USER'S MANUAL

for PC computers

CHAOS DATA ANALYZER

The Professional Version

Julien C. Sprott

Department of Physics
University of Wisconsin
Madison, WI 53706

and

George Rowlands

Department of Physics
University of Warwick
Coventry CV47AL England

CONTENTS

PREFACE

CHAOS DATA ANALYZER: THE PROFESSIONAL VERSION

CHAOS DATA ANALYZER: THE PROFESSIONAL VERSION (CDA PRO) is a research (or teaching) tool consisting of 19 programs that allow you to analyze a record of up to 32,000 data points representing some quantity measured at equally spaced intervals. If the interval is a time, the method is referred to as time-series analysis, and the record is typical of the kind of signal one might display on an oscilloscope or chart recorder.

If you have the regular version of CDA, you will find that CDA PRO includes all the features of that version. In addition, many calculations are faster, and the program can accommodate twice as many data points. There are many new tests, ways to view data, and sample data sets. You will find greatly improved tools for making predictions, for detrending your data, and for generating surrogate data sets to determine the statistical significance of your results.

The major goal of CDA PRO is to detect hidden determinism (chaos) in seemingly random data. Where chaos is found to exist, tools are provided to enable you to determine properties of the equations underlying the behavior.

These tools include calculations of the probability distribution, power spectrum, Hurst exponent, Lyapunov exponent, and various measures of the fractal dimension. The data can be displayed in many ways including time records, phase-space plots, return maps, Poincaré movies, and wavelet transforms. Four different methods including an artificial neural network and a novel technique employing singular value decomposition are used to construct the next few terms in the time series. Thus CDA PRO is a powerful forecasting tool.

The data are read from a disk file provided by the user or produced by the companion program CHAOS DEMONSTRATIONS. Numerous sample data files are included with the program.

All the commands are entered by single keystrokes or by single clicks of a mouse. CDA PRO is written to make it easy for you to move from one analysis program to the next. It has an automatic mode in which all the tests are performed in sequence and a summary of the results is provided.

CDA PRO can also be used as a lecture demonstration tool or student tutorial in a course on nonlinear dynamics or statistical analysis.

This manual was coauthored by J. C. Sprott and George Rowlands, while the software was created by J. C. Sprott. Portions of this work were funded by the U. S. Department of Energy. Computers were provided through a Technology Transfer contract from IBM.

BEFORE YOU BEGIN

Unpacking

Your package for CHAOS DATA ANALYZER: THE PROFESSIONAL VERSION (CDA PRO) is included as a .zip file that contains the program and all necessary files including this *User's Manual*. The User's Manual contains information on how to operate the program and how to execute the commands. It also contains explanations, activities, and a discussion of time-series analysis techniques.

System Requirements

Your computer system and hardware configuration should be any PC computer. The program should run under older versions of Windows, but for more recent versions, support for DOS graphic programs has been removed, in which case you will need to first install the free DOSBox program available from http://www.dosbox.com/.

Licensed Copies

As a registered owner of a single-copy license, you may copy the program onto your hard disk; however, you may not use this program on more than one hard disk at a time.

Each authorized copy may be loaded onto one hard-disk system. The program may be loaded on a network computer system, but one authorized copy is required for each computer operating the program simultaneously.

Terminology

The following terminology is used in this manual:

Press means to strike or press a key. For example, the instruction to press the *<Enter>* key appears as shown below:

Press *<Enter>*

Note: On some keyboards the *<Enter>* key is labeled "Return."

If two keys are used in combination to perform one task, they are presented on the same line. *The first key must be held down while the second key is pressed.* For example, the instruction to hold down the *<Ctrl>* key while pressing the *<Enter>* key would appear as shown below:

Press *<Ctrl> <Enter>*

Type means to strike or press a series of keys. The keys to be typed appear in boldface. For example, the instruction to type the word **start** appears as shown below:

Type **start**

Starting Up

To start CDA Pro,

1. Create a directory (such as CDA) on your hard drive
2. Extract all the files in the .zip file to that directory
3. From that directory, type CDA and press *<Enter>*

Assistance

This program is provided "as is" without technical support with no claim that it will meet your needs. Your use of the program is at your own risk.

ABOUT THE PROGRAM

MENU AND GENERAL OPERATION

After you start CHAOS DATA ANALYZER: THE PROFESSIONAL VERSION (CDA PRO), the program displays a menu screen with the date, time, and copyright information at the top, and a list of 20 options below (see Figure 1). You can try out any of the programs by pressing the key, A to S, corresponding to the program name or you can select a program by moving the highlighted cursor with the arrow keys on your keyboard and then pressing *<Enter>*. You can also use the mouse to select a program. When you exit a program, you will return to this menu screen.

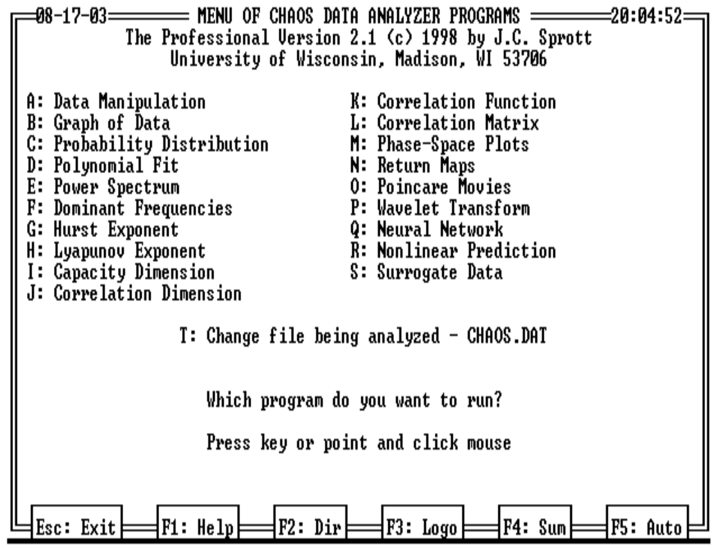

Fig. 1. The opening menu for CDA PRO.

The menu screen also contains a menu item "T: Change file being analyzed," followed by the name of the current file being analyzed (defaults to CHAOS.DAT). You can press T to access a list of the files in the current directory that have the extension .DAT. You can then move the cursor with the arrow keys or mouse and press *<Enter>* or click the left mouse button to select a different file to analyze. You can also cycle through the data files in the current directory using the F command in the Data Manipulation program. You will see the data being graphed as the files are loaded into the program. The data may exceed the graph boundaries or may be compressed into a horizontal line because the graph scale is determined only after the data have been loaded. Once the new data file has been loaded, press X to return to the menu screen.

A number of special commands appear along the bottom of the menu screen.

When you are through using CDA PRO, press <Esc> to exit the program. The right mouse button also acts as an <Esc> key.

To see an abbreviated set of instructions about CDA PRO, press F1.

When you are done reading the instructions, press <Esc> or the right mouse button to return to the menu.

To see a list of the files in the disk directory from which you are running the program, press F2.

Normally, you will not need to do this unless you get an error message during program operation indicating that one of the disk files necessary to run the program is missing.

To display the program logo, press F3.

To return to the menu, press any key.

To see a summary of the analysis performed thus far, press F4.

To cycle the programs sequentially without returning to the menu until the sequence is complete, press F5.

In this mode the programs step through a predetermined sequence, spending about 10 to 20 seconds on each screen, just as if an invisible person was pressing the keys. You can take control of the keyboard or mouse at any time by executing your own commands, but if you wait more than 45 seconds without pressing a key or clicking the mouse, the automatic sequence will resume.

To single-step through the automatic sequence, press *<Tab>*.

To terminate the automatic sequence, press *<Esc>*.

Appendix A describes how to customize the automatic sequence or to make it repeat indefinitely.

The individual programs are controlled by single keystrokes. A number of commands are common to all or many of the programs:

Press E to display a single screen of explanation for the program being executed. Press any key or mouse button to return to the program.

Press P to halt all action on the screen and then press it again to resume where the program left off.

Press R to restart the program from the beginning. Be careful not to do this if you have spent a long time generating a picture that you do not want to destroy.

Press G to switch among the various graphics modes that your computer supports (CGA, EGA, VGA, etc.). When the program starts, it uses the highest resolution color graphics mode available, but you may want to use one of the other modes to suppress the color, to enlarge the text, or to speed some of the slower calculations.

Press C to change among the background colors that are supported by your monitor. Various screens look better against different backgrounds. In addition, if your computer has EGA or VGA graphics, you can cycle among 64 color palettes by pressing + or -.

Press S to toggle the sound on and off in those programs with sound effects.

Press T to control whether or not the points on the screen are connected by lines (Track or Untrack). Note that in the Poincaré Movies program, Track connects each point to the corresponding point at the previous time step.

Most of the programs have a number of different related quantities or graphs that you can examine in succession.

Press V to cycle among these various options.

Press Z to enlarge the graph window to fill the whole screen. This makes the data easier to see, but it slows the calculation slightly in some cases and erases all the text from the screen, including the commands. The command keys are still active, but you have to remember them. Pressing any key that is not a designated command restores the display to its state before zoom enlargement.

Press X or *<Esc>* to exit the program and return to the menu.

In addition to these commands, most programs have parameters that you can vary. Press the letter key corresponding to the parameter to cycle through various values. You can also put the cursor on the parameter you wish to change and then press *<Page Up>* or → to increase its value or <Page Down> or ← to decrease its value. Most of the commands also allow you to move backward through the sequence by holding down the *<Shift>* key.

If your computer has a printer and you are running CDA PRO in Windows 95/98 you can make copies of the screen by pressing the *<Print Screen>* key to capture the screen on the clipboard. Then you can paste the screen image into your word processor or paint program and print it from there. In Windows NT/2000, you can put the CDA PRO in a Window (using Alt-Enter) and then Edit-Copy.

DATA FILES

File Format

The data file you wish to analyze must contain a single sequence of numbers in any ASCII format. Each number must be separated from the one before it by a space (ASCII 32), comma (ASCII 44), carriage return (ASCII 13), or line feed (ASCII 10). The program ignores any spaces before or after the number. The data can be integers (such as 1234), fixed point (such as 1234.0) or floating point (such as 1.234e3), or a combination. The data can be ordinary decimal, hexadecimal (preceded by &H), octal (preceded by &O), binary (preceded by &B), or a combination. Data values should be in the range of -1×10^{18} to 1×10^{18}.

The number of data points that the program can analyze depends upon the free memory available in your computer. The maximum number of data points is 32,000, but if you have less than 640K of RAM or if you have other programs resident in the first 640K of your memory, the number may be smaller. You can use the MEM command at the DOS prompt to see how much of your 640K conventional memory is available to execute the CDA PRO program. If the file contains more data points than your computer can accommodate, the program will ignore points at the end of the file that exceed the memory limit. A warning message will appear in such a case. It is recommended that you use the extension .DAT for data files. Without the .DAT extension, a file cannot be accessed from within the program but only from the command line. See command line options in Appendix A.

Acquiring Data

You can produce data files for analysis using five different methods:

Sample files.

The CDA PRO disk contains a large number of sample data files that are ready to use. A list of these files is included in Appendix A. Most of these files were selected from an ensemble of 26 similar cases chosen such that various moments of their probability distribution are closest to the average of the corresponding moments of the 26 cases. This procedure ensures that the sample cases are typical.

If you start the program in the usual way, it assumes you want to analyze data in the file CHAOS.DAT, which contains 2,000 points of chaotic data from the variable $X(t)$ for the Lorenz Attractor.[1,2] You can examine the other sample files by repeatedly pressing F (for "files") while in the Data Manipulation program or by selecting "T: Change file being analyzed" from the main menu. Alternately, from the DOS prompt, you can type **CDA** followed by a space and the name of the file that you wish to analyze. The extension .DAT is assumed if no extension is given. If you want to analyze a data file that does not have an extension, follow the file name with a period.

To conserve space on your hard disk, you may wish to delete most or all of the sample data files. If you do this, retain the file CHAOS.DAT so that the program starts properly when no data file is specified on the command line.

Files generated by Chaos Demonstrations.

If you have the companion program CHAOS DEMONSTRATIONS,[3] you can use it to produce files that can be read by CDA PRO. Refer to Appendix A of the CHAOS DEMONSTRATIONS User's Manual for an explanation of how to produce data files from CHAOS DEMONSTRATIONS.

Files generated by CDA Pro.

The CDA PRO program can itself generate data files containing the output of its own calculations. For example, typing **CDA CHAOS,LVVVVVVV{CDA}** at the DOS prompt (see command line options in Appendix A) will use the correlation matrix program (with singular value decomposition) to remove noise from the data file CHAOS.DAT and save the new data record in the file CDA.DAT. You can then analyze the new data file with the command **CDA CDA**. Note that CDA.DAT is the default output data file, just as CHAOS.DAT is the default input data file. Thus you can abbreviate the above command as **CDA,LVVVVVVV{}**.

If you want to save the output of the last calculation performed before exiting to DOS, place the output file name in braces as the first (or only) item in the command string. The simplest command would be **CDA ,{},** which would save the most recent calculation to the default output file CDA.DAT. By analyzing the data thus created, you can perform certain calculations that would otherwise be impossible, such as determining the correlation dimension of the data from the model equations in the correlation matrix program. The values saved in the file are single-precision floating point numbers. The number of data points collected is indicated on the lower right of the screen. Any previously existing data in the file are overwritten when the program starts and when it is restarted by pressing the R key.

Files generated by a word processor.

Small data sets can be typed into a file using any word processor, spreadsheet, or editor that can write ASCII files. The DOS editor, EDIT, will suffice. Give the file an appropriate name with the extension .DAT and type in a series of numbers. Press *<Enter>* after each number. Existing files that are not in ASCII format can be modified with a word processor and then saved as ASCII files.

Files generated by another program.

If the program generating the data is one you wrote, you probably know how to modify it to produce a data file in a compatible format. If the data file is inherited from another source and is incompatible, and if simple editing with a word processor is cumbersome, your only choice may be to write a small program that reads the file and then rewrites it to a second file in the required format.

Time-Series Analysis

This section contains general information about time-series analysis strategies and procedures. Experienced users may wish to skip this section and go directly to the description of the tests they wish to perform.

Deterministic Chaos

The objective of experimental data analysis is to find a pattern or structure that models the data. The structure itself may not be simple or obvious, but once found, the original data take on a new simplicity. For example, data that produce a Lissajous figure can be expressed as two Fourier modes with constant amplitude and frequency. Unfortunately, this simple picture is inappropriate for many important areas of science. For example, the turbulent behavior of fluids, such as water, requires a large number of Fourier modes that form a continuum.

In recent years the mathematical study of dynamical systems has shown that the solution of simple, nonlinear equations can exhibit complicated temporal and spatial behavior. Many examples now exist of equations showing a variety of behavior ranging from simply periodic to chaotic. An underlying concept that has emerged from such studies is that of a strange attractor. Simple attractors such as a system in thermodynamic equilibrium (point attractor) or those which decay into periodic states (limit cycles) have been known for some time. A strange attractor has the property of the above examples in that the system decays to (is attracted to) a final state, but this state is not periodic and is extremely complex. In fact, it is chaotic—pseudo-random but arising as the solution of a deterministic set of equations. It is also highly sensitive to initial conditions. These concepts are illustrated in the companion Chaos Demonstrations software[3] and in books[4,5] by the authors.

CDA PRO analyzes data in the form of a time series $X(t)$, for a single variable X, with two assumptions: first, that the presence of a strange attractor can lead to a complicated time series, and second, that such phenomena can be described by simple, nonlinear equations. These equations may be differential equations in which the quantity X varies continuously in time but is sampled at discrete time intervals, or they may be difference equations in which X changes abruptly from one step to the next. The presence of a strange attractor is best revealed from phase-space plots. If the data were given at sufficiently short time intervals and did not have a significant noise component, then the phase-space plot could be constructed with components $X(t)$ and dX/dt, and higher derivatives, and then the whole data record would be represented as a trajectory in this phase space. The procedure would be to consider higher and higher dimensional phase spaces until an increase in the dimension did not change the topology of the underlying structure. For example, a plot in a two-dimensional phase space of the trajectory for a simple loop would still remain a simple loop if plotted in a higher dimensional space.

Once a strange attractor has been identified, its nature can be quantified by calculating various measures of its dimension and its Lyapunov exponent. The dimension is a measure of the complexity of the phase-space trajectory, whereas the Lyapunov exponent is a measure of the sensitivity to initial conditions. To get an accurate measure of an attractor's dimension, it is theoretically necessary to embed it in a space of dimension at least $2d + 1$ where d is the least integral dimension containing the attractor.[6]

There are various (in fact, an infinite number of) ways to define the dimension of an attractor. The simplest are the Hausdorff and capacity dimensions, which describe the geometric dimension of the attractor without regard to how frequently the trajectory visits various locations on the attractor. The correlation dimension is a lower bound on the capacity dimension, but in most cases, it closely approximates the capacity dimension. The correlation dimension is more accurate for small data sets because it weighs more heavily those regions of the embedding space that contain data.[7,8] It has been shown[9] that it is very difficult to calculate a capacity dimension when the dimension is greater than about two. The number of data points required to get a reliable estimate of the dimension is the order of M^d where M has been variously estimated[10] to be in the range of about 10 to 42. More recent estimates[11,12] give a minimum number of data points of approximately $10^{2+0.4d}$. Similar considerations apply for the calculation of the Lyapunov exponent,[13] although it appears that a somewhat smaller embedding dimension suffices in practice. The reliability of the results depends on the system, and no rigorous methods for determining errors are known that work for all cases. Most of the sample data sets provided with this program are not large enough to provide accurate estimates of the dimension and Lyapunov exponent. Estimated errors are provided in these cases and are based on empirical results from the analysis of a large number of systems for which the expected values are known.

Simple Time-Series Expansion

Until recently the usual method of analyzing a time series has been to expand the series in a set of Fourier modes. If the number of modes is small, then the dynamics of the system can be expressed in terms of the dynamics of these modes. Mathematically, this means that the time series for a quantity $X(t)$ may be expressed in the form

$$X(t) = \sum_{m=1}^{M} A_m e^{i\omega_m t} \tag{1}$$

with M being a small integer. The power spectrum $P(\omega) = |\int X(t) e^{i\omega t}\, dt\,|^2$ is then just a series of delta functions at $\omega = \omega_m$ (m taking all integer values between 1 and M) of strength $|A_m|^2$. This method is useful if the power spectrum consists of a finite number of distinct peaks. In practice, the presence of external noise broadens the delta functions, but if the noise is not too strong, the peaks in $P(\omega)$ can still be recognized, and hence A_m and ω_m can be calculated. The above expression for $X(t)$ then gives a smoothed or noise-free representation of the original data.

An important question remaining is how best to represent these smoothed data. One way is to use the phase plane where $X(t)$, for example, is plotted against dX/dt. The latter quantity is readily obtained from Equation 1, but not from the original noisy data. For $M = 1$ (a single mode), such a plot would reveal a simple closed loop, whereas for $M = m_o$, the $(m_o + 1)$-dimensional phase-space portrait, whose ordinates are $X(t)$, dX/dt, ... $d^{m_o}X/dt^{m_o}$, would also be a simple m_o-dimensional loop or torus (depending on whether or not the frequencies are harmonically related). This type of analysis illustrates how a single time-varying quantity $X(t)$ can give multidimensional information.[14,15] However, in a significant number of experimental situations, the power spectrum $P(\omega)$ has little structure. In such a case, an expression in the form of Equation 1 is not appropriate, although a Fourier transform integral does convey some information.

Singular Value Decomposition

An alternative method of analysis is singular value decomposition[16,17] (also called principal component analysis). This method, originally used in statistics to study linear problems, has more recently been applied to inherently nonlinear problems. The method replaces the expansion of $X(t)$ in Fourier modes by an expansion in terms of another complete set of functions. This set is obtained from a numerical analysis of the data and is not imposed, as in the Fourier series representation, from outside. Thus instead of Equation 1 we write

$$X(t) = \sum_{m=1}^{\infty} \psi_m(t) \tag{2}$$

where the functions ψ_m form a complete orthogonal set. By analogy with $P(\omega)$ we define $P_m = [\int X(t)\, \psi_m(t)\, dt]^2$, which is simply $P_m = (\int \psi_m^2\, dt)^2$, and the latter form we can interpret as the probability of the system being in the state ψ_m. Thus, by analogy with Fourier modes, we can anticipate that there are situations where $X(t)$ can be adequately represented by just a few of the ψ_m functions. Then, just as a few peaks in the power spectrum indicate the possibility of an expansion in a small number of Fourier modes, the existence of a few dominant values of P_m indicates the possibility of a useful expansion in a small number of the ψ functions.

Singular value decomposition is a method for the generation of an orthogonal set of such functions. The essentials of the method are as follows. The data are assumed to be known at a finite number, N, of equally spaced intervals of time, Δt. Writing $X_n = X(t = n\Delta t)$ we assume X_n is known for $n = 1, 2,..., N$. From these data we construct a set of M-dimensional vectors V_e defined such that

$$\mathbf{V}_e = \{X_e, X_{e+1}, \ldots, X_{M+e-1}\} \tag{3}$$

We also construct the auto-correlation function defined by

$$C(n) = \sum_{e=1}^{\infty} X_e X_{e+n} \tag{4}$$

Using these values of $C(n)$, we can construct the symmetric $M \times M$ correlation matrix $\underline{M}$ with elements $M_{ep} = C(|e - p|)$. This matrix has M eigenvalues, which we denote by λ_m, and corresponding eigenfunctions α_m. Using these, we define the functions $\psi_m(t)$ such that

$$\psi_m(t = n\Delta t) = \boldsymbol{\alpha}_m \bullet \mathbf{V}_n \tag{5}$$

These functions are orthogonal and normalized such that

$$\frac{1}{N}\sum_{e=1}^{N} \psi_m{}^2(e\Delta t) = \lambda_m \tag{6}$$

which leads to $P_m = \lambda_m^2$. Hence, λ_m is a measure of the importance of the mode ψ_m in the expansion of $X(t)$. Thus, by analogy with the usual procedure of expanding in a finite number of Fourier modes, we now expand in a finite number, d, say, of the values for ψ and select those values for ψ that correspond to the d largest values of λ_m. Thus as an approximation to the original data $X(t)$ we write

$$X_d(t) = \sum_{m=1}^{d} \psi_m(t) \tag{7}$$

Besides giving the best set of orthogonal modes in the sense mentioned above, this method involves some smoothing of the original data, which is desirable since most experimental data have a significant noisy non-deterministic component. A purely random time series leads to a correlation matrix $\underline{M}$, which is diagonal ($C(n) = C_o\delta_{n,o}$) and whose eigenvalues are all equal to C_o. Thus for any other data, the existence of eigenvalues (λ_m) greater than C_o reveals the presence of structure in the data. By limiting the summation in Equation 7 to such eigenvalues, one automatically removes a substantial amount of the noise, so that $X_d(t)$ is a smoothed version of $X(t)$. The singular value decomposition method is identical to the Karhussen-Love expansion and as such was originally suggested by Lumley to study turbulence.[18,19]

A very powerful method of identifying any underlying structure in the original data $X(t)$ is to examine the d-dimensional phase space constructed by means of the functions ψ_1, ψ_2, ..., ψ_d. Such plots reveal the topological structure of the solution. This structure is the same as that obtained by considering a phase space constructed using the vectors **V** as defined above from the original data. However, topological details in the latter phase-spaceplot could be masked because of the presence of noise in the vectors. The topological structure is conserved from a **V**-based phase space to a ψ-based phase space because the ψ functions are constructed as linear combinations of the **V**'s. An additional feature of singular value decomposition is that a two-dimensional plot of ψ_2 versus ψ_1 amounts to a rotation of the attractor in such a way that it is viewed from its broadest side, and thus its structure is most readily apparent.

Given the nature of the ψ functions, it is convenient to assume the equations of motion are of the form

$$\overline{\psi}_m(t+\Delta t) = F_m[\overline{\psi}_n(t)] \tag{8}$$

where the bar over the ψ's is used to denote a value satisfying these model equations and F_m is a nonlinear function of all the ψ_n values from $n = 1$ to d. Guided by the argument that simple nonlinear functions are sufficient to produce chaotic behavior, we assume a form

$$F_m(\overline{\psi}_n) = a_{mo} + \sum_{q=1}^{d} a_{mq}\overline{\psi}_q + \sum_{q,e}^{d} b_{mqe}\overline{\psi}_q\overline{\psi}_e \tag{9}$$

that is a quadratic polynomial characterized by the constants a and b. These constants are determined by obtaining, in the least squares sense, the best fit of the solutions of the equations of motion (Equation 8) to the known form for the ψ_m functions. That is, the quantities J_m, where

$$J_m = \frac{1}{N}\sum_{s=1}^{N}\{\psi_m[(s+1)\Delta t] - F_m[\overline{\psi}_n(s\Delta t)]\}^2 \tag{10}$$

are minimized with respect to the coefficients a and b.

Virtues of Singular Value Decomposition

The original data $X(t)$ are given for a finite time interval only (N is finite). However, once the equations of motion for the ψ_m functions are known, they can be solved for all time. Then, using Equation 7, we have an estimate, $X_d(t)$, for $X(t)$ for all time. Thus, in principle, we have an extrapolation scheme. However, this scheme should be used with caution. If the underlying system is chaotic, then the sensitivity to initial conditions leads us to expect $X_d(t)$ to be a good representation of $X(t)$ only for a short period of time after $t = N\Delta t$. Although the ψ's capture the topology of the original data, solutions of the model equations (Equation 8) do not necessarily provide good topological information.

Topological information depends on sufficient data. For example, a major obstacle in the calculation of fractal information is insufficient data; that is, N is too small.[12] However, the model equations can be used to generate as many data points as required to calculate the dimension and other properties of the attractor. It is also much simpler to calculate Lyapunov exponents from equations than to compute them directly from data. The procedure of first obtaining equations thus leads to a significant simplification in those cases where the equations reasonably replicate the topology of the data.

In singular value decomposition there are two quantities, namely M, the order of the correlation matrix, and d, the number of significant eigenfunctions retained. These parameters must be adjusted to obtain the best fit between the real system under investigation through the data $X(t)$ and the solution of the model Equation 8. Since we envisage the application of this method to situations where the auto-correlation function shows little structure, we hope the complicated time variation can be attributed to the presence of a strange attractor. Then the parameters M and d are chosen to reproduce the topological features of the attractor.

Analysis Strategy

In analyzing data, we are really looking for the simplest way to describe them. The best description depends on the particular problem, and no universal solution exists. Thus, it is best to try the simplest things first. Before you begin, you should look carefully at your data to verify that there are no anomalies such as gaps in the data record, obvious incorrectly recorded data points, and the like.

Another common problem with real experimental data is lack of stationarity. This would show up as a general trend such as you might have with stock market data during a prolonged bull market or with meteorological data during seasonal changes. Such changes may be part of the dynamical system, but they may not be the relevant part, and you may not have sufficient data to resolve slowly varying trends. In such cases, you can "detrend" the data using one of the techniques available in the program such as subtracting from each data point the value obtained from a low-order polynomial or Fourier fit to the data. You can also differentiate the data (that is, do the analysis on the difference of successive data values), although differentiation tends to accentuate the noise and thus is recommended only with accurately measured, highly resolved data. Differentiation works particularly well with data generated from numerical simulations. Your goal is to get a correlation time somewhere in the range of about 1 to 10 time steps.

A less obvious non-stationarity is one in which the data change character at some point in the record. For example, a system with predominately high-frequency structure might change to one with predominately low-frequency structure. Although such behavior can occur in a dynamical system, if it occurs only a few times within the record, you should be suspicious that the record is of insufficient duration to capture these dynamics. One test for stationarity is to split the record into two halves and to verify that the quantities calculated for the first half agree with those calculated for the second half and with those calculated for the entire record.

As you analyze the sample data included with CDA PRO or your own data, look for obvious structure in the views of the graph of data. Look for structure in the phase-space plots, return maps, and Poincaré movies. Look for any sharp peaks in the probability distribution. Examine the data to see whether they can be well fit by a low-order polynomial. If structure is revealed and, for example, you find that the data are fitted sufficiently well by a polynomial or simple closed loop in phase space, there is no need to proceed further. If not, then look at the power spectrum and dominant frequencies. A power spectrum with a few dominant frequencies shows that the data can be well approximated by a Fourier series with just a few terms. In such a case, there is no need to proceed further.

When the above procedures fail to reveal simplicity, you can analyze the data under the assumption that the data represent a strange attractor or other chaotic system, possibly in some high-dimensional phase space. You first need to determine the appropriate embedding dimension in which to reconstruct the attractor. This is best done using the correlation dimension program to look for a clear saturation in the calculated correlation dimension as the embedding dimension is increased. This saturation should occur at a value of no more than about half the embedding dimension. The slope of the correlation sum should exhibit a well-defined plateau when properly embedded. Another indication of the proper embedding is the minimum dimension for which the number of false nearest neighbors falls to zero.

If you cannot find an appropriate embedding dimension, either the system is chaotic and high-dimensional or it is contaminated with noise, a distinction which is largely semantic. You can attempt to remove the noise using one of the techniques provided with the program. However, this puts you on very shaky ground unless you know enough about the system you are analyzing to be sure that there is an underlying low-dimensional dynamical process upon which the noise has been superimposed. Often the noise enters the dynamics in a more subtle way (dynamical noise, rather than measurement noise), and the two cannot be disentangled even in principle.

If you succeed in identifying a proper embedding, you can proceed to quantify the attractor. The correlation dimension and capacity dimension describe its complexity. The Lyapunov exponent provides a measure of the sensitivity of the system to its initial condition. A positive exponent implies chaos, although noise will also produce a positive exponent. A related measure is the growth in the unpredictability of the time series.

Whatever conclusions you draw about the quantitative measures of the attractor should be subjected to the surrogate-data test. The most generally useful method is to Fourier-transform the data, randomize the phases, and then inverse Fourier-transform the result to get a new time series with the same spectral properties as the original but with the determinism removed. Analysis of these surrogate data should provide values that are statistically distinct from those derived for the original data. This test is a very important one and is rarely included in papers claiming observation of low-dimensional chaos in experimental data.

When you perform the analysis on surrogate data, you will of course get different answers than you did when you analyzed your real data. You are then left with the question of whether the difference is statistically significant. One way to address this issue is to generate many surrogate data sets and to see whether the results from your data lie within the range of values corresponding to the surrogates. If they do, then the difference is not statistically significant and your data is indistinguishable from colored (correlated) noise.

If you find a statistically significant attractor dimension less than about five and a Lyapunov exponent that is decidedly positive, it is worthwhile to analyze the data using the methods of singular value decomposition in the correlation matrix program. In practice, with a limited amount of data that may be contaminated by noise, success cannot be guaranteed. This analysis attempts to represent the data in terms of a finite set of orthogonal functions. The number of functions in this set is simply the number of significant eigenvalues of the correlation matrix. Deciding how many eigenvalues are significant requires a subjective judgment, but you should remember that the goal is simplicity, which in this context means a small number of eigenvalues. At least three eigenvalues are needed to give chaos in a system described by differential equations, and in fact, the correlation matrix analysis program assumes that there are only three. Even so, 30 parameters are involved in the fit of the model equations to the data.

You might try correlation matrices with different orders, with the goal of finding a set of model equations whose solution as evidenced by the phase-space plot is a strange attractor that bears some resemblance to the phase-space plot of the original data. Because of what is in effect a change of coordinates, these phase-space plots should be distorted images of one another, but because the changes are linear, the topological features such as holes in phase space are common to both. If these qualitative tests are not satisfied then three significant eigenvalues are not sufficient, and the problem is too difficult to be analyzed by this package. If reasonable agreement between a model and the data is found, the model equations can be viewed as a résumé of the data, and their solution can be used as input for further analytic studies and short-term predictions. You should not, however, expect in general to reconstruct accurate long-term dynamics of the solution by this method. The nonlinear prediction program is better for this purpose.

DATA MANIPULATION

This program allows you to manipulate the data in certain ways prior to analysis (see Figure 2). Press A to take alternate points (every other point). Press D to differentiate the data (take difference between adjacent points). Press I to integrate the data (sum all data points from the beginning of the record to the current point). Press M to smooth the data (replace each data point with the average of itself and its two nearest neighbors). These commands can be executed repeatedly, for example, to take the second or third derivative. Note that these commands are inactive while the graph is being drawn. You can also take only the first N data points by repeatedly pressing N.

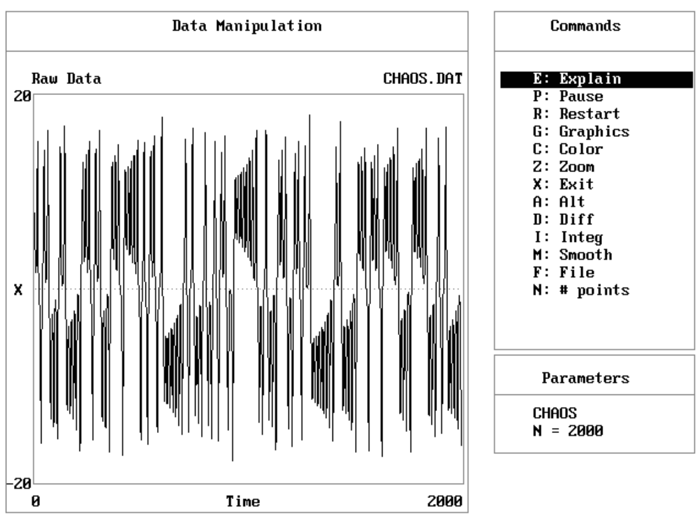

Fig. 2. The Data Manipulation program allows you to perform certain operations on the data before proceeding with the analysis.

Press R to reread the data from the disk in their original form. If you exit the program before all the data are read, only that portion that has been read will be accessible for analysis. The next time you run the program, the data will be read again. The data are plotted while being read from the disk, and thus the autoscaling will not work unless you read in all the data and then press R again to restart.

You may also read a different file by pressing F. The program will cycle among the files in the default directory that have the .DAT extension.

GRAPH OF DATA

This program has four views. In view 1, X(t) is plotted as a function of t. In view 2 (see Figure 3), the value of X(t) is plotted on the vertical axis as a function of X(t – n) on the horizontal axis. View 3 shows a perspective drawing in which X(t) is plotted along one axis, X(t – n) along the second, and X(t – 2n) along the third. View 4 is a stereo plot that allows you, with practice, to see the graph in 3-D. These plots illustrate how a multidimensional phase-space can be constructed from a time series without the necessity of taking derivatives of the data.

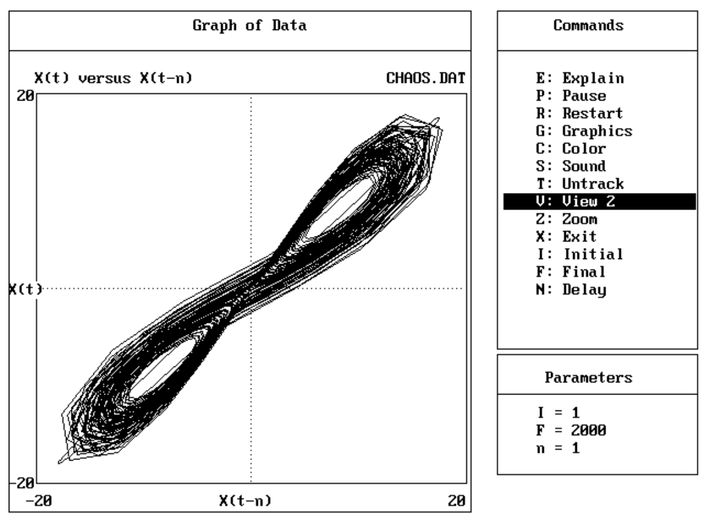

Fig. 3. The Graph of Data program allows you to display the data in a number of different ways, including plotting each data point versus its immediate predecessor.

View 1 mainly confirms that the data record was read properly by the program. The box in the lower right-hand corner gives the number of data points read. Press I or F to begin or end the plot at various times to examine a portion of the data record.

A periodic system will exhibit a closed loop, and a simple chaotic system like the logistic equation[20,21] will produce a plot with discernible structure in views 2 to 4. More complicated cases will fill two- and three-dimensional regions, respectively, with no discernible structure. A fuzzy loop means the system is nearly periodic on a long time scale.

A feature in which the sound frequency is modulated by the data is included mostly for amusement, although in some cases, structure in the data may reveal itself to the ear more easily than to the eye.

PROBABILITY DISTRIBUTION

The probability distribution shows the probability of occurrence of certain data values. You can specify a number of bins between 2 and 512 into which the data points are sorted according to their value. The bins all have the same width and are spread uniformly between the lowest and highest data value. View 1 (see Figure 4) shows a linear plot of the probability, and view 2 shows a logarithmic plot of the probability. View 3 shows the data points plotted versus rank from lowest to highest.

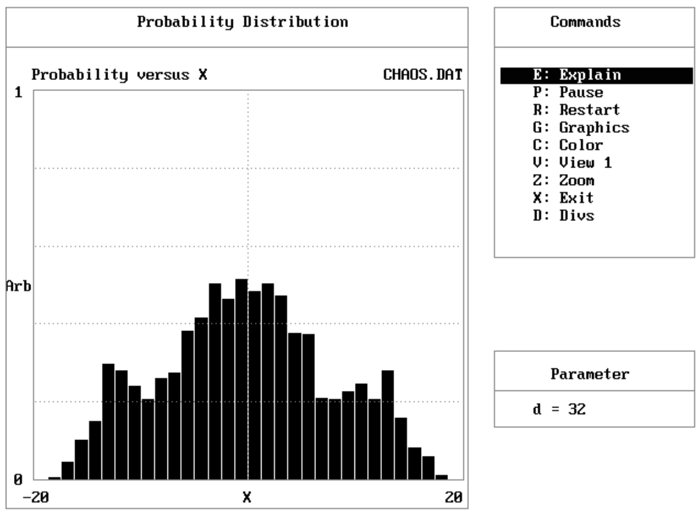

Fig. 4. The Probability Distribution program provides a histogram of the data values and performs various statistical tests on the data.

Random data (noise) will give rise to a Maxwellian distribution, which would show up as a simple curve in all views if sufficient data are available. This can also be the case for chaotic data, but it is not general. The distribution is likely to be a fractal for chaotic data. Periodic data should give a simple histogram with sharp edges.

View 4 attempts to show the fixed points by plotting the number of successive data points within a bin centered on each data point. A system that finds itself near a fixed point will tend to remain there for a long time. View 5 uses symbolic dynamics to calculate the Lempel-Ziv (LZ) complexity[22,23] relative to white noise. View 6 is an IFS clumpiness test[24,25] in which white noise fills the screen uniformly and colored noise or chaotic data produces localized clumps.

POLYNOMIAL FIT

A polynomial of order O is fitted by least-squares to the data record which is assumed to be a function of time (see Figure 5). For this case, time is measured in units such that the data record is of total duration 1.0. Polynomials up to ninth order are allowed, depending on the length of the data record and the amount of free memory. If you want a high-order fit to a record with many data points, you will have to use the A or N command in the Data Manipulation routine to reduce the number of data points to a value that can be accommodated by the memory in your computer.

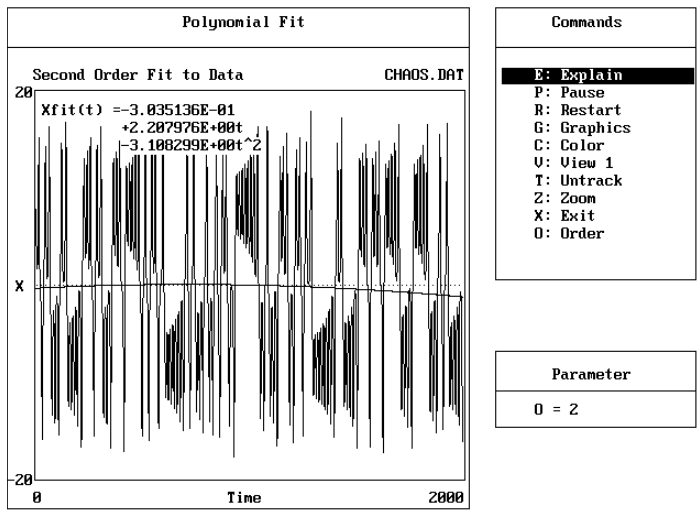

Fig. 5. The data can be fit to polynomials of various orders.

Often chaotic behavior is superimposed on some slowly varying non-chaotic function. Effective study of the chaos requires that the smooth function, which represents the slow trend, be subtracted from the original data before a detailed analysis is attempted. A polynomial fit is one way to accomplish this. View 2 shows the resulting detrended data. You should avoid using a high-order polynomial fit as part of a prediction beyond the end of the data record since the polynomial approximation usually diverges rapidly after the end of the record.

POWER SPECTRUM

This routine performs a fast Fourier transform[26] on the data record and displays the power (mean square amplitude) as a function of frequency (see Figure 6). The power can be plotted on a linear (view 1), log-linear (view 2), or log-log (view 3) scale. The maximum frequency is the Nyquist critical frequency (the reciprocal of twice the interval between the data points). The power is in arbitrary units. The parameter

N controls the number of frequency intervals. Large N improves the resolution but exacerbates the spurious responses.

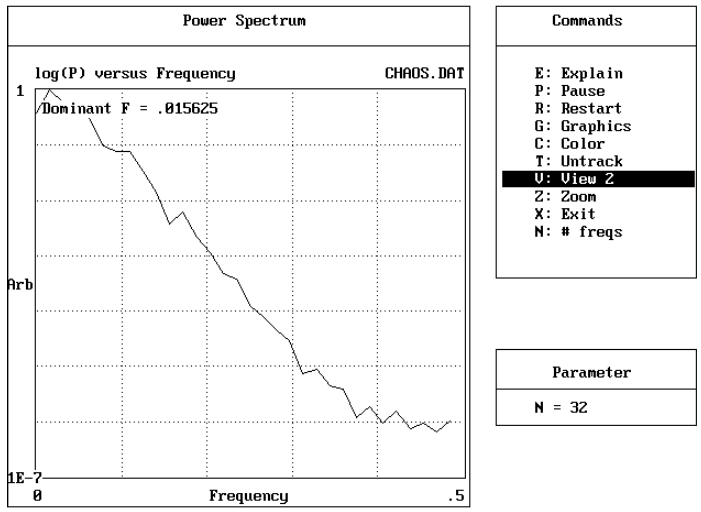

Fig. 6. A fast Fourier transform is used to plot the power spectrum on linear, log-linear, or log-log scales.

The data record is truncated at the largest integer power of 2 and calculated in non-overlapping segments using a Parzen window function[26] to reduce leakage of spectral peaks into nearby frequency intervals.

Random and chaotic data give rise to broad spectra. Periodic and quasi-periodic data will produce a few dominant peaks in the spectrum. Power spectra that are straight lines on a log-linear scale are thought to be good candidates for chaos since noise tends to have a power-law spectrum.

View 4 shows the cumulative periodogram,[27] which is the integral of the power spectrum over frequency. It should follow the 45-degree line if the power spectrum is flat indicating white noise.

DOMINANT FREQUENCIES

This routine is similar to the Power Spectrum routine in that it calculates the power spectrum and displays it in linear (view 1), log-linear (view 2), or log-log (view 3) form. The maximum frequency is the Nyquist critical frequency (the reciprocal of twice the interval between the data points). However, rather than using a fast Fourier transform method, it uses the maximum-entropy (or all poles) method.[28-30]

The maximum-entropy method represents the data in terms of a finite number (N) of complex poles of discrete frequency. By contrast, the fast Fourier transform essentially fits the power spectrum to a polynomial and is better for data whose spectrum is smoothly varying. The maximum-entropy method is good for extracting sharp, discrete lines from an otherwise noisy data record. However, it should be used with caution since it tends to generate spurious peaks at high *N* and can cause extremely sharp peaks to split at modest order.

The method can also be used to detrend the data (view 4) and to predict the next values in the time series (view 5) (see Figure 7).[31] In the former case, the most recent *N* points and the *N* poles are used to predict each next point, and the graph shows the difference between the actual and predicted values. In the latter case the last *N* points in the series are used to predict the next 18 points.

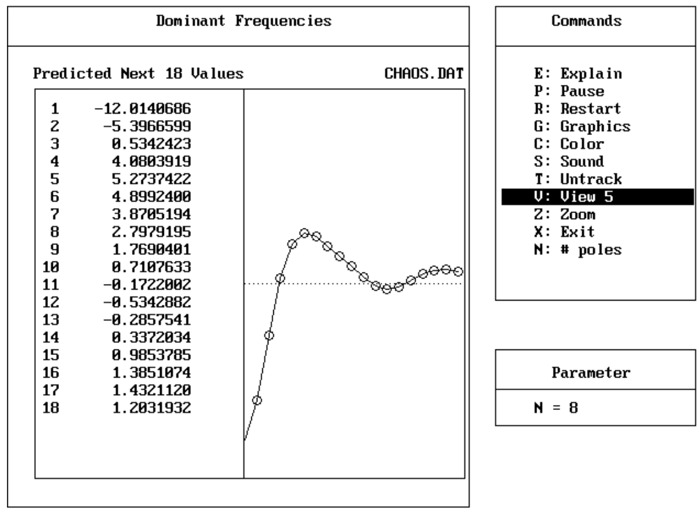

Fig. 7. The maximum entropy method is used to calculate the power spectrum from which a linear prediction of the next 18 values of the time series can be made.

HURST EXPONENT

One common type of time series arises from a random walk, sometimes called Brownian motion. In such a case, the value of *X* on average moves away from its initial position by an amount proportional to the square root of time, and we say the Hurst exponent[32] is 0.5. The root-mean-square displacement (*DX*) is plotted here versus time (tau), using each point in the time series as an initial condition (see Figure 8). The slope of this curve is the Hurst exponent.

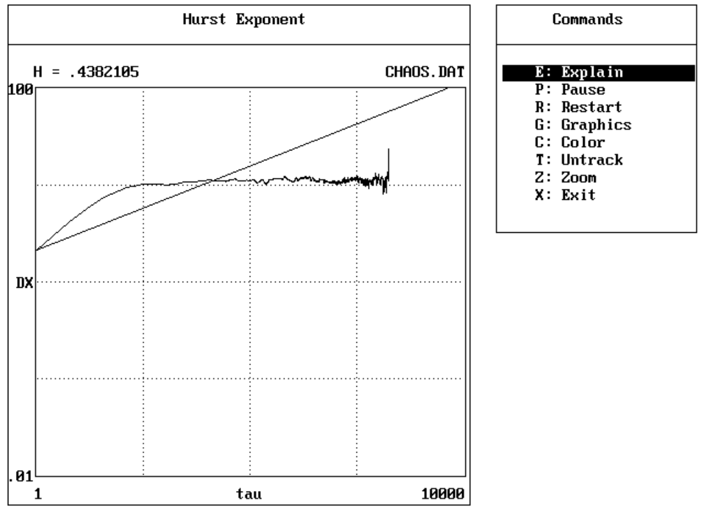

Fig. 8. The Hurst exponent is determined from the slope of the root-mean-square displacement of various initial conditions versus time.

This idea can be generalized to fractional Brownian motion (fBm), which arises from integrating correlated (colored) noise. In such a case, exponents greater than 0.5 indicate persistence (past trends persist into the future), whereas exponents less than 0.5 indicate antipersistence (past trends tend to reverse in the future). Thus, if you have data with a relatively flat power spectrum, you might integrate it (using the Data Manipulation program) and see if the exponent is close to 0.5, which would imply that it is random and uncorrelated.

For real data, the plot of *DX* versus tau seldom falls along a straight line, in which case the Hurst exponent depends upon the time scale. The program approximates the exponent by drawing a line connecting the point at tau = 1 and the point at tau equal to the square root of the total duration of the data record.

LYAPUNOV EXPONENT

The Lyapunov exponent[13,33,34] is a measure of the rate at which nearby trajectories in phase space diverge. Chaotic orbits have at least one positive Lyapunov exponent. For periodic orbits, all Lyapunov exponents are negative. The Lyapunov exponent is zero near a bifurcation. In general there are as many exponents as there are dynamical equations. Only the most positive exponent is calculated here. It is given in units of bits per data sample. Thus a value of +1 means that the separation of nearby orbits doubles on the average in the time interval between data samples. The plot (see Figure 9) shows the separation in units of the diameter of the smallest *D*-dimensional hypersphere that encloses the attractor at *n* successive time steps. The displayed error is estimated from 2.5 times the standard deviation of the slope divided by the square root of the number of trajectories followed.

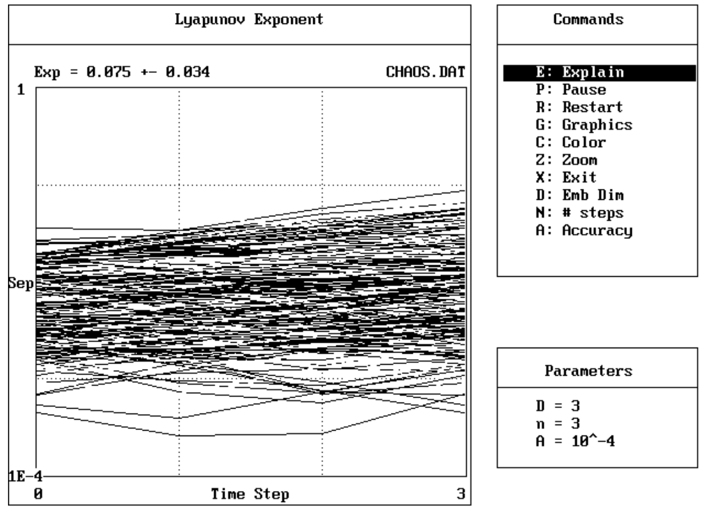

Fig. 9. The largest Lyapunov exponent is estimated by following the evolution of appropriately chosen pairs of data points.

There are three parameters under user control. The embedding dimension *D* should be set somewhat higher than the expected dimension of the attractor. The parameter *n* is the number of sample intervals over which each pair of points is followed before a new pair is chosen. If *n* is too large, the trajectories get too far apart and the exponential divergence of the orbits is lost. If *n* is too small, the calculation becomes very slow. *A* is the relative accuracy of the data below which noise is expected to dominate.

CAPACITY DIMENSION

The capacity dimension[35-37] (similar to the Hausdorff dimension) is calculated by successively dividing the phase space with embedding dimension *D* into equal hypercubes and plotting the log of the fraction of hypercubes that are occupied with data points versus the log of the normalized linear dimension of the hypercubes (view 1) (see Figure 10). The average slope of the line for the two middle segments is taken as the capacity dimension. The relative error is calculated from seven divided by the square root of the number of data points. View 2 shows the slope (capacity dimension) plotted versus the log of the normalized linear dimension of the hypercubes. The dimension is well defined only if there is a plateau region in this graph.

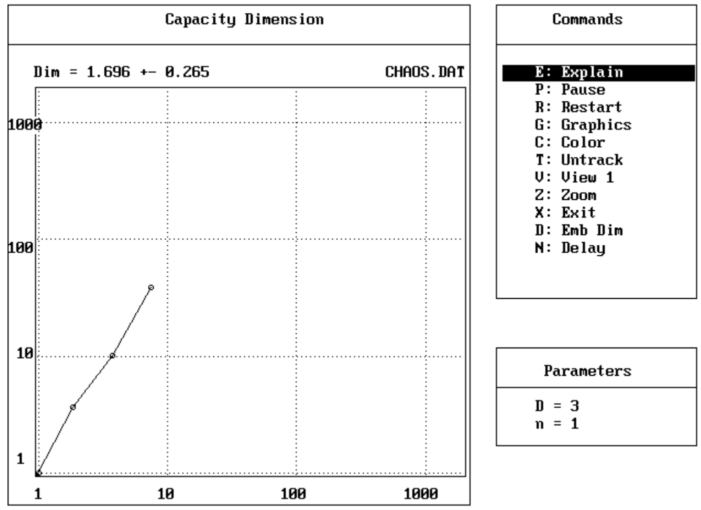

Fig. 10. The capacity dimension is calculated from the slope of the curve produced by a box-counting procedure.

View 3 shows the calculated capacity dimension for various embedding dimensions. As the embedding dimension is increased, the capacity dimension should increase but eventually saturate at the correct value. Many data points are required to get an accurate estimate of the capacity dimension if the dimension is high.[10,38,39] In such a case, the correlation dimension is preferred to the capacity dimension. A dimension greater than about five implies essentially random data.

Correlation Dimension

With each pass through the data, a new data point is taken, and a hyperdimensional sphere of embedding dimension D and radius r is centered on that point. The fraction of subsequent data points in the record within that sphere is then calculated for various values of r, and a plot is made of the log of this number versus the log of the radius (view 1). The correlation dimension[7,8] is taken as the average slope of the cumulative curve over the middle one-quarter of the vertical scale, and the error is taken as half the difference of the maximum and minimum slope over the same range. View 2 shows the slope (correlation dimension) versus the log of the radius of the spheres. View 3 shows the correlation dimension versus the embedding dimension (see Figure 11). View 4 shows the percentage of false nearest neighbors, which should fall to zero at the minimum embedding.[40,41] View 5 shows the entropy,[42-45] which is a measure of the disorder. The entropy is the sum of the positive Lyapunov exponents (base-e), and its reciprocal is roughly the time over which meaningful prediction is possible. View 6 shows the BDS statistic,[46] which is a measure of the deviation from pure randomness.

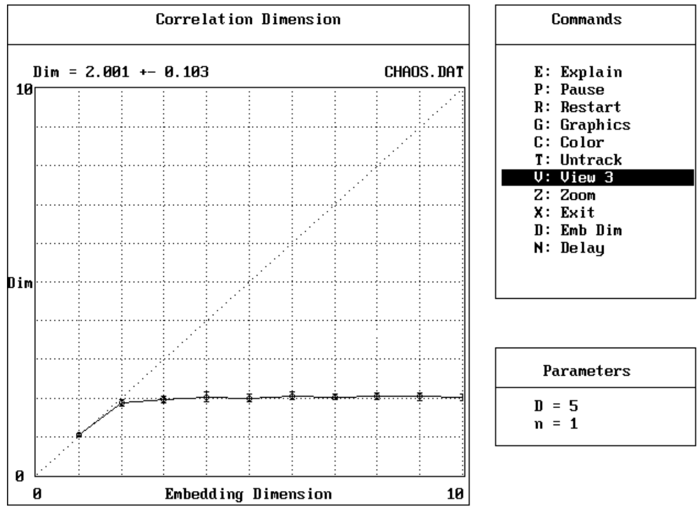

Fig. 11. The correlation dimension is calculated for various embedding dimensions.

A proper correlation dimension requires a plateau on the plot of view 2. As the embedding dimension increases, the correlation dimension should increase but eventually saturate at the correct value. The program calculates embedding dimensions between 1 and 10. A correlation dimension greater than about five implies essentially random data.

CORRELATION FUNCTION

The correlation function (see Figure 12) is obtained by multiplying each $X(t)$ by $X(t - \text{tau})$ and summing the result over all the data points. The sum is then plotted as a function of tau. This gives a measure of how dependent data points are on their temporal neighbors. The value of tau at which the correlation function first falls to $1/e$ is taken here to be the correlation time. The correlation time is generally on the order of the inverse of the Lyapunov exponent.[47,48]

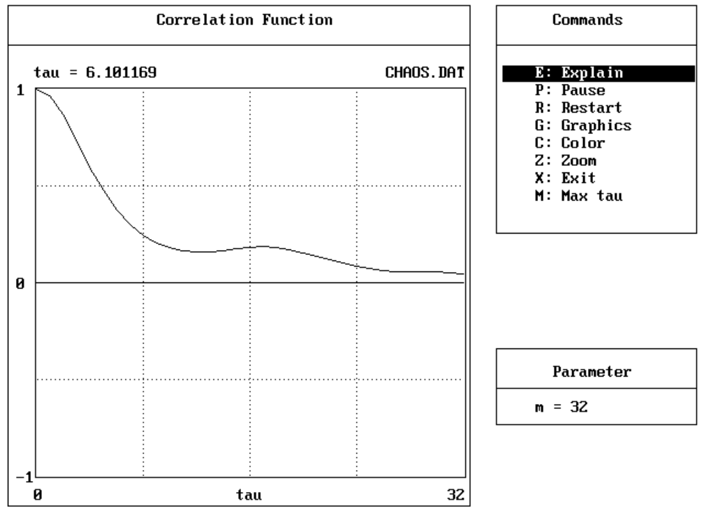

Fig. 12. The correlation function provides a measure of the dependence of the data points on points nearby in time.

Highly random data will have no correlation, and the correlation function will drop abruptly to zero, implying small correlation time. Highly correlated data like the output of a sine wave generator will have a correlation function that varies with tau but whose amplitude only slowly decreases. Chaotic data from difference equations tend to show little correlation, but chaotic data from differential equations may be highly correlated if the sample time is small, since adjacent data points have similar values.

CORRELATION MATRIX

Singular value decomposition (or principal component analysis)[19,49-51] begins with the correlation matrix, which is a two-dimensional ($M \times M$) matrix formed by putting the value of the correlation function with tau = 0 along the diagonal and putting the value of the correlation function with tau = 1 to the right and left of the diagonal and then tau = 2, and so forth until the matrix is filled. The matrix can be as large as 16 × 16.

The eigenvalues of this matrix are displayed in view 1, sorted from largest to smallest. The eigenfunctions corresponding to the three largest eigenvalues are plotted versus time in view 2 (see Figure 13). Views 3-7 show the eigenfunctions plotted versus one another in various ways. The data are then fitted to a linear superposition of the first three eigenfunctions in view 8. The last N data points in the record are fitted to a set of nonlinear model equations (view 9) whose solution is plotted in various ways in views 10-12. The model equations are used to predict the next 18 values in the time series in view 13.

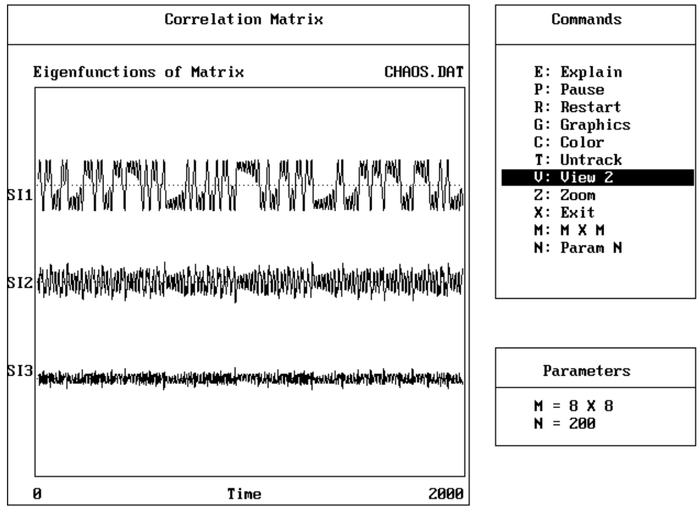

Fig. 13. The dominant eigenfunctions of the correlation matrix are used to form basis functions for nonlinear modeling of the data.

If the data consist of chaotic and random components, this procedure should help extract the chaos from the randomness. The number of significant eigenvalues (view 1), like the correlation dimension, is a measure of the complexity of the system.

PHASE-SPACE PLOTS

The phase-space plots show three views. View 1 is a two-dimensional plot in which the time derivative X' is plotted versus X at each data point (see Figure 14). For this purpose, the derivative is assumed to be given by half the difference of the two data points adjacent to each point. Periodic data should appear as a closed curve on such a plot.

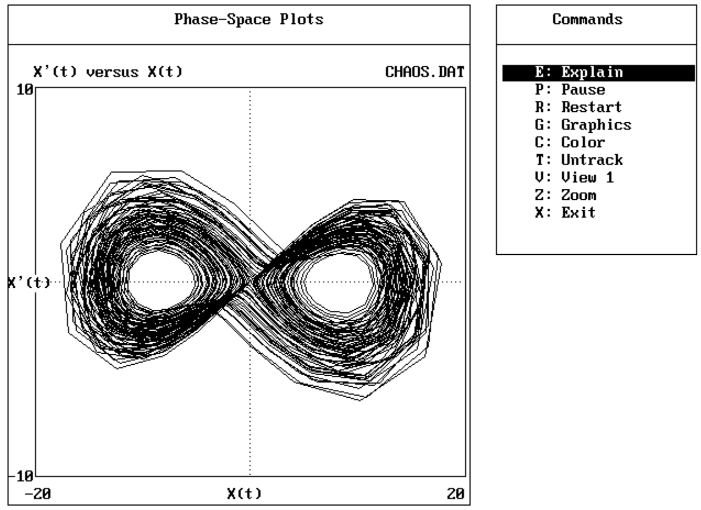

Fig. 14. Various derivatives of the data can be plotted versus the data values to obtain phase-space plots that may reveal the topology of the solution.

View 2 is a three-dimensional plot in which the second derivative *X´´* is plotted along with *X´* and *X* on the three axes of the plot. The second derivative is taken as the difference between the slopes of the lines connecting each data point with its two nearest neighbors. Some cases that are not obviously periodic in two dimensions may reveal their periodicity in three dimensions. View 3 is a stereo plot that allows you, with practice, to see the phase-space trajectory in 3-D.

The phase-space plots contain essentially the same information as views 2 through 4 of the graph of data. If no structure is evident in these plots, try looking at the return maps.

Return Maps

A two-dimensional phase-space plot generally will not distinguish between random and chaotic data. For this purpose, it is useful to take some sort of cross section of the phase plane in order to reduce its dimension by one. After such an operation, chaotic data will often appear in the form of a strange attractor having a fractal structure with fractional dimension. Several such sections are shown here.

View 1 shows the value of *X* at the time at which *X´* equals a constant versus the value of *X* at the previous time at which *X´* equaled the same constant (see Figure 15). The constant can be changed with the f parameter. View 2 shows the value of *X´* when *X* has a particular value (also controlled by *f*) versus the value of *X´* at the previous time at which *X* had the same value. View 3 is the same as view 1 except that only the maxima are plotted, and view 4 is the same as view 1 except that only the minima are plotted. In each case, random data will fill the two-dimensional plane, whereas chaotic data may yield a structure with less than two dimensions. View 5 shows the time intervals between successive crossings of a value that is a fraction *f* of the way from the minimum data value to the maximum data value.

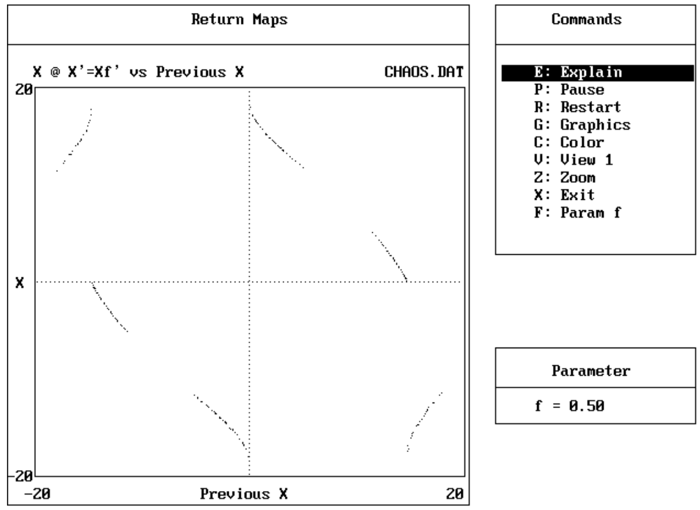

Fig. 15. The data value or its derivative is plotted whenever some particular criterion is met for the value or derivative of the data to produce a plot whose dimension is one less than the dimension of the underlying attractor.

Poincaré Movies

In this case, every *n*-th data point is plotted versus the *m*-th previous point in view 1 and versus the *m*-th and 2*m*-th previous points in view 2. View 3 is a stereo plot that allows you, with practice, to see the graph in 3-D. Then time is advanced by one unit; the points are erased; and the process is repeated. The result is a kind of phase-space trajectory sampled stroboscopically. As with a real strobe, you should adjust *n* (the strobe frequency) to try to make the pattern as stationary as possible. Best results are usually obtained with a small value of *m* but not so small that the plot collapses onto the 45° diagonal. With tracking turned on (press T), a line is drawn to show the displacement of each point from its position at the previous strobe (see Figure 16).

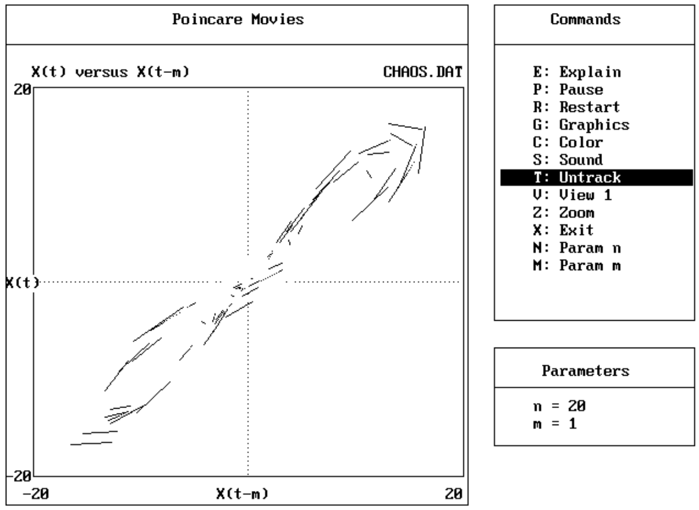

Fig. 16. The Poincaré movie shows an animated display of every n-th data point versus its m-th predecessor as if viewed under a strobe light that flashes every n-th time step.

With random data dominated by noise, no discernible pattern emerges. With periodic data, nearby points move together. With chaotic data, you should observe repeated stretching and folding of the trajectories, causing nearby points to separate. Turn on the sound to produce an audible marker after every cycle through the data.

WAVELET TRANSFORM

In the wavelet transform,[52,53] a narrow function of width $\pm DT$ centered on time T is moved through the time series and convoluted with the data. The amplitude of the convolution is plotted versus DT and T using a color scale. The transform function in view 1 (see Figure 17) is symmetric about T and positive definite, and the transform function in view 2 is antisymmetric with zero mean.

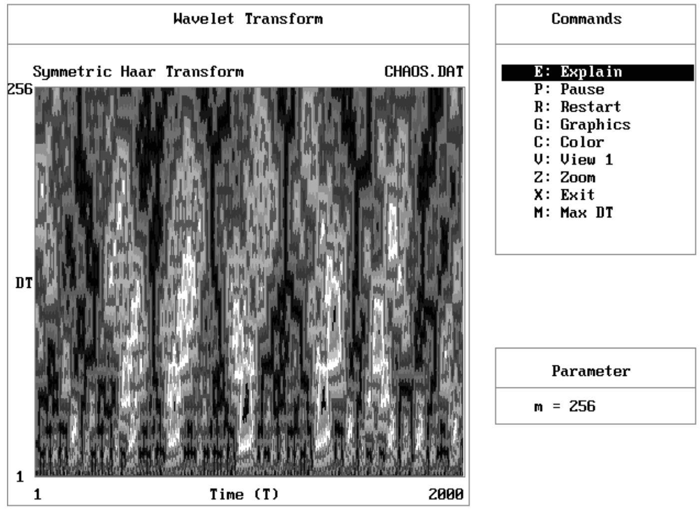

Fig. 17. The wavelet transform attempts to reveal structure in the data localized to a particular time scale and temporal position.

You can think of this process as examining the data with a microscope whose field of view (*DT*) and position (*T*) takes on all possible values. If the structure is confined primarily to a particular range of *DT*, it means most of the variation occurs on that time scale. Systematic variations with *T* can indicate lack of stationarity in the data.

Try rotating the color palette with the **+** and **-** keys to highlight features of the pattern.

NEURAL NETWORK

In this simple artificial neural network,[12,54] each term in the time series is assumed to be given by a superposition of the previous *N* terms with weights determined by a best fit to the existing data. The network "learns" by varying the weights to minimize the error. In this sense, such a network emulates the way the brain works.

Nonlinearity is introduced by "squashing" the input data using a sigmoid function that limits the values to the range -1 to 1. The squashing function used here is tanh(*X*). The network here has only two layers (input and output) with no hidden layers.

View 1 shows the percent error as the network makes its way repeatedly through the data record. View 2 shows the weights of the *N* coefficients (see Figure 18). It provides a crude measure of the determinism, since a chaotic system loses information as it advances in time. View 3 shows the long-term solution, and view 4 shows the short-term solution after the end of the data record based upon the optimized weights. This prediction method complements the others and might be more reliable for certain types of data.

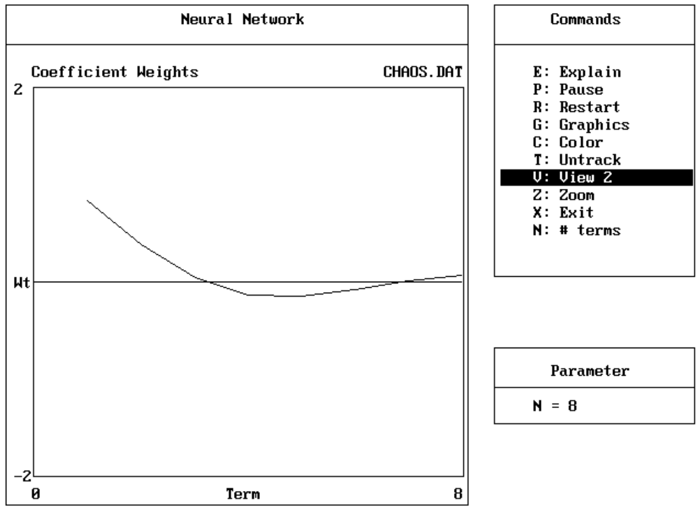

Fig. 18. The neural network provides a means for reducing noise, producing additional data, and predicting the short-term future of a deterministic time series.

NONLINEAR PREDICTION

Perhaps the best measure of determinism in a data record is the extent to which the data can be used to make accurate predictions.[55-58] One way to predict is to search the data record for the nearest *N* points in a *D*-dimensional embedding and see what they did on average for the next few time steps. In view 1, this procedure is carried out for each data point in the time series, and the average prediction error is plotted for the next 18 values (see Figure 19). The average error is expressed as a percentage of the maximum possible error (the range, X_{max} - X_{min}, of the data). For chaotic data, the error should grow exponentially and saturate at a value the order of the standard deviation divided by the range.

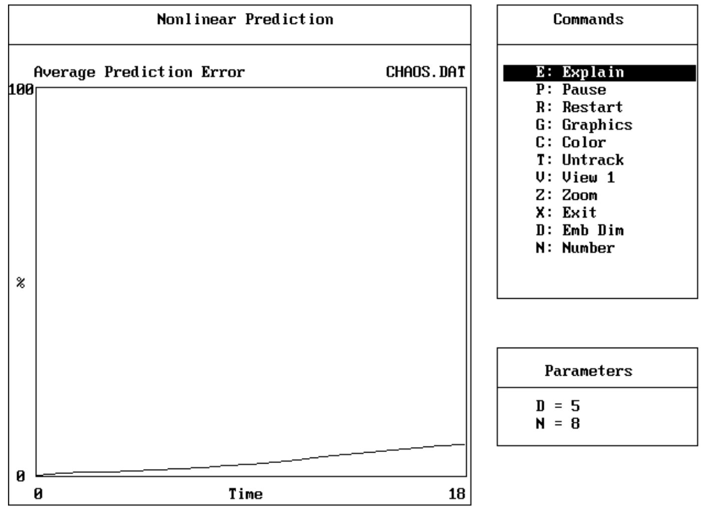

Fig. 19. Nonlinear prediction provides a measure of the degree of determinism in the data, removes noise, and allows short-term predictions.

In view 2, an attempt is made to remove noise from the data[59-63] by replacing each point in the record with a prediction from the average of the closest *N* points to the previous point. The difference between each point and the prediction is also plotted on the same scale to give an indication of the level of uncorrelated noise in the data.

In view 3, the procedure is used to predict the next 18 data points after the end of the record. You may wish to compare this nonlinear prediction with the other prediction methods included elsewhere in the program. Note that colored noise exhibits considerable determinism.[64]

SURROGATE DATA

If you find evidence of determinism in your data, it is prudent to repeat the tests using surrogate data that resemble your data but that lack determinism. If the results are the same, you should suspect the veracity of your conclusion.[65-67] This program generates two such records, both called SHUFFLED.DAT, limited to 16,384 points.

In view 1, the data values are simply shuffled, as you would shuffle a deck of cards. This operation preserves the probability distribution but generally produces a very different power spectrum and correlation function. In view 2, the time series is Fourier transformed, the phase of each Fourier component is set to a random value between 0 and 2π, and then inverse Fourier transformed (see Figure 20). This operation preserves the power spectrum and correlation function but generally produces a different probability distribution. View 3 unshuffles and restores the original data record.

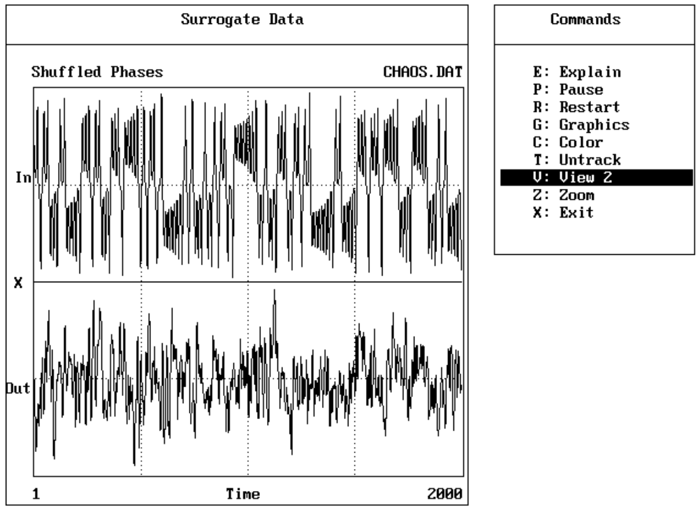

Fig. 20. Surrogate data with the same power spectrum but with the determinism removed can be produced from the original data by taking the Fourier transform, randomizing the phases, and then taking the inverse Fourier transform.

When you exit the Data Shuffle program, subsequent analysis will be on the data in the last view displayed. The shuffled data are not actually written to a disk file unless you start the program with a command such as CDA ROSSLER.DAT, SV{SHUFFLED.DAT}5X. Once you have shuffled the data and exited the Data Shuffle program, you will have to use the Data Manipulation program to recover the original unshuffled data.

SUMMARY OF DATA

When the various programs execute, they save the results of their calculations in a table that you can examine by pressing F4. Only those quantities that have been calculated and have meaningful values are displayed (see Figure 21).

Summary of Data from CHAOS.DAT:

Number of data points	2000	Relative LZ complexity	.1809354
Minimum value	-17.72337	Dominant frequency (FFT)	.015625
Average value	-.2358849	Dominant frequency (MEM)	.04
Maximum value	17.98758	Dominant period (FFT)	64
Range of data	35.71095	Dominant period (MEM)	25
Resolution	1.259995E-2	Hurst Exponent	.4382105
Lower quartile	-5.713983	Largest Lyapunov exponent	0.075±0.034
Median value	-.3580765	Lyapunov exponent (base e)	0.052±0.024
Upper quartile	5.163997	Capacity dimension (approx)	1.807±0.283
Mode (max probability)	-.625	Correlation dim (approx)	2.001±0.103
Average deviation	6.450353	Entropy (approx)	0.350
Standard deviation	7.916132	BDS statistic	.3338559
Variance	62.66514	Correlation time	6.101169
Skewness	4.912616E-2	Predicted next value (Lin)	-12.01407
Kurtosis	-.6851208	Predicted next value (SVD)	-17.19
Pearson's correlation	.9634255	Predicted next value (Net)	-13.88175
Fixed point (estimated)	-.5625651	Predicted next value (NLP)	-12.52904

Fig. 21. At the end of all the calculations, a summary is provided of the numerical values of the various quantities that characterize the data set.

The quantities are as follows:

Number of data points The number of data points N in the record being analyzed. N is an integer with a maximum value of 32,000.

Minimum value The value of the smallest (or most negative) element in the data record.

Average value The average (or mean) value of the elements in the data record:

$$X_{av} = \frac{1}{N}\sum_{i=1}^{N} X_i$$

Maximum value The value of the largest (or most positive) element in the data record.

Range of data Difference between maximum and minimum value of the data.

Resolution An estimate of the resolution of the data obtained by searching for the smallest non-zero difference between any two data points. The actual resolution may be considerably smaller than this value.

Lower quartile The value below which one quarter of the data points lie.

Median value The value for which half of the data points are above and half below.

Upper quartile The value above which one quarter of the data points lie.

Mode (max probability) The peak of the probability distribution function. This value is an approximation based on the midpoint of the most populated bin in the probability distribution calculation. The value will thus depend upon the number of bins chosen (parameter *d*).

Average deviation

$$AD = \frac{1}{N}\sum_{i=1}^{N} \left| X_i - X_{av} \right|$$

Standard deviation

$$\sigma = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N} (X_i - X_{av})^2}$$

Variance

$$Var = \frac{1}{N-1}\sum_{i=1}^{N} (X_i - X_{av})^2$$

Skewness

$$Skew = \frac{1}{N}\sum_{i=1}^{N} \left[\frac{X_i - X_{av}}{\sigma} \right]^3$$

Kurtosis

$$Kurt = \frac{1}{N}\sum_{i=1}^{N} \left[\frac{X_i - X_{av}}{\sigma} \right]^4 - 3$$

Pearson's correlation

A measure of how strongly each data point correlates with its immediate predecessor. A value of +1 implies perfect correlation, and a value of –1 implies perfect anticorrelation. This value is the first term in the correlation function $C(1)$. It is also called the linear correlation coefficient or Pearson's r.[26]

Fixed point — An estimate of the value of X at which one of the fixed (equilibrium) points occurs.

Relative LZ complexity — A measure of the algorithmic complexity of the time series. Maximal complexity (randomness) has a value of 1.0, and perfect predictability has a value of 0. In this calculation, each data point is converted to a single binary digit according to whether its value is greater than or less than the median value, and the algorithm of Kaspar and Schuster[23] is used.

Dominant frequency (FFT) — The frequency at which the power spectrum as determined by a fast Fourier transform of the data has its maximum value. The units are cycles per unit time where the unit of time is the interval between data points. If the data were sampled every second, the frequency would be in Hertz.

Dominant frequency (MEM) — The dominant frequency as determined by the maximum-entropy method. It is usually more accurate than the value determined by the FFT. The units are the same as above.

Dominant period (FFT) — The inverse of the dominant frequency from the fast Fourier transform.

Dominant period (MEM) — The inverse of the dominant frequency from the maximum-entropy method.

Hurst exponent — A measure of the extent to which the data can be represented by a random walk (fractional Brownian motion). The integral of white (uncorrelated) noise corresponds to ordinary Brownian motion and has a Hurst exponent of 0.5.

Largest Lyapunov exponent — An estimate of the rate at which similar trajectories in phase space diverge. The exponent should be zero or negative for periodic data, positive for chaotic data, and infinite for uncorrelated random data. The units are bits (factors of two) per unit time, where the unit of time is the interval between data points. The algorithm is due to Wolf, *et al*.[13]

Largest Lyapunov exponent (base e) — An estimate as explained above, but based on a natural logarithm rather than a base-2 logarithm.

Capacity dimension — An estimate of the fractional dimension of any attractor that results when the data are plotted in a reconstructed phase space with some high embedding dimension. A value is provided only when the calculated capacity dimension is less than half the embedding dimension for some embedding.

Correlation dimension — An estimate of the fractional dimension obtained by counting the data points inside hyperspheres of various radii centered on each data point in a reconstructed phase space with some high embedding dimension. The value should be close to the capacity dimension above, but is generally a more accurate measure. A value is provided only when the calculated correlation dimension is less than half the embedding dimension for some embedding. The algorithm is a variation of the one by Grassberger and Procaccia.[7]

Entropy The sum of the positive Lyapunov exponents (base e). The entropy is a measure of the disorder in the data. The algorithm employed is due to Grassberger and Procaccia.[42]

BDS statistic A measure of the deviation of the data from pure randomness. For an interpretation of the meaning of this measure, consult Brock.[46]

Correlation time A measure of the time over which the data correlates with preceding data. It is taken as the time at which the correlation function first falls to $1/e$ of its fully correlated value.

Predicted next value (Lin) A linear prediction of the next value expected in the time series from the dominant frequencies. This prediction is most accurate when the data consists of a linear superposition of a small number of discrete Fourier modes. The algorithm is due to Press, *et al.*[26]

Predicted next value (SVD) A prediction of the next value expected in the time series from singular value decomposition. This prediction comes from solving a simple set of three model nonlinear equations whose variables are a set of orthogonal basis functions determined optimally from the correlation matrix and whose coefficients (30 in all) are fit to the data. The prediction is reasonable for periodic and simple chaotic functions but comes with no guarantee for real-world data. For more detail, consult Rowlands and Sprott.[51]

Predicted next value (Net) A prediction of the next value from an artificial neural network that is trained on the data.

Predicted next value (NLP) A nonlinear prediction of the next value based on the average next value of similar sequences of data points.

Some of the values have an indicated error tolerance. You should not take these tolerances too literally; rather they serve as a warning against misplaced trust in the exact numerical values. The analysis of chaotic data is far from an exact science.[68-70]

Appendix A: Technical Information

Command-line Options

The **CDA** command will accept on the command line the name of a data file to be analyzed followed by a comma and then a string of characters representing a sequence of keystrokes to be executed automatically when execution begins. This string may contain single-digit numbers that will cause a delay for a corresponding number of seconds. For example, typing **CDA ,A9XB9XC9X** at the DOS prompt will read the default file CHAOS.DAT and will then cycle through the first three programs spending nine seconds on each. You can produce a longer delay by using multiple digits (e.g., 99 gives an 18-second delay). You can display text embedded in quotes. Try a command such as **CDA ,"Let's look at the Lorenz attractor..."3M**.

You can execute DOS commands or user-supplied .BAT, .EXE, or .COM files by embedding them in the command string enclosed within brackets [ ], but external commands require that COMMAND.COM be in the current path.

You can also save the output of a calculation to a disk file by putting a filename in braces on the command line. For example, **CDA ,D{POLYFIT.DAT}** will save the polynomial fit to the data in CHAOS.DAT in the file POLYFIT.DAT. To save the output of the last calculation performed before exiting to DOS, place the filename in braces as the first (or only) item in the command string, such as **CDA ,{RECENT.DAT}**. The next time you start CDA Pro, you can type **CDA RECENT** at the DOS prompt to analyze the file just created.

Several other special commands are supported as follows:

-	waits until space bar is pressed
;	dissolves the screen display
?	initiates the help feature F1
$	displays the disk directory F2
@	displays the program logo F3
!	displays a summary of the data F4
%	executes commands in AUTO.TXT F5
^	prints the screen *<Print Screen>*
.	returns to DOS *<Esc>*

You can cycle backward through the parameters, views, graphics modes, and colors by holding down the *<Shift>* key or by using lower-case letters in the automatic command sequence. The + and - keys change the color palette if your hardware supports it.

Automatic Mode

Placing an R at the end of the command string causes the commands to repeat indefinitely. The special command **CDA ,R** (equivalent to **CDA ,%R**) activates the automatic continuous analysis mode. You can interrupt automatic operation by pressing any key or mouse button, but the cycle will resume after 45 seconds of keyboard and mouse inactivity. You can single-step through the automatic sequence using *<Tab>*. To exit the automatic continuous mode, press *<Ctrl><Home>* and then release them and quickly press *<Esc>*. The diskette may be removed from the drive after the program has begun cycling. The commands performed in the automatic mode are recorded in a file called AUTO.TXT, which you can modify with any word processor that will read and write ASCII files.

You can cause the program to take its commands from a different file that you provide by starting CDA PRO with a command such as **CDA , (MYSCRIPT.TXT)**.

Program Files

CDA.EXE The CDA PRO main program.

AUTO.TXT An ASCII file that controls the automatic program sequence. You can modify this file with most word processors.

EXPLAIN.TXT An ASCII file that contains the text for the explanation screens. You can modify this file with most word processors.

HELP.TXT An ASCII text file containing the text displayed when 1 is pressed.

TUTORIAL.COM A tutorial program that analyzes in turn several of the sample data files included on the disk. Follow the description in Appendix B and press the space bar to advance to each successive screen.

Data Files

ASCIITXT.DAT A file of 1,908 ASCII codes from the opening paragraphs of Gleick's *Chaos: Making a New Science*.[71] This record illustrates one possible way literature might be analyzed and studied using CDA PRO.[72]

BROWNIAN.DAT A file of 2,000 data points from the integral of Gaussian white noise. This record illustrates a one-dimensional random walk (Brownian motion). It should have a $1/f^2$ power spectrum and a Hurst exponent of 0.5.

CANTOR.DAT A file of 2,000 data points randomly selected from a triadic Cantor set on the unit interval ($0 < X < 1$). A graph of such a Cantor set should have a fractal dimension of log(2)/log(3) ≈ 0.631 in a one-dimensional embedding.

CHAOS.DAT A file of 2,000 data points for $X(t)$ at intervals of $\Delta t = 0.05$ for the Lorenz attractor[1,2] $dX/dt = 10(Y - X)$, $dY/dt = 28X$ -

Y - XZ, $dZ/dt = XY - 8Z/3$. The Lorenz attractor is a prototypical and historically important example of a dissipative chaotic flow.

DEVIL.DAT A file of 2,000 data points obtained by integrating the triadic Cantor set of ordered points on the unit interval. The resulting time series is called the Devil's staircase. It has an infinite number of steps but rises only a finite height.

EEG.DAT A file of 2,000 data points representing eight seconds of an electroencephalogram from a 19-year-old, healthy male subject, seated and relaxed with eyes closed, sampled at 250 Hz with electrodes on the left front of the head. Units are 0.1 microvolts. The methods employed in CDA PRO might be of use in detecting neurological disorders[73] (data courtesy of Richard Davidson, University of Wisconsin–Madison).

EKG.DAT A file of 2,000 data points representing about 27 minutes of an electrocardiogram from a 44-year-old, healthy, sleeping male subject. The data represent the interval between successive heartbeats in units of milliseconds. The methods employed in CDA PRO might be of use in detecting cardiac disorders (data courtesy of J. Kenneth Ford, MD, FACC, Western Kentucky Cardiology Consultants, Paducah, KY).

ELNINO.DAT A file of 2,000 data points representing the daily average ocean temperature in degrees Celsius between December 24, 1988 and July 5, 1994 at a depth of 1 meter obtained from a buoy moored in the eastern Pacific on the equator at a longitude of 110° W. Occasional El Niño conditions occur in which a shift in the trade winds alters the tropical Pacific ocean temperature and produces disruptive climatic changes. Some people think these changes are chaotic and thus might be predictable (data courtesy of Dr. Michael J. McPhaden, Director TOGA-TAO Project Office, National Oceanic and Atmospheric Administration, Pacific Marine Environmental Laboratory).

EXPDEV.DAT A file of 2,000 values of the negative natural logarithm of a sequence of random deviates, uniformly distributed over the unit interval (0 to 1). The values are thus distributed with an exp(-X) distribution over the interval $0<X<\infty$. Such a distribution would represent, for example, the time intervals between successive clicks of a Geiger counter or other similar Poisson process.

FEIGEN.DAT A file of 2,000 iterates of the logistic map[20,21] at the Feigenbaum limit $X_{n+1} = 3.5699456X_n(1 - X_n)$. The iterates of this map form a one-dimensional Cantor set at the transition between periodic and chaotic behavior.

HENOISE.DAT A file of 2,000 iterates of X_n from the Hénon map[74] $X_{n+1} = 1 - 1.4X_n^2 + Y_n$, $Y_{n+1} = 0.3X_n$ with 10% Gaussian white noise superimposed. This is a standard example used in noise reduction tests because the extent of the noise reduction is readily apparent in the plots.

HENON.DAT A file of 2,000 iterates of X_{n+1} from the Hénon map[74] $X_{n+1} = 1 - 1.4X_n^2 + Y_n$, $Y_{n+1} = 0.3X_n$. The Hénon map is a two-dimensional generalization of the logistic map. It provides a good calibration of the largest Lyapunov exponent (0.603 bits per iteration[13]), capacity dimension (1.26), and correlation dimension (1.21).[7]

HENROUND.DAT A file of 2,000 iterates of X_n from the Hénon map[74] $X_{n+1} = 1 - 1.4X_n^2 + Y_n$, $Y_{n+1} = 0.3X_n$, but rounded to the nearest multiple of 0.025 (about 1% the range of X). This record illustrates the effect of rounding on experimental data, for example, by an analog-to-digital converter or by the finite-bit representation of the numbers in the computer.

HUMANDNA.DAT A file of 2,000 data points derived from a short segment of the human DNA molecule by letting the orbit walk a unit step in one of four directions (up, down, right, and left) depending upon the next base pair (A ,C, G, and T) in the sequence. The data values represent the magnitude of the displacement from the initial value. If the bases occurred equally and independently, the trajectory would execute a random walk and the Hurst exponent would be 0.5. This is one of many ways one might look for structure in the DNA molecule (data courtesy of Carolyn Barton, Escagenetics, Corp.).

HUMANRND.DAT A file of 2,000 data points representing angles in degrees (0 to 359) around the circumference of a circle produced by a 32-year-old, female psychologist who was attempting to generate a random sequence by clicking the mouse at points on the circle (data courtesy of Deborah Aks, University of Wisconsin-Whitewater).

IKEDA.DAT A file of 2,000 iterates of the real part of the complex variable Z from the Ikeda attractor[75] $Z_{n+1} = 1 + 0.9Z_n \exp[5.6i/(1 + |Z_n|^2)]$. The Ikeda map is a two-dimensional map that arises from a model of a laser.

INVRAND.DAT A file of 2,000 values of the inverse of a sequence of random deviates, uniformly distributed over the unit interval (0 to 1). The values are thus distributed in a highly nonuniform manner over the interval $1<X<\infty$.

LASER.DAT A file of 2,000 data points of the fluctuations in a far-infrared laser, approximately described by three coupled nonlinear ordinary differential equations from the Santa Fe time series competition[68] (data courtesy of Andreas Weigend, University of Colorado).

LCGX.DAT A file of 2,000 data points obtained by dividing by 2048 the successive iterates of the map $X_{n+1} = (157X_n + 1) \bmod 2048$. Such a linear congruential generator[76] is an example of a rather poor pseudo-random number generator typical of those found in primative compilers in the early days of computers.

LOGIT.DAT A file of 2,000 data points obtained from the logistic map[20,21] $X_{n+1} = 4X_n(1 - X_n)$ with the values of X transformed by the function $\ln[X/(1 - X)]$. The values obtained from such a logit transform closely resemble a Gaussian distribution.

LOGMAP.DAT

A file of 2,000 iterates of the logistic map[20,21] $X_{n+1} = 4X_n(1 - X_n)$. The logistic map is a simple one-dimensional chaotic map of interest in population dynamic studies.

LORENZXZ.DAT A file of 2,000 data points at intervals of $\Delta t = 0.1$ for the Lorenz attractor[1,2] $dX/dt = 10(Y - X)$, $dY/dt = 28X - Y - XZ$, $dZ/dt = XY - 8Z/3$ in which simultaneous values of X and Z have been alternately entered into the time series. This example illustrates one possible way to handle multivariate data with CDA PRO. Compare the properties of this time series with the file CHAOS.DAT in which only the X values are included over the same time span.

LOZIMAP.DAT A file of 2,000 iterates of X_n from the Lozi map[77] $X_{n+1} = 1 - 1.7|X_n| + Y_n$, $Y_{n+1} = 0.5X_n$. The Lozi map is a piecewise linear variation of the Hénon map in which the absolute value takes the place of the square.

MACKEY.DAT A file of 2,000 data points from the Mackey-Glass equation[78,79] $dX/dt = aX(t - \tau)/[1 + X(t - \tau)^c] - bX(t)$ with $a = 0.2$, $b = 0.1$, $c = 10$, and $\tau = 100$ at intervals of $\Delta t = 6$. The equation is solved by approximating it as a 3,200-dimension iterated map. [80] The Mackey-Glass equation is a prototypical delay differential equation whose state space is infinite-dimensional but whose attractor has finite dimension.

MSNTEMP.DAT A file of 3,653 values of the mean daily temperature (in °F) in Madison, WI, for the period January 1, 1980 to December 31, 1989. Evidence for simple determinism (beyond the obvious annual cycle) in such data would be of considerable interest to meteorologists (data courtesy of Pam Knox, State of Wisconsin climatologist).

NOISE.DAT A file of 2,000 random numbers with a Gaussian (normal) distribution[26] centered on zero and a standard deviation of 1.0. Such a record illustrates what you would expect for data lacking any deterministic features.

PINKNOIS.DAT A file of 2,000 random numbers with a Gaussian distribution centered on zero with a $1/f$ power spectrum. Such a time record is called $1/f$ noise or pink noise and is very common in nature. It is an example of fractional Brownian motion.

PLASMA.DAT A file of 2,000 data points sampled at 200 kHz, representing 10 milliseconds of raw signal from a magnetic loop (dB/dt) near the surface of the MST reversed field pinch plasma confinement device. [81] The literature is filled with conflicting claims about the existence of low-dimensional chaos in plasma experiments (data courtesy of Christopher Watts, University of Wisconsin–Madison).

RANDOM.DAT A file of 2,000 data points uniformly and randomly distributed over the unit interval (0 to 1). The authors of the algorithm[82] used to produce the data have offered $1,000 to anyone who can convince them that the sequence produced by their routine is not random. Before you try this, you may want to generate a sequence longer than 2,000 points.

ROSSLER.DAT

A file of 2,000 data points for $X(t)$ at intervals of $\Delta t = 0.1$ for the Rössler attractor[83] $dX/dt = -Y - Z$, $dY/dt = X + 0.2Y$, $dZ/dt = 0.2 + Z(X - 5.7)$. The Rössler attractor is a simple and historically important example of a dissipative chaotic flow.

ROSSLERH.DAT A file of 2,000 data points for $X(t)$ at intervals of $\Delta t = 0.1$ for the Rössler hyperchaotic attractor[84] $dX/dt = -Y - Z$, $dY/dt = X + 0.25Y + W$, $dZ/dt = 3 + XZ$, $dW/dt = -0.5Z + 0.05W$. The Rössler hyperchaotic attractor is an early example of a chaotic flow with two positive Lyapunov exponents.

S&P500.DAT A file of 3,000 values of the daily percentage change in the Standard & Poor's Composite Index of 500 Stocks leading up to the 20% drop that occurred on October 19, 1987. Predictions based on such data carry no guarantee, but any profits obtained with these techniques should be shared with the program author (data courtesy of Blake LeBaron, University of Wisconsin–Madison).

SHIFTMAP.DAT A file of 2,000 iterates of the binary shift map $X_{n+1} = 1.9999999999999X_n \bmod 1$. This chaotic map produces uniformly distributed values of X over the unit interval (0 to 1). The use of 1.999... prevents spurious numerical attraction to the unstable fixed point at $X = 0$.

SINE.DAT A file of 2,000 points from $\sin(t/10)$ at integer values of t from 0 to 1999. This record illustrates how a simple periodic function would appear.

SINOISE.DAT A file of 2,000 points from $\sin(t/10)$ at integer value of t from 0 to 1999 with 10% Gaussian white noise superimposed. This case can serve as a test for extracting a periodic signal contaminated by measurement noise.

SPEECH.DAT A file of 2,000 points representing a speech waveform digitized at 8 kHz from a male adult uttering the "ee" vowel sound. The methods employed in CDA PRO might be of use in detecting speech pathologies (data courtesy of Joel MacAuslan, AudioFile, Inc.).

SPROTTA.DAT A file of 2,000 points for $X(t)$ at intervals of $\Delta t = 0.5$ from the simplest known example of a conservative (Hamiltonian) system of ordinary differential equations with quadratic nonlinearities that exhibits chaos[85] $dX/dt = Y$, $dY/dt = -X + YZ$, $dZ/dt = 1 - Y^2$ with an initial condition of (0, 5, 0) This case is known as the Nosé-Hoover oscillator.[86-88] It has a largest Lyapunov exponent (base e) of 0.014 and a Lyapunov dimension of 3.0.

SPROTTB.DAT A file of 2,000 points for $X(t)$ at intervals of $\Delta t = 0.2$ from one of the simplest known examples of a dissipative system of ordinary differential equations with quadratic nonlinearities that exhibits chaos[85] $dX/dt = YZ$, $dY/dt = X - Y$, $dZ/dt = 1 - XY$. This case resembles the Lorenz attractor but is algebraically simpler. It has a largest Lyapunov exponent (base e) of 0.210 and a Lyapunov dimension of 2.174.

STAR.DAT

A file of 2,000 data points representing the temporal intensity variation of a variable white dwarf star from the Santa Fe time series competition.[68] The intensity variation arises from a superposition of relatively independent spherical harmonic multiplets, and there is significant observational noise (data courtesy of Andreas Weigend, University of Colorado).

TENTMAP.DAT A file of 2,000 iterates of the tent map $X_{n+1} = 0.9999999999999999 - 2|X_n|$. This chaotic map produces uniformly distributed values of X over the unit interval (–1 to 1). The use of 0.999... prevents spurious numerical attraction to the unstable fixed point at $X = -1$.

THREESIN.DAT A file of 2,000 points from $\sin(t/2) + \cos(gt/2) + \sin(ht/2 + \pi/4)$ at integer values of t from 0 to 1999 where $g = (\sqrt{5} - 1)/2$ is the inverse of the golden mean and $h = \sqrt{2} - 1$ is the silver mean. The resulting orbit lies on a 3-torus. This example illustrates how a function composed of a small number of periodic terms with non-integrally related frequencies can appear random until subjected to the tests included in this package.

TWOSINE.DAT A file of 2,000 points from $\sin(t/2) + \cos(gt/2)$ at integer values of t from 0 to 1999 where $g = (\sqrt{5} - 1)/2$ is the inverse of the golden mean. The resulting orbit lies on a 2-torus. This example illustrates how a function composed of a small number of periodic terms with non-integrally related frequencies can appear random until subjected to the tests included in this package.

VANDPOL.DAT A file of 2,000 points for $X(t)$ from the Van der Pol equation[89] $d^2X/dt^2 = (1 - X^2)Y - X$ at time intervals of 0.005. This system is typical of a limit cycle whose power spectrum contains a single dominant Fourier frequency component with odd harmonics.

WSTRASS.DAT A file of 2,000 points from the sum for $k = 1$ to 60 of the function $\cos(2t^k)/2^{k/2}$ at equally spaced intervals from $t = 0$ to 2π. This is an approximation to the Weierstrass function,[90,91] which in the limit of $k = 1$ to ∞ is a self-similar analytic function that is everywhere continuous but nowhere differentiable and with a fractal dimension of 1.5. Note that 1.5 is the fractal dimension of the graph of $X(t)$ and not the dimension of the reconstructed time series in a multi-dimensional time-delayed embedding.

APPENDIX B: TUTORIAL

Time-series analysis is in many ways more art than science; methods for quantifying chaos constitute a considerable body of current research. CDA PRO attempts to automate many of these procedures, but the validity and usefulness of the results depend strongly on the skill of the user, especially for small data sets such as the samples included with this package.

This tutorial guides you through a number of typical examples for which data files are provided. We will describe the analysis of each case as if the source of the data were unknown, and our task is to learn as much as we can about the underlying system that produced the data.

The keystrokes required for each test are included in brackets. To execute the tutorial, enter the command in the heading of each example at the DOS prompt. Then enter each keystroke exactly as described as you read along. Alternately, from the DOS prompt, you can type **TUTORIAL** and press *<Enter>*. Then use the space bar to advance to the next screen as you read along. Be careful to wait for each calculation to finish before pressing the next key. To exit the tutorial at any time, rapidly press *<Ctrl>* C several times until the DOS prompt appears. You may then reenter the tutorial by typing the command in brackets in the heading for the case you wish to examine.

Sine Wave [CDA SINE]

The graph of the data [B] and phase-space plot [XM] indicate that the function is periodic. Since it has only a single dominant frequency [XF], the function is sinusoidal. The indicated frequency [X F4] is 0.016 cycles per time step. (The actual value is $1/20\pi = 0.01592$.)

Two Incommensurate Sine Waves [CDA TWOSINE]

The graph of the data [B] and phase-space plot [XM] show no simple periodicity. However, there appear to be two dominant frequencies [XF] having narrow peaks that are not obviously in the ratio of simple integers. Thus we conclude that the function is quasi-periodic and that the phase-space trajectory lies on the surface of a torus. The return map [XN] shows a cross section of the torus. This result is confirmed by the periodic Poincaré movie [XO] when the parameters are adjusted, for example, to n = 22 and m = 3 [NNMM]. Motion on the surface of a torus implies a two-dimensional trajectory as confirmed by the correlation dimension [XJ].

Logistic Map [CDA LOGMAP]

The graph of the data [B] shows no obvious structure, but if each data point is plotted versus the previous point [VT], a simple one-dimensional map is evident. The structure remains one-dimensional if plotted in a three-dimensional embedding [V]. The correlation matrix [XL] can be used to determine the equation of the map. Since the map is one-dimensional, only a single eigenvalue (M=1) is required [MMMMMMMMM]. The corresponding eigenfunction ψ_1 is seen to be a perfect fit to the data [VVVVVVV]. The coefficients of the model equation are easily found [V], and they correspond to $X(t) = 4X(t-1) - 4X(t-1)^2$, which is exactly the equation used to generate the data. The solution of the model equation [VVV] initially agrees with the original data to within round-off error. The sensitivity to initial conditions is clearly evident as the solution of the model equations departs from the data after about a dozen time steps. Final confirmation comes from the correlation dimension [XJ] and the Lyapunov exponent [XH], both of which should be exactly 1.0.

Hénon Map [CDA HENON]

As with the logistic map, the graph of the data [B] shows no structure, but if each point is plotted versus the previous point [VT], a fractal pattern is evident. Thus we are probably dealing with a two-dimensional map. (The dimension of the underlying system of equations is at least the next integer higher than the fractal dimension.) The correlation dimension [XJV] does indeed have a value between 1 and 2. (The accepted value for the correlation dimension of the Hénon map with these parameters is 1.21.) The correlation matrix [XL] can be used to determine a set of equations for the map. In this case, we need to take M=2 [MMMMMMMMMM] since the underlying map is two-dimensional. The plot of ψ_2 versus ψ_1 [VVT] reconstructs the attractor except rotated through 45°. The data fit a simple superposition of $\psi_1 + \psi_2$ [VVVVVT]. The coefficients of the model equations [V] are accurately determined, and the solution of the model equations [VVV] initially agrees well with the data. If the solution to the model equations is hard to distinguish from the actual white line and you have a color or grayscale monitor, try changing the color [C] a few times.

Lorenz Attractor [CDA CHAOS]

The graph of the data [B] strongly suggests chaotic behavior. There is an obvious underlying structure, but it is not simply periodic. The plot of each point versus the previous point [V] shows a smooth variation, indicating that differential equations rather than difference equations govern the behavior. This is confirmed by the correlation function [XK], which falls slowly to zero. (Compare with LOGMAP.DAT which falls abruptly to zero.) The return map [XN] is nearly one-dimensional, which means that the attractor has a dimension close to 2.0. (The dimension of the return map is one lower than the dimension of the attractor.) The capacity dimension [XI] and correlation dimension [XJ] are indeed close to 2.0. (The value expected for the Lorenz attractor is about 2.05.) At least three first-order differential equations are required for a chaotic solution. In this case there are only three equations, as evidenced by the three non-zero eigenvalues of the correlation matrix [XL]. However, a dimension close to 2.0 could also indicate a quasi-periodic solution as with TWOSINE.DAT. This possibility is ruled out by noting that the power spectrum [XEV] is very broad and that there is at most a single (very low) dominant frequency [XFV].

White Noise [CDA NOISE]

The graph of the data [B] and the plot of each point versus the previous point in either two dimensions [VT] or three dimensions [V] show no structure. Similarly, the phase-space plots [XMT] and return map [XN] show no structure. The data do not result from a small number of discrete modes as evidenced by the broad power spectrum [XE] and the absence of any isolated dominant frequencies [XF]. The probability distribution [XC] looks Gaussian, and the correlation matrix [XL] shows many large eigenvalues. The correlation function [XK] falls abruptly to near zero and remains there, and the correlation dimension [XJVV] is on the order of the embedding dimension *D* for all values of *D*. (A truly random uncorrelated process should have an infinite dimension.) These are all indications of random rather than chaotic data. The analysis techniques provided here have little to offer in such a case.

Mean Daily Temperatures [CDA MSNTEMP]

In this example, the challenge is to predict the future by extrapolating from existing data. The raw data [A] shows that (especially in Madison, Wisconsin) it is colder in the winter than in the summer. You could fit a sine wave to the data and remove the seasonal variation, but we are interested in short-term predictions. We might therefore take the derivative [D]. Thus each data point now represents the change in temperature (in degrees Fahrenheit) from the previous day. The new series has no obvious structure [XBVT] apart from the graininess which results from rounding the temperatures to the nearest degree. The dominant frequency [XF] is about 0.2, which implies a five-day period (perhaps the average time for successive fronts to move through the area), though the peak is very broad. You can make a linear prediction of the temperature change for the next 18 days [VVVV], but the result is close to zero because the power spectrum is so broad. The correlation dimension [XJ] is large, and so no simple set of equations would describe the daily variation. Not to be discouraged, you might look at the correlation matrix [XL]. There are many large eigenvalues, suggesting noisy data. However, taking the largest three, fitting the data to the eigenfunctions [VVVVVVV], and calculating the coefficients of the model equations [V] gives a nonlinear prediction [VVVV] of the temperature change for the next 18 days. You may wish to test your predictions against what actually happened. Remember that the values shown are temperature *changes*. The temperature on the last day of the data record was 24, and the next 18 days actually had the following mean temperatures (change from the previous day is in parenthesis): 25(1), 26(1), 36(10), 25(–11), 19(–6), 18(–1), 32(14), 29(–3), 37(8), 28(–9), 38(10), 24(–14), 20(–4), 30(10), 25(–5), 34(9), 42(8), 32(–10).

Standard & Poor's Index of 500 Stocks [CDA S&P500]

Here's another case motivated more by wishful thinking (and perhaps avarice) than by logic. The graph of the data [B] and the plot of each day's percentage change in the stock market versus the change on the previous day [VT] show little structure. The power spectrum [XE] is broad, and there are no pronounced dominant frequencies [XF]. The correlation dimension [XJ] is large (greater than or the order of six), and the correlation matrix [XL] has many large eigenvalues. Nevertheless, you can blindly (and blissfully) continue toward a prediction [VVVVVVVVVVVV]. If you wish to test your prediction, the percentage change on the last day of the data record was –5.2, and the percentage changes on the next 18 days, beginning with "black Monday," were: –20.5, 5.3, 9.1, –3.9, 0.0, –8.3, 2.4, 0.0, 4.9, 2.9, 1.6, –1.9, –0.1, 2.2, –1.6, –2.9, –1.7, 1.2.

APPENDIX C: SUGGESTED ACTIVITIES

After you have mastered the tutorial, you might want to try a few more challenging independent studies of the sample data sets before using the program with your own data. Some of these suggested activities represent topics of current research.

1. Look at the power spectrum for various sample data sets and decide whether the data more nearly fit a straight line on a log-linear scale or on a log-log scale. In general, various kinds of noise generally follow a power-law spectrum ($1/f$, $1/f^2$, etc.) whereas chaotic time series sometimes follow an exponential power law (e^{-f}).
2. Determine how the Lyapunov exponent and the correlation dimension vary with embedding dimension for the Hénon map [HENON.DAT]. Is there an embedding dimension above which the results are insensitive to the embedding? From your results, what would you estimate to be the minimum embedding for a reliable determination of these quantities?
3. Graph the Lyapunov exponent and correlation dimension (including error bars) versus the percentage done for the Lorenz attractor [CHAOS.DAT]. Decide whether you think the values at 100% have converged, or whether more data would be useful. Does it appear to you that the estimated errors encompass the real values that would be obtained with an infinite data record?
4. Determine the effect of differentiating, integrating, and smoothing the Lorenz attractor data [CHAOS.DAT] on the Lyapunov exponent, capacity dimension, and correlation dimension. This is an important practical issue, since experimental data are often subjected to these processes before being recorded.
5. Estimate the location of the fixed points for the logistic map [LOGMAP.DAT], the Hénon map [HENON.DAT], and the Lorenz attractor [CHAOS.DAT] using the probability distribution program. Compare your results with the expected values obtained by solving the respective equations for their time-independent solutions.
6. Assume that the mean daily temperature in Madison, WI [MSNTEMP.DAT] is governed by a linear superposition of sine waves. From the dominant frequencies, make a linear prediction of the

temperature for the first five days of January 1990 for various numbers of poles (2 to 128) and compare the predictions with the actual temperatures (25, 26, 36, 25, 19). Are your predictions statistically better than simply predicting that tomorrow will be just like today?

7. Go to the Data Manipulation program and load the data for the mean daily temperature in Madison, WI [MSNTEMP.DAT]. Smooth the data multiple times, and see if there is any warming or cooling trend over the decade represented. Use the Polynomial Fit program to get a first-order fit to the data, and see by how many degrees the average temperature changed over the decade.
8. Using the correlation matrix analysis with the logistic map data [LOGMAP.DAT], see how many time steps are required for the solution of the model equations to differ by 10% from the input data. Estimate the precision of the input data using the fact that the Lyapunov exponent is 1.0 bit per iteration. This exercise emphasizes the difficulty of making long-term predictions of chaotic systems even under the best of circumstances.
9. Use the correlation matrix analysis with the sine wave data [SINE.DAT] to deduce a set of two linear equations whose solution is a sine wave. From the equations, determine the frequency of the sine wave and compare with the result from a calculation of the dominant frequencies. How are ψ_1 and ψ_2 related to X and dX/dt? Do the equations have a growth or damping term, and if so, what is the time scale for the growth or damping?
10. Use the correlation matrix analysis to predict the change in the S&P 500 index of common stocks [S&P500.DAT] on "black Monday" for various numbers of terms in the correlation matrix and various numbers of data points in the fitting sample. Would you have been a buyer or a seller of stocks when trading opened on black Monday?
11. If you have the Chaos Demonstrations program, use it to generate several time series of chaotic systems not included with CDA Pro, and determine the correlation dimension and Lyapunov exponent of each case.
12. It has been suggested that the largest Lyapunov exponent is the order of the inverse of the correlation time for a chaotic system. Test this prediction using data sets provided with the program and others that you produce.
13. The program provides four different predictors. Take several of the data sets provided with the program and delete the last 18 points from each. Use each of the four methods to make predictions of the deleted points and compare your predictions with the deleted values. Is any one of the predictors generally superior

to the others, or do the results depend on the nature of the time series?

14. See if you can train the neural net program on textual data similar to that in ASCIITXT.DAT so that it produces additional text that mimics human writing. You may want to train the net on your own longer data set and use the phase-plot view to produce a file of predicted values longer than the 18 points to which the predictor defaults.
15. There is a relationship between the slope of the power spectrum and the Hurst exponent for fractional Brownian motion. See if you can discover this relationship by looking at the data in NOISE.DAT, PINKNOIS.DAT, and BROWNIAN.DAT. Look also at other noisy data sets.
16. Using the noise-corrupted sine-wave data [SINOISE.DAT] and Hénon data [HENOISE.DAT], try to remove the noise by the various techniques included with the program. Generate a data file of your best noise-reduced case, produce a graph of it, and compare the graph with graphs of the original data and with the noise-free data in SINE.DAT and HENON.DAT.
17. Examine how the presence of noise [HENOISE.DAT] and rounding [HENROUND.DAT] effect the calculation of the correlation dimension for the Hénon map [HENON.DAT]. Try to remove the noise and rounding errors using one of the methods included in the program and try to get a more reliable estimate of the correlation dimension.
18. It is often the case that several related quantities are measured simultaneously in an experiment. One approach to the analysis of such data using CDA Pro is to generate a single time series in which the simultaneous values appear sequentially. Study the effect of this approach using X and Z data from the Lorenz attractor [LORENZXZ.DAT], comparing in particular the calculated correlation dimension and Lyapunov exponents with the ones calculated from the X data alone [CHAOS.DAT].
19. Compare the results of a mediocre linear congruential uniform random-number generator [LCGX.DAT] with the results of a high quality random-number generator [RANDOM.DAT]. See how many ways you can find in which they significantly differ. Examine the random numbers generated by your favorite compiler for evidence of hidden determinism.

APPENDIX D: REFERENCES

1. E. N. Lorenz, *J. Atm. Sci.* **20**, 130 (1963).
2. C. Sparrow, *The Lorenz Equations: Bifurcations, Chaos, and Strange Attractors* (Springer-Verlag, New York, 1982).
3. J. C. Sprott and G. Rowlands, CHAOS DEMONSTRATIONS PC Version 3.0 (American Institute of Physics, New York, 1994).
4. G. Rowlands, *Non-Linear Phenomena in Science and Engineering* (Ellis Horwood, London, 1990).
5. J. C. Sprott, *Strange Attractors: Creating Patterns in Chaos* (M&T Books, New York, 1993).
6. F. Takens in *Lecture Notes in Mathematics*, **898**, D. A. Rand and L. S. Young, eds. (Springer-Verlag, Berlin, 1981).
7. P. Grassberger and I. Procaccia, *Phys. Rev. Lett.* **50**, 346 (1983).
8. P. Grassberger and I. Procaccia, *Physica D* **13**, 34 (1984).
9. H. S. Greenside, A. Wolf, J. Swift, and T. Pignataro, *Phys. Rev. A* **25**, 3453 (1982).
10. L. A. Smith, *Phys. Lett. A* **133**, 283 (1988).
11. M. A. H. Nerenberg and C. Essex, *Phys. Rev. A* **42**, 7065 (1990).
12. A. A. Tsonis, *Chaos: From Theory to Applications* (Plenum Press, New York, 1992).
13. A. Wolf, J. B. Swift, H. L. Swinney, and J. A. Vastano, *Physica D* **16**, 285 (1985).
14. N. H. Packard, J. P. Crutchfield, J. D. Farmer, and R. S. Shaw, *Phys. Rev. Lett.* **45**, 712 (1980).
15. H. Froehling, J. P. Crutchfield, D. Farmer, N. H. Packard, and R. Shaw, *Physica D* **3**, 605 (1981).
16. D. S. Broomhead and G. P. King, *Physica D* **20**, 217 (1986).
17. R. Vautard and M. Ghil, *Physica D* **35**, 395 (1989).
18. J. L. Lumley, *Stochastic Tools in Turbulence* (Academic Press, New York, 1970).
19. N. Aubry, P. Holmes, J. L. Lumley, and E. Stone, J. *Fluid Mechanics* **192**, 115 (1988).
20. R. M. May, *Nature* **261**, 459 (1976).
21. M. J. Feigenbaum, *J. Stat. Phys.* **19**, 25 (1978).
22. A. Lempel and J. Ziv, *IEEE Transactions on Information Theory* **IT-22**, 75 (1976).
23. F. Kaspar and H. G. Schuster, *Phys. Rev. A* **36**, 842 (1987).

24. I. N. Stewart, *The Dynamics Newsletter* **3**, 4 (1989).
25. H. J. Jeffrey, *Comp. & Graph.* **16**, 25 (1992).
26. W. H. Press, B. P. Flannery, S. A. Teukolsky, and W. T. Vetterling, *Numerical Recipes* (Cambridge University Press, New York, 1986).
27. M. S. Bartlett, *Stochastic Processes* (Cambridge University Press, Cambridge, 1955).
28. F. J. Harris, *Proc. IEEE* **66**, 51 (1978).
29. S. M. Kay and S. L. Marple, *Proc. IEEE* **69**, 1380 (1981).
30. F. Laeri, *Comp. Phys.* **4**, 627 (1990).
31. G. E. P. Box, G. M. Jenkins, and G. C. Reinsel, *Time Series Analysis: Forecasting and Control* (Prentice Hall, Englewood Cliffs, NJ, 1994).
32. J. Feder, *Fractals* (Plenum, New York, 1988).
33. J-P. Eckmann, S. O. Kamphorst, D. Ruelle, and S. Ciliberto, *Phys. Rev. A* **34**, 4971 (1986).
34. R. Stoop and P. F. Meier, *J. Opt. Soc. Am.* B **5**, 1037 (1988).
35. B. B. Mandelbrot, *The Fractal Geometry of Nature* (Freeman, San Francisco, 1982).
36. J. D. Farmer, E. Ott, and J. A. Yorke, *Physica D* **7**, 153 (1983).
37. M. F. Barnsley, R. L. Devaney, B. B. Mandelbrot, H. O. Peitgen, D. Saupe, and R. F. Voss, *The Science of Fractal Images* (Springer-Verlag, New York, 1988).
38. B. Abraham, A. M. Albano, B. Das, G. DeGuzman, S. Young, R. S. Gioggia, G. P. Puccioni, and J. R. Tredicce, *Phys. Lett. A* **114**, 217 (1986).
39. J. W. Havstad and C. L. Ehlers, *Phys. Rev. A* **39**, 845 (1989).
40. M. B. Kennel, R. Brown, and H. D. I. Abarbanel, *Phys. Rev. A* **45**, 3403 (1992).
41. M. B. Kennel and H. D. I. Abarbanel, *Phys. Rev. E* **47**, 3057 (1993).
42. J. D. Farmer, Z. Naturforsch. **37A**, 1304 (1982).
43. P. Grassberger and I. Procaccia, *Phys. Rev. A* **28**, 2591 (1983).
44. G. Benettin, L. Galgani, and J. -M. Strelcyn, *Phys. Rev. A* **14**, 2338 (1976).
45. Ya. B. Pesin, *Russ. Math. Surveys* **32**, 55 (1977).
46. W. A. Brock, *J. Econom. Theory* **40**, 168 (1986).
47. A. Singh and D. D. Joseph, *Phys. Lett. A* **135**, 247 (1989).
48. B. F. Feeny and F. C. Moon, *Phys. Lett. A* **141**, 397 (1989).
49. D. S. Broomhead and R. Jones, *Proc. Roy. Soc. A* **423**, 103 (1989).
50. L. Sirovich and J. D. Rodriguez, *Phys. Lett. A* **120**, 211 (1987).
51. G. Rowlands and J. C. Sprott, *Physica D* **58**, 251 (1992).
52. F. Argoul, A. Arneodo, G. Grasseau, Y. Gagne, E. J. Hopfinger, and U. Frisch, *Nature* **338**, 51 (1989).

53. C. K. Chui, *An Introduction to Wavelet Analysis* (Academic Press, Boston, 1992).
54. A. J. Owens and D. L. Filkin, *International Conference on Neural Networks* **2**, 381 (1989).
55. J. D. Farmer and J. J. Sidorowich, *Phys. Rev. Lett.* **59**, 845 (1987).
56. G. Sugihara and R. M. May, *Nature* **344**, 734 (1990).
57. M. Casdagli, *Physica D* **35**, 335 (1989).
58. R. Wayland, D. Bromley, D. Pickett, and A. Passamante, *Phys. Rev. Lett.* **70**, 580 (1993).
59. E. J. Kostelich and J. A. Yorke, *Physica D* **41**, 183 (1990).
60. J. D. Farmer and J. Sidorowich, in *Evolution, Learning and Cognition*, Y. C. Lee, ed. (World Scientific, Singapore, 1988), p. 277.
61. T. Schreiber and P. Grassberger, *Phys. Lett. A* **160**, 411 (1991).
62. T. Sauer, *Physica D* **58**, 193 (1992).
63. N. Enge, T. Buzug, and G. Pfister, *Phys. Lett. A* **175**, 178 (1993).
64. A. R. Osborne and A. Provenzale, *Physica D* **35**, 357 (1989).
65. M. Casdagli, in *Nonlinear Prediction and Modeling*, M. Casdagli and S. Eubank, eds. (Addison-Wesley, Reading, MA, 1991).
66. J. Theiler, *Phys. Lett. A* **155**, 480 (1991).
67. J. Theiler, S. Eubank, A. Longtin, B. Galdrikian, and J. D. Farmer, *Physica D* **58**, 77 (1992).
68. N. A. Gershenfeld and A. S. Weigend, *The Future of Time Series: Learning and Understanding, Predicting the Future and Understanding the Past: A Comparison of Approaches* (Addison-Wesley, Reading, MA, 1993).
69. H. D. I. Abarbanel, R. Brown, J. J. Sidorowich, and L. Sh. Tsimring, *Rev. Mod. Phys.* **65**, 1331-1392 (1993).
70. E. Ott, T. Sauer, and J. Yorke, eds., *Coping with Chaos* (Wiley, New York, 1994).
71. J. Gleick, *Chaos: Making a New Science* (Viking Penguin, New York, 1987).
72. N. K. Hayles, ed., *Chaos and Order: Complex Dynamics in Literature and Science* (University of Chicago Press, Chicago, 1991).
73. E. Basar (ed.), *Chaos in Brain Function* (Springer-Verlag, Berlin, 1990).
74. M. Hénon, *Comm. Math. Phys.* **50**, 69 (1976).
75. K. Ikeda, *Opt. Commun.* **30**, 257 (1979).
76. R. L. Bowman, *Comput. & Graphics* **19**, 315 (1995).
77. R. Lozi, J. Phys. **39**, 9 (1978).
78. M. C. Mackey and L. Glass, *Science* **197**, 287 (1977).
79. L. Glass and M. C. Mackey, *From Clocks to Chaos: The Rhythms of Life* (Princeton University Press, Princeton, NJ, 1988).
80. P. Grassberger and I. Procaccia, *Physica D* **9**, 189 (1983).

81. C. Watts, D. E. Newman, and J. C. Sprott, *Phys. Rev. E* **49**, 2291 (1994).
82. W. A. Press and S. A. Teukolsky, *Comp. Phys.* **6**, 522 (1992).
83. O. E. Rössler, *Phys. Lett. A* **57**, 397 (1976).
84. O. E. Rössler, *Phys. Lett. A* **71**, 155 (1979).
85. J. C. Sprott, *Phys. Rev. E* **50**, R647 (1994).
86. S. Nosé, *J. Chem. Phys.* **81**, 511 (1984).
87. W. G. Hoover, *Phys. Rev. A* **31**, 1695 (1985).
88. H. A. Posch, W. G. Hoover, and F. J. Vesely, *Phys. Rev. A* **33**, 4253 (1986).
89. B. Van der Pol, *Phil. Mag.* **2**, 978 (1926).
90. T. Tel, *Z. Naturforsch.* **43A**, 1154 (1988).
91. M. V. Berry and Z. V. Lewis, *Proc. R. Soc. London, Ser. A* **370**, 459 (1980).

INDEX