball size ϵ cannot be too large, because nonlinearities tend to grow with ϵ, making eq. (2) a less accurate approximation of the dynamics at the point x. In most cases, the maximum value of ϵ that yields a good linear approximation depends on the reference point.

These issues are important, but they will not be considered further in this paper. Instead, we are interested in problems that limit one's ability to determine the dynamics from experimental data, even in cases where local linear models like eq. (2) in principle should be good approximations. There are many statistical difficulties arising from the presence of noise and the fractal character of the attractor that deserve more careful attention than they have received in the dynamics literature.

2.1. *Ill conditioned least squares models*

Most low dimensional chaotic attractors have a fractal, striated structure—points tend to form a Cantor set of layers. The layers can be difficult to distinguish because of the limited resolution and size of typical data sets; instead they resemble a curve. Figure 1 shows a portion of the attractor reconstructed from an x-coordinate time series generated by the Hénon map [15]

$$x_{n+1} = 1 - \alpha x_n^2 + y_n,$$
$$y_{n+1} = \beta x_n \tag{3}$$

with the usual parameters $\alpha = 1.4$, $\beta = 0.3$.

If we use eq. (2) to approximate the dynamics in this region of the plane, then we must find the best 2×2 matrix that fits the data. However, all the observations lie nearly along a straight line. In this case, we should change coordinates so that one axis is parallel to the line containing the observations. The map stretches points along this line by an amount that can be determined readily from the images of the observations. The map contracts points in the orthogo-

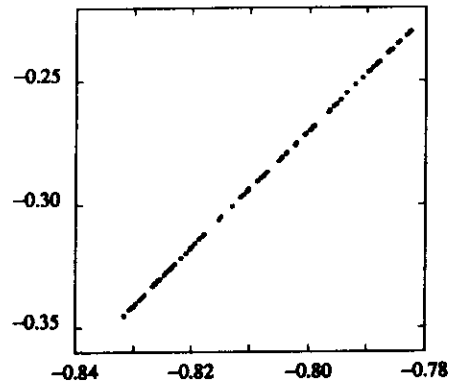


Fig. 1. A typical portion of the Hénon attractor where the points are nearly collinear.

nal direction, but this cannot be quantified readily from these data. It becomes more difficult to estimate the rate of contraction of nearby layers of points as the noise level increases, because noise obscures the structure at small scales. The Lorenz [16] and Rössler [17] attractors are two other familiar examples of attractors where similar problems occur: the points contained in a small box around a typical trajectory tend to be coplanar.

This kind of attractor structure leads to *ill conditioned* least squares problems; i.e., the covariance matrix of the observations is nearly singular. The concept is important because the numerically computed solution of an ill conditioned least squares problem has a large relative error. In this case, ill conditioning means that the Jacobian matrix A in eq. (2) cannot be estimated accurately.

The *singular value decomposition* can be used to detect situations like these. Let $\{x_i\}_{i=1}^n$ be the collection of points plotted in fig. 1, let $\bar{x}$ be their mean, and let X be the $n \times m$ matrix whose ith row is $x_i - \bar{x}$. (We assume $n \geq m$, where $m = 2$ is the embedding dimension for the Hénon data.) Then there exist an $n \times m$ matrix U, an $m \times m$ diagonal matrix Σ, and an $m \times m$ matrix V such that

$$X = U\Sigma V^{\mathrm{T}}. \tag{4}$$

The columns of U and V form an orthonormal basis for the columns and rows of X, respectively [18]. Equation (4) is the *singular value decomposition* of X and can be arranged so that the diagonal elements σ_i of Σ satisfy $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_m \geq 0$. These are the *singular values* of X and correspond to the nonnegative square roots of the eigenvalues of the covariance matrix $X^{\mathrm{T}}X$.

The singular value decomposition allows one to determine whether the least-squares problem in eq. (2) is singular: if the rank of X is r, then only the first r singular values are positive. There is good numerical software for computing the singular value decomposition of a matrix [18]. However, the numerically computed singular values are rarely zero due to roundoff error (even for a singular matrix), but they are small. The *condition number* κ is defined as the ratio σ_1/σ_r. Large values of κ correspond to ill conditioned least-squares problems in that upper bounds for the relative error of the solutions are proportional to κ and sometimes to κ^2. (The relevant theorems are technical and will not be stated here; consult [19] or [18] for more details.)

A difficult question is how to decide whether a problem is ill conditioned when the data in X are known only to a finite accuracy. Some criteria are given in [18] and [19]. Numerical experiments by the author on the Hénon map and laboratory data sets suggest that least squares solutions to eq. (2) are most accurate if the condition number of the problem is of order 10 or less. If the condition number of the original problem with m coordinates (corresponding to the embedding dimension) is larger than 10, then the problem is solved with fewer coordinates. In other words, if $\sigma_1/\sigma_{k+1} > 10$ but $\sigma_1/\sigma_k \leq 10$ then the least-squares problem is solved by first projecting the rows of the observation matrix X onto the subspace spanned by the first k columns of V. In this case, the matrix A computed in eq. (2) has rank k.

The choice of 10 as the upper limit for the condition number is heuristic. A suitable value de-

pends on the noise level and the size of the balls containing the observations for the least squares. If the data are noisy and the fractal character of closely spaced trajectories cannot be distinguished, then 10 probably is a good choice. In this case, the points are smeared along a single thick line, and one cannot estimate the rate at which nearby trajectories are contracted together. A small upper bound like 10 for the condition number usually prevents this.

The points in fig. 1 form an ill conditioned set in that the condition number $\kappa = 348$. The span of the first column of V essentially contains all the observations. So we project them onto the corresponding one dimensional subspace, compute an estimate of the expansion in this direction, and change coordinates back to get the matrix A (whose rank is 1) in eq. (2).

Ill conditioned problems become more common as the embedding dimension increases. For example, the author took a Hénon map time series of 65 536 values to which 0.1% noise had been added and embedded it in 4 dimensions. The resulting least squares problems were ill conditioned (i.e., the condition number of the 4-dimensional least squares problems was larger than 10) about 2/3 of the time.

The dynamics underlying the time series are also important. In some cases, like the Lorenz flow, initial conditions approach the attractor very rapidly, and most of the observations tend to lie along a single sheet. When this happens, it is usually possible to obtain accurate information only about the expansion rates of points on the attractor.

Farmer and Sidorowich [4] have advocated a noise reduction procedure that exploits information about the expanding and contracting directions to locate a new, less noisy trajectory close to the observed one. The idea is to see how a small uncertainty grows in the expanding direction upon forward iteration of the map and how the uncertainty grows in the contracting direction on backward iteration. When the underlying dynamical system is known exactly, their procedure can produce a trajectory of unlimited accuracy. (A similar approach can be used to prove shadowing theorems for numerically generated trajectories of the Hénon map [20]). However, the method requires information about the contracting as well as the expanding directions at each point on the attractor. The above discussion suggests that it may be difficult to adapt their procedure to the case where the dynamics must be estimated from the data, because accurate information about the contracting directions may be unobtainable.

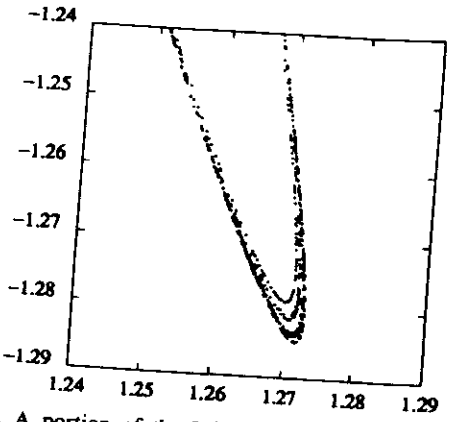


Fig. 2. A portion of the Hénon attractor. Although the points appear to fall along a parabolic curve, the dynamics are well approximated by a linear map.

The singular value decomposition provides a straightforward solution to the problem of ill conditioning, as long as one is interested in a linear model of the dynamics. However, some authors have suggested the use of quadratic and other nonlinear models for local approximations of the dynamics [21,22]. Local nonlinear models complicate the question of the best choice of variables, and the distribution of the observations can be misleading.

Consider the problem of finding the best fit for the collection of points in fig. 2 to their images. Should the linear model of eq. (2) be used with two orthogonal coordinates, or would it be better to use a quadratic model in one variable, i.e., fit the data with a parabola in some suitable coordinates? In fact, these data are generated from

the Hénon map (3), and eq. (2) provides an excellent fit. Despite the appearance of the attractor, the dynamics locally are well described by a linear map.

2.2. Outliers and influential points

Figure 3a shows a set of points selected for a least-squares fit of eq. (2). The data are generated from the Hénon map (3). In this example, the computer is instructed to take the 100 points on the reconstructed attractor closest to the reference point[#2]. Figure 3b shows the same data after 1% uniformly distributed random noise is added. All the observations lie in a box whose sides are less than 2.4% of the attractor extent in length, so eq. (2) provides a good linear approximation of the dynamics. The condition number of the associated least squares problem is about 5, so it is not ill conditioned. Nevertheless, there is a potentially serious problem in estimating the Jacobian matrix.

In this example, the strips of points will be stretched in one direction and pushed together after one iteration of the Hénon map, as illustrated in figs. 3c, d. The problem is that the three rightmost points in figs. 3a, b are relatively far removed from the rest. Here the amount of stretching along the strips is estimated using 100 points, but in the noisy example, the amount of contraction is estimated using only the rightmost 3 points.

These 3 observations are examples of *influential points*. In other words, the determinant of the 2×2 matrix computed from the data is particularly sensitive to small changes in the 3 rightmost points in figs. 3a, b, and its least squares estimate from noisy data is unreliable because the sample is so small.

[#2] The reference point is one of the three points on the right-hand side of the plot in fig. 3a, and distances are measured with the maximum norm. Although the distribution of these points may look peculiar, entirely similar situations are encountered with the use of other norms.

Influential points may be outliers, arising from relatively large "glitches" in the data. More frequently, they result from the fractal structure of the attractor. Figure 4 is a schematic illustration of a common situation. The reference point is located on one of the frequently visited portions of the attractor, represented by the closely spaced curves. (Noise may obscure the attractor structure at the smallest scales, so to the observational accuracy, the closely spaced curves may look like a single thick one.) Regardless of the details of the nearest neighbor lookup strategy, the point selection algorithm often finds a set of points whose distribution is similar to that enclosed in the circle. Suppose the map stretches points along the curves and squeezes the curves together after one iteration. Although there may be a very good linear approximation of the dynamics near the reference point, the rate of contraction is estimated using only a few points chosen from the upper (thin) curve.

The notion of an influential point is a heuristic one, and there is no formal statistical definition. Unlike ill conditioning, influential points do not necessarily affect the accuracy of least squares solutions. For example, the data in figs. 3a, c are known to the numerical precision of the computer. Here the least-squares estimate of the determinant is accurate, even though the three rightmost points are influential. The estimate is less accurate in the noisy case, and the influential points give it a large variance.

For this reason, it is a good idea to check for influential points, particularly when dealing with noisy input data. (See [23] or [24] for extensive discussions of this topic.) Let X be the $n \times m$ matrix of observations as described in section 2.1. The corresponding *prediction matrix* is defined as

$$P = X(X^{\mathrm{T}}X)^{-1}X^{\mathrm{T}}. \tag{5}$$

It can be shown [24,25] that the ith diagonal element p_{ii} of P satisfies $0 \leq p_{ii} \leq 1$ and that $1/p_{ii}$ can be regarded as the number of points in

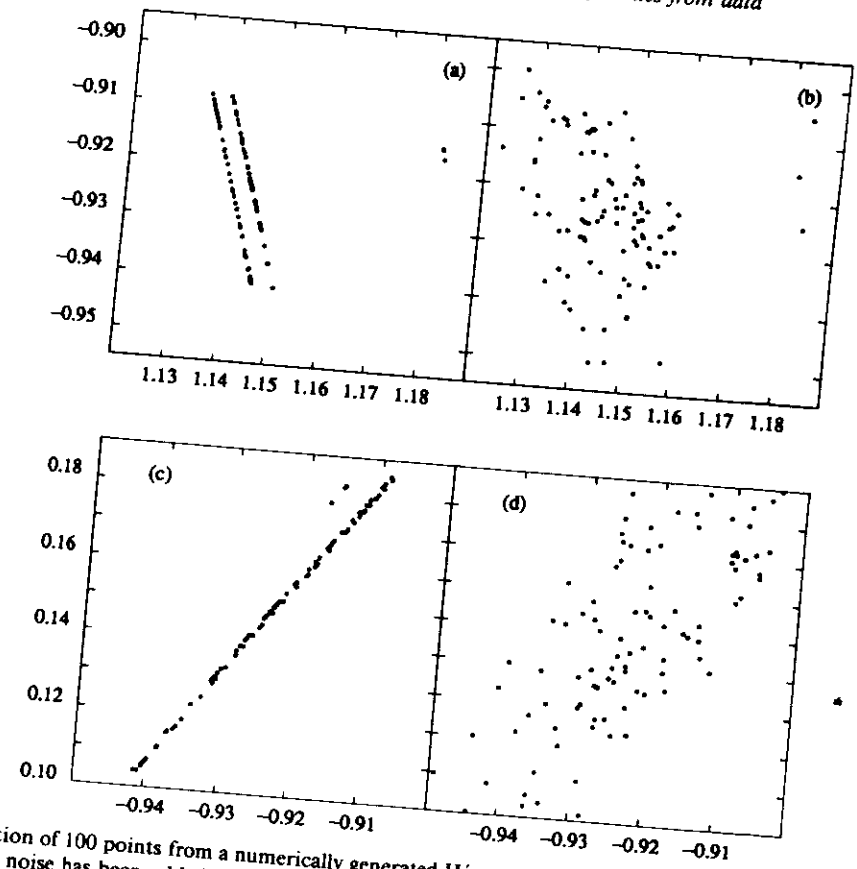


Fig. 3. (a) A collection of 100 points from a numerically generated Hénon attractor. (b) The same points after 1% uniformly distributed random noise has been added. (c), (d) The corresponding images of the points.

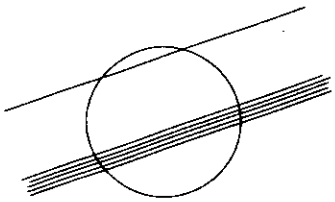


Fig. 4. Schematic illustration of how influential points can arise in least squares problems. The solid curves represent portions of the attractor. The point selection algorithm picks observations from within the circle. The points on the upper curve are influential points.

the fit that determine the corresponding observation y_i. Thus the rows in X for which the corresponding value of p_{ii} exceeds some value, say $p_{ii} > 0.2$, may be influential points [24,26].

However, it is important to note that not all points with a large value of p_{ii} have a disproportionate effect on the least-squares solution, and conversely. There is no single, clear-cut answer to the question of what should be done with influential points in the context of computing local approximations of dynamical systems. In the situation illustrated in figs. 3, the three influential points might be discarded. Although information about the rate of contraction is lost, it may not be readily determinable anyway, particularly from noisy data. (The resulting Jacobian matrix would have rank 1, as described in section 2.1.)

In many statistical applications, it is unwise simply to discard influential points. Graphical analysis of the data and the residuals is generally

recommended. However, this is impractical in applications like Lyapunov exponent estimation and noise reduction in chaotic data, where thousands of multivariate least-squares problems are solved as the attractor is traversed.

In the case of chaotic data, however, one can make reasonable assumptions about the process that generates the observations—points tend to expand in one direction and contract in others. The discarded influential points usually involve only a loss of information about the local contraction rate.

Nevertheless, it is important to assess how much the influential points affect the analysis of a given data set, such as the estimation of the Lyapunov exponents. One procedure is to calculate the Lyapunov exponents first without discarding any influential points. Then the calculation is repeated, perhaps by discarding influential points for which the corresponding value of p_{ii} is particularly large, say $p_{ii} > 0.5$. Next, some smaller values of p_{ii} can be tried (the smaller the value of p_{ii}, the more likely the points are deemed to be influential). The negative Lyapunov exponents are the most likely to be affected by the different rejection criteria. This procedure may yield a useful estimate of the accuracy with which the negative Lyapunov exponents have been calculated.

Influential points are common. For example, in the numerical experiment on the Hénon map described at the end of section 2.1, about 2/3 of the least squares problems had at least one influential point. In this experiment, a point was classified as influential if $p_{ii} > 0.25$.

It should be noted that the prediction matrix P in eq. (5) can be computed inexpensively from the QR decomposition of X. See [18] for details.

The above analysis can be extended to include the image points y_i. Figure 5 is a schematic illustration of the case where a collection of points initially are close together on the attractor but diverge into two or more groups later. This can happen when some trajectories nearly cross each other due to a poor time-delay embedding or due to the complex structure of the attractor. (For example, this situation can arise on the Lorenz attractor where trajectories cross from one lobe to the other.)

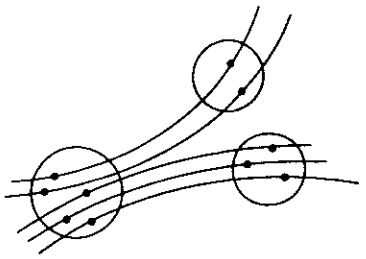


Fig. 5. Schematic diagram of diverging trajectories on an attractor.

One possibility is to form the $n \times 2m$ *augmented* matrix $(X : Y)$ [24]. The ith row of this matrix has as its first m entries the vector $x_i - \bar{x}$ and the vector $y_i - \bar{y}$ as the last m entries. (The bar denotes the mean of the corresponding observations.) The augmented prediction matrix $P_{X,Y}$ is the same as in eq. (5), except that the matrix $(X : Y)$ replaces X. A small value for the ith diagonal entry of $P_{X,Y}$ may indicate that the pair (x_i, y_i) is particularly influential in the fit, perhaps because it is on a diverging trajectory as described above. The use of this procedure may obviate the use of a "global" and a "local" embedding dimension as discussed for instance in [21].

Another alternative is to replace least squares estimation with a different minimizing function, a topic that will not be considered here. The use of so-called "robust statistics" offers many interesting possibilities for the analysis of chaotic experimental data. The monograph by Tong [27] considers the application of robust statistics to certain nonlinear time series models.

2.3. The errors-in-variables problem

Let us consider the problem of fitting a straight line to a collection of data. The classical least-squares problem assumes that

$$y_i = ax_i + b + \epsilon_i, \tag{6}$$

where each observation y_i is a linear function of the independent variable x_i, and the random variables ϵ_i are normally distributed with mean 0 and variance σ^2. In eq. (6), it is assumed that the only error occurs in the measurement of y_i—the values x_i are known exactly.

However, this assumption does not hold when one deals with noisy input data. Instead, all the observations are measured with error. Here the classical least-squares problem is replaced by

$$y_i = a(x_i + \delta_i) + b + \epsilon_i, \tag{7}$$

the so-called *errors in variables* model. Here δ_i and ϵ_i are independent, normally distributed random variables. It can be shown that the classical least-squares estimate of the slope is *biased*; i.e., the slope a is underestimated by an amount that depends on the variance of the δ_i and is independent of the number of observations. Similar results can be obtained in the multivariate case. (The book by Fuller [28] contains a detailed discussion of this problem and an extensive bibliography.)

Each matrix in eq. (2) computed from noisy data is inherently inaccurate as long as ordinary least squares is used to estimate it. This may be important for example in Lyapunov exponent calculations. The standard algorithm [29] for estimating the Lyapunov exponents involves the construction of an orthonormal set of vectors (the Lyapunov basis). At each iteration, the vectors are multiplied by the tangent map (here the matrix A, estimated using a ball of points around the current trajectory point), and the resulting collection of vectors is orthonormalized again. The normalization coefficients are then saved to get an estimate of the Lyapunov exponents. (See [30,31,3] for details.)

This statistical bias may create systematic errors in the estimation of the Lyapunov exponents. In the one dimensional case, for example, the Lyapunov exponent of a chaotic time series probably would be underestimated because the slope of the regression line is biased toward zero. A similar effect may hold in higher dimensions. As a numerical experiment, the author generated a time series of 32 768 x-coordinates from the Hénon map (3) with the usual parameters $\alpha = 1.4$, $\beta = 0.3$. The largest Lyapunov exponent λ_1 obtained from the numerically generated time series, using a procedure similar to that described in [30], is 0.608 bits/iteration. (This is very close to the value $\lambda_1 = 0.607$ bits/iteration obtained by direct iteration of the analytically determined tangent map.) The procedure was repeated after adding 1% uniformly distributed random noise to the original data. The estimated value of λ_1 for the noisy data is 0.562 bits/iteration, an underestimate corresponding to a relative error of 7%.

Noise introduces a tradeoff between a small ball size (needed for an accurate linear approximation of the dynamics) and a large variance in the data relative to the size of the ball. The maximum size of the neighborhood that allows a good local linearization depends on the particular dynamical system, but typically it is not larger than a few percent of the attractor extent. In this case, a noise level of 1% may correspond to a large relative variance, which can obscure the dynamics.

The above numerical experiment to estimate the largest Lyapunov exponent was done with a data set that is easy to analyze: there are 32 000 data points; the attractor is low dimensional (the pointwise dimension [14] of the Hénon attractor is about 1.25); the noise level is relatively small and uniform; and it is possible to obtain a good linearization in a fairly large neighborhood around each point (5% of the attractor extent or so). Nevertheless, noise limits

the accuracy of the Lyapunov exponent calculation. In contrast, laboratory data are more difficult to handle, because they typically generate higher-dimensional attractors, the time series are often relatively short, they may contain occasional large "glitches," and the noise levels may be higher. Even with very large data sets, noise limits one's ability to extract dynamical information like Lyapunov exponents.

3. Noise reduction using periodic orbits

Several procedures for reducing noise have been suggested in cases where the time series is generated by a low dimensional dynamical system [2,4,22,5,6]. All of them share some common features. Typically a *time delay embedding* is used to reconstruct the underlying attractor [8]. A local model of the dynamics, similar to eq. (2), is constructed in each of a set of small neighborhoods covering the attractor[#3]. The observed trajectories in these neighborhoods are then adjusted slightly so that they better satisfy some criterion which depends on the local approximations to the dynamics.

Although these methods can be effective in reducing noise, they have drawbacks. For example, the local approximations to the dynamics usually are determined by fitting the observed points to the model with ordinary least squares. Because all the observations are measured with error, the ordinary least squares solutions for the parameters of the model are biased, as described in section 2.3. Moreover, the adjustment of the trajectories may involve the composition of long sequences of the statistically determined maps, and this may lead to systematic errors.

Recent work has shown that saddle periodic orbits govern the dynamics on typical low-dimensional chaotic attractors [32,33]. In many cases, it is possible to locate saddle orbits on attractors reconstructed from experimental data by finding recurrent trajectories. These can be used to estimate the topological entropy [34] and other invariants associated with the dynamics [35,36].

In this section we describe how saddle periodic orbits can be used to reduce the noise in experimental data. The idea is to linearize around the saddle orbit to find a model that is linear in the parameters like eq. (2). However, the model describes a functional relationship between each recurrent orbit and its successor. That is, the parameters of the map and the adjustments to the observed orbits are determined simultaneously. Only *one* map is calculated for each recurrent orbit. Moreover, the procedure operates directly on sections of the time series without using a time-delay embedding of the attractor. Because only one map is calculated in order to adjust all the points on each of the trajectories near the saddle orbit, the procedure is computationally efficient—over 10 times faster than previous methods.

3.1. Description of the algorithm

The procedure examines the dynamics near saddle periodic orbits. Although saddle orbits are unstable, nearby initial conditions may be pushed away relatively slowly (this is often true of experimental data as discussed below). In such cases, trajectories near the saddle orbit loop around it almost periodically for a time. We say that x_i is an (m,ϵ) *recurrent point* if it returns within ϵ of itself after m iterations of the map, i.e., $\|f^m(x_i) - x_i\| < \epsilon$.

In a time delay embedding, the presence of an (m,ϵ) recurrent point means that there may be consecutive sections of m time series values that are nearly the same. Figure 6 shows a portion of a time series record from an oscillating chemical experiment analyzed in [34]. The reconstructed attractor has many points that recur after $m = 375$ time steps. The marked sections in fig. 6 show some of the respective pieces of the time

[#3] Cawley and Hsu [6] construct a local projection onto the attractor instead of a map.

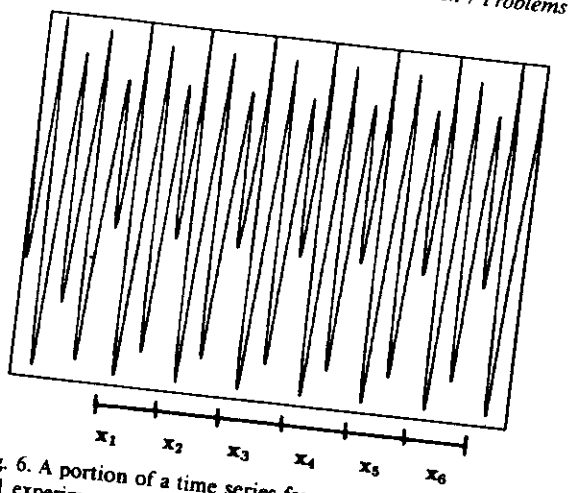


Fig. 6. A portion of a time series from an oscillating chemical experiment. The notation x_i refers to the ith section of the time series which is nearly periodic.

series. These sections appear to correspond to trajectories that loop around a saddle periodic orbit on the attractor. Here x_1 marks the first section near the saddle orbit, and x_2, x_3, etc. refer to successive sections of m values.

We will treat each section as a vector of m components. Each successive section is a function F of the previous one. We write $x_{i+1} = F(x_i)$, where F represents the dynamics near the saddle periodic orbit. This is a natural place to consider the linearized map, as follows.

Let x_f denote the saddle periodic orbit. That is, x_f represents the periodic sequence of m time series values that would be generated if the initial condition were chosen exactly on the saddle periodic orbit and the sampling interval were exactly $1/m$ times the period of the saddle orbit. If x_i and x_{i+1} are sufficiently close to x_f, then

$$x_{i+1} - x_f = F(x_i) - x_f \approx A(x_i - x_f), \qquad (8)$$

where A is the Jacobian matrix of partial derivatives of F evaluated at x_f [37]. Although the map F and its derivatives are unknown, they can be approximated from the data as described in section 2, using a model similar to eq. (2).

In the case of experimental data representing discrete observations from a flow, the recurrence time m can be large. For the data in fig. 6, we have $m = 375$. It is impractical to compute an $m \times m$ approximation of the Jacobian matrix in this example, because it requires an estimate of 140 000 parameters from a small number of recurrent trajectories.

There is considerable redundancy in the observations, however, and the singular value decomposition can be exploited to reduce the problem to the estimation of a $d \times d$ matrix where $d \ll m$ [#4]. Suppose $\{x_i\}_{i=1}^{p}$ is a collection of successive m-recurrent trajectories around some saddle orbit. Let $\bar{x}$ denote the mean of these observations. Let X be the $m \times p$ matrix whose ith column contains $x_i - \bar{x}$, and consider its singular value decomposition given in eq. (4). The columns of U form an orthonormal basis for the columns of X. Moreover, the sum of squares $\sum_{i=1}^{p} \sigma_i^2$ of the singular values equals the total variance σ_{tot}^2 in the observations x_i^* [19].

We retain only the first d singular vectors (the first d columns of U), where d is the smallest number such that

$$\sum_{i=1}^{d} \sigma_i^2 \geq C\sigma_{\text{tot}}^2 \qquad (9)$$

and project each column of the observation matrix X onto the subspace spanned by them. Here C is a number between 0 and 1, representing some fraction of the total variance. (In general, one can experiment with different values of C to determine the effects of retaining different numbers of singular vectors. In the results described below, we illustrate the analysis with $C = 0.5$ and $C = 0.75$.) In the new coordinates, the m-vector x_i is replaced by a d-vector v_i, whose entries correspond to the components of x_i along each of the d most significant singular vectors. The mean of the original observations is mapped to the origin. The saddle periodic point x_f in the

[#4] Broomhead et al. [12] describe a similar procedure for estimating the local dynamics in large dimensional systems.

original coordinates is now a d-vector v_f in the new coordinates. We let $v_f = 0$ be the initial guess.

The dynamics near the saddle orbit are modeled as in eq. (8), with respect to the basis given by the first d columns of U. The saddle orbit corresponding to the fixed point is near but not exactly at the origin in the new coordinates, and each of the projected observations still contains some error because of the noise. Although the covariance matrix for the remaining errors is not known, we assume it is a multiple of the identity matrix. This is equivalent to the following three assumptions: (1) the errors in the vectors v_i are independent and identically distributed; (2) the variance of the error in each coordinate of each point v_i is the same (this is reasonable since all the observations come from the same time series); (3) any correlation between the errors in different coordinates is negligible.

Following Schwetlick and Tiller [38], we determine values $\hat{A}$ of the Jacobian, $\hat{v}_f$ of the fixed point, and $\hat{v}_i$ of the observations in eq. (8) that minimize

$$S = \sum_i w\|\hat{v}_i - v_i\|^2 + \sum_j \|(\hat{v}_{j+1} - \hat{v}_f) - \hat{A}(\hat{v}_j - \hat{v}_f)\|^2 \qquad (10)$$

It can be shown that these are unbiased estimates under mild additional hypotheses [28]. The first summation in eq. (10) is over all the observations near the fixed point. (Here w is a weighting parameter for the original observations. In the results reported below, we have set $w = 1$.) The second sum is only between those pairs of observations for which one point is the image of the other.

The sum of squares in eq. (10) is similar in spirit to the minimization attempted in an earlier paper by Kostelich and Yorke [2]. The difference here is that the entries of the matrix A are not determined separately from the adjusted observations $\hat{v}_i$. That is, instead of determining A from linear least squares first, then adjusting the observations v_i, we treat both the observations and the entries of A as unknowns in the same nonlinear minimization problem. As mentioned above, we hope to get a better (unbiased) estimate of A, corresponding to the dynamics around the saddle orbit.

Equation (10) involves a relatively small number of successive sections of the original time series. If the procedure is applied to the data in fig. 6, for instance, the first sum in eq. 10 runs from $i = 1$ to 6 and the second sum from $j = 1$ to 5.

Schwetlick and Tiller [38] describe a Gauss–Newton method for minimizing S. It is also possible to proceed more simply and use the Polak–Ribiere conjugate gradient method described in [39]. A suitable initial guess for A usually can be obtained by ordinary least squares, and we set $v_f = 0$ initially. After the points $\hat{v}_i$ are determined, we change coordinates back to obtain the adjusted recurrent orbits.

We conclude this introduction with some remarks about finding recurrent orbits from a given experimental time series. The basic procedure is simple. First reconstruct the attractor using a time delay embedding. For each point on the attractor, find the neighboring points and see whether they return after some multiple of m time steps. (Depending on the sampling time and the underlying dynamics, reasonable values for m might range from 1 to 1000 or more.) One must also verify that successive sections of the corresponding m values in the original time series also remain close.

The size ϵ of the neighborhoods must be not be chosen so large that unrelated trajectories are considered close to the same saddle orbit, and ϵ cannot be so small that recurrent points are missed. (For instance, noise might knock a recurrent point out of a small neighborhood of the saddle orbit.) In the results reported below, different values of ϵ have been used to locate the recurrent orbits, depending on the data set. These values were chosen by trial and error to avoid

the problems just mentioned.

3.2. Results using experimental data

One difficulty that arises in the discussion of various noise reduction schemes is how to quantify the amount of noise removed from a chaotic experimental data set when the "true" dynamics are unknown. The power spectra of chaotic data have a broadband component, and it is difficult to determine how much is attributable to the dynamics and how much comes from the noise.

For the sake of comparison, we consider the two Couette–Taylor data sets previously analyzed in [2]. The first of these is degenerate in some sense because it comes from wavy Taylor vortex flow, whose dynamics are equivalent to a stable limit cycle instead of a saddle orbit. However, in this case power spectra do give a reliable indication of the noise level, since the flow is periodic.

The data set consists of 32 768 time series values that are measurements of one velocity component at equally spaced time intervals in wavy Taylor vortex flow [2]. Nearly all the observations are recurrent with a recurrence time corresponding to 30 time steps. (In general, the recurrence time is not an integer multiple of the sampling time, which always creates some small phase errors. In this data set, the observations can be divided into 5 groups of recurrent orbits—each around the same limit cycle—whose trajectories are nearly in phase.)

For each group of recurrent orbits, the singular value decomposition and the change to singular basis coordinates are done as described in section 3.1. The noise reduction is accomplished by minimizing the sum of squares in eq. (10) in the subspace spanned by first d singular vectors. We illustrate the results obtained when $d = 2$ (accounting for about half the total variance, i.e., $C \approx 0.5$ in eq. (9)) and $d = 5$ (corresponding to $C \approx 0.75$).

Figure 7a shows the power spectrum of the raw data. The power spectrum of the data adjusted with respect to only the first 2 singular basis vectors (fig. 7b) is comparable to that of the data after processing by the Kostelich–Yorke algorithm (see fig. 4 in [2]). A careful examination of the spectra in fig. 7a, b reveals a small peak near 0.3 times the Nyquist frequency in the original data that is missing from the processed data (this also occurs in [2]). However, when the first 5 singular basis vectors are used, the peak remains and the noise floor rises only slightly, as illustrated in fig. 7c. This suggests that the flow may be weakly quasiperiodic. In such cases, the present method may preserve the dynamics more faithfully than the procedure described in [2], provided that d is large enough.

The method is computationally efficient, since only one map needs to be computed for each set of recurrent trajectories. In this example, the minimization of eq. (10) corresponds to the simultaneous adjustment of dozens of trajectories representing hundreds of time series values. On a Silicon Graphics Personal Iris workstation, it takes about 8 seconds to determine all the recurrent orbits in the data set and about 43 seconds to minimize the sum of squares in eq. (10) for each of the 5 groups of recurrent trajectories. All but 1100 of the time series values are adjusted in about 50 seconds of computer time for the case $d = 5$. (In contrast, the noise reduction method described in [2] requires 15 minutes of CPU time.)

We next consider the application of the method to a time series of 32 768 values in a weakly turbulent Couette–Taylor flow [40]. (This is the same data set described in fig. 5 of [2].) A careful examination of the data set suggests that almost 31 000 of the values lie on a single recurrent orbit[#5].

The singular value decomposition and the change of coordinates to the corresponding ba-

[#5] The recurrence time corresponds to approximately 1153 time steps. The search for recurrent orbits was done by embedding the attractor in 6 dimensions and setting ϵ to 8% of the attractor extent.

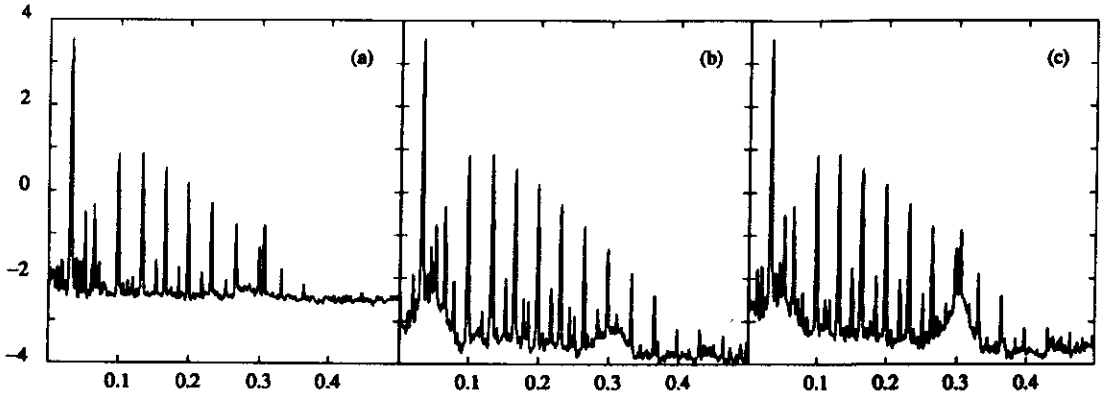


Fig. 7. (a) Power spectrum from wavy Taylor vortex flow. (b), (c) Power spectrum after noise reduction using the first 2 and the first 5 singular basis vectors. The vertical axis is the base-10 logarithm of the power spectral density; the horizontal axis is in multiples of the Nyquist frequency.

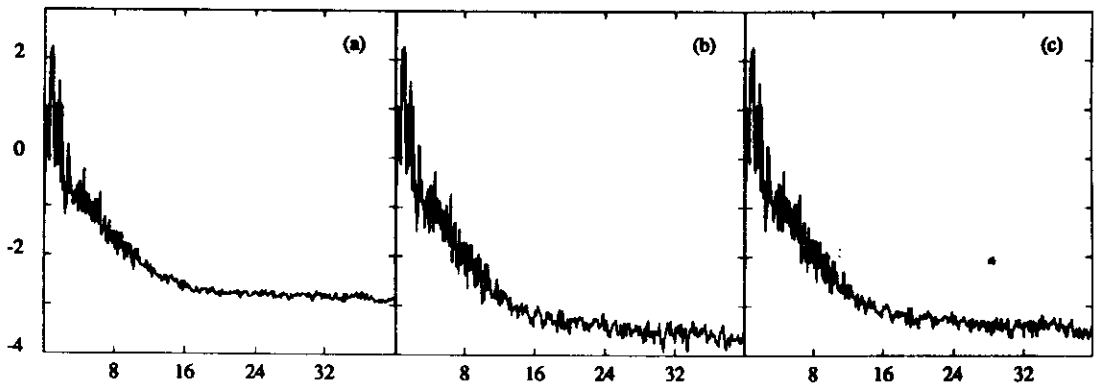


Fig. 8. Power spectra for the weakly chaotic Taylor–Couette data set described in [2]. (a) Power spectrum for the raw data. (b), (c) Power spectra using the noise reduction procedure described in the text with the first 2 and first 5 singular basis vectors, respectively.

sis is computed as described in section 3.1. As above, the noise reduction is done using the first $d = 2$ and $d = 5$ singular vectors (this captures about 50% and 75% of the variance respectively in the recurrent trajectories, like in the periodic case).

Figure 8 shows the resulting power spectra (the units on the plots are the same as those in [40]). The corresponding phase portraits closely resemble those in fig. 5 of [2] and are omitted. The results are comparable to those obtained from the Kostelich–Yorke algorithm [2].

The consistency of the statistical relationship can be checked in a cursory way by plotting the estimated saddle orbit $\hat{v}_f$ in the original time series coordinates. Figure 9 shows the saddle orbit obtained when the sum of squares is minimized using the first 5 singular vectors [#6].

However, in the $d = 5$ case, the power spectral density associated with higher frequencies in the processed data is somewhat larger than that in [2]. Because the data are chaotic, it is difficult to decide whether this higher "noise floor" corresponds to a more faithful preservation of the

[#6] The computer time required is the same as for the periodic data.

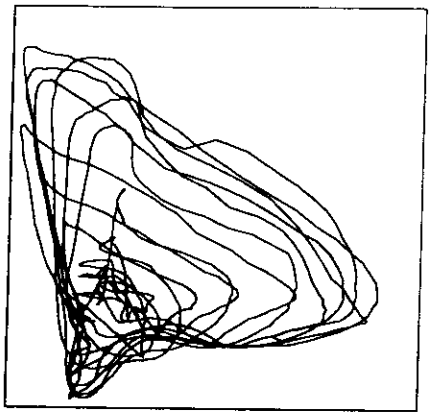


Fig. 9. The saddle orbit estimated from the chaotic Couette–Taylor data.

dynamics, a less effective noise reduction strategy, or some combination of the two.

The calculation of the singular value decomposition is subject to outliers and influential points as described in section 2.2. There are some relatively large "glitches" in these experimental data that could affect the resulting singular vector basis, thus limiting the effectiveness of the noise reduction procedure. The author tried iterating the process (i.e., using the output of one run—where most of the glitches are gone—as the input of the next), but the resulting power spectra are very similar.

Finally, the author checked the effects of different starting values for the matrix A and fixed point v_f in eq. (8). Sometimes the starting guess for A came from ordinary linear regression, and sometimes the initial guess was the identity matrix. Likewise, the fixed point v_f was originally set to zero (corresponding to the mean of the observed trajectories), and sometimes it was chosen slightly away from 0. In all cases, the numerical method converged to the same values of A, v_f, and adjusted observations $\tilde{v}_i$.

4. Conclusion and remarks

The noise reduction scheme described here is applicable only to recurrent trajectories near a periodic orbit. Points that are not near a periodic saddle orbit will not be adjusted. A noise reduction method like that in [2] can be used to adjust such non-recurrent points.

In some cases, where the data set is either strongly quasiperiodic or highly chaotic, few points are recurrent, and the method described here is not useful. For example, the distance between two typical nearby points on the Hénon attractor [15] approximately doubles after each iteration of the map. For this reason, few recurrent trajectories can be found in a moderately-sized data set.

However, recurrent trajectories are common in systems that mimic many kinds of experimental data. For instance, the author has found that slightly more than half the points are recurrent in a time series of 65 000 x coordinates obtained by numerical integration of the Lorenz equations [16] with the usual parameters[#7].

The recurrent trajectories near periodic orbits on an attractor are a natural place to find an accurate linear approximation of the dynamics. The time series values comprising the trajectories can be treated as a sequence of long vectors. The singular value decomposition can be exploited to project them onto a low dimensional subspace where a least-squares minimization method can be used to determine an approximate Jacobian matrix and a more self-consistent set of slightly adjusted observations. The adjusted time series is obtained by a simple change of coordinates. In contrast to ordinary least squares, the procedure uses the functional relationship between the recurrent trajectories in the least squares minimization. There is no artificial distinction between dependent and

[#7] The attractor is embedded in 4 dimensions and ϵ is set to 3% of the attractor extent in the search for recurrent points.

independent variables, which may avoid the bias resulting from errors in measurement. The method is computationally efficient, since only one map needs to be determined for each set of recurrent orbits.

Acknowledgements

The author thanks H. Tong and two anonymous referees for helpful comments. This work is supported by the NSF Applied and Computational Mathematics Program under grant No. DMS-9017174.

References

[1] E.J. Kostelich and J.A. Yorke, Phys. Rev. A 38 (1988) 1649.
[2] E.J. Kostelich and J.A. Yorke, Physica D 41 (1990) 183.
[3] J.-P. Eckmann and D. Ruelle, Rev. Mod. Phys. 57 (1985) 617.
[4] J.D. Farmer and J.J. Sidorowich, Physica D 47 (1991) 373.
[5] T. Schreiber and P. Grassberger, A simple noise reduction method for real data, preprint.
[6] R. Cawley and G.-S. Hsu, A local geometric projection method for noise reduction in chaotic maps and flows, preprint.
[7] B.-L. Hao, ed. Chaos II (World Scientific, Singapore, 1990).
[8] F. Takens, in: Dynamical Systems and Turbulence, eds. D.A. Rand and L.-S. Young Lecture Notes in Mathematics, Vol. 898 (Springer, New York, 1981) p. 366.
[9] T. Sauer, J.A. Yorke and M. Casdagli, J. Stat. Phys. (1992), to appear.
[10] J. Theiler, J. Opt. Soc. Am. A 7 (1990) 1055.
[11] D.S. Broomhead and G.P. King, Physica D 20 (1986) 217.
[12] D.S. Broomhead, R. Indik, A.C. Newell and D.A. Rand, Nonlinearity 4 (1991) 159.
[13] A. Fraser and H.L. Swinney, Phys. Rev. A 34 (1986) 1134.
[14] J.D. Farmer, E. Ott and J.A. Yorke, Physica D 7 (1983) 153.
[15] M. Hénon, Commun. Math. Phys. 50 (1976) 69.
[16] E.N. Lorenz, J. Atmos. Sci. 20 (1963) 130.
[17] O.E. Rössler, Phys. Lett. A 57 (1976) 397.
[18] J.J. Dongarra, C.B. Moler, J.R. Bunch and G.W. Stewart, LINPACK User's Guide (Society for Industrial and Applied Mathematics, Philadelphia, 1979).
[19] G.W. Stewart, Introduction to Matrix Computations. (Academic Press, New York, 1973).
[20] S.M. Hammel, J.A. Yorke and C. Grebogi, Bull. Am. Math. Soc. 19 (1988) 465.
[21] R. Brown, P. Bryant and H.D.I. Abarbanel, Phys. Rev. A 43 (1991) 2787.
[22] M. Casdagli, Physica D 35 (1989) 335.
[23] F.R. Hampel, E.M. Ronchetti, P.J. Rousseeuw and W.A. Stahel, Robust Statistics: The Approach Based on Influence Functions. (Wiley, New York, 1986).
[24] S. Chatterjee and A.S. Hadi, Sensitivity Analysis in Linear Regression. (Wiley, New York, 1988).
[25] H. Tong, Non-linear Time Series. (Oxford Univ. Press, Oxford, 1990).
[26] P. Huber, Robust Statistics. (Wiley, New York, 1981).
[27] H. Tong, Threshold models in nonlinear time series analysis, Lecture Notes in Statistics, No. 21 (Springer, Heidelberg, 1983).
[28] W.A. Fuller, Measurement Error Models. (Wiley, New York, 1987).
[29] G. Benettin, L. Galgani, A. Giorgilli and J.M. Strelcyn, Meccanica 15, 9 (1980).
[30] J.-P. Eckmann, S.O. Kamphorst, D. Ruelle and S. Ciliberto, Phys. Rev. A 34 (1986) 4971.
[31] M. Sano and Y. Sawada, Phys. Rev. Lett. 55 (1985) 1082.
[32] D. Auerbach, P. Cvitanović, J.-P. *Eckmann, G. Gunaratne and I. Procaccia, Phys. Rev. Lett. 58 (1987) 2387.
[33] C. Grebogi, E. Ott and J.A. Yorke, Phys. Rev. A 37 (1988) 1711.
[34] D.P. Lathrop and E.J. Kostelich, Phys. Rev. A 40 (1989) 431.
[35] N.B. Tufillaro, H.G. Solari and R. Gilmore, Phys. Rev. A 41 (1990) 5717; Phys. Rev. Lett. 64 (1990) 2350.
[36] L. Smith, Quantifying Chaos with Predictive Flows and Maps: Locating Unstable Periodic Orbits. in: Measures of Complexity and Chaos, eds. N.B. Abraham et al. (Plenum, New York, 1989).
[37] See for instance J.M.T. Thompson and H.B. Stewart, Nonlinear Dynamics and Chaos (Wiley, Chicester, 1986) Ch. 8.
[38] H. Schwetlick and V. Tiller, Technometrics 27 (1985) 17.
[39] W.H. Press, B.P. Flannery, S.A. Teukolsky and W.T. Vetterling, Numerical Recipes (Cambridge Univ. Press, Cambridge, 1986).
[40] A. Brandstäter and H.L. Swinney, Phys. Rev. A 35 (1987) 2207.

Physica D 41 (1990) 183–196
North-Holland

NOISE REDUCTION: FINDING THE SIMPLEST DYNAMICAL SYSTEM CONSISTENT WITH THE DATA

Eric J. KOSTELICH[a,b,1] and James A. YORKE[a,c]
[a] *Institute for Physical Science and Technology, University of Maryland, College Park, MD 20742, USA*
[b] *Center for Nonlinear Dynamics, Department of Physics, University of Texas, Austin, TX 78712, USA*
[c] *Department of Mathematics, University of Maryland, College Park, MD 20742, USA*

Received 3 March 1989
Revised manuscript received 11 October 1989
Accepted 18 October 1989
Communicated by R. Westervelt

A novel method is described for noise reduction in chaotic experimental data whose dynamics are low dimensional. In addition, we show how the approach allows experimentalists to use many of the same techniques that have been essential for the analysis of nonlinear systems of ordinary differential equations and difference equations.

Introduction

Numerical computation and computer graphics e been essential tools for investigating the be- ior of nonlinear maps and differential equa- s. The pioneering work of Lorenz [25] was le possible by numerical integration on a com- r, allowing him to take nearby pairs of initial litions and compare the trajectories. Hénon discovered the complex dynamics of his cele- ed quadratic map with the aid of a pro- mmable calculator. A variety of classical and ern techniques has been exploited to find peri- orbits, their stable and unstable manifolds basins of attraction [26], fractal dimension and Lyapunov exponents [10, 31, 37]. In cases, numerical methods can establish rig- ly the existence of initial conditions whose tories have essentially the same intricate ure that one sees on a computer screen [18].

Until recently, experimentalists have not been able to apply most of these methods to the analysis of experimental data, since they do not in general have explicit equations to model the behavior of their apparatus. In cases where it is possible to find accurate models of the physical system, quantitative predictions about the behavior of actual experiments are possible [17]. However, all that is available in a typical experiment is the time-dependent output (e.g. voltage) from one or more probes, which is a function of the dynamics.

One fundamental problem in the analysis of experimental data concerns the correspondence between the dynamics that governs the behavior of the apparatus and the discretely sampled time series that comprises the data. Another question is how to minimize the effect of noise. In this paper, we show how the *time delay embedding method*, now commonly used to reconstruct an attractor from experimental data, yields a novel procedure for reducing noise in data whose dynamics can be characterized as low dimensional. Moreover, we

ient address: Department of Mathematics, Arizona State ity, Tempe, AZ 85287, USA.

show how the approach can be extended to allow experimentalists access to many of the analytical tools mentioned above.

Section 2 reviews the time delay embedding method and some of its applications. Section 3 introduces some of the problems associated with traditional filters and outlines our noise reduction method.

2. The time delay embedding method

As stated in section 1, one problem in analyzing experimental data is how to relate the measurements with the dynamics. Before the early 1980's, power spectra were the principal method for analyzing such data. For instance, Fenstermacher et al. [13] relied heavily on power spectra to detect transitions from periodic to weakly turbulent flow between concentric rotating cylinders. However, Fourier analysis alone is inadequate for describing the dynamics.

Other methods also have been used to analyze time series output from dynamical systems. Lorenz [25] used next amplitude maps to describe some features of the dynamics; that is, he plotted z_{n+1} against z_n where z_n is the nth relative maximum of the third coordinate of the numerically calculated solution. Such maps are often useful, not only for investigating features of the Lorenz attractor [32], but also for instance in experiments on intermittency in oscillating chemical reactions [30].

In the past decade, the time delay embedding method has come into common use as a way of reconstructing an attractor from a time series of experimental data. In this approach, one supposes that the dynamical behavior is governed by a solution traveling along an attractor[*1] (which is not observable directly). However, one assumes there is a smooth function that maps points on the attractor to real numbers (the experimental measurements). In the embedding method, one generates a set of m-dimensional points whose coordinates are values in the time series separated by a constant delay [11]. For example, when $m = 3$, the reconstructed attractor is the set of points $\{x_i = (s_i, s_{i+\tau}, s_{i+2\tau})\}$ where τ is the time delay. Takens [34] has shown that under suitable hypotheses, this procedure yields a set whose properties are equivalent to those of the original attractor provided that the embedding dimension m is large enough.

In principle, the embedding method allows one to study the dynamics in detail. The earliest applications may be called *static* in that the analysis focuses on the geometric properties of the set of points on the reconstructed attractor. For example, phase portraits and Poincaré sections are used in ref. [5] to help determine the transition between quasiperiodic and chaotic flow in a Couette–Taylor experiment. Another important application is the estimation of attractor dimension from experimental data, for which there is a large literature [27]. In addition, various information theoretic notions can be used to find good choices of embedding dimension and time delay [15].

More recent applications of the embedding method are quite different in nature and can be called *dynamic* in that information about the dynamics is stored in the computer for analysis. With each data vector x_i, one stores the "next" vector, for example, $x_{i+\delta}$ for some $\delta > 0$. This makes it possible to compute a linear approximation of the dynamics in a neighborhood of x_i, assuming that there is a low-dimensional dynamical system underlying that data[*2]. In particular, a linear approximation provides an estimate of the Jacobian of the map at x_i [11]. Eckmann et al. [10] use linear maps computed in this way to integrate a set of variational equations and find the positive Lyapunov exponents[*3].

[*1] Existing numerical methods require the attractor to be low dimensional.

[*2] This material was first presented by D. Ruelle at a Nobel symposium in 1984.

[*3] Wolf et al. [37] have proposed a different method in which nearby pairs of points are followed to estimate the largest Lyapunov exponent.

In fact, the time delay embedding method pro-ides a powerful set of tools for analyzing the lynamics, the breadth of which may not have een realized by Eckmann and Ruelle. In the emainder of this paper, we discuss two novel pplications that are possible, specifically:

(1) *Noise reduction*. Since one can approximate ıe dynamics at each point, it becomes possible to entify and correct inaccuracies in trajectories ising from random errors in the original time ries. Numerical evidence suggests that the noise duction procedure described below improves the curacy of other analyses, such as Lyapunov ponents and dimension calculations.

(2) *Simplicial approximations*. Linear approxi-tions can be computed at each point on a grid a neighborhood of the attractor to form a ıplicial approximation of the dynamical system. s can be used to locate unstable periodic orbits r the attractor.

consider noise reduction in section 3.

Noise reduction

ıe ability to extract information from time-ing signals is limited by the presence of noise. :nt experiments to study the transition to tur-ıce in systems far from equilibrium, like those enstermacher et al. [13], Behringer and Ahlers ınd Libchaber et al. [24], succeeded largely ıse of instrumentation that enabled them to tify and reduce the noise. However, it is often ısive and time consuming to redesign experi-al apparatus to improve the signal to noise

important question, therefore, is how the mental data can be filtered or otherwise ocessed before it is analyzed further. One on approach is to use Fourier analysis: one model the noise as a collection of high-ncy components and subtract them from a spectrum (or Fourier transform) of the in-ta. The transform can be inverted to yield a new time series with some of the high-frequency components removed. This is the basic idea behind Wiener and other bandpass filters [29].

However, as noted previously, power spectral analysis is insufficient to characterize the dynamics when the data are chaotic. Since the power spectrum of a low-dimensional chaotic signal resembles that of a noisy one, the suppression of certain frequencies can alter the dynamics of the filtered output signal. Badii et al. [1] have shown that a simple low-pass filter effectively introduces an extra Lyapunov exponent that depends on the cutoff frequency. If the cutoff frequency is sufficiently low, then the filter can increase the fractal dimension of the reconstructed attractor. This result also has been confirmed by Mitschke et al. [28] with data from an electronic circuit.

We now consider a different approach and show how the time delay embedding method can be exploited to reduce the noise, at least in cases where the time series can be viewed as a dynamical system with a low-dimensional attractor. Our objective is to use the dynamics to detect and correct errors in trajectories that result from noise. This is done in two steps once an embedding dimension m and a time delay τ have been fixed.

In the first step, we consider the motion of an *ensemble* of points in a small neighborhood of each point on the attractor in order to compute a linear approximation of the dynamics there. In the second step, we use these approximations to consider how well an *individual* trajectory obeys them. That is, we ask how the observed trajectory can be perturbed slightly to yield a new trajectory that satisfies the linear maps better. The trajectory adjustment is done in such a way that a new time series is output whose dynamics are more consistent with those on the phase space attractor.

This approach is fundamentally different from traditional noise reduction methods. Because we consider the motion of points on a phase space attractor, we are using information in the original signal that is not localized in a time or frequency domain. Points that are close in phase space correspond to data that in general are widely and

irregularly spaced in time, due to the sensitive dependence on initial conditions on chaotic attractors. In contrast, Kalman [4] and similar filters examine data that are closely spaced in time; bandpass filters operate in the frequency domain.

4. Eckmann–Ruelle linearization

The discrete sampling of the original signal means that the points on the reconstructed attractor can be treated as iterates of a nonlinear map f whose exact form is unknown. We assume that f is nearly linear in a small neighborhood of each attractor point x and write

$$f(x) \approx Ax + b \equiv L(x)$$

for some $m \times m$ matrix A and m-vector b. (The matrix A is the Jacobian of f at x.)

This approximation, which we call the *Eckmann–Ruelle linearization* at x, can be computed with least-squares methods similar to those described in refs. [11, 10]. Given a reference point x_{ref}, let $\{x_i\}_{i=1}^{n}$ be a collection of the n points which are closest to x_{ref}. With each point x_i we store the next point (i.e., the image of x_i), denoted y_i[*4]. The kth row a_k of A and the kth component b_k of b are given by the least-squares solution of the equation

$$y_k = b_k + a_k \cdot x, \qquad (1)$$

where y_k is the kth component of y and the dot denotes the dot product. Fig. 1 illustrates the idea[*5].

[*4] The points x_i are points on the attractor which are *not* consecutive in time. The subscript i merely enumerates all the points on the attractor contained within a small distance ϵ of x_{ref}. In this notation, x_i and y_i are consecutive in time.

[*5] Farmer and Sidorowich [12] observe that the Eckmann–Ruelle linearization can be used for prediction. Given a reference point x_i, find the Eckmann–Ruelle linearization $A_i x + b_i$, compute $x_{i+1} = A_i x_i + b_i$, and repeat the process to get the predicted trajectory.

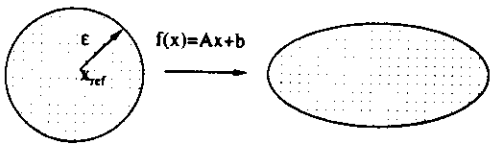


Fig. 1. Schematic diagram for the first stage of the noise reduction method. A collection of points in an ϵ-ball about the reference point x_{ref} is used to find a linear approximation of the dynamics there.

We mention three difficulties in computing the local linear approximations in the subsections below.

4.1. Ill conditioned least squares

There is a particular problem when one tries to compute solutions to eq. (1) with a finite data set of limited accuracy that has not been addressed in previous papers [10, 31]. Suppose for example that all the points in a neighborhood of x_{ref} lie nearly along a single line, i.e., the attractor appears one dimensional within the available resolution. Although it is possible to measure the expansion along the unstable manifold at x_{ref}, there are not enough points in other directions to measure the contraction. Hence it is not possible to compute a 2×2 Jacobian matrix accurately. Any attempt to do so will result in an estimate of the Jacobian whose elements have large relative errors. This kind of least-squares problem is *ill conditioned*.

The ill conditioning can be avoided by changing coordinates so that the first vector in the new basis points in the unstable direction[*6]. A one-dimensional approximation of the dynamics is computed using the new coordinates; that is, we approximate the dynamics only along the unstable manifold. We recover the matrix A by changing coordinates back to the original basis.

For example, if we are working in the plane and the unstable direction is the line $y = x$, then we rotate the coordinate axes by 45°. The dynamics are approximated by a one-dimensional linear map

[*6] This is done by computing the right singular vectors [9] of the $n \times m$ matrix whose jth row is x_j. The procedure is called *principal component analysis* in the statistical literature.

computed along the line $y = x$. Then we rotate back to the original coordinates. (The resulting matrix A has rank 1 in this example.) This approach substantially enhances the robustness of the numerical procedure.

4.2. Finding nearest neighbors

A second problem is finding an efficient way to locate all of the points closest to a given reference point. The dynamical embedding method imposes stringent requirements on any nearest-neighbor algorithm. The storage overhead for the corresponding data structures must be small, because there are tens of thousands of attractor points. The algorithm must be fast, since there is one nearest-neighbor problem for each linear map to be computed.

We solve this problem by partitioning the phase space into a grid of boxes that is parallel to the coordinate axes. Each coordinate axis is divided into B intervals. (Fig. 2 illustrates the grid in two dimensions.) Each point on the attractor is assigned a box number according to its coordinates. For example, a point on the plane whose first coordinate falls in the jth interval (counting from 0) along the x axis and whose second coordinate falls in the kth interval along the y axis is assigned to box number $kB + j$. The list of box numbers is sorted, carrying along a pointer to the original data point. Given a reference point x_{ref}, its box number is found using the above formula. A binary search in the list of box numbers then locates the address of x_{ref} and all the other points in the same box number. The search is extended if necessary to adjacent boxes.

$B^2 - B$	$B^2 - B + 1$	$B^2 - B + 2$	...	$B^2 - 1$
⋮	⋮	⋮	⋮	⋮
B	$B + 1$	$B + 2$	...	$2B - 1$
0	1	2	...	$B - 1$

Fig. 2. Box numbering scheme in two dimensions. The attractor is normalized to fit in the unit square. The bottom row of boxes rests against the x axis and the leftmost row of boxes against the y axis.

Only a crude partition is needed for this algorithm to work efficiently (typically we choose $B = 40$), and the grid is extended only to the first three coordinate axes. When the embedding dimension is larger than three, a preliminary list of nearest neighbors is obtained using only the first three coordinates of each attractor point. The final list is extracted by computing the distances from x_{ref} to each point in the preliminary list.

Although there are circumstances where this algorithm can perform poorly (e.g., when most of the attractor points are concentrated in a handful of boxes), the distribution of points on typical attractors is sufficiently uniform that the running time is very fast. Memory use is also efficient: a set of N attractor points requires $3N$ storage locations. In contrast, the tree-search algorithm advocated in ref. [12] requires several times more storage (although the lookup time is probably slightly less). Because $N \approx 10^5$ in typical applications, we believe that the box-grid approach (or some variant) is the most practical. A survey of other nearest-neighbor algorithms is given in ref. [3].

4.3. Errors in variables

There is a potential difficulty in the use of ordinary least squares to compute the linear maps. In the usual statistical problem of fitting a straight line, one has observations (x_i, y_i) where x_i is known exactly and y_i is measured. One assumes that $y_i = a_0 + a_1 x_i + \epsilon_i$, where the ϵ_i are independent errors drawn from the same normal distribution. (Analogous assumptions hold in the multivariate case.) In the present situation, however, both x_i and y_i are measured with error. It can be shown that the ordinary least-squares method produces biased estimates of the parameters a_0 and a_1 in this case [16, 23]. In practice this does not seem to be a serious problem, but statistical procedures to handle this situation (the so-called "errors in variables" methods) may provide

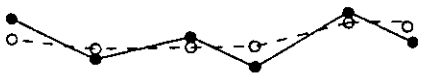


Fig. 3. Schematic diagram of the trajectory adjustment procedure. The trajectory defined by the sequence $\{x_i\}$ is perturbed to a new trajectory given by $\{\hat{x}_i\}$ which is more consistent with the dynamics. In this example we show what the perturbed trajectory might look like if the dynamics were approximately horizontal translation to the right.

an alternative approach to noise reduction. We consider this question in the appendix.

5. Trajectory adjustment by minimizing self-inconsistency

The Eckmann–Ruelle linearization procedure described above is computed and the resulting maps are stored for a sequence of reference points along a given trajectory (for the results quoted here, the sequence usually contains 24 points). We now consider how to perturb this trajectory so that it is more consistent with the dynamics. The objective is to choose a new sequence of points $\hat{x}_i$ to minimize the sum of squares

$$\sum w\|\hat{x}_i - x_i\|^2 + \|\hat{x}_i - L_{i-1}(\hat{x}_{i-1})\|^2 + \|\hat{x}_{i+1} - L_i(\hat{x}_i)\|^2, \quad (2)$$

where $L(x_i) = A_i x_i + b_i$, w is a weighting factor, and the sum runs over all the points along the trajectory*7. Eq. (2) can be solved using least squares. Heuristically, eq. (2) measures the self-inconsistency of the data, assuming that the linear approximations of the dynamics are accurate. See fig. 3. We say the new sequence $\{x_i\}$ is more *self-consistent*.

*7 In the results described in this paper, the Eckmann–Ruelle linearization procedure is done using a collection of points within a radius of 1–6% of each reference point, depending on the embedding dimension, the dimension of the attractor, and the number of attractor points. This results in collections of 50–200 points per ball, which gives reasonably accurate map approximations without making the computer program too slow. The weighting factor w is set to 1.

The trajectory adjustment can be iterated. That is, once a new trajectory $\hat{x}_i$ has been found, one can replace each x_i in eq. (2) by $\hat{x}_i$ and compute a new sequence $\{\hat{\hat{x}}_i\}$.

We place an upper limit on the distance a point can move. Points which seem to require especially large adjustments can be flagged and output unchanged. (This may be necessary if the input time series contains large "glitches" or if nonlinearities are significant over small distances in certain regions of the attractor.)

When the input is a time series, we modify the above procedure slightly since we require a time series as output. The trajectory adjustment is done so that changes to the coordinates of x_i (corresponding to particular time series values) are made consistently for all subsequent points whose coordinates are the same time series values. For example, suppose the time delay is 1 and the embedding dimension is 2. Then trajectories are perturbed so that the second coordinate of the ith point is the same as the first coordinate of the $(i+1)$st point. That is, when $x_i = (s_i, s_{i+1})$ is moved to the point $\hat{x}_i = (\hat{s}_i, \hat{s}_{i+1})$, we require that the first coordinate of $\hat{x}_{i+1}$ be $\hat{s}_{i+1}$.

6. Results using experimental data

We note that the attractor need not be chaotic for this noise reduction procedure to be effective. Fig. 4a shows a phase portrait of noisy measurements of wavy vortex flow in a Couette–Taylor experiment [20]. This flow is periodic, so the attractor is a limit cycle (widened into a band because of the noise) and the power spectrum consists of one fundamental frequency and its harmonics above a noise floor. See fig. 4b. Figs. 4c, 4d show the same data after noise reduction. The noise reduction procedure makes the limit cycle much narrower, and the noise floor in the power spectrum is reduced by almost two orders of magnitude. However, no power is subtracted from any of the fundamental frequencies, and in fact some harmonics are revealed which previously were obscured by the noise.

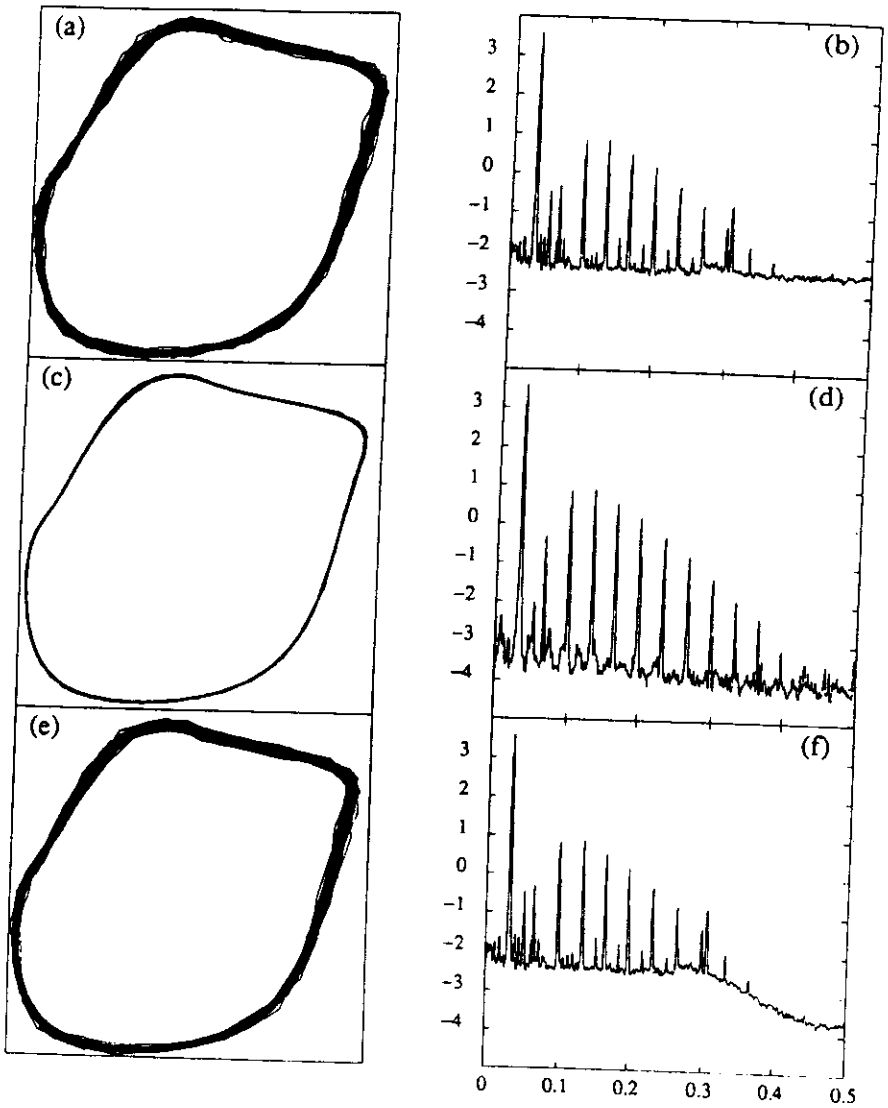


Fig. 4. Phase portraits and power spectra for measurements of wavy vortex flow in a Couette–Taylor experiment. (a), (b) Phase portrait and power spectrum before noise reduction is applied; (c), (d) after noise reduction; (e), (f) after a low-pass filter is applied to the original data. The vertical axis in (b), (d) and (f) is the base-10 logarithm of the power spectral density; the horizontal axis is in multiples of the Nyquist frequency.

These results are significantly different from those obtained by low-pass filtering. Figs. 4e, 4f show the phase portrait and power spectrum when the original data are passed through a 12th-order Butterworth filter with a cutoff frequency of 0.35. The dynamical noise reduction procedure is more effective than low-pass filtering since the noise appears to have a broad spectrum.

However, the dynamical noise reduction method appears to subtract power from a mode whose fundamental frequency is approximately 0.3 times the Nyquist frequency. We do not know exactly

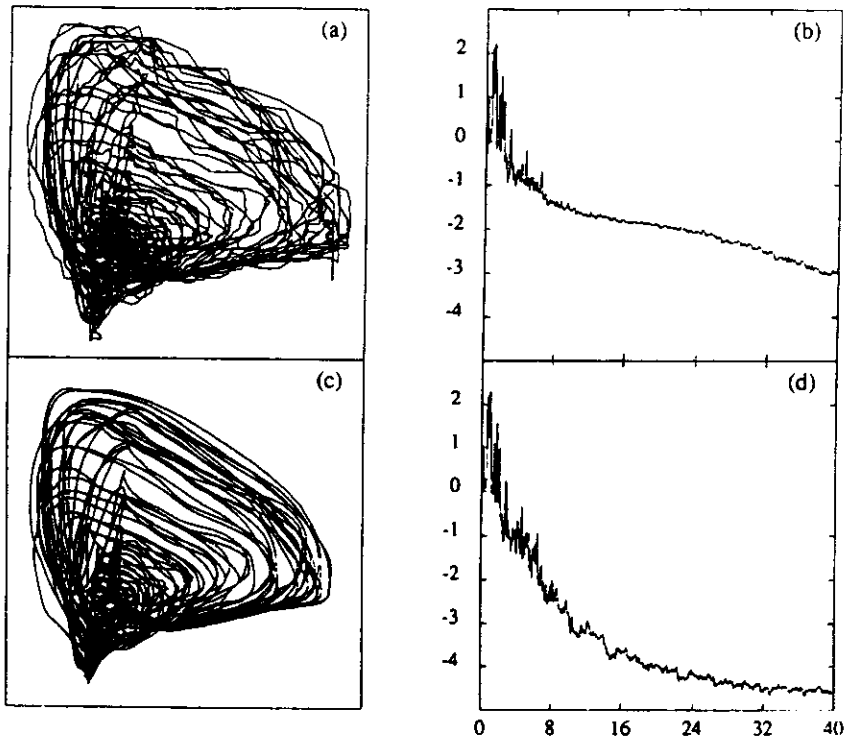


Fig. 5. Phase portraits and power spectra for measurements of weakly chaotic flow in a Couette–Taylor experiment. (a), (b) Phase portrait and power spectrum before noise reduction is applied; (c), (d) after noise reduction. The units for the power spectrum plots are the same as those in ref. [5].

why this occurs. However, this peak corresponds to the rotation frequency of the inner cylinder and may result from a defect in the Couette–Taylor apparatus [33]. We do not consider this to be a serious problem, because the power associated with this mode is several orders of magnitude smaller than that of the wavy vortex flow.

We emphasize that our objective is to find a simple dynamical system that is consistent with the data. It is possible for this method to eliminate certain dynamical behavior from an attractor if those dynamics have very small amplitude, as fig. 4f shows. This situation is most likely to arise when there are not enough data to distinguish such dynamics from random noise. In the present example, the noise reduction procedure reveals the limit cycle behavior quite well[*8].

The results obtained by applying the method to chaotic data from the Couette–Taylor fluid flow experiment described in ref. [5] are shown in fig. 5. Fig. 5a shows a two-dimensional phase portrait of the raw time series at a Reynolds number $R/R_c = 12.9$, which corresponds to weakly chaotic flow [5]. The corresponding phase portrait from the filtered time series is shown in fig. 5b. Figs. 5c, 5d show

[*8] We have not attempted to find the smallest amplitude at which the noise reduction procedure can distinguish quasiperiodic from periodic flow. In general this will depend on the amount of data, the sampling rate, the embedding dimension, and other factors.

the power spectra for the corresponding time series[*9].

It is difficult to estimate how much noise is removed from the data in this example on the basis of power spectra. One problem is that the transition from quasiperiodic to weakly chaotic fluid flow is marked by a sudden rise in the noise floor in the power spectrum (cf. fig. 3 in ref. [5]). Hence one cannot determine how much of the noise floor is due to deterministic chaos and how much results from broad-band noise. The noise reduction procedure described here has the effect of reducing the power in the high-frequency components of the signal. One question therefore is whether reducing the high-frequency noise corresponds to discovering the true dynamics which have been masked by noise. We believe that the answer is yes, based on those cases where there is an underlying low-dimensional dynamical system. However, in chaotic processes some high-frequency components remain, because they are appropriate to the dynamics.

7. Numerical experiments on noise reduction

One important question is how much noise this method removes from the data. The power spectra above suggest that the method eliminates most of the noise, but it is impossible to give a precise estimate for typical chaotic experimental data.

However, the Hénon map [19] provides a convenient way to quantify the noise reduction, because it can be written as a time delay map of the form

$$x_{i+1} = f(x_i, x_{i-1}) = 1 - \alpha x_i^2 + \beta x_{i-1}. \qquad (3)$$

We use eq. (3) to generate a time series as follows (with the standard parameter values $\alpha = 1.4$, $\beta = 0.3$). We choose an initial condition and discard the first 100 iterates. The next 32 768 iterates are stored, and a time series is generated by adding a uniformly distributed random number to each iterate. This simulates a time series with *measurement noise*, i.e., a time series where noise results from errors in measuring the signal, not from perturbations of the dynamics.

We measure the improvement in the signal after processing by considering the *pointwise error*

$$e_i = \|x_{i+1} - f(x_i, x_{i-1})\|,$$

i.e., the distance between the observed image and the predicted one. Let the *mean error* be

$$E = \left(\frac{\sum e_i^2}{N}\right)^{1/2},$$

the rms value of the pointwise error over all N points on the attractor. We define the *noise reduction* as

$$R = 1 - E_{\text{fitted}}/E_{\text{noisy}},$$

where the mean errors are computed for the adjusted and original noisy time series, respectively. The quantity R is a measure of the self-consistency of the time series. (In other words, R measures how much better on the average the output attractor obeys eq. (3) as one hops from point to point.)

When 1% noise is added to the input as described above, the noise reduction (measured with the actual map) is 79%[*10]. Nearly identical results are obtained when the input contains only 0.1% noise. In addition, noise levels can be reduced almost as much in cases where the noise is added to the dynamics, i.e., where the input is of the form $\{x_{i+1} : x_{i+1} = f(x_i + \eta_i, x_{i-1} + \eta_{i-1}),\ \eta_i, \eta_{i-1}$ random$\}$. When the program is run on noiseless input, the mean error in the output is 0.025% of the attractor extent, which suggests that errors

[*9] The time series consists of 32 768 values, from which an attractor is reconstructed in four dimensions. Linear maps are computed using 50–100 points in each ball. Trajectories are fitted using sequences of 24 points.

[*10] The pointwise error is measured using eq. (3). However, the attractor can be embedded in more than two dimensions when performing the noise reduction.

arising from small nonlinearities are negligible when the input contains enough points.

8. Simplicial approximations of dynamical systems

Recent work has shown that simplicial approximations of dynamical systems can reproduce the behavior of the original system to high accuracy [36]. (See also ref. [35] for a bilinear approach.) In particular, the fractal structure of the original attractors and basin boundaries is preserved over many scales. Such approximations can yield significant computational savings, especially when the original system consists of ordinary differential equations.

This approach can be extended in a natural way to generate simplicial approximations of the dynamics on attractors reconstructed from experimental data. Our objective here is to find an approximate dynamical system in a neighborhood of the attractor as follows.

A *simplex* in an m-dimensional space is a triangle with $m + 1$ vertices. Suppose the map is known at each point on a grid. Then there is a unique way to extend the map linearly to the interior of the simplex S whose vertices are grid points. Given a point P in the interior of S, let $\{b_i\}_{i=0}^{m}$ be its corresponding *barycentric coordinates* (see ref. [36] for an algorithm to compute them). Let $f(v_i)$ be the map at the ith vertex. The dynamical system at P is iterated by computing

$$\Phi(P) = \sum_{i=0}^{m} b_i f(v_i). \tag{4}$$

We apply this method to experimental data by finding a linear approximation of the dynamics at each vertex v_i with the least-squares method described above, using a collection of points in a small ball around v_i. The maps are stored and retrieved using a hashing algorithm similar to that described in ref. [36]. This yields a piecewise linear approximation of the dynamics from a set of experimental data which can be analyzed with the methods that previously were available only to theorists[*11].

We illustrate the approach using a time series of 32 768 values from the Hénon map with $\alpha = 1.2$, $\beta = 0.3$ using eq. (3) and adding 0.1% noise as described above. The original attractor is shown in fig. 6a. We take a grid of points which are spaced at 1% intervals (this and subsequent distances are expressed as a fraction of the original attractor extent). The time series is embedded in two dimensions, and a linear approximation of the dynamics is computed at each grid point for which 50 or more attractor points can be collected with a ball of radius 0.03; the set of such grid points is shown in fig. 6b. We take an initial condition near the original attractor and show the first 3000 iterates using eq. (4) in fig. 6c. Although some defects are visible, the attractor produced by the approximate dynamical system looks almost identical to the original one.

One application of simplicial approximations is the location of periodic saddles and the estimation of the largest eigenvalue of the corresponding Jacobian. That is, if x is a periodic point of period p, then we find the eigenvalue of $Df^p(x)$ of largest modulus, where $Df^p(x)$ refers to the matrix of partial derivatives of the pth iterate of the map f evaluated at x.

Given an initial guess for x, one can apply Newton's method using the maps computed at the grid points and eq. (4) to locate the saddle using the simplicial approximations. Likewise, eq. (3) can be used to locate the corresponding "exact" saddle. Saddle orbits up to period 8 have been computed in this way. In all cases, the saddle point for the simplicial approximation is within 2% of the corresponding saddle point for the Hénon map. Table 1 shows the largest eigenvalues of the saddle orbits. (The columns labeled $m = 2$ and $m = 3$ refer to the embedding dimension used to reconstruct the attractor.) In most cases, the

[*11] This approach is less ambitious than that of Crutchfield [8], who attempts to find a single set of nonlinear difference equations that creates the observed attractor.

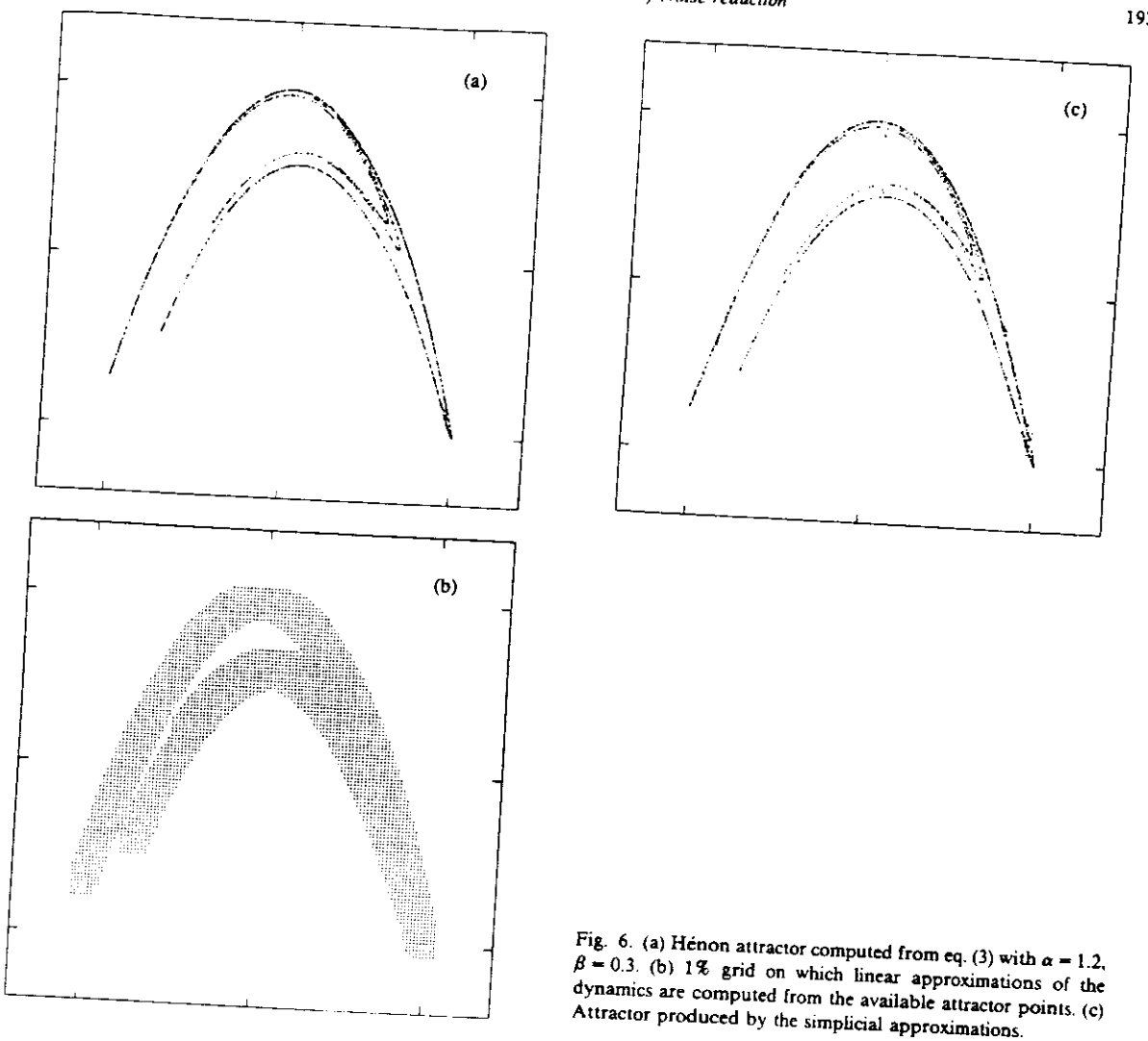


Fig. 6. (a) Hénon attractor computed from eq. (3) with $\alpha = 1.2$, $\beta = 0.3$. (b) 1% grid on which linear approximations of the dynamics are computed from the available attractor points. (c) Attractor produced by the simplicial approximations.

lative error is only a few percent, and in no case ceeds 25%. (The largest relative error is for the riod 8 saddles, where one finds the eigenvalue of : product of 8 Jacobians computed from the st squares.)

This method can be extended to experimental :a sets. However, there are relatively stringent uirements on the data that can be handled: the e series must be long enough to trace out many ectories near the principal unstable saddle or-, and the noise level must be low. (Presumably, sy data can be preprocessed using the approach described in section 4.) The current computer implementation uses a large amount of disk space to store the linear map approximations at the grid points.

We have constructed a simplicial approximation for an attractor obtained from a Belousov–Zhabotinskii chemical reaction [7, 30]. The attractor is reconstructed in three dimensions from a set of 32 768 measurements of bromide ion concentration. The phase portrait is shown in fig. 7a.

Linear approximations of the dynamics are computed at each point of a grid consisting of 50

Table 1
The largest eigenvalues of the Jacobian of the periodic orbits located using the simplicial approximation of the Hénon attractor.

Period	$m = 2$	Exact	$m = 3$
1	1.793	1.695	1.757
2	2.178	2.199	2.183
4	4.226	4.329	4.051
6	10.38	10.70	9.626
6	10.38	11.32	12.12
8	25.80	24.88	30.25
8	20.02	20.60	20.38
8	17.70	24.32	21.70

intervals along each coordinate axis for which 50 or more attractor points can be located within an 8% radius of the grid point. This produces a database of 59 550 maps. We observe from graphical evidence that many trajectories approach what appears to be a period-3 saddle in the middle of the attractor. Using initial guesses from some of the trajectories, we apply Newton's method to locate the saddle orbit shown in fig. 7b. Moreover, we obtain estimates of the Jacobian Df of the map evaluated at a point on the saddle orbit. The eigenvalues of Df are estimated as $\lambda_1 = 1.14$, $\lambda_2 = 0.102$, and $\lambda_3 = -1.53$. These quantitative results confirm that the orbit is a saddle since $\lambda_1 > 0 > \lambda_3$. (Note that one expects $\lambda_2 = 0$ for a flow generated from a set of differential equations.)

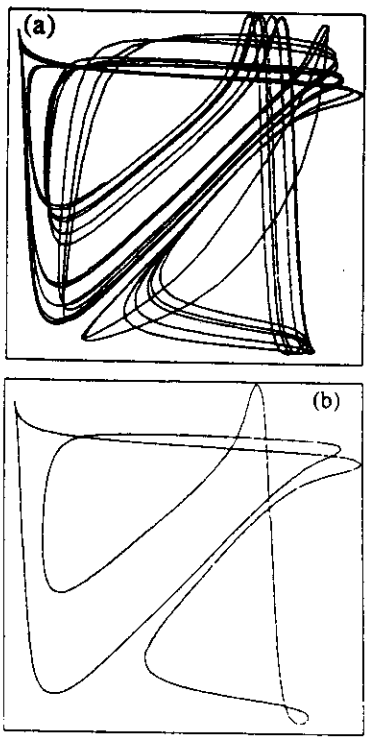


Fig. 7. (a) The attractor reconstructed from a time series of bromide ion concentrations in a Belousov–Zhabotinskii chemical reaction. (b) The period-3 saddle orbit.

9. Conclusion

Methods for approximating the dynamics of attractors reconstructed from experimental data provide powerful tools. Most of the same procedures that have been so important for theoretical insight, such as Poincaré maps, unstable fixed points and their manifolds, basin boundaries, and the like, are now available to experimenters, at least in cases where the dynamics are low dimensional. There is little doubt that these tools will lead to breakthroughs in the understanding of a wide variety of physical systems. However, considerable effort is needed before we learn which kinds of systems will benefit most from these types of analyses. Significant improvements in technique will certainly extend the applicability of dynamical embedding methods, for example to higher-dimensional attractors.

Appendix

In this appendix we outline a possible alternative noise reduction method based on the theory of least squares when all the quantities in the regression are measured with error.

In ordinary least squares, the variables in the problem fall into two classes: the *independent* variables, which are known exactly, and the *dependent* variables, which are observations assumed to be functions of the independent variables. The dependent variables are subject to random errors that are assumed independent and identically distributed (i.i.d.).

On an attractor reconstructed from experimental data, we assume that the mapping which takes points in a sufficiently small ball to their images is approximately linear. However, the locations of all the points are subject to small random errors because of the noise. Hence one cannot describe the points as independent variables and their images as dependent variables. The usual least-squares method produces a biased estimate of the linear map, and this bias does not decrease if more observations are added [16, 23].

The so-called "errors in variables" least-squares methods can be used to handle the latter problem. This approach can be used to obtain both an estimate of the linear map as well as estimates of the "true" values of each of the observations.

At first this appears to be an underdetermined problem: from n pairs of observations one wants to compute the parameters of the functional relation between them as well as estimates of the n actual pairs[*12]. However, it is possible to solve this problem by making some assumptions about the errors [16, 23].

In our case, we *assume* that the errors in the location of each point and its image are i.i.d. In particular, we let the covariance matrix of the errors in the variables be the identity matrix. This assumption is valid whenever the noise is independent of the dynamics[*13].

We illustrate the procedure for the case where we are given a collection of n points (in $\mathbf{R}^m$) and their images. Following Jefferys [21], we form a set of n equations of condition given by

$$f_i(x_i) = x_{n+i} - Ax_i - b_i \equiv x_{n+i} - L(x_i), \qquad (5)$$

where x_i is the ith point, x_{n+i} is its observed image, A is an $m \times m$ matrix, and b is an m-vector. The goal is to find estimates of L (i.e., A and b), together with perturbations $\hat{v}$, such that

$$f_i(x_i + \hat{v}_i) = (x_{n+1} + \hat{v}_{n+i}) - L(x_i + \hat{v}_i) = 0$$

and such that the quadratic form

$$S_0 = \tfrac{1}{2}\hat{v}^t\sigma^{-1}\hat{v} \qquad (6)$$

is minimized. The superscript t denotes transpose and σ is the covariance matrix of the observations (which we assume is the identity matrix here).

This minimization problem can be solved using Lagrange multipliers (see refs. [21, 22] for a numerical algorithm). The solution gives A and b together with estimates $x_i + \hat{v}_i$ of the "true" observations. It can be shown [16] under fairly mild hypotheses that the estimates of L and the observations are the best in the class of linear estimators.

One way to approach noise reduction is to extend eq. (5) to include several iterations of the observed points. Given a collection of points in a ball, together with the next p iterates of each point, the method above is used to find a collection of linear maps $L_1, L_2, \ldots, L_p$ approximating the dynamics. The method also finds estimates of the actual observations. In this approach, therefore, the calculation of the maps and the adjustment of the trajectories is done in one step. Moreover, each point and its image exactly satisfy a linear relationship.

Of course, p cannot be too large, because nonlinear effects eventually will become significant when the dynamics are chaotic. On the other hand, eq. (5) provides a natural way to include quadratic or other nonlinear terms.

We have written a computer program to implement this alternative noise reduction algorithm. So far, the results of this approach have not been as good as those from the method described in the main part of the paper, but further refinement should improve them.

[*12] In the statistical literature, the problem is said to be *underidentified*.

[*13] Dynamical noise (i.e., each point is perturbed slightly before iterating) yields a covariance matrix which depends on the point. However, as long as the dynamical noise is small, our assumptions about the covariance matrix of the errors should not compromise the accuracy of the method.

Acknowledgements

Dan Lathrop provided invaluable assistance in finding periodic orbits in the Hénon and BZ attractors. We thank Bill Jefferys for useful discus-

sions and computer software for the errors in variables least-squares problem. Andy Fraser, Randy Tagg and Harry Swinney all provided helpful suggestions. This research is supported by the Applied and Computational Mathematics Program of the Defense Advanced Research Projects Agency (DARPA-ACMP) and by the Department of Energy Office of Basic Energy Sciences.

References

[1] R. Badii, G. Broggi, D. Derighetti, M. Ravani, S. Ciliberto and A. Politi, Phys. Rev. Lett. 60 (1988) 979.
[2] R.P. Behringer and G. Ahlers, J. Fluid Mech. 125 (1982) 219;
G. Ahlers and R.P. Behringer, Phys. Rev. Lett. 40 (1978) 712.
[3] J.L. Bentley and J.H. Friedman, ACM Comput. Surv. 11 (1979) 397.
[4] S.M. Bozic, Digital and Kalman Filtering (Edward Arnold Publishers Ltd., London, 1979).
[5] A. Brandstäter and H.L. Swinney, Phys. Rev. A 35 (1987) 2207.
[6] M. Casdagli, Nonlinear prediction of chaotic time series, preprint (December 1987).
[7] K.C. Coffman, Ph.D. Thesis, University of Texas at Austin (1987).
[8] J.P. Crutchfield and B. McNamara, Complex Systems 1 (1987) 417;
H.D. Abarbanel, R. Brown and J.B. Kadtke, Prediction and system identification in chaotic nonlinear systems: time series with broadband spectra, preprint (January 1989).
[9] J.J. Dongarra, C.B. Moler, J.R. Bunch and G.W. Stewart, LIN-PACK User's Guide (Society for Industrial and Applied Mathematics, Philadelphia, 1979).
[10] J.-P. Eckmann, S.O. Kamphorst, D. Ruelle and S. Ciliberto, Phys. Rev. A 34 (1986) 4971.
[11] J.-P. Eckmann and D. Ruelle, Rev. Mod. Phys. 57 (1985) 617.
[12] J.D. Farmer and J.J. Sidorowich, Phys. Rev. Lett. 59 (1987) 845.
[13] P.R. Fenstermacher, H.L. Swinney and J.P. Gollub, J. Fluid Mech. 94 (1979) 103.
[14] W. Franceschini and L. Russo, J. Stat. Phys. 25 (1981) 757.
[15] A. Fraser and H.L. Swinney, Phys. Rev. A 34 (1986) 1134.
[16] W.A. Fuller, Measurement Error Models (Wiley, New York, 1987).
[17] E.G. Gwinn and R.M. Westervelt, Phys. Rev. A 33 (1986) 4143.
[18] S.M. Hammel, J.A. Yorke and C. Grebogi, J. Complexity 3 (1987) 136; Bull. Am. Math. Soc. 19 (1988) 465.
[19] M. Hénon, Comm. Math. Phys. 50 (1976) 69.
[20] D. Hirst, Ph.D. Dissertation, University of Texas (December 1987).
[21] W.H. Jefferys, Astron. J. 85 (1980) 177.
[22] W.H. Jefferys, Astron. J. 86 (1981) 149.
[23] M.G. Kendall and A. Stuart, The Advanced Theory of Statistics, Vol. 2 (Griffin, London, 1961) p. 375.
[24] A. Libchaber, S. Fauve and C. Laroche, Physica D 7 (1983) 73.
[25] E.N. Lorenz, J. Atmos. Sci. 20 (1963) 130.
[26] S.W. MacDonald, C. Grebogi, E. Ott and J.A. Yorke, Physica D 17 (1985) 125.
[27] G. Mayer-Kress, ed., Dimensions and Entropies in Chaotic Systems (Springer, Berlin, 1986), and references therein.
[28] F. Mitschke, M. Möller and W. Lange, Phys. Rev. A 37 (1988) 4518.
[29] L.R. Rabiner and B. Gold, Theory and Application of Digital Signal Processing (Prentice-Hall, Englewood Cliffs, NJ, 1975).
[30] J.-C. Roux, Physica D 7 (1983) 57;
J.-C. Roux, R.H. Simoyi and H.L. Swinney, Physica D 8 (1983) 257.
[31] M. Sano and Y. Sawada, Phys. Rev. Lett. 55 (1985) 1082.
[32] C. Sparrow, The Lorenz Equations: Bifurcations, Chaos, and Strange Attractors (Springer, New York, 1982).
[33] R. Tagg, private communication.
[34] F. Takens, in: Dynamical Systems and Turbulence, Springer Lecture Notes in Mathematics, Vol. 898, eds. D.A. Rand and L.-S. Young (Springer, Berlin, 1980) p. 366.
[35] B.H. Tongue, Physica D 28 (1987) 401.
[36] F. Varosi, C. Grebogi and J.A. Yorke, Phys. Lett. A 124 (1987) 59.
[37] A. Wolf, J.B. Swift, H.L. Swinney and J.A. Vastano, Physica D 16 (1985) 285.

PHYSICAL REVIEW E VOLUME 48, NUMBER 3 SEPTEMBER 1993

Noise reduction in chaotic time-series data: A survey of common methods

Eric J. Kostelich
Department of Mathematics, Box 871804, Arizona State University, Tempe, Arizona 85287

Thomas Schreiber*
Niels Bohr Institute, Blegdamsvej 17, DK-2100 Copenhagen Ø, Denmark
(Received 23 April 1993)

This paper surveys some of the methods that have been suggested for reducing noise in time-series data whose underlying dynamical behavior can be characterized as low-dimensional chaos. Although the procedures differ in details, all of them must solve three basic problems: how to reconstruct an attractor from the data, how to approximate the dynamics in various regions on the attractor, and how to adjust the observations to satisfy better the approximations to the dynamics. All current noise-reduction methods have similar limitations, but the basic problems are reasonably well understood. The methods are an important tool in the experimentalist's repertoire for data analysis. In our view, they should be used more widely, particularly in studies of attractor dimension, Lyapunov exponents, prediction, and control.

PACS number(s): 05.45.+b

I. INTRODUCTION

Noise limits one's ability to extract quantitative information from time-varying signals. In many cases, one is interested in a time series of data produced by a system whose underlying behavior can be characterized as low-dimensional chaos. Two important measures associated with the dynamics are the dimension of the attractor (which describes approximately the number of degrees of freedom) and the Lyapunov exponents (which quantify the sensitivity of the process to initial conditions).

There is much literature on physical systems that exhibit chaotic behavior and on methods for estimating the dimension of an attractor reconstructed from time-series data (see, for instance, the papers in [1,2]). Algorithms for estimating Lyapunov exponents have been described by Wolf *et al.* [3] and by Eckmann *et al.* [4], among others.

Both the dimension and the Lyapunov exponents describe a kind of scaling behavior in the limit as the distances between points on the attractor approach zero [5]. Each of them is sensitive to the presence of small amounts of noise. Simple numerical experiments show that a noise level of 1% of the time-series extent makes it impossible to measure the correlation dimension of the attractor using distances less than 3% of the attractor extent [6]. Noise in laboratory data may completely obscure the underlying fractal structure unless the data are preprocessed to reduce the noise [7].

There are several aspects to the noise-reduction problem. One question concerns the circumstances under which low-pass filtering and similar methods are justified for the analysis of chaotic laboratory data. Another question involves ways in which the dynamics can be used to identify and correct errors in the observations arising from noise, at least in cases where the behavior of the process can be characterized as low-dimensional chaos. Different approaches have been suggested to the latter question. Examples include Kostelich and Yorke [6,8], Schreiber and Grassberger [9], Sauer [10], Cawley and Hsu [11], and Farmer and Sidorowich [12].

We do not want to attempt a quantitative comparison of these noise-reduction methods. Instead, it is our opinion that the methods have important similarities—indeed, one can view them as variations on a common theme. All of the methods must address three basic problems: (1) they must have a strategy for reconstructing the underlying attractor from the observed time series; (2) they must estimate the local dynamical behavior by choosing a class of models and fitting the parameters statistically; and (3) they must adjust the observations to make them more consistent with the models. For this reason, all of the methods have similar advantages and shortcomings. Nevertheless, the various methods perform well in many cases. In our opinion, experimentalists and others should take advantage of these ideas in their analysis of data.

In the next section, we summarize briefly one popular method for reconstructing an attractor from a time series of data and show how noise complicates the problem of computing the dimension of the attractor. We discuss improved embedding theorems in Sec. III and show how the effects of noise can be reduced with better embedding methods. In Sec. IV, we discuss how to approximate the dynamics using the data. We show how the observations can be adjusted to reduce noise in Sec. V and consider error measures in Sec. VI.

II. THE EFFECT OF NOISE: AN EXAMPLE USING TIME-DELAY EMBEDDINGS

The first step in any analysis of chaotic data is to reconstruct an attractor from the data. This is the *em-*

1063-651X/93/48(3)/1752(12)/$06.00

bedding problem, which can be summarized briefly as follows. Suppose one has an apparatus whose behavior can be described in principle by a set of differential equations. There might be a large number of equations, and one might not know exactly what they are. (For example, the Belousov-Zhabotinskii chemical reaction exhibits complex dynamics and more than 30 intermediate chemical species have been identified in it. However, many of the associated rate constants are not known [13,14].) After a time, the system reaches an asymptotic state in which the behavior is governed by the attractor A for the underlying set of equations. The experimental data are functions of points on A. The embedding problem is the following: given a time series of measurements $\{s_i\}$, how can a set be reconstructed that is equivalent in an appropriate mathematical sense to A? In particular, there should be a one-to-one correspondence between points on A and points on the reconstructed attractor, and information about the derivatives of the flow should be preserved.

The *Takens time-delay-embedding method* [15] is probably the most common attractor reconstruction method in the literature. (We discuss other strategies in Sec. III.) Given the time series $\{s_i\}$, the reconstructed attractor consists of the m vectors

$$\mathbf{x}_i = (s_i, s_{i+\tau}, s_{i+2\tau}, \ldots, s_{i+(m-1)\tau}) ,$$

where τ is the *time delay* and m is the *embedding dimension*. Takens showed that if the embedding dimension is sufficiently large, then the set $\{\mathbf{x}_i\}$ has the same dimension and Lyapunov exponents as the original attractor for "most" choices of the time delay [15]. (We give a more precise statement below.)

One of the oldest applications of time delay embeddings is to the estimation of the dimension of the attractor. There are many notions of dimension; see [7] and the references therein for more details. In some sense, the dimension describes the number of degrees of freedom associated with the dynamics.

One popular definition is the *correlation dimension*, introduced by Grassberger and Procaccia [16]. Suppose the reconstructed attractor consists of N points. Let

$$C(\epsilon) = \frac{2P(\epsilon, N)}{N(N-1)} , \qquad (1)$$

where $P(\epsilon, N)$ is the number of pairs of points on the attractor whose distance apart is less than ϵ. Then $C(\epsilon)$ is a number between 0 and 1. In the limit as $\epsilon \to 0$ and $N \to \infty$, and in the absence of noise, we have $C(\epsilon) \sim \epsilon^d$, where d is the *correlation dimension* of the attractor.

In practice, the finite number of data points and noise in the data place a lower bound on the values of ϵ that can be used to estimate d. Typically, one computes $C(\epsilon)$ for each ϵ in a decreasing sequence. The dimension d is estimated by using linear regression over a range of $(\ln \epsilon, \ln C(\epsilon))$ pairs. One can compare the scaling relations over various ranges by performing the linear regression over different subsets of the ϵ's and looking for consistent values of d.

Ideally, the values for d should be relatively constant for a large interval of ϵ values. For example, if $C(\epsilon)$ has been computed for a decreasing sequence of 30 values of ϵ, one might perform the linear regression over the largest five ϵ's to get an estimate $\hat{d}(\epsilon_1, \ldots, \epsilon_5)$ of the dimension. One could repeat the calculation using the second largest five ϵ's to get $\hat{d}(\epsilon_2, \ldots, \epsilon_6)$, and so on. In this way, one would get estimates of the derivative $d[\ln C(\epsilon)]/d(\ln \epsilon)$.

In a good experimental data set, one might find reasonably constant values of $\hat{d}$ for ϵ ranging from 10% down to 1% or less of the attractor extent. However, a "plateau" in the values of d often is not apparent. The values of $\hat{d}$ may yield a plot that is qualitatively similar to Fig. 1.

In their analysis of weak turbulence in a Couette-Taylor fluid flow experiment, Brandstater and Swinney [17] tried to estimate the correlation dimension over various ranges of ϵ and divided their plot into four regions labeled A–D, as shown in Fig. 1. In region A the lack of data points is the dominant feature. Therefore, the values of $\hat{d}$ are subject to large fluctuations and cannot be used to estimate the dimension reliably. The behavior in region B is attributed to the noise, which "smears out" the fractal structure below a certain level of resolution. One wants to see the behavior in region C for as large a range of ϵ values as possible. The behavior in region D reflects the lack of scale invariance since ϵ is of the order of the size of the entire attractor.

The data in Fig. 1 consist of a time series of 40 000 values from a laser experiment by Flepp and co-workers [18]. The curves illustrate attempts to estimate the correlation dimension of the data before and after a nonlinear noise-reduction method was applied. The region labeled C on each curve corresponds to an approximate power-law relationship between the correlation $C(\epsilon)$ defined in Eq. (1) and the ball size ϵ. Before noise reduction (represented by diamonds), a scaling region is difficult to discern, because the noise obscures the fine scale structure up to 1/16 of the attractor extent. A wide range of estimates of the attractor dimension are possible with the noisy data, depending on the range of ϵ used. In this example, it is difficult simply to determine whether the process is low dimensional using only the noisy data. Hence, even small levels of noise significantly complicate esti-

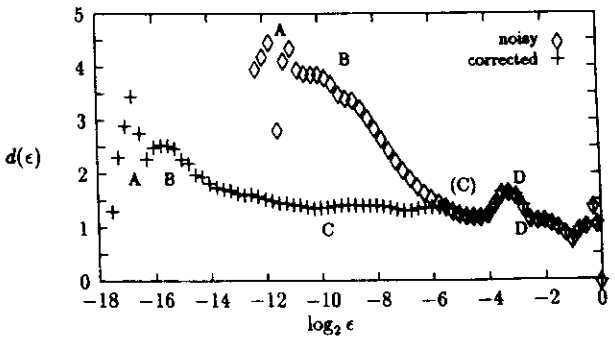


FIG. 1. A typical plot of correlation dimension as a function of the range of distances ϵ used to estimate the scaling exponent. Noise obscures the fine scale structure up to 1/16 of the attractor extent, making an accurate estimate of the attractor dimension impossible.

mates of dimension, a quantity that in principle should be straightforward to measure.

A similar situation arises in the estimation of Lyapunov exponents. In his numerical study of the Lorenz equations, Abarbanel [19] found that the negative Lyapunov exponent cannot be determined when the noise level is as small as 10^{-4}, and none of the exponents can be determined with satisfactory precision in the presence of 1% noise. This is true even if the minimum required embedding dimension is known. (Higher embedding dimensions lead to the additional problem of spurious negative exponents.)

Although dimension estimation provides an illustrative example, there are many other applications in which noise is a serious concern. The ability to calculate accurate approximations to the dynamics allows one to make short-term predictions of the process [20–23] or to determine small changes to an accessible system parameter in order to stabilize the process around a saddle periodic orbit. (The so-called *control of chaos* is a subject of considerable recent interest [24].) Obviously, noise limits one's ability to estimate the dynamics, and a good noise-reduction procedure is essential in such applications.

III. NOISE REDUCTION AND EMBEDDING TECHNIQUES

A. Generalizations of the Takens embedding theorem

The Takens time-delay-embedding method discussed in the preceding section is one of several approaches to the problem of attractor reconstruction. A fundamental objective is to find a reconstruction of the attractor that minimizes the effects of noise in the input time signal.

Casdagli *et al.* [25] consider notions of distortion and noise amplification in attractor reconstruction. Because of the noise, each point in an m-dimensional time-delay reconstruction of the attractor can be regarded as a random m vector. The mean of the underlying density is a nearby "true" vector if the noise has mean zero and is uncorrelated with the signal. Their objective is to try to choose the time delay in such a way that the variance of the random observations around the true state does not have unacceptably large components along regions of interest on the attractor.

Sauer, Yorke, and Casdagli [26] have proved important generalizations of the Takens time-delay-embedding theorem which are especially relevant to the analysis of experimental data. Their work justifies the use of filtering in the reconstruction of attractors to reduce the effect of noise. In the rest of this section, we summarize their main results and related work.

An embedding is a reconstruction for which there is a one-to-one mapping to the original attractor that preserves information about the derivatives of the original flow. This implies that an embedding produces a reconstructed set with the same dimension and Lyapunov exponents as the original attractor.

The Takens embedding theorem asserts that a one-to-one mapping exists for a generic set of time delays, provided that the embedding is done in enough dimensions. How many are necessary?

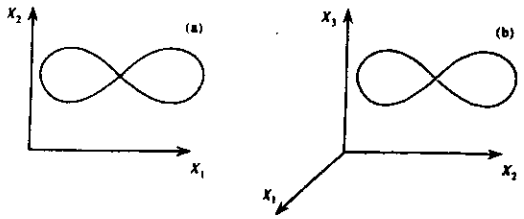


FIG. 2. (a) No small perturbation of the reconstruction function eliminates the self-intersection in the limit cycle. (b) Although the limit cycle intersects itself, arbitrarily small perturbations will remove the self-intersection.

Suppose that the input time series is periodic (a sine wave, for instance). Dynamically this corresponds to a limit cycle, which can be embedded in two dimensions. However, in a two-dimensional Takens time-delay reconstruction, the measurement function may produce a self-intersecting limit cycle, as shown in Fig. 2(a). No small perturbation of the measurement function removes the self-intersection. It is still possible for the curve to intersect itself in three dimensions, but there is a set of arbitrarily small perturbations to the measurement function that produce a one-to-one mapping between the underlying attractor and its reconstruction.

This example provides a heuristic motivation for the rigorous result proved by Sauer, Yorke, and Casdagli [26]: if the box-counting dimension [27] of the attractor is d, then m dimensions suffice to produce an embedding, where $m > 2d$. For a limit cycle ($d = 1$), three dimensions are sufficient. In fact, almost every reconstruction in three dimensions avoids a self-intersection of the sort depicted in Fig. 2(b). This notion of probability one can be made precise in a way that will not be discussed here (see [26] for details). Moreover, typical choices of time delay are likely to produce an embedding. (In the example of a limit cycle, one has an embedding unless the time delay is an integer multiple of the period of the signal.)

An embedding preserves characteristic quantities such as the dimension. This does not mean that reliable estimates of these numbers can be extracted from a given data set because different reconstructions are not equally sensitive to noise and sparse data.

B. Noise reduction with filtered embeddings

The Takens time-delay-embedding method, although popular, is not the only way to reconstruct an attractor. Suitable linear combinations of input time series values can reduce the noise in the reconstructed attractor. Let

$$\mathbf{x}_i = B \begin{pmatrix} s_i \\ s_{i+1} \\ \vdots \\ s_{i+w-1} \end{pmatrix} \tag{2}$$

where B is an $m \times w$ matrix. This sequence of vectors

produces an embedding if the rank of B is sufficiently large and if B does not collapse periodic points of the underlying attractor of integral period less than or equal to w [26]. For example, B can be chosen to yield a Fourier embedding (*e.g.*, $w = 64$ and $m = 16$, so that each point on the reconstructed attractor is a suitably truncated fast Fourier transform of a window of 64 time-series values). Sauer [10] has shown that this "low-pass" embedding can be used to preprocess data with 100% additive Gaussian noise.

Broomhead and King [28] have suggested the use of singular value analysis to reconstruct the attractor. (This is an alternative choice for B.) One can choose a window of the original time series and project it onto the subspace spanned by the corresponding singular vectors. Heuristically, each window of the time series is projected onto the subspace that contains the largest fraction of the total variance of the data. The remaining orthogonal directions presumably contain most of the noise. (See [28] for details.) Albano *et al.* [29] used a similar procedure to reduce the noise and estimate the Grassberger-Procaccia dimension of the attractor reconstructed from time-series data. Other applications of singular value decomposition to noise reduction are described in later sections.

Another alternative is to use a Takens time-delay reconstruction after applying a suitable filter to the entire time series. For example, one can compute the power spectrum of the input signal, make some assumptions about the frequency distribution of the noise, remove or suppress those frequency components, and invert the transform. This is the idea behind traditional bandpass filters (see [30] for a concise description of the Wiener filter).

Traditional filters work well in situations where most of the noise is restricted to certain frequencies. However, the time series of many chaotic signals have broadband components [31,17]. Thus bandpass filters will cause distortion because some of the suppressed frequency components are part of the dynamics. If it is not done carefully, bandpass filtering can fail to give an embedding. If *infinite input response* filters are used, the dimension of the reconstructed set may deviate considerably from that of the original attractor. See [32] for details.

Despite these caveats, we recommend the use of these preprocessing methods because they can reduce the noise significantly. The numerical algorithms are widely available and easily implemented, even on small computers. It is often desirable to try different attractor reconstructions from the same data because the methods distort regions of interest on the attractor in different ways. Depending on the application, one reconstruction may be better than another.

C. Multiple-probe measurements

Multiple redundant measurements can be exploited to reduce noise. For example, several measurements can be taken during the fundamental period of a driven system; several devices can be operated simultaneously (e.g., velocity data can be taken at several nearby locations in a flow) [33] or the data can be taken at time intervals that are short compared to the correlation time of the signal (oversampling).

Suppose there are L probes and at time i we record $s_i^{(1)}, \ldots, s_i^{(L)}$. Each of the L time series contains the same dynamical information and comparable amounts of noise (we assume the measurement errors are independent). The series with the largest variance has the smallest *relative* noise level (highest signal-to-noise ratio). Following the above considerations about embeddings, almost every nonsingular linear combination $V_i = \sum_{k=1}^{L} a_k s_i^{(k)}$ contains the same information about the dynamics. (One need not restrict attention to linear combinations of the time series for attractor reconstruction, but they are convenient.)

A useful strategy is to pick the linear combination which has the highest signal-to-noise ratio. It can be found by maximizing the variance of the V_i subject to the constraint $\sum_{k=1}^{L} a_k^2 = 1$. The requisite vector is the eigenvector corresponding to the largest eigenvalue of the $L \times L$ covariance matrix Γ whose (k,l)th entry is $\Gamma_{kl} = \langle s^{(k)} s^{(l)} \rangle - \langle s^{(k)} \rangle \langle s^{(l)} \rangle$, where the angular brackets denote the average value over all time steps i. This procedure is similar to the singular value decomposition and is also called principal component analysis [34].

IV. MODELING DYNAMICS FROM DATA

The embedding methods described in the preceding section are useful preprocessing tools. However, they are linear methods applied to portions of the time series that are restricted in either the time or frequency domain. They do not exploit the underlying dynamical behavior to identify and correct errors in the observations.

The objective of every noise-reduction method is to find a simpler dynamical system that is consistent with the data. There are three aspects to the problem: (1) the nature of the noise, (2) the class of models to be fitted using a statistical method, and (3) the adjustment of the observations to be more consistent with the model. We discuss these in turn in the following subsections.

A. Measurement error and dynamical noise

Measurement noise refers to the corruption of observations by errors which are independent of the dynamics. The dynamics satisfy $\mathbf{x}_i = \mathbf{f}(\mathbf{x}_{i-1})$, but we measure $\mathbf{x}_i + \eta_i$. [More typically, the measurements consist of scalars $s_i = h(\mathbf{x}_i) + \delta_i$, where h is a smooth function that maps points on the attractor to real numbers and the δ_i are independent and identically distributed random variables.] *Dynamical noise*, in contrast, is a feedback process wherein the system is perturbed by a small random amount at each time step:

$$\mathbf{x}_i = \mathbf{f}(\mathbf{x}_{i-1} + \eta_{i-1}). \tag{3}$$

Dynamical and measurement noise are two notions of the error that may not be distinguishable *a posteriori*

based on the data only. Both descriptions can be consistent to some extent with the same signal. The *shadowing* problem addresses the question of whether there is a trajectory near the observed one, possibly for a different nearby initial condition or for a slightly different map.

Suppose the system contains dynamical noise and that there is a map $\tilde{\mathbf{f}}$ near $\mathbf{f}$ in some metric for which $\mathbf{x}_i = \tilde{\mathbf{f}}(\mathbf{x}_{i-1})$, $i = 1, \ldots, n$. Then the observations can be considered as the exact output of the unknown but slightly different map $\tilde{\mathbf{f}}$, or they can be considered as the noisy output from the known map $\mathbf{f}$. Often one of the two points of view permits a simpler description than the other. For instance, if we detect correlations between δ_i and $\mathbf{x}_{i+\tau}$ for some τ, then we can reject the hypothesis that the noise arises from independent measurement errors. Provided that the series does not contain *exactly* the same phase-space point twice, we can construct a smooth function that interpolates between all measured points; relative to this function, the data form a noise-free orbit.

Strictly speaking, it is not necessary to think of this problem only in terms of separating a deterministic signal from some random fluctuations. It is possible that the "noise" might arise from a high-dimensional, deterministic dynamical system. The noise-reduction problem therefore is a question of how to separate the low-dimensional dynamics from a complex signal. This requires one to choose a class of models for the dynamics and to fit a model to the data in the regions of interest on the attractor. The following sections describe some approaches to this problem.

B. Time ordering and estimation of the dynamics

The overall objective of noise-reduction methods is to find a simple dynamical system that is consistent with the data. Therefore, one must find an approximation to the dynamics from a class of models and adjust the observations to satisfy the dynamical approximations better. Although these two steps are not independent of each other, we will discuss them separately, pointing out along the way where they overlap.

A basic problem is how to exploit the time ordering of the data to produce a more self-consistent trajectory and reduce the noise. Suppose that m dimensions suffice to reconstruct the attractor. In principle, one can write

$$s_{m+1} = f(s_1, \ldots, s_m) + \eta_{m+1} \tag{4}$$

where η_{m+1} denotes a noise term and the s_i are the time series values (possibly preprocessed using the methods outlined above). The subscripts denote the time ordering of the data; i.e., s_{i+1} is the observation immediately following s_i. We call Eq. (4) a *forward-in-time* representation of the dynamics. The prediction problem consists of finding an approximation $\hat{f}$ of f to get an estimate $\hat{s}_{m+1}$ of s_{m+1}, viz. $\hat{s}_{m+1} = \hat{f}(s_1, \ldots, s_m)$.

Typically, an ensemble of nearby trajectories is used to find $\hat{f}$ from a class of models. In some sense, $\hat{s}_{m+1}$ can be regarded as a "maximum likelihood" estimate of s_{m+1}.

This produces a naive (but unsatisfactory) scheme for noise reduction: start with the first m time-series values. Output a prediction $\hat{s}_{m+1}$ for s_{m+1}. Next, use the original observations $s_2, \ldots, s_{m+1}$ to determine $\hat{s}_{m+2}$, and so on. This scheme adjusts all but the first m time-series values.

Can this process be iterated to reduce the noise? That is, if we take the output from the first run and use it as the input to a second run, will the resulting time series be less noisy? The answer is *no* if the underlying time series is chaotic, because errors in $s_1, \ldots, s_m$ typically are amplified due to the sensitivity to initial conditions. For the same reason, the output time series drifts away from the original. (The same is true if we write s_i as a function of the succeeding m observations and work backwards in time.)

One approach which largely avoids this problem was suggested by Schreiber and Grassberger [9]. We review it here to outline the main ideas; a refinement of their procedure is described in a later section. The objective of the procedure is to use past and future values to adjust one or more observations in the middle. Let

$$0 = F(s_1, \ldots, s_m, s_{m+1}) + \eta_{m+1} \tag{5}$$

express the functional dependence of past and future values up to a noise term. If m dimensions suffice to embed the attractor, then any one of the observations in Eq. (5) is an implicit function of the others. We call Eq. (5) an *implicit* representation of the dynamics. Schreiber and Grassberger [9] use a linear approximation for F to find the least-squares estimate

$$\hat{s}_{m/2} = \sum_{\substack{k=1 \\ (k \neq m/2)}}^{m+1} a_k s_k + b \tag{6}$$

for the value in the middle of the sequence. This method uses information from both the future and the past to adjust the observations.

The map is determined in the following way. Given $s_1, \ldots, s_{m+1}$, one locates several closely matching sequences of $m+1$ observations. The middle value in each sequence is expressed as a linear combination of the others as in Eq. (6). The coefficients (except for $a_{m/2}$) are determined from a least-squares fit. Notice that the coefficient $a_{m/2}$ must be excluded to prevent a trivial fit (wherein $a_k = 0$ for $k \neq m/2$ and $a_{m/2} = 1$).

In a similar way, one can use $s_2, \ldots, s_{m+2}$ to determine a new set of coefficients to get an estimate $\hat{s}_{m/2+1}$ for the next output point, and so on to the end of the time series. This procedure adjusts all but the first and last $m/2$ values. The output is a less noisy time series if the linearization in Eq. (6) is an accurate approximation of the dynamics.

Because past and future values are used at each step, the procedure can be iterated without the rapid drift due to the sensitive dependence on initial conditions. (One must be careful to avoid especially large corrections, which can arise because the linearization is a poor approximation of the dynamics or because of occasional "glitches" in the data.) Schreiber and Grassberger's method is straightforward to program, runs quickly on a desktop workstation, and can reduce the noise by a factor of 10 in some cases [9].

Schreiber [35] proposed a very simple variant of the above method which gives stable results especially for very short and noisy data: the linear approximation (6) is replaced by a constant, which can be determined with much less data. The resulting algorithm can be coded in a few lines and is quite efficient.

C. Eckmann-Ruelle linearisation

One can work directly with each m-dimensional point on the reconstructed attractor to determine a local linear approximation of the dynamics. As above, the subscript denotes the time ordering of the data: $\mathbf{x}_{i+1}$ is the point immediately following $\mathbf{x}_i$ in time. The dynamics are given by $\mathbf{x}_{i+1} = \mathbf{f}(\mathbf{x}_i)$ for an unknown function $\mathbf{f}$.

One assumes that $\mathbf{f}$ is at least piecewise differentiable and considers a low-order polynomial expansion of $\mathbf{f}$. Eckmann and Ruelle [36] suggested that the local linear approximation $\mathbf{f}(\mathbf{x}_{\text{ref}}) = A\mathbf{x}_{\text{ref}} + \mathbf{b}$ can be computed from the data using least squares. Here A is an approximation of the Jacobian matrix of partial derivatives of $\mathbf{f}$ evaluated at $\mathbf{x}_{\text{ref}}$.

Suppose that $\mathbf{x}_{\text{ref}}$ has k neighbors within a suitably small neighborhood U. Linear regression is used to find the matrix A and vector $\mathbf{b}$ that minimize the sum of squares $\sum_j \|\mathbf{x}_{j+1} - (A\mathbf{x}_j + \mathbf{b})\|^2$, where the sum runs over all indices j such that $\mathbf{x}_j \in U$. A different A and $\mathbf{b}$ are computed for each neighborhood on the reconstructed attractor.

The noise reduction scheme described in Kostelich and Yorke [6] uses Eckmann-Ruelle linearization to approximate the dynamics at each point on the reconstructed attractor. To avoid the drift due to the sensitivity on initial conditions discussed above, it must be followed by a separate trajectory adjustment step in which a nearby time series is computed that is more consistent with the linearization of the dynamics at each point. (This is outlined in Sec. V.)

There are several difficulties with this kind of function approximation that have been considered in detail in [37]. Two of the more important considerations are the following: (1) The accuracy of the approximation depends on how well a linear map describes the dynamics. Some regions of the attractor may contain few observations. A larger ball size increases the number of available points but makes nonlinearities more prominent. This situation becomes more common as the dimension of the attractor increases. (2) The linear least-squares procedure produces biased estimates of the matrix A and vector $\mathbf{b}$ because errors in measurement are present in all the observations; i.e., there is error in both the original points $\{\mathbf{x}_j\}$ and their successors $\{\mathbf{x}_{j+1}\}$. Ordinary least squares produces unbiased estimates only when the errors are confined to the latter set.

It is difficult to make general statements about the effect of these problems in practice. Much depends on the dimension of the attractor, the number of data points, the noise level, etc. Nevertheless, Eckmann-Ruelle linearization has useful applications both in noise reduction and Lyapunov exponent estimation.

D. Local projective maps

An alternative approach uses geometrical considerations to reduce the noise. Since the reconstructed attractor is a subset of a smooth manifold in an m-dimensional phase space, one can estimate the local tangent plane at each point with singular value decomposition. Let $\{\mathbf{x}_i\}_{i=1}^k$ be the set of all points in a small neighborhood of some reference point with mean $\bar{\mathbf{x}}$, and let X be the $k \times m$ matrix whose ith row is $\mathbf{x}_i - \bar{\mathbf{x}}$. Then

$$X = U^T \Sigma V \tag{7}$$

where the columns of U and V form an orthonormal basis for the columns and rows of X, respectively [38]. The entries σ_i of the diagonal matrix Σ are the *singular values* of X: they are the non-negative square roots of the eigenvalues of the covariance matrix $X^T X$. (The decomposition can be arranged so that $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_m \geq 0$.) The sum $\sigma_1^2 + \cdots + \sigma_m^2$ equals the total variance in the observations $\mathbf{x}_i$ [38]. The columns of the orthonormal matrix V are the corresponding singular vectors.

The noise spreads out the observations in all directions, but most components lie along a lower-dimensional hyperplane through $\bar{\mathbf{x}}$. The components in the remaining orthogonal directions presumably are mostly noise. Thus one can ask how many dimensions account for a certain fraction of the total variance. For example, if p is the integer such that $\sigma_1^2 + \cdots + \sigma_p^2$ is about 95% of the total variance, then the span of the first p singular vectors is a good approximation to the tangent plane of the attractor in a neighborhood of $\bar{\mathbf{x}}$. (Obviously, the observations must lie in a small ball for this to be true. Whether one considers 95% of the variance or some other fraction depends upon the estimated underlying noise level. See [39] for details.)

Cawley and Hsu [11] and Sauer [10] suggested that noise in the observations can be reduced by projecting the observations onto the subspace spanned by a suitable collection of singular vectors at each point on the attractor. (Alternatively, instead of projecting each observation directly onto the subspace, one can move the observation only part of the way. This minimizes possible corruption of the data due to a poor approximation to the tangent plane.)

Figure 3 is a schematic diagram of the process. The method is purely geometric; the time ordering of the data is not used to compute the map.

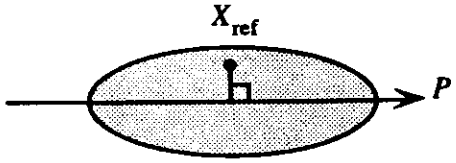


FIG. 3. Schematic diagram of the noise-reduction process in one dimension using singular value decomposition. The line P is the span of the first singular vector. The ellipse indicates that noise spreads out the observations in a neighborhood of the reference point. The reference point (and possibly some nearby points) are projected orthogonally onto P in order to reduce the noise.

When the attractor is reconstructed using a Takens time-delay embedding, the projections typically produce m different corrections for each time series value. Thus it is necessary to incorporate a trajectory adjustment step to compromise between the projections.

E. Local projections with constraints

Grassberger *et al.* [40] have considered a more general procedure in which the projective maps and the corrections to the observations are computed in one step as part of a least-squares minimization problem. Like the projective method above, this method circumvents the problem of biased estimates of derivatives arising from errors in all of the observations and obviates a separate trajectory adjustment step. In addition it exploits the time ordering of the data in a clever way. Figure 4 was obtained using the method described in this section; see [41] for the parameters used.

As above, one assumes that when the embedding dimension m is higher than strictly necessary, the observations can be projected onto a d-dimensional manifold, which is assumed to be approximately linear in a suitably small neighborhood of the reference point. Thus the components of each observation that lie in a q-dimensional subspace perpendicular to the tangent plane of the manifold are zeroed out by the projection, where $q = m - d$.

However, Grassberger *et al.* [40] suggested that the projection should be done in such a way that changes are made only to a central block of coordinates in each observation. (In three dimensions, only the middle value of each observation would be adjusted. If $m = 7$, then one might try to project the middle three observations while leaving the first two and last two essentially unchanged.) Large corrections to coordinates at the end of each observation will propagate due to the sensitive dependence on initial conditions. Similarly, large corrections in the first few coordinates will grow in backwards time. Hence this method uses information from the future and past dynamics to find the best correction to the middle coordinate values. When only the middle value of each observation is adjusted, the method becomes very similar to the one by Schreiber and Grassberger discussed in Sec. IV B.

Let $\mathbf{x}_{\mathrm{ref}} = (s_i, \ldots, s_m)$ be an $(m+1)$ vector whose coordinates are consecutive values from the input time series. (We assume that the attractor can be embedded in m or fewer dimensions.) As above, we let U denote a suitably small neighborhood of $\mathbf{x}_{\mathrm{ref}}$. The components of each $\mathbf{x}_j \in U$ correspond to portions of nearby trajectories on the reconstructed attractor, each of which lies on a manifold of dimension $m - q$. An implicit functional relationship between the components of each $\mathbf{x}_j$ is defined by $\mathbf{F}(\mathbf{x}_j) = \mathbf{0}$, where $\mathbf{F}$ is a vector function with q components. (This can be regarded as a set of q constraints.)

The relation $\mathbf{F}$ is assumed to be approximately linear in U. For each observation $\mathbf{x}_j \in U$, one seeks a vector θ_j, an $(m+1) \times q$ matrix A and a q-vector $\mathbf{b}$ such that (1) $A^T(\mathbf{x}_j + \theta_j) + \mathbf{b} = \mathbf{0}$; (2) $A^T P A = I$; and (3) $\sum \theta_j^T P^{-1} \theta_j$ is a minimum, where the sum is over all the points in U. The corrections θ_j, the matrix A, and the vector $\mathbf{b}$ can be computed in a single minimization problem. The matrix P is a fixed $(m+1) \times (m+1)$ diagonal weight matrix. The entries of P are chosen to penalize large corrections to the first and last few coordinates of each point in the neighborhood. The columns of A are the eigenvectors corresponding to the q smallest eigenvalues of the weighted covariance matrix of the observations. Additional details are given in [40].

Intuitively, the columns of A form an orthogonal basis for a subspace that is perpendicular to the tangent plane of the manifold containing the observations. One seeks the smallest corrections that project each point onto the tangent plane, but distances are measured with a metric determined by the weight matrix P [40]. (The columns of A are not an orthonormal basis for the perpendicular subspace unless P is the identity matrix, in which case the method becomes identical to the one by Cawley and Hsu discussed above.)

Finally, we note that a similar approach, in which a local linear approximation and the adjustments to the observations are computed in one step, was outlined in [37] for cases where the observations are near saddle periodic orbits.

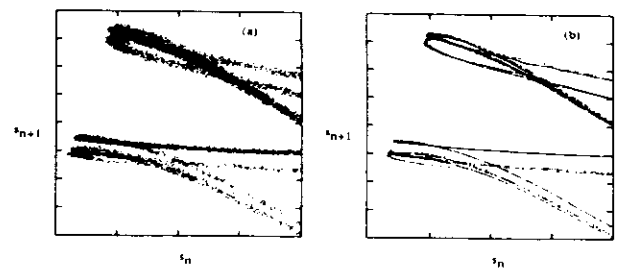


FIG. 4. Enlargements of phase portraits of the Zürich NMR laser data (same data as in Fig. 1): (a) unprocessed data, (b) the same region after noise reduction with constrained projections (Sec. IV E). This is a case where the phase portrait shows effective noise reduction.

F. Global function fits and other methods

Global models try to find one function $\hat{f}$ which gives the best fit to f, where the sum of squares is taken over all the data. A classical method is to express $\hat{f}$ as a linear combination of a set of k basis functions:

$$\hat{f} = \sum_{j=1}^{k} \alpha_j \psi_j. \tag{8}$$

The coefficients α_j are then chosen to minimize the rms approximation error, which is a linear optimization problem that can be solved by standard techniques. A popular choice of functions ψ_j (at least in the context of chaotic dynamics) are radial basis functions [42,43]: $\psi_j(\mathbf{x}) = \phi(\|\mathbf{c}_j - \mathbf{x}\|)$. All basis functions thus have the same functional form and are distinguished only by the different center points $\mathbf{c}_j$. A variety of choices is possible for $\phi(r)$: examples include Gaussian functions, exponential functions, low-order polynomials, and rational functions. The coefficients usually are obtained by minimizing

$$E^2 = \frac{1}{N}\sum_i \left[s_{i+m+1} - \hat{f}(s_i, \ldots, s_{i+m})\right]^2$$

where the sum runs over all the time series values in the data set. The numerical method of choice for this least-squares problem is singular value decomposition; see Press *et al.* [30].

The success of the method depends on the appropriate choice of the form of the basis functions (like the width of the Gaussian functions whenever they are used). In addition, one needs a strategy for choosing the centers $\mathbf{c}_j$ and a criterion for deciding how many basis functions are required. More basis functions leads to closer approximations of the input time series. Of course the original map f is not known, so as more centers are chosen we model more details of the noisy data. When the number of basis functions equals the number of input points, we have an interpolation between all the observations: the interpolation is exact on the observations, the noise is interpreted as part of the dynamics, and the function $\hat{f}$ cannot be used for trajectory adjustment. More material on radial basis-function approximation can be found in [21,42,44].

Another method of global nonlinear function approximation is the modulation by neural networks. In experienced hands they have proven capable of learning nonlinear functions useful for time-series analysis. Weigend, Huberman, and Rumelhart [45] successfully applied this technique to predict sunspot numbers. Neural networks were used successfully in a recent Santa Fe Institute time series prediction contest [23], but the performance of the different networks varied widely. Further references about function approximation by neural networks can be found in [46].

Genetic algorithms under certain circumstances can learn to make predictions [47], but so far seem of limited use for noise reduction. They do not make predictions for every reference point, but only for certain regions in phase space.

Although global function fits have some appeal, the choice of basis functions induces some bias. For instance, the accuracy of the dynamical approximations at each point on the attractor depends in a nontrivial way on both the shape of the basis functions, the distribution of the centers, and the curvature of the trajectories. In the case of neural nets the nature of this bias is unknown. The main advantage of global models is that they can provide stable fits even for small amounts of data.

Finally we remark that straightforward global approximations work mainly for forward-in-time fits as in Eq. (4). In many cases the dynamical equations cannot be solved globally to give a unique value for coordinates other than s_{m+1}, even when the map is invertible. For example, the Hénon map is $s_3 = 1 - as_2^2 + bs_1$. If we try to solve for s_2 we obtain $s_2 = \pm\sqrt{1 + bs_1 - s_3}/a$, which usually has two branches. On a noisy orbit the square root can even become imaginary.

V. TRAJECTORY ADJUSTMENT

Every noise-reduction method requires some strategy to adjust the observations to respect the dynamics more closely. Two requirements must be fulfilled by the new orbit: it should be more consistent with the estimated dynamics than the original signal and it should remain close to the original signal. (In cases where a Takens time-delay embedding is used to reconstruct the attractor, the observations should be adjusted so that the output is a scalar time series.)

Usually the new orbit is obtained as the result of some optimization procedure. It is not possible to satisfy all the criteria simultaneously, so one must make compromises in any particular algorithm. For example, a given point on the attractor may belong to several neighborhoods. Hence, different maps could be used to adjust the point, each of which moves the point in a different way. An open question concerns the best way to reconcile all the possible adjustments.

A. One-step methods

The trajectory adjustment step is trivial if the dynamics are represented according to the Schreiber-Grassberger scheme in Eq. (6). The central coordinate $s_{m/2}$ is well controlled in both the stable and the unstable directions on the attractor, because past and future values are used to adjust it. Thus the estimate $\hat{s}_{m/2}$ can be taken as a replacement for the observed value. The value of the coefficient $a_{m/2}$ must be fixed between 0 and 1 to avoid a trivial least-squares problem. Hence a smaller value can be used to lend more weight to the dynamics, and a value closer to 1 can be used to obtain smaller corrections. The method automatically yields a scalar signal when the attractor is reconstructed with a Takens time-delay embedding.

Only slightly more work is needed in the constrained projective procedure described in Sec. IV E. In this case, proposed corrections are obtained for *all* components of the observations instead of a single central coordinate. In a Takens time-delay embedding, the same scalar measurement appears as a coordinate in several vectors, so the different proposed corrections can be averaged together. (See [40] for more details.) The same considerations apply to the geometric method of Cawley and Hsu [11] and Sauer [10].

B. Two-step methods

Forward-in-time reconstructions require an explicit nontrivial adjustment step, particularly when the attractor is reconstructed using Takens time-delay embeddings. Some methods require the new orbit to fulfill the dynamics exactly. This is reasonable when the dynamics are known with high accuracy. Algorithms of Hammel [48] and Farmer and Sidorowich [12] are examples in this category. When the map is known, it is possible to refine trajectories to arbitrarily high precision.

This is not the case in laboratory situations because the dynamics must be estimated from noisy data. Instead, one must minimize the deviation from the individual adjustments subject to the constraint that the

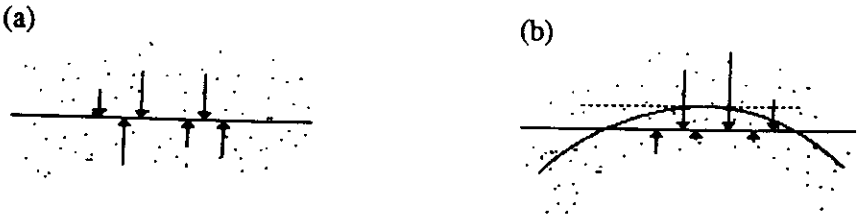


FIG. 5. Consider a noisy one-dimensional set. In the absence of curvature (a), the mean of the corrections is zero. If the underlying manifold is curved (b), projection onto the fitted straight line yields a nonzero mean correction, which can be subtracted to come closer to the desired tangent.

distance from the original orbit must not be too large. Kostelich and Yorke [6] sought the trajectory $\{\hat{x}_{i+k}\}_{k=0}^{p}$ that minimizes the sum of squares

$$\sum_{k=0}^{p} \|\hat{\mathbf{x}}_{i+k} - \mathbf{f}(\hat{\mathbf{x}}_{i+k-1})\|^2 + w\|\hat{\mathbf{x}}_{i+k} - \mathbf{x}_{i+k}\|^2 + \|\hat{\mathbf{x}}_{i+k+1} - \mathbf{f}(\hat{\mathbf{x}}_{i+k})\|^2 \quad (9)$$

where $\mathbf{f}$ denotes the estimated dynamics at each point. Equation (9) expresses the idea that the distances should be small between each fitted point and its preimage, each fitted point and the original observation, and the fitted point and its image. Since their method uses a Takens time-delay reconstruction of the attractor, the minimization in Eq. (9) is done so that the output consists of a sequence of scalar values. The distances between each point and its image are weighted twice as heavily as the distance to the original observation when the parameter w is set to 1. Larger values of w may be appropriate in cases where the input contains large, isolated errors in the data so that the trajectory is not moved excessively to accommodate an occasional erroneous observation.

The minimization problem is more complicated if the model $\mathbf{f}$ is not a piecewise linear function. Strategies for this case have been discussed by Davies [49]. If the nonlinear minimization problem is solved by gradient descent, the weight w can be absorbed into the stepsize for the descent.

We remark that no matter how the sum of squares in Eq. (9) is minimized, the Jacobian of $\mathbf{f}$ must be available. (This is trivial for piecewise linear approximations of the dynamics.) In contrast, the one step methods described above do not need the Jacobian function. This can make one-step methods numerically more robust. For instance, errors in the coefficients computed using the method of Schreiber and Grassberger are not important as long as they predict $s_{m/2}$ reasonably well. The method of Kostelich and Yorke depends more sensitively on the linear maps because they are also used for trajectory adjustment.

C. Recommended improvements

In this section we mention some modifications which were omitted from the description of the algorithms for clarity. Their implementation is not required in order to reduce the noise in most cases, but they can enhance the performance of the methods considerably.

All of the noise-reduction methods occasionally may make anomalously large corrections to an observation. This happens mainly at points where the stable and unstable manifolds are almost tangential. They can also be caused by singular least-squares fits or by a large, isolated error in the data. Since it is not possible to determine the underlying problem with an automatic computer program, most implementations restrict the maximum size of any correction. Kostelich and Yorke [6] did not alter such points at all, but simply flagged them in the output. Alternatively, such points can be moved only some fraction of the computed distance; these observations can be corrected gradually in subsequent iterations over the data set.

Locally linear and projective models introduce errors due to the presence of small nonlinearities, such as the curvature of the attractor. This can be detected because the curvature error is systematic and makes the mean of the corrections nonzero. Forcing the corrections to all points in a small neighborhood to have zero mean partially compensates for this. Sauer [10] proposed this modification, which becomes particularly important when one attempts to remove very small noise levels and curvature errors dominate. Figure 5 illustrates how curvature causes a nonzero mean correction. If this trend is subtracted, the linear approximation approaches the desired tangential form (dashed line).

VI. HOW MUCH NOISE IS TAKEN OUT?

This is a fundamental question. The answer depends partly on the assumptions one makes about the nature of the noise in the input signal. Two error measures have been used for the development of algorithms. When the noise-free signal is known, the error in the input and output is given by the rms distance between the data s_i and the original noise free signal s_i^0:

$$E_0 = \left(\frac{1}{N} \sum_{i=1}^{N} (s_i - s_i^0)^2 \right)^{1/2}.$$

A similar quantity $\hat{E}_0$ can be computed for the cleaned data $\hat{s}_i$. If $\hat{E}_0 < E_0$, then the noise has been reduced. A slightly weaker error measure can be computed when the exact dynamical evolution equation $\mathbf{x}_i^0 = \mathbf{f}(\mathbf{x}_{i-1}^0)$ is known. It measures the deviation from deterministic behavior according to the equation

$$E_{\rm dyn} = \left(\frac{1}{N} \sum_{i=1}^{N} \|\mathbf{x}_i - \mathbf{f}(\mathbf{x}_{i-1})\|^2 \right)^{1/2} .$$

As before, the dynamical error can be compared before and after the noise-reduction procedure has been applied.

These are natural measures when the original trajectory or the "true" dynamics are known. However, other points of view are possible. For example, the output from a noise-reduction scheme might be considered as a cleaner set of data for a slightly different value of the parameter. It is also possible for a noise-reduction procedure to translate the data slightly, resulting in large values for $E_{\rm dyn}$. Nevertheless, the data might still be regarded as a less noisy realization of the dynamics in slightly different coordinates.

The main disadvantage of these quantities is that in a typical experimental situation neither the noise-free data nor the exact dynamics is known. Thus we must employ criteria which are accessible given only the data. Figure 4 shows an example where the processed data "looks less noisy," but we need a more objective, quantitative criterion.

One reasonable assumption is that the data are composed of a deterministic part plus random noise. Thus we can measure the success of noise-reduction algorithms by testing whether the signal has become more deterministic and the subtracted noise is random. Sections VI A and VI B outline methods to test this assumption.

A. Correlation sums

A widely used tool for detecting and quantifying low-dimensional behavior is the correlation sum C defined in Eq. (1). This quantity is used to estimate the correlation dimension d of an attractor. Here we are interested in the scale-dependent effective dimension $d(\epsilon) = d[\ln C(\epsilon)]/d(\ln \epsilon)$. For an infinite amount of noiseless data, $d = \lim_{\epsilon \to 0} d(\epsilon)$.

For laboratory data, this scaling of interpoint distances exists only over a finite range of length scales. As pointed out in Sec. II, the scaling breaks down for larger distances (region D in Fig. 1). At small scales two effects can be seen. When points become sparse, the effective dimension of the attractor fluctuates wildly and no scaling is found (region A). If enough pairs are still found with a distance smaller than the noise level, one measures an effective dimension $d(\epsilon)$ that is close to the dimension of the whole phase space (usually given by the embedding dimension, region B).

If the data set is too short, then region B may not be observable because points become sparse at or above the noise level. This implies that it is not possible to gather enough information on small scales to determine whether a point is displaced by noise and to correct the error. In this case, nonlinear noise reduction tends to break down.

If there are enough points, one finds a crossover region between the noise dominated length scales (region B) where $d(\epsilon)$ equals the embedding dimension and the "correct" scaling (region C) where $d(\epsilon) \approx d$. The length scale at which this crossover occurs is proportional to the noise level. (For Gaussian noise, this observation can be quantified [50].) If one succeeds in reducing the noise in a chaotic time series, then this region is pushed to a smaller scale, as seen in Fig. 1. In the best case, the region is reduced to a scale where spareseness is the dominating effect.

B. Prediction error

If a noise-reduction procedure successfully removes most of the noise, then the processed data set should have better short-term predictability then the raw data set. In other words, the data should be more self-consistent. This can be quantified by computing the *out-of-sample* prediction error with some nonlinear predictor before and after noise reduction. In the simplest case, the data are split into two parts, one of which must be long enough to fit a nonlinear mapping $\hat{\mathbf{f}}(\mathbf{x})$. (One can use the same maps as in the noise reduction itself, but this does not have to be the case.) The prediction error

$$E_p = \left(\frac{1}{N_p} \sum_{i=N-N_p+1}^{N} \|\mathbf{x}_i - \hat{\mathbf{f}}(\mathbf{x}_{i-1})\|^2 \right)^{1/2}$$

is computed over the remaining N_p data points. It is essential to take the out-of-sample error because one can always find a mapping which yields zero in-sample prediction error simply by interpolating between the data points.

If the data set is too short to be split into two sufficiently long parts, then *take-one-out* statistics can be used to obtain an out-of-sample error [7,51]. For each reference point $\mathbf{x}_i$, a predictor $\hat{\mathbf{f}}_i$ is fitted using all the data in the neighborhood except $\mathbf{x}_i$. The resulting map $\hat{\mathbf{f}}_i$ is used only to predict $\mathbf{x}_i$. In this way, one obtains

$$E_p = \left(\frac{1}{N} \sum_{i=1}^{N} \|\mathbf{x}_i - \hat{\mathbf{f}}_i(\mathbf{x}_{i-1})\|^2 \right)^{1/2}$$

as an estimate of the out-of-sample prediction error. This error measure is computed before and after noise reduction. (More material on error measures for nonlinear time-series prediction can be found in [22,23].)

Three effects contribute to the one-step prediction error in low-dimensional chaotic systems.

(1) The observed value $\mathbf{x}_i$ is contaminated by noise, inducing an error proportional to the noise level.

(2) The data values used as arguments of the prediction function are contaminated with noise, leading to an error approximately proportional to the noise level. The proportionality constant is determined by the Jacobian determinant of $\hat{\mathbf{f}}$.

(3) The fitted prediction function is only an approximation to the true dynamics. The accuracy of the fit can be a complicated function of the noise level.

For dissipative maps, effect (1) usually dominates (2). The size of effect (3) is mostly unknown. However, we expect the deviation of the predictor from the true dynam-

ics to increase monotonically with the noise level. In the presence of all three effects, E_p will increase faster than linearly with the noise level. If we compare E_p before and $\hat{E}_p$ after noise reduction we only obtain an upper bound on the amount of noise reduction. However, if $\hat{E}_p < E_p$, then the data have been rendered more self-consistent.

C. Power spectra

Chaotic data are characterized by broadband power spectra [31,17]. Thus it is generally not possible to reduce noise in such data using methods based on power spectra. However, there are cases where signal and noise can be readily distinguished in some part of the spectrum, for example, at high frequencies or around a dominant frequency. Nonlinear noise-reduction methods should suppress noise throughout the spectrum, and their effects also should be visible in these places.

Let us take densely sampled flow data as an example. High-frequency noise can be removed with a simple low-pass filter. Nonlinear noise-reduction methods tend to suppress the high frequencies as well. However, some of the high frequencies are part of the dynamics. Nonlinear noise-reduction methods also can remove some of the lower-frequency noise that is not part of the dynamics. Such distinctions cannot be made with a low-pass or Wiener filter [6].

We do not recommend the power spectrum as a measure of the amount of noise reduction in chaotic time-series data, because it is essentially a linear method. As we discuss below, however, the power spectrum is a useful characterization of the corrections applied to the input data. Most nonlinear methods have limiting cases where they behave like linear filters (for example, when large neighborhoods are used to fit in local linear models or when inappropriate basis functions are used to fit global models). Successful nonlinear noise reduction should make sure we do not pick one of these cases.

D. Consistency tests

So far we have only described tests which can be applied to the cleaned signal in order to check whether it is likely to be low-dimensional deterministic chaos. If the input data consist of uncorrelated noise added to a deterministic signal, then the corrections applied to the data should resemble a random process. One can exploit all that is known about the sources of noise in the data. In most cases, it has only short correlation times. The spectrum will not always be flat, but if it has unexpected features, then one must consider whether the method used to process the data is appropriate.

For example, it is a good idea to check the distribution of the corrections to the data. Measurement errors are expected to have a normal distribution, whereas discretization errors are uniformly distributed in $[0, 2^{-r-1}]$ if the data are stored as r-bit integers.

In addition, one should look for cross correlations between the signal and the corrections. They should be small if the noise-reduction procedure has successfully removed random measurement errors. Significant cross correlations can arise for different reasons, including some threshold in the measurement device, fluctuations in the scale of measurement ("multiplicative noise"), varying noise levels in different regions of the phase space, and corruption of the data due to an inappropriate noise-reduction method.

VII. CONCLUSIONS

The basic problems involved in the analysis of chaotic time-series data are reasonably well understood. Several methods for reducing noise are available, provided that the underlying dynamics are low dimensional. The underlying attractor can be reconstructed from the data in several possible ways, some of which can act as linear filters to lower the noise level.

More sophisticated noise-reduction methods are available which exploit the local dynamical behavior to identify and correct errors arising from noise. A variety of methods exists for estimating the dynamics. One-step methods, such as those suggested by Grassberger and coworkers [7,40], adjust the observations as part of the process of determining a local linear model for the dynamics. Projective schemes, such as that implemented by Cawley and Hsu [11], move the observations onto a subspace that approximates the tangent plane to the manifold containing the attractor at each point. Kostelich and Yorke [6,8] compute an estimate of the Jacobian matrix of partial derivatives of the map at each point on the attractor.

Trajectory adjustment refers to the process by which the input data are changed to be more consistent with the estimated dynamics. Although the true dynamics are not known in most experimental situations, there are several ways to measure the self-consistency of the data and to check whether the noise level has been reduced.

Even low levels of noise can complicate the estimation of basic quantities such as attractor dimension and Lyapunov exponents. For this reason, the use of noise-reduction methods is strongly recommended in any analysis of chaotic time-series data.

ACKNOWLEDGMENTS

The authors thank L. Flepp, J. Simonet, and E. Brun for the data used in Figs. 1 and 4. E.K. is supported by the National Science Foundation Applied and Computational Mathematics Program under Grant No. DMS-9017174. T.S. receives a European Communities grant within the framework of the SCIENCE program, Contract No. B/SCI*-900557.

* Permanent address: Physics Department, University of Wuppertal, D-5600 Wuppertal 1, Gauss-Strasse 20, Germany.

[1] B.-L. Hao, *Chaos II* (World Scientific, Singapore, 1990).
[2] *Dimensions and Entropies in Chaotic Systems*, edited by G. Mayer-Kress (Springer-Verlag, Berlin, 1986).
[3] A. Wolf, J. B. Swift, H. L. Swinney, and J. A. Vastano, Physica D **16**, 285 (1985).
[4] J.-P. Eckmann, S. O. Kamphorst, D. Ruelle, and S. Ciliberto, Phys. Rev. A **34**, 4971 (1986).
[5] A readable discussion of the definitions can be found in D. Gulick, *Encounters with Chaos* (McGraw-Hill, New York, 1992).
[6] E. J. Kostelich and J. A. Yorke, Physica D **41**, 183 (1990).
[7] P. Grassberger, T. Schreiber, and C. Schaffrath, Int. J. Bifurcation Chaos **1**, 521 (1991).
[8] E. J. Kostelich and J. A. Yorke, Phys. Rev. A **38**, 1649 (1988).
[9] T. Schreiber and P. Grassberger, Phys. Lett. A **160**, 411 (1991).
[10] T. Sauer, Physica D **58**, 193 (1992).
[11] R. Cawley and G.-H. Hsu, Phys. Rev. A **46** 3057 (1992); R. Cawley and G.-H. Hsu, Phys. Lett. A **166** 188 (1992).
[12] J. D. Farmer and J. Sidorowich, in *Evolution, Learning and Cognition*, edited by Y. C. Lee (World Scientific, Singapore, 1988); Physica D **47** 373 (1991).
[13] J. C. Roux, R. H. Simoyi, and H. L. Swinney, Physica D **8**, 257 (1983).
[14] K. Coffman, W. D. McCormick, Z. Noszticzius, R. H. Simoyi, and H. L. Swinney, J. Chem. Phys. **86**, 119 (1987).
[15] F. Takens, in *Dynamical Systems and Turbulence*, edited by D. A. Rand and L.-S. Young, Lecture Notes in Mathematics Vol. 898 (Springer-Verlag, New York, 1981), p. 366.
[16] P. Grassberger and I. Procaccia, Phys. Rev. Lett. **50**, 346 (1983).
[17] A. Brandstater and H. L. Swinney, Phys. Rev. A **35**, 2207 (1987).
[18] L. Flepp, R. Holzner, E. Brun, M. Finardi, and R. Badii, Phys. Rev. Lett. **67**, 2244 (1991); M. Finardi, L. Flepp, J. Parisi, R. Holzner, R. Badii, and E. Brun, *ibid.* **68**, 2989 (1992).
[19] H. D. I. Abarbanel, in [22].
[20] J. D. Farmer and J. Sidorowich, Phys. Rev. Lett. **59**, 845 (1987).
[21] M. Casdagli, Physica D **35**, 335 (1989).
[22] *Nonlinear Modeling and Forecasting*, edited by M. Casdagli and S. Eubank, Santa Fe Institute Studies in the Science of Complexity, Proc. Vol. XII (Addison-Wesley, Reading, MA, 1992).
[23] *Predicting the Future and Understanding the Past: A Comparison of Approaches*, Proceedings of the NATO ARW on Time Series Analysis and Forecasting held in Santa Fe, NM, 1992, edited by A.S. Weigend and N.A. Gershenfeld, (Addison-Wesley, Reading, MA, 1993).
[24] See for example F. Romeiras, C. Grebogi, E. Ott, and W. P. Dayawansa, Physica D **58**, 165 (1992); E. Ott, C. Grebogi, and J.A. Yorke, Phys. Rev. Lett. **64** 1196 (1990); G. Nitsche and U. Dressler, Physica D **58** 153 (1992); W. L. Ditto, S. N. Rauseo, and M. L. Spano, Phys. Rev. Lett. **65**, 3211 (1990).
[25] M. Casdagli, S. Eubank, J. D. Farmer, and J. Gibson, Physica D **51**, 52 (1991).
[26] T. Sauer, J. A. Yorke, and M. Casdagli, J. Stat. Phys. **65**, 579 (1991).
[27] The box-counting dimension can be defined in a manner analogous to the correlation dimension. Let $N(\epsilon)$ be the minimum number of cubes of side length ϵ needed to cover the attractor. Then $N(\epsilon) \sim \epsilon^d$ as $\epsilon \to 0$, where d is the box-counting dimension. For more details, see J. D. Farmer, E. Ott, and J. A. Yorke, Physica D **7**, 153 (1983).
[28] D. Broomhead and G. P. King, Physica D **20**, 217 (1986).
[29] A. M. Albano, J. Muench, C. Schwartz, A. I. Mees, and P. E. Rapp, Phys. Rev. A **38**, 3017 (1988).
[30] W. H. Press, B. P. Flannery, S. A. Teukolski, and W. T. Vetterling, *Numerical Recipes* (Cambridge University Press, Cambridge, 1988).
[31] P. R. Fenstermacher, H. L. Swinney, and J. P. Gollub, J. Fluid Mech. **94**, 103 (1979).
[32] R. Badii, G. Broggi, B. Derighetti, M. Ravani, S. Ciliberto, A. Politi, and M. A. Rubio, Phys. Rev. Lett. **60**, 979 (1988); R. Badii *et al.*, *ibid.* **60**, 979 (1988); M. Muldoon (unpublished); Milan Paluš and Ivan Dvořák, Physica D **55**, 221 (1992).
[33] P. L. Read (unpublished).
[34] R.W. Preisendorfer, *Principal Component Analysis in Meteorology and Oceanography* (Elsevier, Amsterdam, 1988).
[35] T. Schreiber, Phys. Rev. E **47**, 2401 (1993); A. Pikovsky, Sov. J. Commun. Technol. Electron. **31**, 81 (1986).
[36] J.-P. Eckmann and D. Ruelle, Rev. Mod. Phys. **57**, 617 (1985).
[37] E. J. Kostelich, Physica D **58** 138 (1992).
[38] See, for example, G. W. Stewart, *Introduction to Matrix Computations* (Academic, New York, 1973).
[39] D. Broomhead, R. Jones, and G. P. King, J. Phys. A **20**, L563 (1987).
[40] P. Grassberger, R. Hegger, H. Kants, C. Schaffrath, and T Schreiber, Chaos (to be published).
[41] H. Kantz, T. Schreiber, I. Hoffmann, T. Buzug, G. Pfister, L.G. Flepp, J. Simonet, R. Badii, and E. Brun, Phys. Rev. E **48**, xxxx (1993).
[42] D. Broomhead and D. Lowe, Complex Syst. **2**, 321 (1988).
[43] J. Holzfuss, and J. B. Kadtke, Int. J. Bifurcation Chaos (to be published).
[44] See L. A. Smith, Physica D **58** 50 (1992), and references therein.
[45] A. S. Weigend, B. A. Huberman, and D. E. Rumelhart, Int. J. Neural Syst. **1** 193 (1990); A. S. Weigend, B. A. Huberman, and D. E. Rumelhart, in [22].
[46] S. A. Solla, in [22].
[47] J. L. Breeden and N. Packard, Technical Report No. CCSR-92-11, 1992 (unpublished).
[48] S. M. Hammel, Phys. Lett. A **148**, 421 (1990); M. E. Davies, *ibid.* **169** 251 (1992).
[49] M. E. Davies, Int. J. Bifurcation Chaos (to be published); M. E. Davies (unpublished).
[50] T. Schreiber, Phys. Rev. E (to be published).
[51] B. Efron, *The Jackknife, the Bootstrap and Other Resampling Plans* (SIAM, Philadelphia, 1982).