17
Nonlinear Techniques

17.1 INTRODUCTION

As was discussed in Chapter 8, most of the analytical tools we introduced throughout the text are based on dynamical processes generated by linear time invariant (LTI) systems. In general, the success of most of these time series analysis methods in physiology is surprising considering that physiological processes are known to include significant nonlinearities. The explanation for this relative success is perhaps due to cases in which physiological systems can be studied in a state where the linear behavior is most prominent, or where a limited range of a measured property is considered in which a linear process is a good approximation of the system's behavior. On the other hand, there are many examples where inclusion of nonlinear dynamics is critical for understanding the physiology. The well-known Hodgkin and Huxley model (Hodgkin and Huxley, 1952) or higher-order kernels in the auditory response (e.g., Recio-Spinoso et al., 2005) are just a few examples. It is also very likely that our perception of the success of linear analysis is biased because many interrelationships in the nervous system remain undiscovered because the linear analysis techniques currently in use (e.g., correlation) fail even to detect them. In a general sense, there is a significant need for novel signal processing tools for studying nonlinear relationships in physiology as well as a critical necessity to evaluate the tools that have been developed over the past decades.

The purpose of this chapter is to introduce a few of the nonlinear analysis tools that are available to analyze and to describe biomedical signals. First, we explore some of the characteristics that distinguish linear from nonlinear systems by analyzing a few simple examples, including the so-called logistic equation. In a second step, we look into the failure and success of different techniques in characterizing and quantifying time series generated by nonlinear dynamical systems. An overview of metrics that have been developed to characterize the nonlinear dynamics of time series is given in Section 17.5. Students with more interest in nonlinear

systems are refered to Peitgen et al. (1992), Kaplan and Glass (1995), and Strogatz (1994). These authors provide excellent introductions to nonlinear dynamics and chaos theory with numerous practical examples.

17.2 NONLINEAR DETERMINISTIC PROCESSES

The purpose of many experiments is to find direct cause-effect relationships: so-called deterministic relationships in which the past uniquely determines the current state of a system. While some systems such as a swinging pendulum behave predictably, developments in the stock markets or the weather do not seem to be so predictable. We might therefore conclude ***incorrectly*** that simple deterministic systems are predictable, whereas involvement of more complex processes puts that predictability at risk. In the following, we show that even simple, deterministic processes can display surprisingly unpredictable behavior. For instance, a time series generated by a simple difference equation, such as the logistic equation:

$$x_i = ax_{i-1}(1 - x_{i-1}) \tag{17.1}$$

in which each point x_i depends only on a quadratic function of its previous value, can exhibit behavior ranging from stable, or oscillatory, to very erratic. Examples of time series x_i generated with different values of parameter a in Equation (17.1) are shown in Figure 17.1A–C.

One way to understand this system is to investigate its convergence properties. In these examples, it can be seen that the number of possible final states of the system critically depends on the value of a. For some values of this parameter, the system converges to a single steady-state solution (Fig. 17.1A), while for other values the system generates anything from a handful to a seemingly infinite number of states (Fig. 17.1B, C). If we plot the final states generated by Equation (17.1) against different values of a, we obtain the so-called final state diagram shown in Figure 17.1D.

This diagram, which can be produced with MATLAB script pr17_1.m, shows the following:

1. The behavior of the logistic equation converges to a single value for $a < 3$ (e.g., Fig. 17.1A)
2. Stable periodic behavior with two values occurs for $3 < a < 3.4495$ (e.g., Fig. 17.1B)
3. A subsequently increasing number of final states occur with increasing values of a

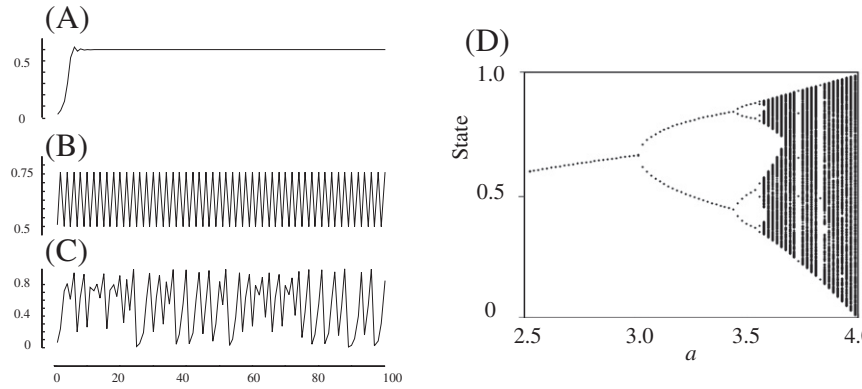

Figure 17.1 Characteristics of time series created with the logistic equation (Equation (17.1)). (A) The time series converges to a single value for $a = 2.50$. (B) For $a = 3.24$, there is oscillatory behavior between two states. (C) Chaos at $a = 4$. (D) One of the icons of chaos: the final state diagram showing the period-doubling route to chaos. In this graph, final states are plotted against the value of a in the logistic equation. The logistic equation (a quadratic iterator) transitions to oscillatory behavior at the bifurcation $a = 3$. For $a > 3.569 \ldots$, the system transitions to chaotic behavior. Feigenbaum (1983) discovered that the ratio of two successive ranges over which the period doubles is a constant universally encountered in the period-doubling route to chaos (Feigenbaum's number: 4.6692 . . .). The MATLAB script pr17_1.m can be used to create the final state diagram. (From van Drongelen et al. (2005), Seizure Prediction in Epilepsy. In *Neural Engineering* (ed. He, B), Kluwer Academic, New York.)

This characterization of the long-term behavior of the system, or bifurcation diagram in Figure 17.1D, shows a transition from stable to so-called chaotic behavior for a above values of 3.569. . . . The transition pathway, from simple periodic behavior into an unpredictable regime shown in the diagram of Figure 17.1D, is called the period-doubling route to a ***chaos*** (Fig. 17.1C). The logistic equation is not an exceptional case: many more examples of fairly simple systems showing complex behavior can be found. The seminal example, a simplified and deterministic model of a weather system consisting of a set of only three nonlinear differential equations, showed similarly dramatic unpredictability (Lorenz, 1963). We can compare these unpredictable processes to rolling a die or drawing a numbered lotto ball; they all show random behavior that can be characterized by measuring the probabilities of the various outcomes. In principle, if one knew precisely all the positions and mechanical parameters of the elements in a lotto drawing, one would be able to calculate the end result. It is surprising that in spite of this "in-principle-predictability," randomness seems inherently associated with these types of deterministic pro-

cesses. Interestingly, some very complex phenomena, such as tides, that depend on many other processes (position of the moon, the wind, details in the coastline, etc.) can be fairly predictable. From the earlier examples, we therefore conclude the following:

1. That the level of complexity in a time series does not necessarily correspond with the level of complexity of the underlying process
2. That deterministic systems do not always show predictable behavior

One might counter that the second conclusion simply represents lack of knowledge of the system and that one should be able to precisely compute the behavior of a system if the equations governing its dynamics (such as Equation (17.1)) are known. In principle this is correct, but there are serious practical problems with this approach. Usually there is a degree of unavoidable uncertainty that prevents us from knowing all aspects of the past and present states of a dynamical system. And, even if we do know all this, any knowledge, measurement, or computation of a system state is associated with a degree of precision which limits our exact knowledge of the initial and subsequent condition of an evolving process. Finally, it appears that in some systems with *nonlinear dynamics*, minute errors, or perturbations (of the order of magnitude of a rounding error of a computer or even smaller) generate huge differences in the predicted outcomes even over short prediction windows. This difference can grow ***disproportionately*** toward the same order of magnitude as the predicted values — that is, the evolution and outcome of certain types of processes may depend critically on initial conditions (see the example in Figs. 17.4D–F). This dependence is sometimes referred to as the "butterfly effect": as Lorenz pointed out, a perturbation as small as the flapping wings of a butterfly could influence the development of a tornado on another continent. Of course, sensitivity to perturbations also exists in linear systems. However, the error in a linearly evolving process grows ***proportionally*** with the predicted values.

17.3 LINEAR TECHNIQUES FAIL TO DESCRIBE NONLINEAR DYNAMICS

Linear techniques were designed to detect properties or relationships within or between time series generated by linear systems. Therefore we may safely assume that these techniques perform well in signals with a strong linear component. In contrast, we may suspect that these techniques would fail in signal analysis when the data originate from a nonlinear dynamical system. To explore our expectations and suspicions, let's

consider an example of the application of the autocorrelation function (Chapter 8) to three distinct time series:

A predominantly linear relationship:	$x_i = ax_{i-1} + N_{i-1}$	(17.2a)
A nonlinear relationship (logistic equation):	$x_i = ax_{i-1}(1 - x_{i-1})$	(17.2b)
A completely random relationship:	$x_i = N_{i-1}$	(17.2c)

with N being a random process with zero mean and a standard deviation of one.

In Figure 17.2 we can observe a sample of each of these time series, their associated autocorrelation functions, and their so-called return plots. The return plot depicts the relationship between successive values of the time series x_{i+1} and x_i. As expected, the autocorrelation in Figure 17.2A2 indicates a relationship between subsequent points in the time series

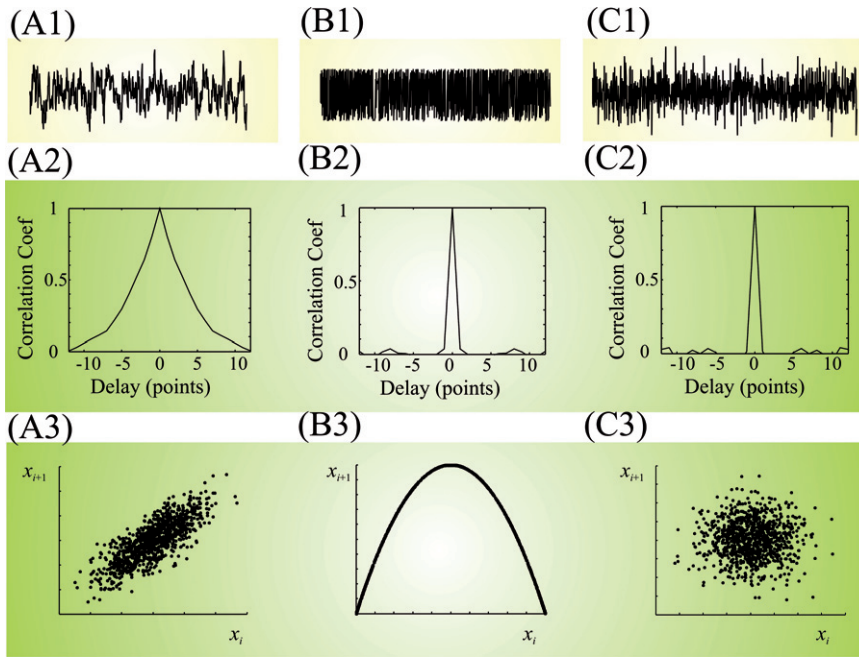

Figure 17.2 Three different time series generated with Equations (17.2a–c). The waveform in (A1) is determined by a linear function; the signal in (B1) is from a nonlinear dynamical system (the logistic equation); (C1) is a random time series. The graphs in (A2) to (C2) show the corresponding autocorrelation functions. The scatter plots in (A3) to (C3) depict $x_{i+1} = f(x_i)$. The graphs in this figure can be reproduced by MATLAB programs pr17_2.m to pr17_5.m.

generated by Equation (17.2a). While the autocorrelation reasonably detects no point-to-point relationship within the random series (since there is none to detect), it fails to show the relatively simple, and completely deterministic, connection between past and current values in the nonlinear logistic equation (Figs. 17.2B2, C2). On the other hand, the return plots (Fig. 17.2A3, B3, and C3) clearly distinguish between the linear, quadratic, and random relationships. The important conclusion here is that in both the first two examples (Equations (17.2a) and (17.2b)) there are deterministic components, but only the first can be detected by linear analysis tools.

In case of the linear iterator in Equation (17.2a), we added the random term N_{i-1} to perturb the system. Without this random term, the time series reflecting the system is drawn to an equilibrium that ends all dynamics (compare the MATLAB scripts pr17_2.m and pr17_3.m); in this case the autocorrelation is similar to the one in Figure 17.2A2 but the noise in the return plot is absent, resulting in points determined by the equation $x_i = ax_{i-1}$ and most points being around (0, 0). In the quadratic relationship explored in MATLAB script pr17_4.m (Equation (17.2b)), the system's behavior remains dynamic and the addition of a significant noise term is not required and may even lead to instability.

An important characteristic of a linear system is that a sine wave input generates a sine wave output with the same frequency in which only the phase or amplitude may change. A nonlinear system, in contrast, may react very differently and alter the waveform and frequency of the output signal with respect to the input. This nonlinear property of a frequency change is also poorly detected by the correlation function. An example is shown in Figure 17.3, summarizing a correlation study using a sinusoidal

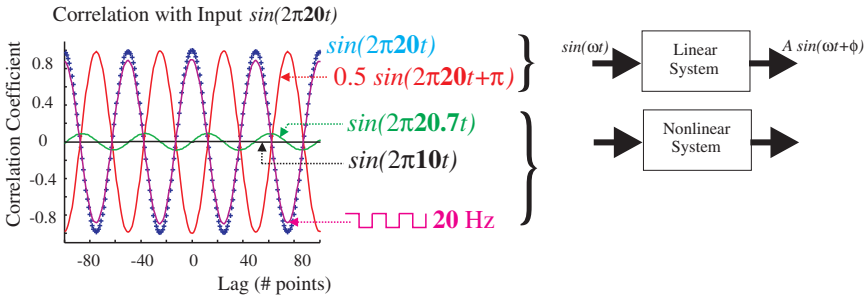

Figure 17.3 A correlation study between sine waves. The correlation between the input and output of a system shows a relationship if the frequency (here 20 Hz) remains unaltered. A minor (20.7 Hz) or larger (10 Hz) change in frequency as compared to the 20 Hz causes correlation values to drop significantly. However, a rectangular pulse of 20 Hz representing a seriously distorted sine wave generates values close to the autocorrelation. The graphs in this figure can be obtained from MATLAB script pr17_6.m.

input of 20 Hz ($\sin(2\pi 20t)$). As expected, the autocorrelation (waveform indicated with blue + in Fig. 17.3) detects the correlation. Cross-correlation between sine waves of the same frequency of 20 Hz, but with different amplitude and phase, also works well (red waveform in Fig. 17.3). However, as soon as we assume a nonlinear system that alters the frequency from 20 Hz to 20.7 Hz (green waveform in Fig. 17.3) or even 10 Hz (black waveform in Fig. 17.3), the cross-correlation procedure applied to the time series does not detect any relationships.

17.4 EMBEDDING

In the previous section, we confirmed our suspicion that linear analysis techniques may perform poorly when studying nonlinear dynamics. The return plots in Figure 17.2, however, were fairly effective in showing the relationship within the different types of time series. In the return plot, we depicted the relationship between two points delayed by a single sample point; this approach can be extended both to include more points and delays. Such a multidimensional version of the scatter plot is the conceptual basis for a powerful technique in the analysis of dynamical systems, the so-called ***embedding*** procedure. Embedding of a time series x_t ($x_1, x_2, x_3, \ldots, x_N$) is done by creating a set of vectors X_i such that

$$X_i = [x_i, x_{i+\Delta}, x_{i+2\Delta}, \ldots, x_{i+(m-1)\Delta}] \tag{17.3}$$

where Δ is the delay in number of samples and m is the number of samples (dimension) of the vector. When embedding a time series, we must choose the dimension m of X_i and the delay Δ, such that each vector X_i represents values that reveal the topological relationship between subsequent points in the time series. The number of samples in the embedded vector is usually chosen to be large enough to cover the dominant frequency in the time series, but m should not be so large that the first and last values in the epoch are practically unrelated. The evolution of the system can now be represented by the projection of the vectors X_i onto a trajectory through multidimensional space, often referred to as ***phase space*** or ***state space***. If the multidimensional evolution converges to a subspace within the phase space, this subspace is called the ***attractor*** of the system. The construction and characterization of system attractors play a major role in the analysis of time series. As was proven mathematically, the attractor characterized by embedding a single variable (e.g., a single channel of EEG or ECoG) can characterize the nonlinear system that generated the time series (Takens, 1981). Measures that are commonly used to describe the attractor in phase space are ***dimension***, ***entropy***, and ***Lyapunov exponents***. For the dimension and entropy measures, several "flavors" exist and a

multitude of algorithms for each of these metrics has been developed over the past decades.

Examples of time series and a two-dimensional embedding are shown in Figure 17.4. The upper time series (Fig. 17.4A) is an example of the swing of a pendulum and the associated plot shows a strict relationship between past and future points. The next example (Fig. 17.4B) shows a random time series where the embedded vector shows no specific relation between successive points. The example in Figure 17.4C is from the logistic Equation (17.1). Interestingly, from visual inspection the time series generated by the random process and the logistic iterator does not seem that different. However, by plotting x_t versus x_{t-1}, one can see that one time series shows a random relationship and the next has a fairly simple attractor characterized by a the quadratic relationship from Equation (17.1). The time series embedding in Figure 17.4D is characterized by more complex relationships of a type often referred to as a strange attractor.

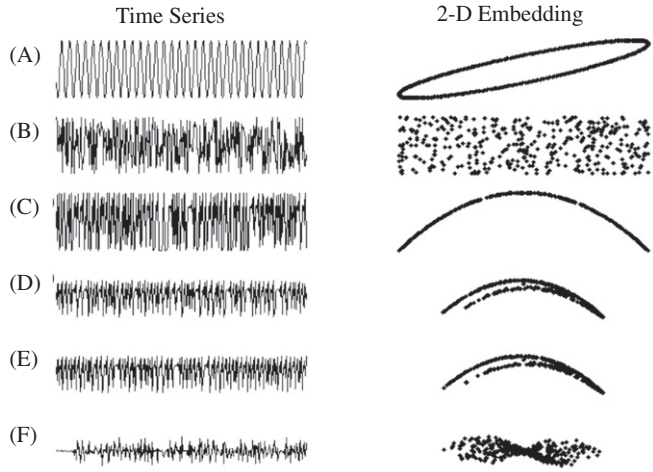

Figure 17.4 Examples of time series (left column) and embedding in two dimensions (right column). (A) Sinusoidal signal. (B) Random signal. (C) Time series determined by the logistic equation ($x_t = 4x_{t-1}[1 - x_{t-1}]$; $x_0 = 0.397$). (D, E) Two examples of a Henon map ($x_t = y_{t-1} + 1 - ax^2{}_{t-1}$; $y_t = bx_{t-1}$, $a = 1.4$, $b = 0.3$). The initial conditions differ between (D) $x_0 = 0$; $y_0 = 0$ and (E) $x_0 = 10^{-5}$; $y_0 = 0$. (F) The difference between (D) and (E) shows that initially both time series develop along a similar path (difference $\rightarrow$ 0). However, after ~25 iterations the difference in initial condition causes disproportionate difference in the values of the time series. This figure can be produced with MATLAB script pr17_7.m. (From van Drongelen et al. (2005), Seizure Prediction in Epilepsy. In *Neural Engineering* (ed. He, B), Kluwer Academic, New York.)

This strange attractor represents a more intricate geometry than that of the curved line in the quadratic relationship, but it is more confined in space than the random process, which covers the whole area of the plot. Both time series in Figures 17.4D and E are examples of time series generated by the Henon map, a classic chaotic iterator that defines the co-evolution of two variables x_t and y_t. Both plots in Figures 17.4D and E show x_t, but with only slightly different initial conditions: (0, 0) in Figure 17.4D and (10^{-5}, 0) in Figure 17.4E. The difference between the two closely related time series in Figure 17.4D and E is shown in Figure 17.4F, clarifying the sensitivity to a small perturbation (in this example 10^{-5}). Initially the difference between the two time series is small, but after 25 iterations the difference grows disproportionately until this error is of the same order of magnitude as the time series amplitudes themselves (Figs. 17.4D and E). This phenomenon illustrates the point that even with knowledge of initial conditions to a precision of 10^{-5} ***chaotic processes are only weakly predictable*** (i.e., the observed values may deviate considerably after only a few time steps).

An illustration of embedding of a measured EEG signal during an epileptic seizure is shown in Figure 17.5. To demonstrate the principle of embedding, we show a two-dimensional depiction despite the fact that two dimensions are likely insufficient to capture the full dynamics of the EEG.

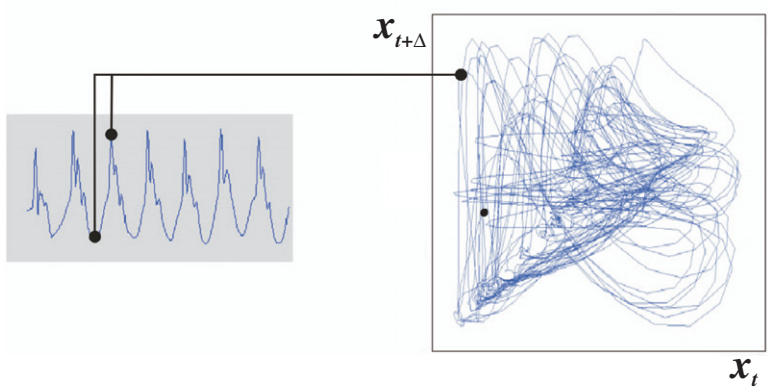

Figure 17.5 An example of embedding of an EEG signal during an epileptic seizure in two dimensions. Two points: $x_{t+\Delta}$ and x_t of the time series are plotted as one single point in a two-dimensional state space diagram. By embedding all subsequent pairs in the same manner, a two-dimensional projection of the attractor is obtained. (From van Drongelen et al. (2005), Seizure Prediction in Epilepsy. In *Neural Engineering* (ed. He, B), Kluwer Academic, New York.)

17.5 METRICS FOR CHARACTERIZING NONLINEAR PROCESSES

17.5.1 Attractor Dimension

Measures of dimensionality are used to characterize the geometry of an attractor in space. Several flavors of the dimension metric are currently in use. An overview of the relationships between the different dimension measures (the so-called Renyi dimensions) would be beyond the scope of this chapter and can be found in Peitgen et al. (1992). Theoretically central among these measures is the capacity dimension *D_Cap* of an attractor, which can be estimated with a box-counting algorithm. This metric formalizes our observation that while random fluctuations fill state space, chaotic attractors are limited to a restricted subspace (Fig. 17.4). The procedure estimates the space that is occupied by the attractor in terms of the number of hypercubes, or "boxes," $N(s)$ with size s in which points of the attractor are located (Fig. 17.6A):

$$D\_Cap = \lim_{s \to 0} \frac{\log_{10} N(s)}{\log_{10}(1/s)} \tag{17.4}$$

We will not provide a mathematical proof of this equation, but its rationale can easily be determined with a few examples (Fig. 17.6A). A single line segment of 1 m can be subdivided into 10 units ($N(s) = 10$) of 0.1 m (i.e., scale $s = 0.1$); this results in a dimension of one:

$$\frac{\log_{10} 10}{\log_{10}(1/0.1)} = 1$$

A surface of 1×1 m^2 includes 100 boxes ($N(s) = 100$) of 0.1×0.1 m^2 (i.e., scale $s = 0.1$); applying this to Equation (17.4), the dimension is two:

$$\frac{\log_{10} 10}{\log_{10}(1/0.1)} = 2$$

Critically, this technique also works for different scales. For instance, a cube of $1 \times 1 \times 1$ m^3 can be subdivided into 1000 small cubes of $0.1 \times 0.1 \times 0.1$ m^3, and 1,000,000 small cubes of $0.01 \times 0.01 \times 0.01$ m^3, and so on. In this example, the number of small cubes versus the inverse of the size (s) scales as $(1/s)^3$, the power being the capacity dimension of the cube:

$$\frac{\log_{10} 1000}{\log_{10}(1/0.1)} = 3 \quad \text{and} \quad \frac{\log_{10} 1000{,}000}{\log_{10}(1/0.1)} = 3$$

For different sizes of s, the value of $N(s)$ scales according to a power law: $N(s) \propto (1/s)^{D\_Cap}$. Applying the same box counting and scaling procedure for more irregular structures, such as an attractor embedded in a hypercube, can generate a ***noninteger value*** in between 2 and 3 for the dimension. The smaller the size of the box in the counting procedure, the more precisely the area/volume/and so on covered by the attractor can be described. Unfortunately, a reliable small-box count necessarily requires an attractor that is known in great detail (i.e., many points are available to characterize the attractor's space). For measured time series, such large data sets are often not available. The use of larger boxes is easier to accomplish but reflects the attractor's dimension less precisely. For this reason, the capacity dimension is not attractive for application to measured time series. Another measure that is related to *D_Cap* is the information dimension. This measure relates to the entropy, the distribution, and local density of the attractor's points in space. In box-counting terms, one counts the number of boxes occupied in space and weights the box by the number of points it includes. Like capacity dimension, the computational burden of estimating information dimension prevents it from being frequently used in experimental work.

The most popular dimension measure is the so-called correlation dimension. A metric derived from the so-called correlation integral:

$$C(s) = \left[\frac{1}{N(N-1)}\right] \sum_{i \neq j} U(s - |X_i - X_j|) \tag{17.5}$$

with U = Heaviside (unit step) function, and N = the number of points. The term $|X_i - X_j|$ denotes the distance between the points in state space. The summation (Σ) and the Heaviside function count the vector pairs (X_i, X_j) with an interpoint distance smaller than the threshold s, because $U(.)$ is one if this distance is smaller than s, and zero in all other cases:

$$U(s - |X_i - X_j|) \begin{cases} 1 & \text{for} \quad s - |X_i - X_j| > 0 \rightarrow |X_i - X_j| < s \\ 0 & \text{otherwise} \end{cases}$$

The value of $C(s)$ in Equation (17.5) is a measure of the number of pairs of points (X_i, X_j) on the reconstructed attractor whose distance is smaller than a set distance (Fig. 17.6B). The expression in Equation (17.5) is applied to discrete time data, therefore the correlation integral contains a summation rather than an integral. In the examples in Figure 17.6B, it can be seen that a line structure within a circle of radius s creates a set of pairs satisfying $|X_i - X_j| < s$ that are proportional to s, while a two-dimensional distribution creates a number of pairs proportional with s^2. Generally, for a large number of points (N) and small distances (s), $C(s)$ scales according

to a power law $C(s) \propto s^{D\_Cor}$, where *D_Cor* is the correlation dimension of the attractor.

17.5.2 Kolmogorov Entropy

Another metric that can characterize the dynamics of an attractor is the order-2 Kolmogorov entropy. Order-2 Kolmogorov entropy is a measure of the rate at which information about the state of a system is lost, and it can be estimated by examination of two initially close orbits in an attractor. The idea is that by selecting two neighboring points on an attractor, the evolution of one of the trajectories is informative about the other trajectory as long as they stay close. As soon as the trajectories diverge, the information about one trajectory relative to the other is lost. The time interval (t) required for the orbits to diverge beyond a set distance satisfies a distribution:

$$C(t) \propto e^{-KEt} \tag{17.6}$$

where *KE* is the Kolmogorov entropy. Schouten et al. (1994) described an efficient maximum-likelihood method of estimating *KE*. Their method assumes a time series of N points that is uniformly sampled at intervals of t_s; under these assumptions, Equation (17.6) becomes

$$C(b) = e^{-KEt_s b} \tag{17.7}$$

where b represents the number of time steps required for pair separation beyond the set criterion. They then show that the maximum likelihood estimate of the Kolmogorov entropy KE_{ml} (in bits per second) is

$$KE_{ml} = -\frac{1}{t_s}\left[\log_2\left(1 - \frac{1}{b_{avg}}\right)\right] \tag{17.8}$$

where b_{avg} is the average number of steps required for initially close pairs to diverge. The diagram in Figure 17.6C shows the principle of determining Kolmogorov entropy from nearby trajectories in an attractor.

To collect the necessary b's in Equation (17.8), methods of choosing nearby independent points as well as determining the divergence threshold are needed. Schouten et al. (1994) suggested estimating these from the data in the following way. First, the data are demeaned and divided by (normalized to) the average absolute deviation (x_{abs}) of the demeaned data:

$$x_{abs} = \frac{1}{N}\sum_{i=1}^{N}|x_i|$$

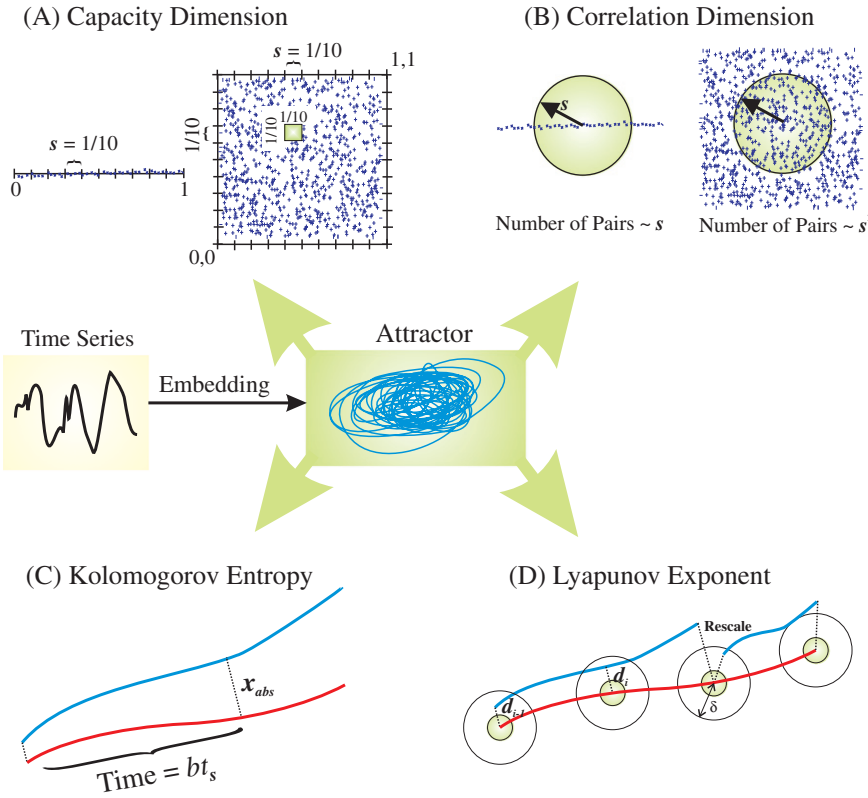

Figure 17.6 Simplified diagrams that reflect the algorithms to estimate measures that characterize an attractor. Two metrics (capacity (A) and correlation dimension(B)) reflect the distribution of the attractor in space; the other two measures — Kolmogorov entropy (C) and Lyapunov exponent(D) — quantify the divergence of initially close trajectories.

where N is the number of sample points. The value x_{abs} is then used as an estimate of the divergence threshold (i.e., a value of 1 in the normalized data set). Second, the number of cycles in the time series is estimated as half of the number of zero crossings; this is used to calculate the number of samples/cycle m, which is used as the independence criterion. This criterion indicates that two points that are separated by at least m samples belong to a different cycle in the time series and can therefore be considered independent. Therefore, the algorithm proceeds by selecting a pair of samples in the data at randomly chosen time steps i and j; if they are separated by at least m time steps ($|i - j| \geq m$), then they are considered to be independent and therefore eligible for use in the following calculations.

The largest of m absolute differences between pairs of values starting at i and j constitute the ***maximum norm***, which is the distance metric used in this algorithm:

$$d = \max(|x_{i+k} - x_{j+k}|) \tag{17.9}$$

for $0 \le k \le m - 1$; if $d \le 1$ (remember that this threshold of 1 corresponds to a threshold of x_{abs} in the non-normalized data), the samples are considered nearby. Finally, having found a pair of randomly chosen, nearby, independent data points, the number of steps b needed for them to diverge (such that at least one pair exceeds the criterion) — that is, $|x_{i+m-1+b} - x_{j+m-1+b}| > 1$ — can be added to the set used to calculate b_{avg}.

The preceding thresholds for determining independence and divergence work reasonably for many data sets, but we must stress that x_{abs} and m are heuristics that provide reasonable guidelines; they may yield better results for some data sets if modified by a factor of order unity.

17.5.3 Lyapunov Exponent

To begin with a trivial statement: an attractor would not be an attractor if there were not attraction of trajectories into its space. On the other hand, an attractor would not represent a chaotic process if neighboring trajectories within its space did not diverge exponentially fast. The Lyapunov exponent describes speed of attraction (convergence) or divergence of trajectories in each dimension of the attractor. We indicate the exponent in the *ith* dimension as λ_i, describing the rate at which the distance between two initially close trajectories changes over time as an exponent: e^{λ_i}. A value of $\lambda_i > 0$ indicates there is divergence and $\lambda_i < 0$ indicates convergence in the *ith* dimension. In two dimensions, the sum of the two exponents determines how a surface in the *ith* and $(i + 1)^{th}$ dimension evolves: $e^{\lambda_i} e^{\lambda_{i+1}} = e^{\lambda_i + \lambda_{i+1}}$. In three dimensions, three Lyapunov exponents describe the evolution of a cube, and the sum of all Lyapunov exponents indicates how a so-called hypercube evolves in a multidimensional attractor. To show divergence, and the chaotic signature of sensitivity to initial conditions, the largest Lyapunov exponent determined in an attractor of a chaotic process must be >0. Therefore the characterization of recorded signals by the Lyapunov exponent is usually focused on the largest exponent. The largest exponent describes the expansion along the principal axis (p_i) of the hypercube over a given time interval t. Formally, the exponent (λ_i) is calculated as

$$\lambda_i = \lim_{t \to \infty} \frac{1}{t} \log_2 \left[\frac{p_i(t)}{p_i(0)} \right] \tag{17.10}$$

Wolf et al. (1985) developed an algorithm to estimate the largest Lyapunov exponent in a measured time series. The algorithm described by Wolf et al. (1985) was revised for application to EEG and ECoG time series by Iasemidis et al. (1990). The procedure is shown in Figure 17.6D: the principle is to select a trajectory in the embedded time series and to determine a second point nearest to the starting point of this fiducial trajectory (red in Fig. 17.6D). The nearest point must not be too close because then the pair might be dynamically equivalent and only separated by some measurement noise (symbolized by the small green circle in Fig. 17.6D). Subsequently, the trajectories from this and the starting point are followed for a fixed time interval. The initial distance d_0 and the distance d_1 after time interval are measured. If the distance d_1 is smaller than a preset value (the larger circle with radius δ in Fig. 17.6D), the procedure is repeated. Figure 17.6D shows an example of two initially close trajectories (blue and red) and their start and end positions. If the distance between the end positions grows larger than the preset value δ, an attempt is made to rescale the distance by searching for a new point closer to the reference trajectory. Because we want to determine the Lyapunov exponent in a given dimension, we must stay within that same dimension and the rescaling procedure must find a new neighboring point that satisfies this condition. In reality this is the critical component of the algorithm because in measured signals a nearby point in the same dimension may not be available, making the rescaling a challenge. This procedure of measuring the interdistance at the start and end of these trajectories is repeated k times to cover the measured attractor from t_0 to t_k, and the largest Lyapunov exponent (λ_{max}) is calculated as

$$\lambda_{\max} = \frac{1}{t_k - t_0} \sum_{i=1}^{k} \log_2 \left[\frac{d_i}{d_{i-1}} \right] \tag{17.11}$$

Both the value of the largest Lyapunov exponent and the Kolmogorov entropy describe how quickly nearby trajectories diverge and therefore relate directly to predictability of the underlying process. For the Kolmogorov entropy estimation, the interpoint distance is set and the time of divergence is measured, whereas for estimation of the largest Lyapunov exponent, it is the other way around. For the Kolmogorov entropy estimation, close trajectories are selected randomly, while for the Lyapunov exponent, the procedure covers the attractor sequentially (Figs. 17.6C, D). Large values of both measures indicate an important divergence of trajectories that are initially close. As in the example of the Henon map in Figures 17.4 D, E, and F, small perturbations or inaccuracies in the initial state or in the calculation of subsequent values in a time series will create large differences after only a few iterations, thus limiting the potential for accurate prediction over a longer interval.

17.5.4 Surrogate Time Series

An important question when applying nonlinear time series analysis to recorded data is the nature of the underlying process. The algorithms for computing nonlinear metrics are constructed such that they will provide an estimate even when the underlying process is actually random or linear. Therefore, assigning a dimension value, Lyapunov exponent, and so on is not a guarantee that the time series is actually generated by a nonlinear dynamical process. To determine whether a data set contain nonlinearities, several methods have been developed in which surrogate data sets are generated and compared against the measured data (e.g., Kaplan and Glass, 1995). If we want to test linearity versus nonlinearity, we can compute one of the nonlinear measures for both the measured time series and for a surrogate time series generated by some linear model of the system.

A common approach is to estimate the linear model that generates the surrogate time series from the measured data itself. Subsequently, the values of the nonlinear measure obtained from the real data and a set of surrogate time series are compared. The null hypothesis is that the value of the computed nonlinear measure can be explained from the linear model, and if the null hypothesis is rejected, a nonlinear process may have generated the original data. The procedure to obtain surrogate data depends on the null hypothesis at hand. If the null hypothesis is that the data originate from a purely random process, a random shuffle of the measured data is sufficient to generate a surrogate time series. Another commonly applied null hypothesis is to assume that the underlying process is stationary, linear, and stochastic. A commonly applied technique to obtain surrogate time series satisfying this hypothesis is to compute the fast Fourier transform (FFT) followed by a randomization of the phase. The inverse FFT generates a surrogate time series representing linearly correlated noise with the same power spectrum and autocorrelation as the original signal but with the higher-order timing relationships destroyed. Methods of surrogate time series comparison provide a relatively robust technique for the task of making the presence of underlying nonlinearity plausible. Although nonlinearity is a prerequisite for existence of chaos, similarly objective tests to demonstrate an underlying chaotic process in measurements do not exist.

17.6 APPLICATION TO BRAIN ELECTRICAL ACTIVITY

Extraction of nonlinear metrics from brain activity reflected in EEG and ECoG (Chapter 1) time series has been used to anticipate or detect epilep-

tic seizures or to describe sleep stages. Automated detection of sleep stages is based on the idea that the EEG rhythm during different stages of vigilance will be reflected in the metrics for dimensionality, entropy, and so on. The assumption behind anticipating epileptic events is based on the hypothesis that seizures are preceded by a so-called pre-ictal state in which the processes leading to the ictal state (the seizure) take place. Although the use of nonlinear metrics in predicting epileptic events is still somewhat controversial, it does appear that at least in a number of cases a pre-ictal state can be detected prior to the clinical onset of the seizure. The epoch over which such detection occurs varies from seconds to hours prior to the clinical seizure. Further details of this topic may be found in an overview by van Drongelen et al. (2005).