学位論文

Reconstruction of the Bifurcation Structure of a Dynamical System from Time Series Data

（時系列データに基づいた力学系の分岐構造の再構成）

２０００年１月

大阪大学大学院 基礎工学研究科

システムス人間系専攻 生物工学分野

EPIFANIO BAGARINAO, JR.

エピファニョ バガリナオ ジュニア

Thesis Summary

In this study, a new and equally important theme in nonlinear time series analysis is considered – the study of bifurcations. Tools needed to analyze bifurcation structures from time series have been developed. The algorithms are based on a geometrical interpretation of the problem in terms of *projection region*. In this formulation, the projection region is defined as the region in the parameter space of a predetermined family of functions, which models the system, with similar bifurcation structures as the system. For parameter values within this region, the dynamics of the model is therefore the same as that of the given system. Thus, one can take the bifurcation structure of the model in this region as that of the original system. This allows the reconstruction of the bifurcation structure without knowing *a priori* the explicit form of the differential or difference equations describing the dynamics. With this, time series at different parameter values have been used to obtain a suitable family of predictor functions, which exhibit qualitatively similar bifurcations as the given system.

Instead of having a large introductory chapter, the preparatory materials are divided into three chapters. In chapter 1, a summary of the different themes of nonlinear time series analysis is given. A brief literature survey of related works is also presented. The problem is then introduced and defined. In chapter 2, several well-known techniques for the construction and quantification of the attractor of a dynamical system from time series are briefly reviewed. In particular, the estimation of the embedding dimension and the time delay is considered. A method to analyze bifurcation structures of predictor functions and the classification of the different bifurcations considered in this study are also given. In chapter 3, the different models used in this study are introduced. The use of the likelihood function to estimate model parameters for chaotic time series and the application of the nonlinear autoregressive (NAR) models and neural networks are discussed. A detection algorithm for nonlinear behavior in the observed data is also presented as one important application of NAR models. The examples included are not intended for comparison purposes but rather as specific examples of the model's application.

Chapters 4 to 7 contain the main results of the study. In chapter 4, the reconstruction problem is defined more precisely. The basic assumptions of the problem are presented. The concept of projection region is discussed and a reconstruction template is outlined. The linear manifold approximation of the projection region is considered. An example of its application to the reconstruction problem using the Hénon map is given at the end of the chapter.

In chapter 5, the conditions under which the linear manifold approximation holds are determined numerically using several well-known dynamical systems. The results have indicated that for a class of maps, referred to as linear-in-parameter (LIP) maps, the linear approximation always holds. Moreover, the results showed that as long as the parameter values are not widely apart, the LIP approximation is still valid even for non-LIP maps. A scheme is also developed to distinguish time series generated by this class of maps.

In chapter 6, the feasibility of reconstruction is explored when the linear manifold approximation is insufficient. A more general one-dimensional reconstruction algorithm is considered. The algorithm employed principal curves to approximate the one-dimensional projection region. The algorithm is tested using time series from systems exhibiting stable limit cycles and stable equilibrium points. In particular, the FitzHugh-Nagumo equations and the Lorenz equations are used to illustrate the algorithm's application. Possible extension to higher dimensional reconstruction is also discussed in the later part of the chapter.

In chapter 7, the effect of noise in the reconstruction problem is investigated. In particular, the reconstruction of bifurcation diagrams using time series from randomly forced maps is considered. Orthogonal polynomials are used as predictor functions. The proposed algorithm is applied to the reconstruction of the bifurcation diagrams of different dynamical systems. The results have indicated that the algorithm is robust to noise and works well even for a limited number of time series.

Finally, the conclusion is presented in chapter 8. The results are summarized and the different issues appearing in the course of the simulations are discussed. New interesting problems for possible extension of this work are also noted. The success of the simulation promises potential practical applications especially in the analysis of systems where first principle modeling is extremely difficult if not impossible.

Contents

Chapter 1

Introduction

Studies in nonlinear time series analysis have provided reliable techniques for the evaluation of signals from dynamical systems. Some of these techniques are used to gain insights into the unknown physical processes, to do prediction, as well as to determine invariants associated to the dynamics of the system. Others are employed to determine whether irregularities in signals are due to the intrinsic nonlinearity of the system or are caused by extrinsic random processes impinging on the system. Several others are applied to build models capturing the dynamics of the system from the observed data. In this study, new methods to uncover the underlying mechanisms of dynamical systems using time series are introduced. This set of algorithms can be used to evaluate how sensitive the system is to the values of its parameters and how the system's behavior changes as the parameters are varied. Since these issues are best explored by means of bifurcation diagrams (BDs), this thesis describes the algorithms in constructing BDs from time series.

Extracting physically interesting and useful information from the observed data is the primary goal of time series analysis. Nonlinear time series analysis has particularly provided additional tools for the characterization of irregular, broadband signals that are products of nonlinear dynamical systems. Without these tools, these signals are incomprehensible observations rather than vital sources of physically interesting information. Since these signals are prevalent in nature, their evaluation is very important and relevant. A few examples include the electrical activities within the brain, the beating of the heart, the spiking of neurons, the spread of epidemics, the swings in animal populations, and the changes in global climate. The analyses of these signals can be loosely grouped into: 1) the reconstruction and quantification of attractors, and 2) model-building and prediction, which includes parameter estimation using time series.

Dissipative systems are typically characterized by the presence of attracting sets or *attractors* in the phase space. An attractor is a bounded subregion of the phase space of a dynamical system to where regions of initial conditions of nonzero volumes eventually converge with increasing time. It can be a point in the phase space, of dimension equal to 0, or a closed curve, of dimension equal to 1. Some other attracting sets can be very irregular and in fact, can have a dimension which is not an integer value. Such sets are called *fractals*, and when they are attractors they are referred to as strange attractors. Strange attractors can be characterized by a spectrum of dimension values such as *box-counting dimension*, *information dimension*, and *correlation dimension*. The motion on a strange attractor can also display *sensitivity to initial conditions* such that the distance between neighboring points on the attractor can grow exponentially with time. This motion is referred to as being *chaotic*. The existence of chaos means that small errors grow exponentially in time that long term prediction becomes impossible. A quantitative description of the sensitivity to initial condition is provided by the *Lyapunov exponents*, quantities characterizing the stretching of infinitesimal displacements in a strange attractor.

The reconstruction of attractors by *delay embedding* [17, 36, 37, 48, 50, 65, 79], and their quantification in terms of dimensions[30, 80], Lyapunov exponents[19, 28, 72, 85], among other attractor invariants[35, 38] have yielded means of revealing intrinsic nonlinear behavior of a dynamical system from time series[11, 51, 68, 70, 78]. For example, the presence of a positive Lyapunov exponent or a fractional value of the attractor's dimension affirms the nonlinear nature of the system. These invariants can also be used to identify systems in a manner similar to natural frequencies of some physical systems. These quantities, however, do not give a complete description of the dynamics *per se* and thus, a different set of methods is required.

On the other hand, the goal of model-building is to construct a template of the dynamics using the observed data. Loosely speaking, this can be done by obtaining an appropriate set of coefficients in a predetermined class of functional forms such that the resulting function captures the dynamics of the system under study. The model is used either to represent the global behavior of the observed data or to describe the local dynamics in the reconstructed attractor or a mixture of both. The use of nonlinear autoregressive models [4, 5, 23, 62, 63], the measure-based functional reconstruction of Giona [34], radial basis functions[2, 22], and neural networks [3, 14, 18, 54] are among the many functions that represent global models describing the dynamics in the whole phase-space. On the other hand, local linear maps using neighboring points [55], local averaging [66], and the use of higher-order polynomials [31] whose coefficients

are fitted using near neighbors are just a few of the many local modeling approaches. The effectiveness of the model is measured by its prediction performance, defined as the ability of the model to give accurate values at steps forward in time.

The description of dynamical systems in terms of invariants or sets of coefficients in a predetermined class of functions is already sufficient to solve many significant problems. However, these invariants, though they remain constant with coordinate transformation, are not robust to changes in system parameters. Model coefficients also vary from one observation to another when observations are taken at different parameter values. Thus, problems that involve changes in parameters require a broader framework than the above description. This framework is provided by the study of *bifurcations*. A bifurcation is a qualitative change in the dynamics, for example from a stable behavior to an unstable behavior, which occurs as a system parameter varies. The knowledge of the bifurcation structure of a dynamical system is therefore important in order to understand the system's response to the changes in parameter values. This is particularly necessary for the case of nonlinear systems where small perturbations, for parameter values near critical points, can cause dramatic changes in the system's output. The study of bifurcations from time series, however, has received less attention in the past years. It is only recently that this problem is addressed rigorously [7, 8, 9, 10, 22, 49, 81, 82]. The problem is that the analysis requires *a priori* knowledge of the dynamics in the form of differential or difference equations [39, 84] that can prove difficult to construct even for simple systems. This makes the problem of reconstructing bifurcation structures from time series a formidable task.

In this thesis, a new and equally important theme in nonlinear time series analysis, *the study of bifurcations*, is introduced. The tools developed can be used to analyze time series measured at different parameter values. The motivation of the study is to unveil the bifurcation structure of the system using the observed data. In particular, the study aims to: 1) know the sequence of bifurcations that the system undergoes as the parameters are changed; and 2) uncover behaviors of the system, which may be present but not readily observed. To achieve these goals, the problem of reconstructing bifurcation diagrams is systematically investigated. Methods to obtain qualitatively the same BD as that of the given system using time series at a finite number of parameter values are presented. The reconstruction does not assume any knowledge of the explicit form of the dynamical system (differential/difference equation). Instead, time series at different parameter values are used to obtain a suitable family of predictor functions, which exhibits qualitatively similar bifurcations as the given system. The BD of this family of predictor functions on some parameter region, termed as *projection region*, is then

regarded as the reconstructed BD. In other words, the projection region is the region in the parameter space of the model with similar bifurcation structure as the system. For parameter values within this region, the dynamics of the model is therefore the same as that of the given system. Thus, one can take the BD of the model in this region as the reconstructed BD. The problem therefore is to determine the projection region using parameter values computed from the available time series.

In chapter 2, an overview of the different techniques to quantify and reconstruct attractors is presented. These algorithms are readily available in several review papers[1, 35] and published materials[61] and thus, only the general idea of each algorithm is outlined. In chapter 3, the different models used in the simulations are presented. In particular, the use of log-likelihood function for parameter estimation, the use of neural networks and nonlinear autoregressive models as parametrized predictor functions, and the detection of nonlinearities in noisy time series are examined. In chapters 4 to 7, the main problem of the study – bifurcation analysis using time series is tackled in depth and the necessary tools in reconstructing the BDs are described in detail. Finally in chapter 8, the results are summarized and recommendations are suggested.

Chapter 2

Overview

In this chapter, several well-known techniques for the reconstruction and quantification of the attractor of a dynamical system from time series are briefly reviewed. In particular, the estimation of the embedding dimension and the time delay is considered. These are important quantities in the BD reconstruction problem. A method to analyze bifurcation structures of predictor functions and the classification of the different bifurcations considered in this study is given in the last section.

2.1 Phase-Space Reconstruction

The basis of phase-space reconstruction is provided by Takens' theorem [79]. Accordingly, one can define an equivalent dynamics in terms of the observation $\{y_t,\ 0 \leq t \leq \tau\}$ such that

$$Y_{t+\Delta} = \mathbf{F}(Y_t) \tag{2.1}$$

where $\mathbf{F} : \mathbb{R}^d \to \mathbb{R}^d$ is a diffeomorphism, $Y_t = (y_t, \ldots, y_{t-(d-1)\Delta})$ is a vector of delayed coordinates, d is the dimension of the reconstructed phase-space (embedding dimension), and Δ is the time delay. Note that the form of $\mathbf{F}(Y_t)$ is such that all the components of the left hand-side vector are completely determined by the components on the right-hand side. This allows Eq. (2.1) to be written in the form

$$y_{t+\Delta} = g(y_t, y_{t-\Delta}, \cdots, y_{t-(d-1)\Delta}) \tag{2.2}$$

where $g : \mathbb{R}^d \to \mathbb{R}^1$ is called the *predictor function*.

Various approaches have been employed to approximate the predictor function $g(Y_t)$. These include local linear approximations[31], linear interpolation [55], nonlinear autoregressive models

[4, 5, 23, 62, 63], radial basis functions [2, 22], neural networks [3, 14, 18, 54], among others. However, finding an appropriate model out of the huge number of possible candidates proves to be extremely difficult. These issues will be explored in the next chapter.

The determination of the appropriate phase-space requires the proper choice of the embedding dimension d and time delay Δ. This has been a subject of such an intense interest that more reliable techniques are becoming available now. In the following, an overview of some of these techniques are given.

2.1.1 Embedding dimension

In Takens' theorem, the condition to have a diffeomorphism is that the embedding dimension d must be greater than the attractor dimension d_A, that is, $d \geq 2d_A + 1$. However, d_A is usually unknown and one should therefore be able to estimate the embedding dimension from the time series themselves. There are many available approaches to do this. Three of the most commonly used approaches are summarized below.

Singular-value decomposition. Suppose that the true embedding dimension necessary to unfold the dynamics is given by d. Now suppose that the available time series is embedded in a d_E-dimensional reconstructed phase-space such that $d_E > d$. The $d_E - d$ dimensions in this space are only populated by noise. One can then form the sample covariance matrix given

$$C = \frac{1}{N}\sum_{n=1}^{N}(Y_n - Y_{\text{ave}})(Y_n - Y_{\text{ave}})^T \tag{2.3}$$

with

$$Y_{\text{ave}} = \frac{1}{N}\sum_{n=1}^{N} Y_n. \tag{2.4}$$

This matrix will have d eigenvalues resulting from the real signal and $d_E - d$ eigenvalues due to noise. If the contamination is relatively high, the noise may be seen as filling all these $d_E - d$ dimension in a uniform manner. One can then examine where the eigenvalues start to plateau and the number of eigenvalues before the plateau may be taken as the desired dimension d.

Saturation of system invariants. If the attractor is properly unfolded, then any property associated to the attractor, which depends on the distance between two points lying on the attractor, should not vary with further increase in the embedding dimension. One of the examples is given by the number density, the number of points on the orbit within a radius r defined by

$$n(r, \mathbf{x}) = \frac{1}{N}\sum_{k=1}^{N}\theta(r - |Y_k - \mathbf{x}|), \tag{2.5}$$

with $\theta(u) = 0$ for $u < 0$ and $\theta(u) = 1$ for $u > 0$. The average over all points of powers of $n(r, \mathbf{x})$ is

$$C_q(r) = \frac{1}{M} \sum_{j=1}^{M} [n(r, \mathbf{x}_j)]^{(q-1)}. \tag{2.6}$$

One can then plot $C_q(r)$ as a function of d and determine when the slope of its logarithm as a function of $\log(r)$ becomes independent of d.

False nearest neighbors. This approach employs the basic question addressed in the embedding theorem: at what embedding dimension will the orbit cease to cross itself? The basic idea is thus to find the appropriate dimension where self intersection is not observed. This can be done by examining false neighbors, that is, points which appear to be neighbors due to the projection onto a lower dimensional space. By successively increasing the embedding dimension, these points will separate as the attractor is properly unfolded. The dimension where the number of these points goes to zero is then considered as the desired embedding dimension.

2.1.2 Time delay

Another important parameter in reconstructing the phase-space is the time delay Δ. For very small values of Δ, the coordinates $y_{t+(n+1)\Delta}$ and $y_{t+n\Delta}$ are too close to each other that from any practical point of view, they do not provide two independent coordinates. On the other hand, too large values of Δ will make $y_{t+(n+1)\Delta}$ and $y_{t+n\Delta}$ completely unrelated to each other. Thus, an intermediate value of Δ is desired.

As a first approximation, the *linear autocorrelation function* given by

$$C_L(\tau) = \frac{\frac{1}{N}\sum_{n=1}^{N}(y_{n+\tau} - \bar{y})(y_n - \bar{y})}{\frac{1}{N}\sum_{n=1}^{N}(y_n - \bar{y})^2} \tag{2.7}$$

where

$$\bar{y} = \frac{1}{N} \sum_{n=1}^{N} y_n \tag{2.8}$$

can be used to obtain a good hint for the choice of Δ. In this approach, the time delay Δ is given by the value of τ where $C_L(\tau)$ first passes through zero. The resulting delayed coordinates are independent in a linear fashion.

To account for the nonlinearities, one can use the *mutual information* criterion. The mutual information of two measurements a_i and b_k is defined by

$$I_{AB}(a_i, b_k) = \log_2 \left[\frac{P_{AB}(a_i, b_k)}{P_A(a_i) P_B(b_k)} \right], \tag{2.9}$$

where the probability of observing a out of the set of all A is $P_A(a)$, the probability of finding b in a measurement of B is $P_B(b)$, and the joint probability of the measurement of a and b is $P_{AB}(a,b)$. The average mutual information between measurements of any value a_i from system A and b_k from system B is the average of all possible measurements of $I_{AB}(a_i,b_k)$,

$$I_{AB} = \sum_{a_i,b_k} P_{AB}(a_i,b_k) I_{AB}(a_i,b_k). \tag{2.10}$$

One can think of set A as the sets of measurements y_n, and set B as measurements at a later time τ. In this way, the average mutual information can be defined as a function of the delay as

$$I(\tau) = \sum_{n=1}^{N} P(y_n, y_{n+\tau}) \log_2 \left[\frac{P(y_n, y_{n+\tau})}{P(y_n) P(y_{n+\tau})} \right], \tag{2.11}$$

and $I(\tau) \geq 0$. A candidate time delay is given by the location of the first minimum of $I(\tau)$ as a function of τ.

2.2 Bifurcation Analysis

When the dynamical system depends on some parameter $\mathbf{c} \in \mathbb{R}^p$, the time series output of the system will also depend on $\mathbf{c}$ and Eq. (2.1) can be rewritten to include explicitly the dependence on the parameters:

$$Y_{t+\Delta} = \mathbf{F}(Y_t; \mathbf{c}). \tag{2.12}$$

The predictor function $g(Y)$ also depends on $\mathbf{c}$, i.e., $g(Y;\mathbf{c})$. It is of interest to study the different behaviors of $g(Y;\mathbf{c})$ as function of the parameters.

Consider a p-parameter family of maps of $\mathbb{R}^d$ into $\mathbb{R}^d$

$$Y \mapsto \mathbf{F}(Y,\mathbf{c}), \quad Y \in \mathbb{R}^d, \quad \mathbf{c} \in \mathbb{R}^p \tag{2.13}$$

where $\mathbf{F}$ is $\mathbf{C}^r$ on some sufficiently large open set in $\mathbb{R}^d \times \mathbb{R}^p$. Equation (2.13) can be written explicitly as follows:

$$\begin{pmatrix} y_t \\ y_{t-\Delta} \\ \vdots \\ y_{t-(d-1)\Delta} \end{pmatrix} = \begin{pmatrix} g(y_{t-\Delta}, \cdots, y_{t-d\Delta}; \mathbf{c}) \\ y_{t-\Delta} \\ \vdots \\ y_{t-(d-1)\Delta} \end{pmatrix}. \tag{2.14}$$

The fixed points of Eq. (2.14) are given by the fixed points of $g(Y;\mathbf{c})$ which can be computed for a given $\mathbf{c}$ by finding the zeros of the function

$$z(y) = y - g(y, \cdots, y\ ; \mathbf{c}). \tag{2.15}$$

Now suppose that Eq. (2.14) has a fixed point at $(Y, \mathbf{c}) = (Y_o, \mathbf{c}_o)$, i.e.,

$$\mathbf{F}(Y_o, \mathbf{c}_o) = Y_o. \tag{2.16}$$

To determine the stability of the fixed point, the associated linearized map is examined. This is given by

$$\xi \mapsto D_y\mathbf{F}(Y_o, \mathbf{c}_o)\xi, \quad \xi \in \mathbb{R}^d. \tag{2.17}$$

Or explicitly,

$$\begin{pmatrix} \xi_t \\ \xi_{t-\Delta} \\ \vdots \\ \xi_{t-(d-2)\Delta} \\ \xi_{t-(d-1)\Delta} \end{pmatrix} = \begin{pmatrix} \frac{\partial g}{\partial y_{t-\Delta}} & \frac{\partial g}{\partial y_{t-2\Delta}} & \cdots & \frac{\partial g}{\partial y_{t-(d-1)\Delta}} & \frac{\partial g}{\partial y_{t-d\Delta}} \\ 1 & 0 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & 0 \\ 0 & 0 & \cdots & 1 & 0 \end{pmatrix} \times \begin{pmatrix} \xi_{t-\Delta} \\ \xi_{t-2\Delta} \\ \vdots \\ \xi_{t-(d-1)\Delta} \\ \xi_{t-d\Delta} \end{pmatrix} \tag{2.18}$$

where the partial derivatives of $g(Y; \mathbf{c})$ are evaluated at the fixed point $(Y_o, \mathbf{c}_o)$. It can be shown that the eigenvalues of $D_y\mathbf{F}(Y_o, \mathbf{c}_o)$, that is, $\det|D_y\mathbf{F}(Y_o, \mathbf{c}_o) - \gamma| = 0$, are equal to the roots of a polynomial of degree d and coefficients which are equal to the partial derivatives of $g(Y; \mathbf{c})$ evaluated at the fixed point, that is,

$$P(\gamma) = \sum_{i=0}^{d} a_i \gamma^{d-i} \tag{2.19}$$

where $a_0 = -1$ and $a_i = \partial g(Y_o; \mathbf{c}_o)/\partial y_{t-i\Delta}$ for $i = 1, \cdots, d$. When the fixed point is *hyperbolic*, that is, none of the roots of $P(\gamma)$ have unit modulus, the stability of the linearized equation implies the stability of the predictor function. This means that if all the roots of $P(\gamma)$ lie within the unit circle in the complex plane, the fixed point $(Y_o, \mathbf{c}_o)$ is stable; otherwise, it is unstable. When the fixed point is *nonhyperbolic*, the stability of the map cannot be determined from the stability of the linearized equation. Moreover, changes in $\mathbf{c}$ can result to bifurcations. The simplest ways in which a fixed point of a map can be nonhyperbolic are the following:

1. $P(\gamma)$ has a single root equal to 1 with the remaining $d-1$ roots having moduli not equal to 1.

2. $P(\gamma)$ has a single root equal to -1 with the remaining $d-1$ roots having moduli not equal to 1.

3. $P(\gamma)$ has two complex conjugate roots having modulus 1 (which are not one of the first four roots of unity) with the remaining $d-2$ roots having moduli not equal to 1.

In the following, the possible bifurcations that can take place when the above conditions hold are enumerated. Without loss of generality, the analysis can be reduced to the study of a parametrized family of maps on the one-dimensional center manifold. Refer to Wiggins [84] for a detailed discussion.

The saddle-node bifurcation. A general one-parameter family of $\mathbf{C}^r$ $(r \geq 2)$ one-dimensional maps $x \mapsto f(x,\mu)$, $x \in \mathbb{R}^1$, $\mu \in \mathbb{R}^1$ undergoes a *saddle-node* bifurcation at $(x,\mu) = (0,0)$ if

$$f(0,0) = 0, \tag{2.20}$$
$$\frac{\partial f}{\partial x}(0,0) = 1, \tag{2.21}$$
$$\frac{\partial f}{\partial \mu}(0,0) \neq 0, \tag{2.22}$$
$$\frac{\partial^2 f}{\partial x^2}(0,0) \neq 0. \tag{2.23}$$

The transcritical bifurcation. A general one-parameter family of $\mathbf{C}^r$ $(r \geq 2)$ one-dimensional maps $x \mapsto f(x,\mu)$, $x \in \mathbb{R}^1$, $\mu \in \mathbb{R}^1$ undergoes a *transcritical* bifurcation at $(x,\mu) = (0,0)$ if

$$f(0,0) = 0, \tag{2.24}$$
$$\frac{\partial f}{\partial x}(0,0) = 1, \tag{2.25}$$
$$\frac{\partial f}{\partial \mu}(0,0) = 0, \tag{2.26}$$
$$\frac{\partial^2 f}{\partial x \partial \mu}(0,0) \neq 0, \tag{2.27}$$
$$\frac{\partial^2 f}{\partial x^2}(0,0) \neq 0. \tag{2.28}$$

The pitchfork bifurcation. A general one-parameter family of $\mathbf{C}^r$ $(r \geq 3)$ one-dimensional maps $x \mapsto f(x,\mu)$, $x \in \mathbb{R}^1$, $\mu \in \mathbb{R}^1$ undergoes a *pitchfork* bifurcation at $(x,\mu) = (0,0)$ if

$$f(0,0) = 0, \tag{2.29}$$
$$\frac{\partial f}{\partial x}(0,0) = 1, \tag{2.30}$$
$$\frac{\partial f}{\partial \mu}(0,0) = 0, \tag{2.31}$$
$$\frac{\partial^2 f}{\partial x^2}(0,0) = 0, \tag{2.32}$$
$$\frac{\partial^2 f}{\partial x \partial \mu}(0,0) \neq 0, \tag{2.33}$$
$$\frac{\partial^3 f}{\partial x^3}(0,0) \neq 0. \tag{2.34}$$

The period-doubling bifurcation. A general one-parameter family of $\mathbf{C}^r$ $(r \geq 3)$ one-dimensional maps $x \mapsto f(x,\mu)$, $x \in \mathbb{R}^1$, $\mu \in \mathbb{R}^1$ undergoes a *pitchfork* bifurcation at $(x,\mu) = (0,0)$ if

$$\begin{aligned} f(0,0) &= 0, && (2.35)\\ \frac{\partial f}{\partial x}(0,0) &= -1, && (2.36)\\ \frac{\partial f^2}{\partial \mu}(0,0) &= 0, && (2.37)\\ \frac{\partial^2 f^2}{\partial x^2}(0,0) &= 0, && (2.38)\\ \frac{\partial^2 f^2}{\partial x \partial \mu}(0,0) &\neq 0, && (2.39)\\ \frac{\partial^3 f^2}{\partial x^3}(0,0) &\neq 0, && (2.40) \end{aligned}$$

where f^2 denotes the *second iterate* of f.

The Naimark-Sacker (Hopf) bifurcation. Consider a one-parameter family of maps $x \mapsto f(x,\mu)$, $x \in \mathbb{R}^2$, $\mu \in \mathbb{R}^1$ having a fixed point at $(x,\mu) = (0,0)$, i.e., $f(0,0) = 0$, with the matrix $D_x f(0,0)$ having two complex conjugate eigenvalues $\lambda(0)$ and $\bar{\lambda}(0)$, with $|\lambda(0)| = 1$. Moreover, let the eigenvalues satisfy

$$\lambda(0)^n \neq 1, \quad n = 1,2,3,4. \tag{2.41}$$

Under these conditions, the truncated normal form of $f(x,\mu)$ can be written in polar coordinates as

$$\begin{aligned} r &\mapsto r + (d\mu + ar^2)r,\\ \theta &\mapsto \theta + \phi_0 + \phi_1 \mu + br^2. \end{aligned} \tag{2.42}$$

Note that $r = 0$ is a fixed point of Eq. (2.42) that is

asymptotically stable	for $d\mu < 0$
unstable	for $d\mu > 0$
unstable	for $\mu = 0, \quad a > 0$
asymptotically stable	for $\mu = 0, \quad a < 0.$

The following lemmas are proved in [84]:

Lemma 1: $\left\{(r,\theta) \in I\!R^+ \times S^1 | r = \sqrt{\frac{-\mu d}{a}}\right\}$ *is a circle which is invariant under the dynamics generated by Eq. (2.42).*

Lemma 2: *The invariant circle is asymptotically stable for* $a < 0$ *and unstable for* $a > 0$.

Finally, the four possible cases for the bifurcation of an invariant circle from a fixed point are listed below:

Case 1: $d > 0$, $a > 0$. In this case, the origin is an unstable fixed point for $\mu > 0$ and an asymptotically stable fixed point for $\mu < 0$ with an unstable invariant circle for $\mu < 0$.

Case 2: $d > 0$, $a < 0$. In this case, the origin is an unstable fixed point for $\mu > 0$ and an asymptotically stable fixed point for $\mu < 0$ with an asymptotically stable invariant circle for $\mu > 0$.

Case 3: $d < 0$, $a > 0$. In this case, the origin is an asymptotically stable fixed point for $\mu > 0$ and an unstable fixed point for $\mu < 0$ with an unstable invariant circle for $\mu > 0$.

Case 4: $d < 0$, $a < 0$. In this case, the origin is an asymptotically stable fixed point for $\mu > 0$ and an unstable fixed point for $\mu < 0$ with an asymptotically stable invariant circle for $\mu < 0$.

The two sections in this chapter outline the possible approach to analyze bifurcation structure from time series. Roughly, the initial task is to reconstruct the phase-space of the dynamics. This involves the estimation of the embedding dimension and the time delay. An appropriate predictor function can then be used to specify the dynamics in the reconstructed phase-space. The bifurcation structure of the predictor function is then analyzed. This means extracting the bifurcation points and classifying these points according to the above definitions. A bifurcation diagram can then be constructed to reflect the different bifurcations of the system. Note that the resulting BD will depend on the predictor function employed. Choosing an appropriate predictor function for the given time series will be the topic of the next chapter.

Chapter 3

Model Identification

3.1 Introduction

In the past decades, a wide range of theoretical models have been proposed and a number of methods have been presented for the reconstruction of the system dynamics from observation data. Several commonly used models include linear autoregression (AR) models, nonlinear autoregressive (NAR) models, neural network (NN) models, radial basis function (RBF) models, among others. Each of these models has its own advantages and disadvantages. In this chapter, the different models used in this study will be introduced. The use of the likelihood function to estimate model parameter values for chaotic time series will be investigated in section 3.2. This method can be applied when a model is already known and it is desired to estimate the associated parameters for the given time series. The application of the NAR and NN models as predictor functions will be discussed in sections 3.3 and 3.4, respectively. In the last section, NAR models are used for the identification of experimental data taken from the electrical discharge of electric fish. The emphasis is on the detection of nonlinear behavior in the observed data. The examples that follow for each model are not intended for comparison purposes but rather as specific examples of the model's application. Thus, different systems are used for each model.

3.2 The Likelihood Function

One important issue to consider in using the likelihood function for parameter estimation is the behavior of the likelihood function itself. Note that the estimate of the parameter by maximum likelihood method is of questionable value unless the likelihood function is unimodal and concentrated near the mode. The first thing to consider therefore is to examine if the

likelihood function is indeed well-behaved near the true parameter values for chaos models. The present claim so far is that "chaos models lead to chaotic likelihood functions"[13]. However in this section, a counter example showing the contrary is presented. Numerical integration is done by employing *local linearization method* [74, 75]. For the calculation of the likelihood function, the *innovations approach* [33] is employed. Both the theoretical formalism based on [64] and [33] and the results of the numerical simulations are presented.

The problem at hand can be posed as follows: Consider the continuous stochastic dynamical system described by the vector (Itô) stochastic differential equation

$$d\mathbf{x}(t) = \mathbf{f}[\mathbf{x}(t), t]dt + \mathbf{G}[\mathbf{x}(t), t]d\beta(t), \quad t \geq t_o \tag{3.1}$$

where $\mathbf{x}(t)$ and $\mathbf{f}[\mathbf{x}(t), t]$ are n-vectors, $\mathbf{G}[\mathbf{x}(t), t]$ is an $n \times r$ matrix, and $\{\beta(t),\ t > t_o\}$ is an r-vector Brownian motion process with $E\left[d\beta(t)d\beta(t)^T\right] = Q(t)dt$[1]. Suppose that continuous observations are taken and is given by

$$d\mathbf{z}(t) = \mathbf{h}[\mathbf{x}(t), t]dt + d\eta(t) \tag{3.2}$$

where $\mathbf{z}(t)$ and $\mathbf{h}[\mathbf{x}(t), t]$ are m-vectors, and $\{\eta(t), t \geq t_o\}$ is a Brownian motion process with $E\left[d\eta(t)d\eta(t)^T\right] = R(t)dt$. Let $Y(\tau)$ be a realization of the observation, that is,

$$Y(\tau) = \{z(s),\ t_o \leq s \leq \tau\}. \tag{3.3}$$

The continuous estimation problem consists of computing an estimate of the unobserved state $\mathbf{x}(t)$ based on the realization $Y(\tau)$. Furthermore, for parametrized models, this can be extended to estimating the parameters using the same observation data. The problem therefore is twofold: to estimate the parameter values of the known model and to estimate the unobserved state of the system.

The complete solution of the estimation problem can be obtained if the *conditional probability density* $p[\mathbf{x}, t|Y(\tau)]$ of $\mathbf{x}(t)$, given $Y(\tau)$, is known. This is because $p[\mathbf{x}, t|Y(\tau)]$ embodies all statistical information about $\mathbf{x}(t)$ which is contained in the available observation and the initial condition $p(\mathbf{x}, t_o)$. Note that for $t = \tau$, the problem is a *filtering problem*, for $t < \tau$, a *smoothing problem*, and for $t > \tau$, a *prediction problem*. It is also well-known that the optimal estimate for the unobserved state $\mathbf{x}(t)$ is given by the *conditional mean* $\hat{\mathbf{x}}(t|\tau) = E\left[\mathbf{x}(t)|Y(\tau)\right]$ [47]. Thus the estimation problem involves the determination of both the conditional probability density

[1]The notation $E[\cdot]$ will be used to denote statistical expectation, that is, if X is a random variable and $f(x)$ its density, then $E[X] = \int_{-\infty}^{\infty} xf(x)dx$.

and the conditional mean. For nonlinear cases, it is difficult to obtain these two values since the distribution of the observed data is not always known. However, a method called the innovations approach will facilitate the computation of these two quantities from the observed data. This will be discussed in the next subsection.

3.2.1 The innovations approach

Suppose that observations (data) of the form

$$\mathbf{z}(t) = \mathbf{y}(t) + \mathbf{v}(t),\ 0 \leq t \leq T, \tag{3.4}$$

where $\mathbf{y}(t)$ and $\mathbf{v}(t)$ are m-dimensional statistically independent vector processes, are given. Furthermore, $\mathbf{v}(t)$ is a white Gaussian noise with the following properties:

$$E[\mathbf{v}(t)] = 0 \tag{3.5}$$

$$E\left[\mathbf{v}(t)\mathbf{v}^T(s)\right] = I\delta(t-s). \tag{3.6}$$

Also, the signal process, $\mathbf{y}(t)$, which is not necessarily Gaussian, is characterized by the following:

$$E[\mathbf{y}(t)] = 0, \tag{3.7}$$

$$\int_0^T E\left[\mathbf{y}(t)\mathbf{y}^T(t)\right] dt < \infty, \tag{3.8}$$

or more strictly

$$|\mathbf{y}(t)| \leq M < \infty,\ 0 \leq t \leq T. \tag{3.9}$$

As mentioned earlier, the estimation problem involves the approximation of a random process, say $\mathbf{x}(t)$, satisfying Eq. (3.1) from the observation $\mathbf{z}(t)$. Also, $\mathbf{x}(t)$ is related to the signal process $\mathbf{y}(t)$ via

$$\mathbf{y}(t) = \mathbf{h}[\mathbf{x}(s),\ s \leq t] \tag{3.10}$$

and obeys the following properties:

$$E[\mathbf{x}(t)] = 0 \tag{3.11}$$

$$\int_0^T E\left[\mathbf{x}(t)\mathbf{x}^T(t)\right] dt < \infty. \tag{3.12}$$

The problem is to find the optimal estimate $\hat{\mathbf{x}}(t|\tau)$ of $\mathbf{x}(t)$ given $\{\mathbf{z}(s),\ 0 \leq s < \tau \leq T\}$ which is readily obtained from the conditional mean of $\mathbf{x}(t)$, that is,

$$\hat{\mathbf{x}}(t|\tau) = E[\mathbf{x}(t)|\mathbf{z}(s),\ 0 \leq s < \tau]. \tag{3.13}$$

See for example Neveu [60] and Doob [27]. However, explicit formulas for $\hat{\mathbf{x}}(t|\tau)$ are not readily available since in general the distribution of $\mathbf{y}(t)$ is not known. It is therefore important to find more explicit formulas for an optimal estimate of $\mathbf{x}(t)$.

In the innovations approach, the observation process $\mathbf{z}(t)$ is transformed into a white Gaussian noise called the *innovations process* $\nu(\cdot)$. This transformation can be used to obtain the estimate

$$E[\mathbf{x}(t)|\nu(s),\ 0 \leq s < \tau] \tag{3.14}$$

since the distribution of the innovations is known. Furthermore, the innovations process $\nu(t)$ and the observation process $\mathbf{z}(t)$ are equivalent in the sense that there exists a causal and causally invertible transformation from $\{\mathbf{z}(s),\ 0 \leq s < \tau \leq T\}$ to $\{\nu(s),\ 0 \leq s < \tau \leq T\}$ such that

$$\hat{\mathbf{x}}(t|\tau) \doteq E[\mathbf{x}(t)|\mathbf{z}(s),\ 0 \leq s < \tau] \tag{3.15}$$

$$= E[\mathbf{x}(t)|\nu(s),\ 0 \leq s < \tau]. \tag{3.16}$$

The optimal estimate $\hat{\mathbf{x}}(t|\tau)$ in terms of the innovations is now given by

$$\hat{\mathbf{x}}(t|\tau) = \int_0^\tau E\left[\mathbf{x}(t)\nu^T(s)|\nu(\sigma),\ 0 \leq \sigma < s\right]\nu(s)ds \tag{3.17}$$

where

$$\nu(t) = \mathbf{z}(t) - \tilde{\mathbf{y}}(t|\tau),\ 0 \leq t < \tau \tag{3.18}$$

and $\tilde{\mathbf{y}}(t|t)$ is the least-square estimate of $\mathbf{y}(t)$ in terms of $\{\mathbf{z}(s),\ 0 \leq s < \tau\}$. Note that the above integral is a special integral called an *Itô integral.* The proofs are discussed in [33].

3.2.2 Derivation

The use of the likelihood function in parameter estimation from observation data has been widely known. The idea is that the observed data is fitted to a known model by varying the model parameters. The parameters that yield the maximum likelihood are chosen to be the best estimate. Thus, this approach involves the optimization of the likelihood function of the observed data relative to the model parameters. However, it is always difficult to obtain the likelihood function for a given observation since in most cases the distribution of the observed data is not known. In this case, the use of the innovations approach becomes handy by converting the observation process to the innovations process with a known distribution as discussed in the previous subsection.

A special case of Eq. (3.2) is when the observation process is linear in $\mathbf{x}(t)$, that is, $\mathbf{h}[\mathbf{x}(t), t] = \mathbf{C}\mathbf{x}(t)$ where $\mathbf{C}$ is called the *observation matrix*. Rewriting Eq. (3.2) yields

$$d\mathbf{z}(t) = \mathbf{C}\mathbf{x}(t) + \sigma_\epsilon d\mathbf{w}_o(t) \tag{3.19}$$

where $d\mathbf{w}_o(t)$ is Gaussian with unit variance. If $d\mathbf{z}(t_1), d\mathbf{z}(t_2), \cdots, d\mathbf{z}(t_n)$ are taken as samples from this model, the probability density $p(d\mathbf{z}_1, d\mathbf{z}_2, \cdots, d\mathbf{z}_n|\theta)$, where θ is some parameter, is difficult to obtain since the distribution of $\mathbf{z}(t)$ is not known when $\mathbf{f}(\cdot)$ in Eq. (3.1) is nonlinear. However, using the relation

$$p(d\mathbf{z}_1, d\mathbf{z}_2, \cdots, d\mathbf{z}_n|\theta) = p(d\mathbf{z}_n|d\mathbf{z}_{n-1}, \cdots, d\mathbf{z}_1, \theta)p(d\mathbf{z}_{n-1}, d\mathbf{z}_{n-2}, \cdots, d\mathbf{z}_1|\theta) \tag{3.20}$$

and taking the log of the probability density, a relation between the original density in terms of the relative densities can be written as follows:

$$\log p(d\mathbf{z}_1, d\mathbf{z}_2, \cdots, d\mathbf{z}_n|\theta) = \sum \log p(d\mathbf{z}_i|d\mathbf{z}_{i-1}, \cdots, d\mathbf{z}_1, \theta). \tag{3.21}$$

Upto this part, the derivation applies for a general case. Note that since the distribution of $\mathbf{z}(t)$ is not known, it is difficult to obtain the explicit form of the relative probability densities on the right hand side of Eq. (3.21) making the likelihood approach difficult to apply for nonlinear cases. However, with the use of the innovations approach and Eq. (3.18), Eq. (3.19) can now be written in terms of the innovation as

$$d\mathbf{z}(t) = \mathbf{C}\tilde{\mathbf{x}}(t|\tau)dt + d\nu(t) \tag{3.22}$$

without affecting the problem in any way. $d\nu(t)$ represents the innovation and $\mathbf{C}\tilde{\mathbf{x}}(t|\tau)$ is the least squares estimate of $\mathbf{C}\mathbf{x}(t|\tau)$ in terms of $\{\mathbf{z}(s),\ 0 \leq s < \tau\}$. With this, the distribution of $d\mathbf{z}(t_k)$, which is the sampled version of $d\mathbf{z}(t)$, can now be obtained. Since the term $\mathbf{C}\tilde{\mathbf{x}}(t_k)dt_k$ is by definition a constant, the mean is simply given by $E[d\mathbf{z}(t_k)] = \mathbf{C}\tilde{\mathbf{x}}(t_k)dt_k$ and the variance is also given by $\mathrm{Var}[d\mathbf{z}(t_k)] = \sigma_\epsilon^2 dt_k$.

Discretizing the above relation, one obtains

$$\mathbf{z}_k = \mathbf{C}\mathbf{x}_{k|k-1} + \nu_k \tag{3.23}$$

where $\mathbf{z}_k = \Delta\mathbf{z}(t_k)$ which approximates $d\mathbf{z}(t_k)$, $\mathbf{C}\mathbf{x}_{k|k-1} = E[\mathbf{C}\mathbf{x}_k|\mathbf{z}_{k-1}, \cdots, \mathbf{z}_1]$ which approaches $\mathbf{C}\tilde{\mathbf{x}}(t_k)dt_k$ as Δt approaches zero, and ν_k is a discrete-time Gaussian white noise with variance $\sigma^2_{k|k-1} = E[\mathbf{z}_k - \mathbf{C}\mathbf{x}_{k|k-1}]^2$ which approaches $\sigma_\epsilon^2 dt_k$ for very small Δt. The conditional density of ν_k can now be written as follows

$$p(\nu_k|\mathbf{z}_{k-1}, \cdots, \mathbf{z}_1, \theta) = \frac{1}{\sqrt{2\pi\sigma^2_{k|k-1}}} \exp\left[-\frac{(\mathbf{z}_k - \mathbf{C}\mathbf{x}_{k|k-1})^2}{2\sigma^2_{k|k-1}}\right]. \tag{3.24}$$

Using the equivalence relation of $\mathbf{z}_k$ and ν_k, then

$$\sum(-2)\log p(\mathbf{z}_k|\mathbf{z}_{k-1},\cdots,\mathbf{z}_1,\theta) = \sum(-2)\log p(\nu_k|\mathbf{z}_{k-1},\cdots,\mathbf{z}_1,\theta) \tag{3.25}$$

Equation (3.24) together with Eq. (3.25) can now be used to find more explicit formulas for the relative probabilities of $\mathbf{z}_k$. This leads to the following relations

$$\begin{aligned}(-2)\log p(\mathbf{z}_1,\mathbf{z}_2,\cdots,\mathbf{z}_n|\theta) &= \sum(-2)\log p(\mathbf{z}_k|\mathbf{z}_{k-1},\cdots,\mathbf{z}_1,\theta)\\ &= \sum\left[\log\sigma^2_{k|k-1}+\frac{(\mathbf{z}_k-\mathbf{C}\mathbf{x}_{k|k-1})^2}{\sigma^2_{k|k-1}}\right]+n\log 2\pi\end{aligned}$$

from which the likelihood function can be computed. For this result to be useful in estimating the likelihood function, the conditional variance $\sigma^2_{k|k-1}$ and mean $\mathbf{x}_{k|k-1}$ should be estimated first. To do this, a nonlinear Kalman filtering scheme is employed.

3.2.3 Local linearization scheme

The state representation of the stochastic process given by Eq. (3.1) can be written using *local linearization scheme* [74] as

$$\mathbf{x}_{t+1} = \mathbf{A}(\mathbf{x}_t)\mathbf{x}_t + \mathbf{B}(\mathbf{x}_t)\sigma_w\mathbf{w}_{t+1} \tag{3.26}$$

where

$$\mathbf{A}(\mathbf{x}_t) = 1 + \mathbf{J}_t^{-1}\{e^{\mathbf{J}_t\Delta t}-1\}\frac{\mathbf{f}(\mathbf{x}_t)}{\mathbf{x}_t} + \mathbf{J}_t^{-2}\{e^{\mathbf{J}_t\Delta t}-1-\mathbf{J}_t\Delta t\}\frac{\sigma_w^2\mathbf{H}_t}{2\mathbf{x}_t} \tag{3.27}$$

$$\mathbf{B}(\mathbf{x}_t) = \sqrt{\frac{e^{2\mathbf{J}_t\Delta t}-1}{2\mathbf{J}_t}}. \tag{3.28}$$

To compute for the likelihood, one needs to find estimates of $\mathbf{x}_{k|k-1}$ and $\sigma^2_{k|k-1}$ denoted by $\hat{\mathbf{x}}_{k|k-1}$ and $\hat{\sigma}^2_{k|k-1}$ respectively. To do this, the relative expectation of Eq. (3.26) is taken:

$$\mathbf{x}_{k|k-1} = \int \mathbf{x}_k p(\mathbf{x}_k|\mathbf{z}_{k-1},\cdots,\mathbf{z}_1,\theta)d\mathbf{x}_k \tag{3.29}$$

where $p(\mathbf{x}_k|\mathbf{z}_{k-1},\cdots,\mathbf{z}_1,\theta)$ is the *state prediction density function.* This has no exact formula. However, $\mathbf{A}(\mathbf{x}_{k-1})\mathbf{x}_{k-1|k-1}$ could be taken as a reasonable estimate where

$$\mathbf{x}_{k-1|k-1} = \int \mathbf{x}_{k-1} p(\mathbf{x}_{k-1}|\mathbf{z}_{k-1},\cdots,\mathbf{z}_1,\theta)d\mathbf{x}_{k-1} \tag{3.30}$$

and $p(\mathbf{x}_{k-1}|\mathbf{z}_{k-1},\cdots,\mathbf{z}_1,\theta)$ is the *filtered density.* The estimate can now be written as

$$\hat{\mathbf{x}}_{k|k-1} = \mathbf{A}(\mathbf{x}_{k-1})\hat{\mathbf{x}}_{k-1|k-1} \tag{3.31}$$

and the innovation at a point k is

$$\hat{\nu}_k = \mathbf{z}_k - \mathbf{C}\mathbf{A}(\mathbf{x}_{k-1})\hat{\mathbf{x}}_{k-1|k-1}. \tag{3.32}$$

A filtering procedure is used to carry out the approximation approach, that is, to obtain $\hat{\mathbf{x}}_{k|k}$ from the previous prediction $\hat{\mathbf{x}}_{k|k-1}$ and $\hat{\nu}_k$ by using the relation

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\hat{\nu}_k \tag{3.33}$$

where $\mathbf{K}_k$ is the *correction gain matrix* obtained by minimizing the estimation error variance of the state, that is, trace$E\left[(\mathbf{x}_k - \mathbf{x}_{k|k})(\mathbf{x}_k - \mathbf{x}_{k|k})^T\right]$. After several matrix manipulation, the value of $\mathbf{K}_k$ which minimizes the variance is given by the relation

$$\mathbf{K}_k = \mathbf{P}_k\mathbf{C}^T(\mathbf{C}\mathbf{P}_k\mathbf{C}^T + \sigma_\epsilon^2)^{-1} \tag{3.34}$$

where $\mathbf{P}_k = E\left[(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k-1})^T\right]$. Furthermore, let $\mathbf{V}_k = E\left[(\mathbf{x}_k - \mathbf{x}_{k|k})(\mathbf{x}_k - \mathbf{x}_{k|k})^T\right]$ then

$$\mathbf{V}_k = \mathbf{P}_k - \mathbf{K}_k\mathbf{C}\mathbf{P}_k. \tag{3.35}$$

The evolution of the $\mathbf{P}_k$ operator is obtained via

$$\begin{aligned}\mathbf{P}_{k+1} &= E\left[(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k})(\mathbf{x}_{k+1} - \hat{\mathbf{x}}_{k+1|k})^T\right] \\ &= E\left[\{\mathbf{A}(\mathbf{x}_k)(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}) + \mathbf{B}(\mathbf{x}_k)\sigma_w\mathbf{w}_{k+1}\}\{\mathbf{A}(\mathbf{x}_k)(\mathbf{x}_k - \hat{\mathbf{x}}_{k|k}) + \mathbf{B}(\mathbf{x}_k)\sigma_w\mathbf{w}_{k+1}\}^T\right].\end{aligned}$$

Assuming $\mathbf{A}(\mathbf{x}_k)$ is constant on $[k, k+1)$ then

$$\mathbf{P}_{k+1} = \mathbf{A}(\mathbf{x}_k)\mathbf{V}_k\mathbf{A}(\mathbf{x}_k)^T + \mathbf{B}(\mathbf{x}_k)\Sigma_w\mathbf{B}(\mathbf{x}_k)^T. \tag{3.36}$$

$\mathbf{A}(\mathbf{x}_k)$ and $\mathbf{B}(\mathbf{x}_k)$ can be approximated by $\mathbf{A}(\hat{\mathbf{x}}_{k|k})$ and $\mathbf{B}(\hat{\mathbf{x}}_{k|k})$ respectively. Then the actual value of the prediction estimate is calculated by using the following recursion relations:

$$\hat{\mathbf{x}}_{k+1|k} = \mathbf{A}(\hat{\mathbf{x}}_{k|k})\hat{\mathbf{x}}_{k|k} \tag{3.37}$$

$$\hat{\mathbf{x}}_{k|k} = \hat{\mathbf{x}}_{k|k-1} + \mathbf{K}_k\hat{\nu}_k \tag{3.38}$$

$$\mathbf{K}_k = \mathbf{P}_k\mathbf{C}^T(\mathbf{C}\mathbf{P}_k\mathbf{C}^T + \sigma_\epsilon^2)^{-1} \tag{3.39}$$

$$\mathbf{P}_{k+1} = \mathbf{A}(\hat{\mathbf{x}}_{k|k})\mathbf{V}_k\mathbf{A}(\hat{\mathbf{x}}_{k|k})^T + \mathbf{B}(\hat{\mathbf{x}}_{k|k})\Sigma_w\mathbf{B}(\hat{\mathbf{x}}_{k|k})^T \tag{3.40}$$

$$\mathbf{V}_k = \mathbf{P}_k - \mathbf{K}_k\mathbf{C}\mathbf{P}_k \tag{3.41}$$

$$\hat{\sigma}_{k|k-1}^2 = \mathbf{C}\mathbf{P}_k\mathbf{C}^T + \sigma_\epsilon^2 \tag{3.42}$$

with initial conditions $x_{o|o}$ and V_o. The variable θ specifies the system parameters, system error variance and observation error variance.

3.2.4 Example: the Lorenz equations

In this section, the method presented above is applied to estimate the parameters of the Lorenz equation [57]:

$$\begin{aligned} \frac{dx}{dy} &= -ax + ay \\ \frac{dy}{dt} &= -xz + rx - y \\ \frac{dz}{dt} &= xy - bz \end{aligned} \tag{3.43}$$

given a time series generated from it. Integration was done by using the local linearization scheme as given by Eq. (3.26). The system noise variance was set to zero. Noise was added to the observation data with variance σ_ϵ equal to 0.0001. The observation matrix is given by $\mathbf{C} = (1\ 0\ 0)^T$ to observe only the z-component of the signal. The values of the parameters used were $a = 10$, $b = 8/3$, and $r = 28$. The observation data is shown in Fig. 3.1. The behavior of the likelihood function near the true parameter values was investigated. This was done by varying one parameter while fixing the value of the other parameters.

Although the state of the system was generated without system noise, the numerical algorithm used in integrating Eq. (3.43) may possibly affect the computed state especially for chaos models. To verify this, the behavior of the likelihood function as a function of the system noise variance was investigated. This means identifying the preferred model of the maximum likelihood approach when numerical errors are present. This was done by setting all other parameter values to their true values and varying the system noise variance $\hat{\sigma}_w$ in estimating the likelihood function. Ideally, the maximum is expected to occur at zero variance since the data was generated using this value. However, if artificial noise is introduced due to numerical integration, then the likelihood will peak at a different value.

Figure 3.2 shows the behavior of the prediction error as a function of the approximated noise variance. From the figure, as the variance of the system noise is increased, there is a corresponding decrease in the prediction error especially near the initial position. This implies that although the time series was generated without system noise, the maximum likelihood approach prefers a model in which the system noise variance is nonzero. This behavior can be attributed to the errors introduced by the numerical method used in integrating the differential equation.

Another important issue to consider is the choice of the values of x_o and $v_{o|o}$ which correspond to the initial values in the iterative filtering scheme. The effect of the initial values is shown in

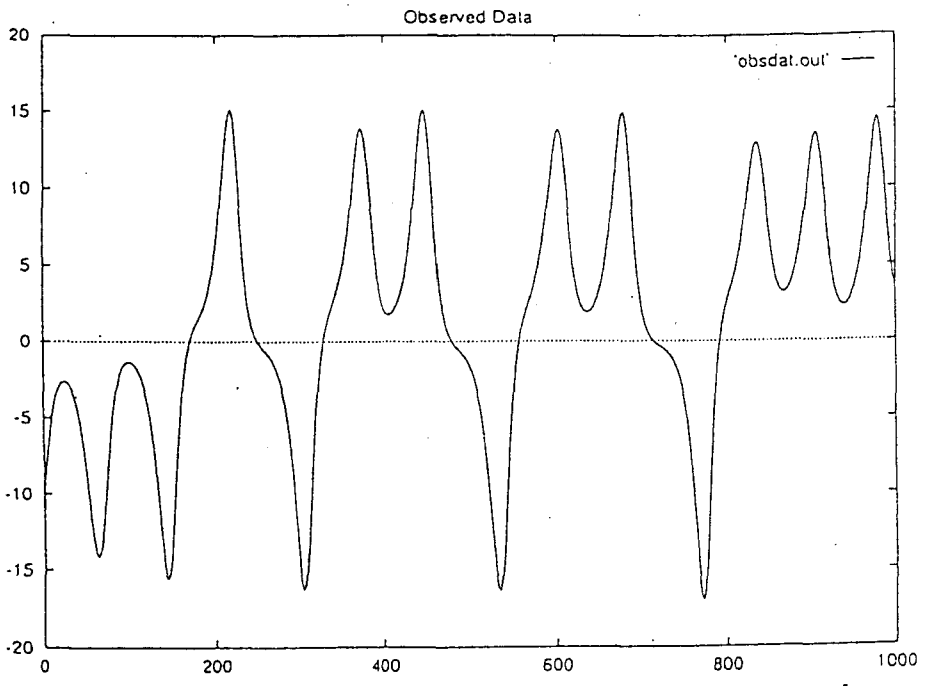

Figure 3.1: Observation data used in the analysis generated from the Lorenz system with $a = 10$, $b = 8/3$, and $r = 28$.

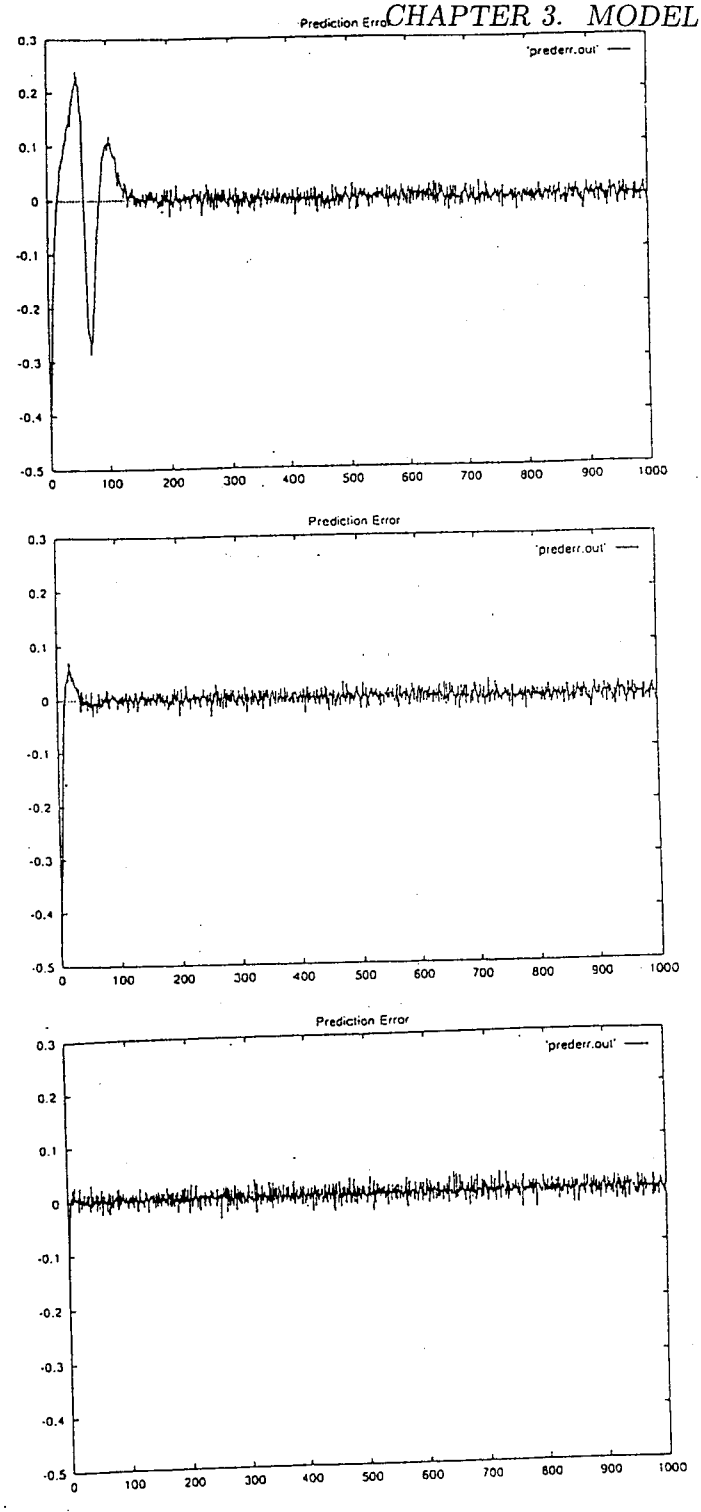

Figure 3.2: Prediction error with varying noise variance. $\sigma_\varepsilon = 10^{-4}$, $\sigma_w = 0.0$, and $x_o = -10$. Top: $\hat{\sigma}_w = 10^{-8}$. Middle: $\hat{\sigma}_w = 10^{-4}$. Bottom: $\hat{\sigma}_w = 10^{-2}$.

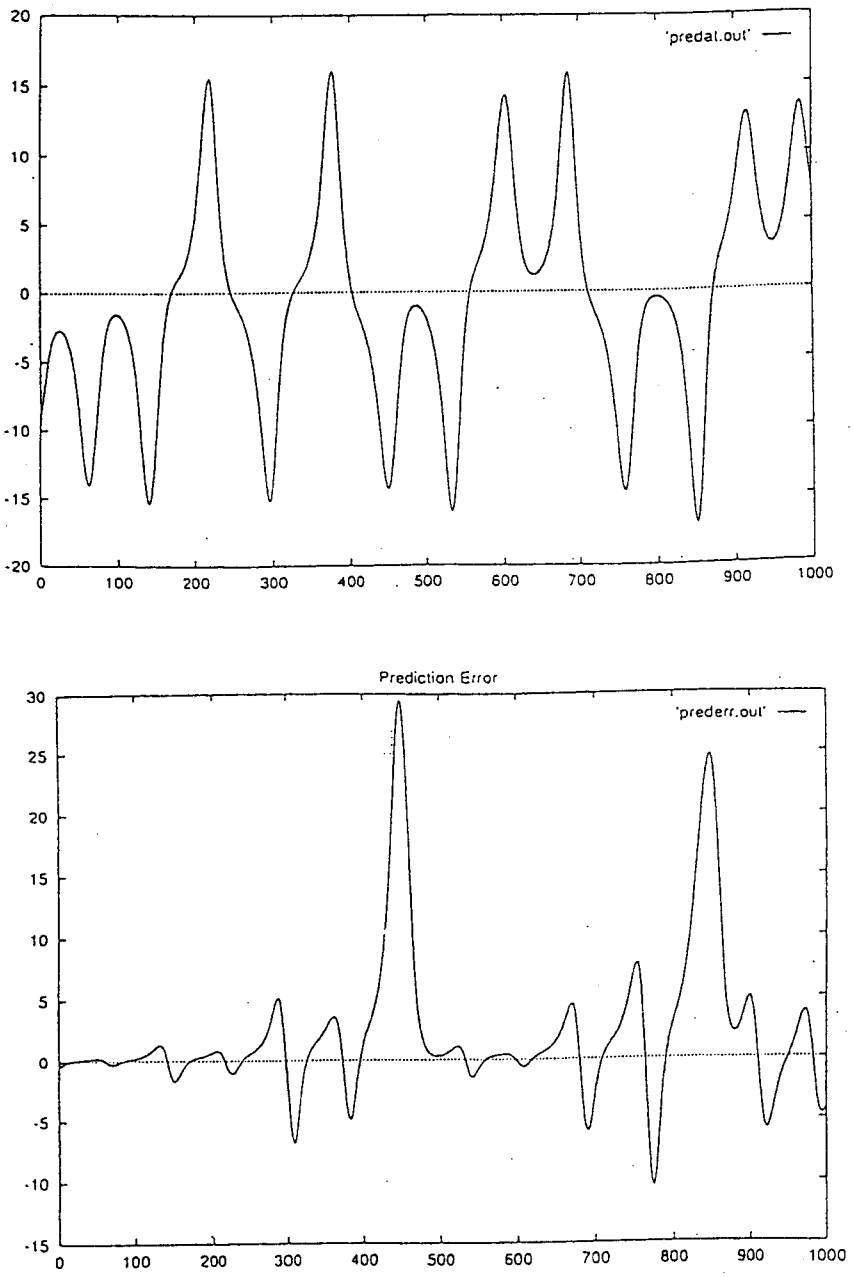

Figure 3.3: Effects of the initial values. Top: The behavior of the predicted value when the initial values of x_o and $v_{o|o}$ are chosen inaccurately. Bottom: Difference between the observed series and the predicted series shown in the top figure.

Fig. 3.3. Figure 3.3(top) shows the behavior of the predicted value when the initial values of x_o and $v_{o|o}$ are chosen inaccurately. The difference, which is relatively significant, between the time series and the predicted data is shown in Fig. 3.3(bottom). Since these two values also affect the computation of the likelihood function and in most cases, are always unknown, then these can be added as parameters to the system to be approximated using the same approach.

Figure 3.4 shows the log-likelihood function for different initial positions as the system noise variance is varied. x_o in the figure refers to the true initial position. Note that for initial positions different from the true one, the log-likelihood function minimizes at different noise variance. This means that poor choice of the initial position will lead to poor estimate of the true system noise variance. However, a closer examination of the plot reveals that the true initial value gave the most minimum log-likelihood. This is clearly evident in Fig. 3.5 where the minima in Fig. 3.4 are plotted against the initial position. Here, the minimum occurs very near to the true initial position value. Furthermore, a plot of the estimated variance vs. the value of the initial position (Fig. 3.6) reveals that the minimum variance is very close to the true variance and occurs at x_o.

This section illustrates how to get the log-likelihood function for nonlinear systems. The results are in complete agreement with the results of [64] also using the same example. They demonstrate the applicability of the log-likelihood function to estimate the model parameters, including the initial values of the state of the system and the system noise, even for chaotic nonlinear systems.

3.3 Neural Networks

In this section, the neural network paradigm is discussed and its merits to approximate the predictor function $g(\cdot)$ are closely examined. The use of neural networks is motivated by the following reasons:

- Neural networks are inherently nonlinear. In general, time series data are generated by nonlinear dynamical systems, therefore neural networks are good candidate models.

- Neural networks solve a given task by "learning" the solution [67] from a known set of examples called the *training set*. In this case, the available time series will serve as the training set and the network is tasked to learn the underlying mechanism that generates the data.

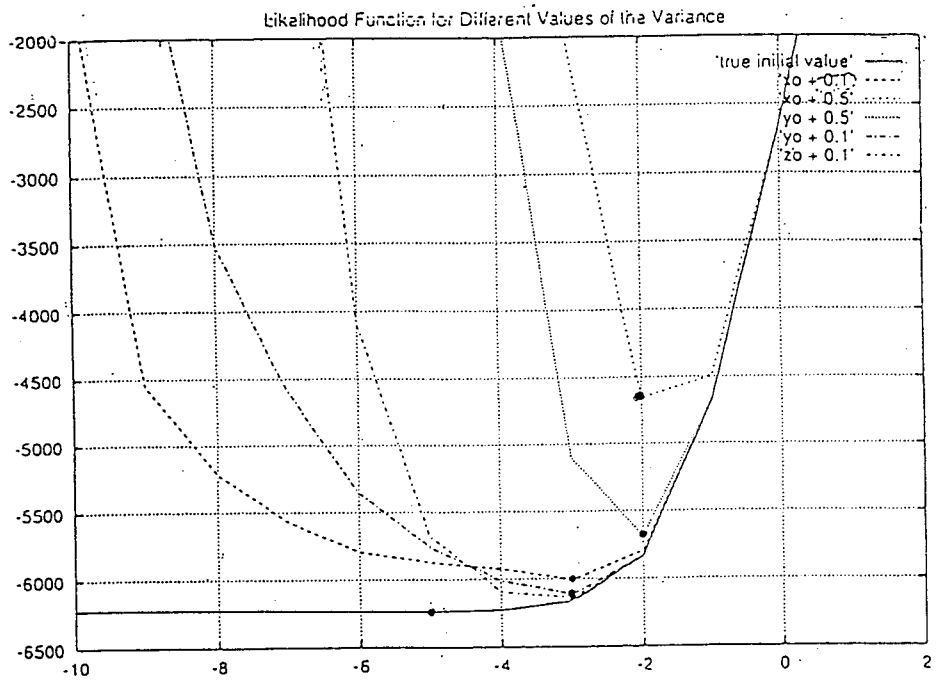

Figure 3.4: The log-likelihood as a function of system noise variance for different initial positions. The y-axis is the value of the log-likelihood and the x-axis is the noise variance. The true value of the variance is 10^{-5}.

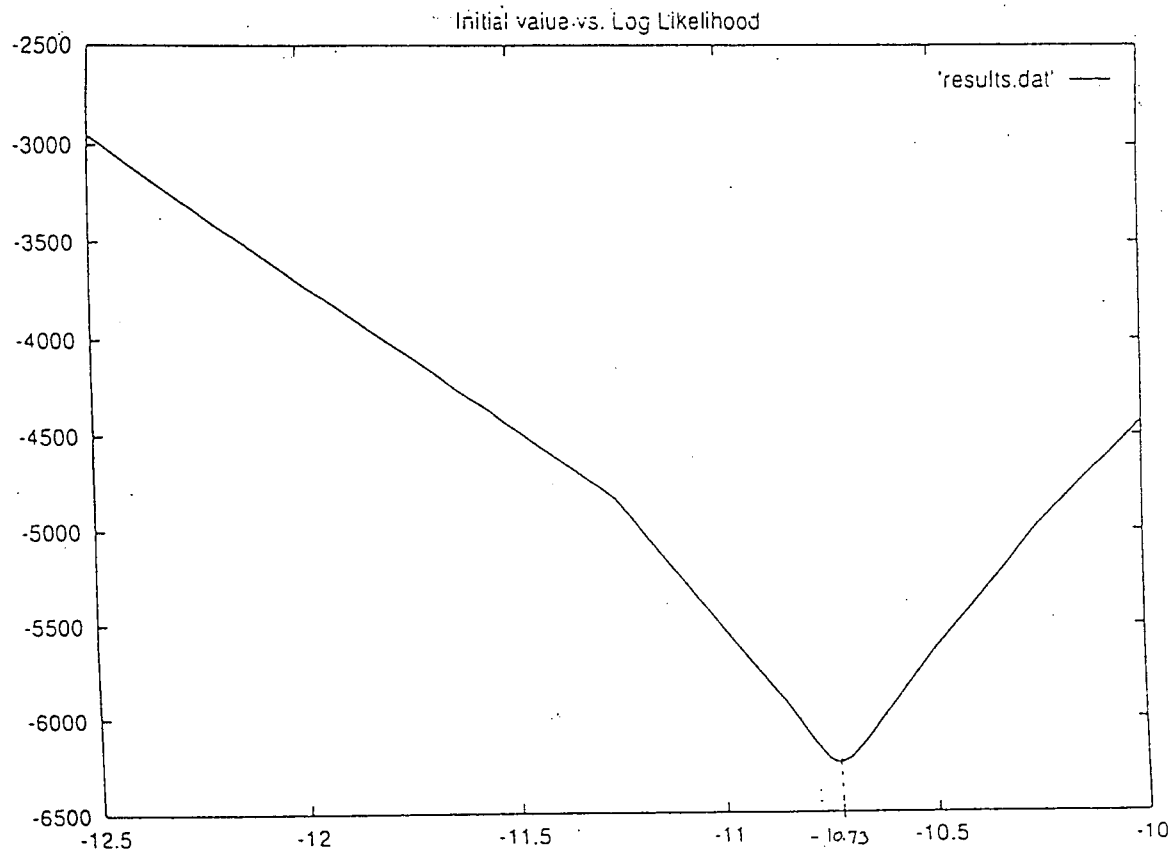

Figure 3.5: Locations of the minima in Fig. 3.4 as a function of initial values. The y-axis is the value of the log-likelihood function and the x-axis gives the value of initial position. Each point corresponds to different system variance. The minimum occurs at -10.73 which is very close to the true value equal to -10.7266.

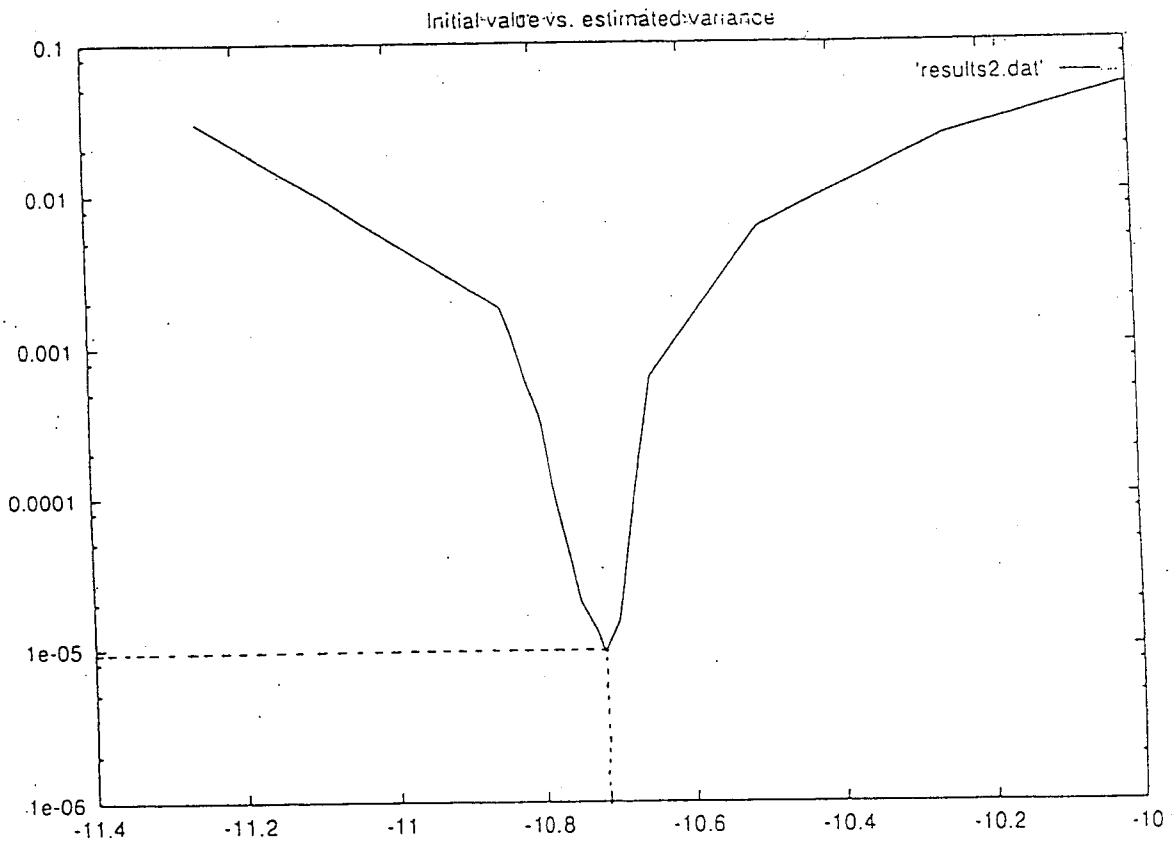

Figure 3.6: Estimated variance, where the log-likelihood is minimum, plotted against the value of the initial position. The true values are: -10.7299 for the initial position and 10^{-5} for the variance. The minimum of the plot occurs at -10.73 with variance equal to 9.3×10^{-6}.

- Neural networks are *universal approximators* [44], which means that they can approximate any continuous mapping to within a desired accuracy.

3.3.1 Mathematical formulation

A feedforward neural network transforms a set of input variables into a set of output variables in a nonlinear fashion. The precise form of the transformation is governed by a set of parameters called the *weights*. The process of determining these parameters is called *training* or *learning*. For a three-layer feedforward neural network (see Fig. 3.7), the output of the jth neuron in the hidden layer is given by

$$z_j = g_{\mathrm{HL}}\left(\sum_{i=1}^{d} w_{ij}^{\mathrm{IH}} x_i\right) \tag{3.44}$$

where d is the number of neurons in the input layer, the w_{ij}^{IH}s are the *interconnection* weights between input and hidden layer neurons and $g_{\mathrm{HL}}(\cdot)$ is the *activation function* of the neurons in the hidden layer. The output of the kth neuron in the output layer is expressed as

$$y_k = g_{\mathrm{OL}}\left(\sum_{j=1}^{m} w_{jk}^{\mathrm{HO}} z_j\right) \tag{3.45}$$

where $g_{\mathrm{OL}}(\cdot)$ is the activation function of this neuron, m is the number of neurons in the hidden layer, and the w_{jk}^{HO}s are the weight of interconnections between the neurons in the hidden layer and that of the output layer.

Training or *learning* involves a minimization process of an *error function* usually of the form

$$e_{\mathrm{NN}} = \sum_{n=1}^{N}\sum_{k=1}^{H} (y_k^n - t_k^n)^2 \tag{3.46}$$

where N is the number of examples used in training the network, H is the number of neurons in the output layer, y_k^n is the output of the kth neuron in the output layer during the presentation of the nth example and t_k^n is the desired output for that particular neuron. The set of weight values that minimizes e_{NN} represents the network solution to the problem. For convenience in notation, these weight values are collected into a $p \times 1$ vector $\mathbf{w}$ whose elements are $(w_{ij}^{\mathrm{IH}} : i = 1, \cdots, d;\ j = 1, \cdots, m;\ w_{jk}^{\mathrm{HO}} : j = 1, \cdots, m;\ k = 1, \cdots, H)$ and $p = m(d + H)$.

For a prediction problem, the output of the network is set to correspond to some future value in a given time series whereas the inputs correspond to previous values of the said series. Therefore for this case, Eq. (3.45) together with Eq. (3.44) will become

$$\begin{aligned} x_i &= g_{\mathrm{OL}}\left(\sum_{k=1}^{m} w_{jk}^{\mathrm{HO}}\, g_{\mathrm{HL}}\Big(\sum_{j=1}^{d} w_{ij}^{\mathrm{IH}} x_{i-j}\Big)\right) \\ &= \hat{g}(x_{i-1}, \cdots, x_{i-d}; \mathbf{w}). \end{aligned} \tag{3.47}$$

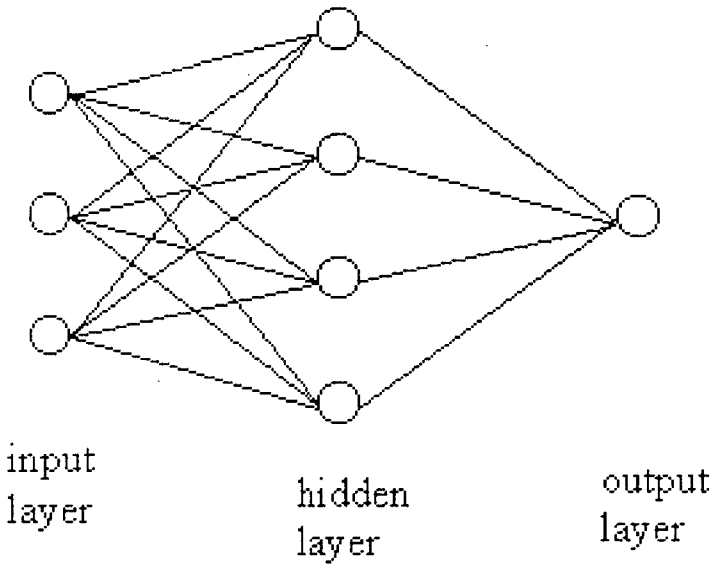

Figure 3.7: Architecture of a three-layer fully-connected neural network.

Equation (3.47) can be used to approximate the predictor function $g(\cdot)$. Therefore, the problem of finding $g(\cdot)$ is reduced to finding the appropriate parameters $\mathbf{w}$ minimizing the error function e_{NN} in Eq. (3.46).

Two important quantities relating neural network to estimation problems are worth discussing here. The first is the *training error* defined as the value of the error function Eq. (3.46) when training is stopped. Since the error function is the difference between the desired output and the actual network output, this quantity provides a measure of the closeness of the network's output to the desired output. For noiseless data, the very small values of e_{NN} are desired to reproduce the training set accurately. However for noisy data, zero values of e_{NN} will only reproduce the noisy data. And thus, it is desirable to not let e_{NN} approach zero.

The second value of merit is the *mean square difference* defined by

$$\beta = \frac{1}{N}\sqrt{\sum[g(\cdot) - \hat{g}(\cdot)]^2} \tag{3.48}$$

where $g(\cdot)$ is the original map and $\hat{g}(\cdot)$ is the neural network approximated map. This quantity gives the difference between the map approximated by neural network and the actual map. Note that even though the training error is zero for noiseless data, β may not be necessarily zero. This is because the training error only represents the closeness of the network output to that of the training set which is only a finite sample of the actual map. The closer the value of β to zero, the closer the learned map to the actual map. This means a better generalization capability of

the trained network which in turn produces better prediction. In the next two subsections, the behavior of these two quantities as the noise variance increases will be examined.

3.3.2 The circle map

In the simulation conducted, time series were generated using well-known chaotic maps. The first map used was the circle map which is given by

$$\theta_{n+1} = [\theta_n + p_1 + p_2 \sin(\theta_n)] \bmod 2\pi + \varepsilon_{n+1}. \tag{3.49}$$

where ε_{n+1} is Gaussian white noise with variance σ. The parameters p_1 and p_2 were set to 0.9 and 3.14 respectively. A three-layer fully-connected neural network was used. The number of input units was set to 2 corresponding to θ_n and a bias neuron with output always equal to 1. The number of output neuron was set to one to correspond to θ_{n+1}. The number of hidden units was set to 22 since this gave the minimum training error for the noiseless time series. Several time series, having 1000 data points, were generated corresponding to different noise variance. One network was trained for each of them.

To compare the convergence of the error function [Eq. (3.46)] with respect to the noise variance, the number of iteration was fixed to 1000. The initial values of the weights for each training were randomly chosen between 1.0 and -1.0. For each time series, the training error is obtained and plotted against the noise variance. This is shown in Fig. 3.8. From the figure, it is obvious that as the noise variance is increased, the training error also increases. This is due to the fact that the map to be approximated by the network becomes rougher as compared with increasing noise. However, this does not mean that the approximated map, when the noise variance is high, is no better compared to that when the noise variance is low. In fact, there are instances when the opposite is true, that is, better approximation occurs when the variance is high. This is clearly seen in Fig. 3.9 which shows the behavior of β as the noise variance σ is increased. From the figure, there are times when β falls as the variance is increased. This indicates that the learned map is closer to the actual map. However, no direct relation between generalization error and noise variance can be readily concluded. This is due to the fact that the approximated maps vary considerably in regions not covered by the generated time series. For example, in Fig. 3.10 when $\theta_n > 2.55$, the three maps coincide but for values less than 2.55, the three diverge away from each other.

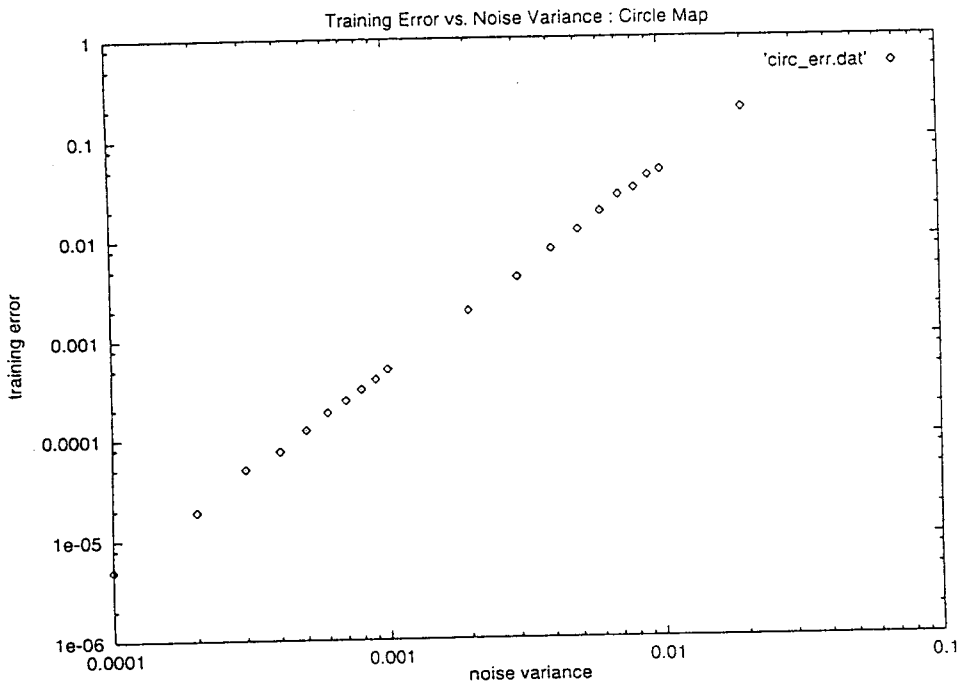

Figure 3.8: The circle map. Training error as a function of noise variance. As the variance increases, the convergence of the error function decreases corresponding to increasing error.

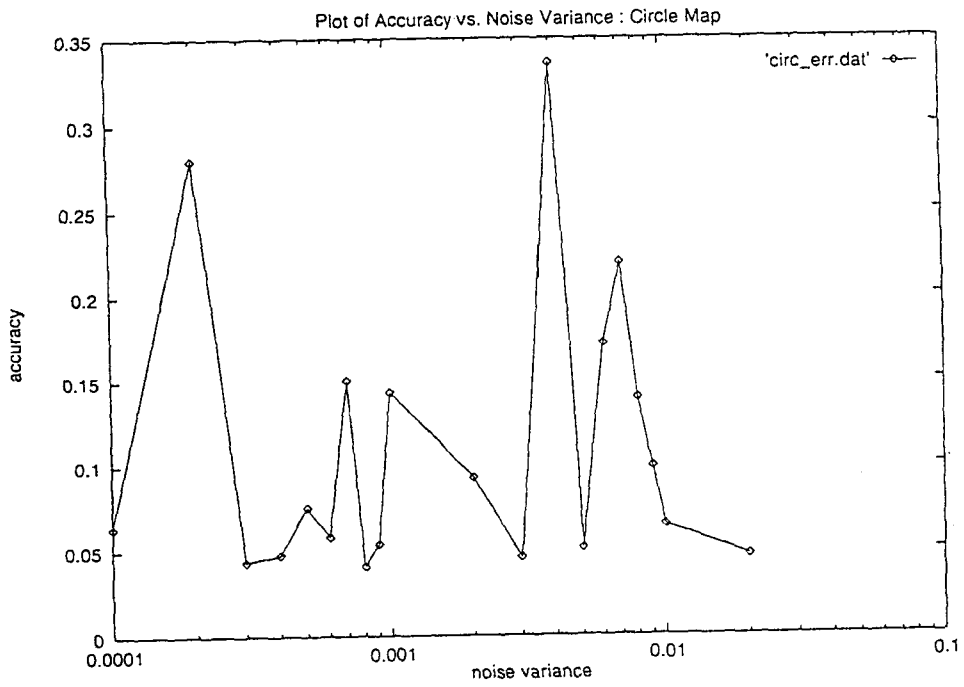

Figure 3.9: The accuracy, β, as a function of the noise variance.

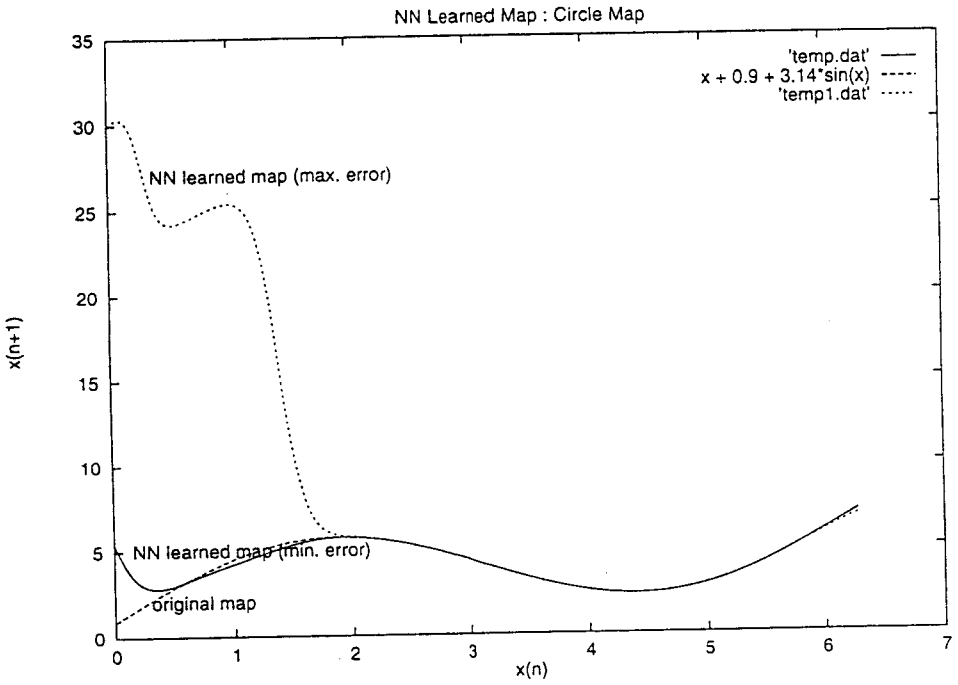

Figure 3.10: The functions learned by the network corresponding to the maximum and minimum training errors as compared to the original function.

3.3.3 The Hénon map

The second map is the Hénon map [43] which is given by

$$x_{n+1} = 1.0 - p_1 x_n^2 + p_2 x_{n-1} + \omega_{n+1} \tag{3.50}$$

where ω_{n+1} is a Gaussian white noise with variance σ. The values of p_1 and p_2 used were 1.3 and 0.25, respectively. Twenty time series were generated corresponding to 20 different values of the variance. Each time series has 1000 data points. Figure 3.11 shows a representative return map of the generated time series. From this figure, it is obvious that the samples covered only a small part of the entire map. A fully-connected feedforward neural network with 3 input units, 18 hidden units, and 1 output unit was used. The input units correspond to x_n, x_{n-1} and 1 bias neuron and the output corresponds to x_{n+1}. The number of hidden units was chosen to minimize the training error for the case when the variance is zero.

In the same manner as the circle map, the number of iteration was fixed to 1000 and the initial weight values were the same for all the training sets. The resulting plot for the training error as a function of the noise variance is shown in Fig. 3.12. Again, the same relation is obtained as that of the circle map: an increasing training error with increasing noise variance. The effect of noise on the generalization capability of the trained network is also investigated in the same manner as the previous case. Here, the mean square difference β is defined to be the volume between the two surfaces within the given domain of their respective variables. Again, no direct relation can be obtained between the noise variance and generalization error β (Fig. 3.13). However, there are instances wherein higher noise variance yields better generalization in the same manner as that of the circle map. A comparison of the NN learned map and the actual map is shown in Fig. 3.14.

3.4 The Orthogonal Approach

Aside from neural networks, other parametrized families of predictor functions can be used to approximate the dynamics in the reconstructed phase-space from the given time series. In this section, the use of orthogonal basis functions, such as orthogonal polynomials, is considered. The idea is to construct orthogonal functions from the given time series and use these functions to span the desired predictor function.

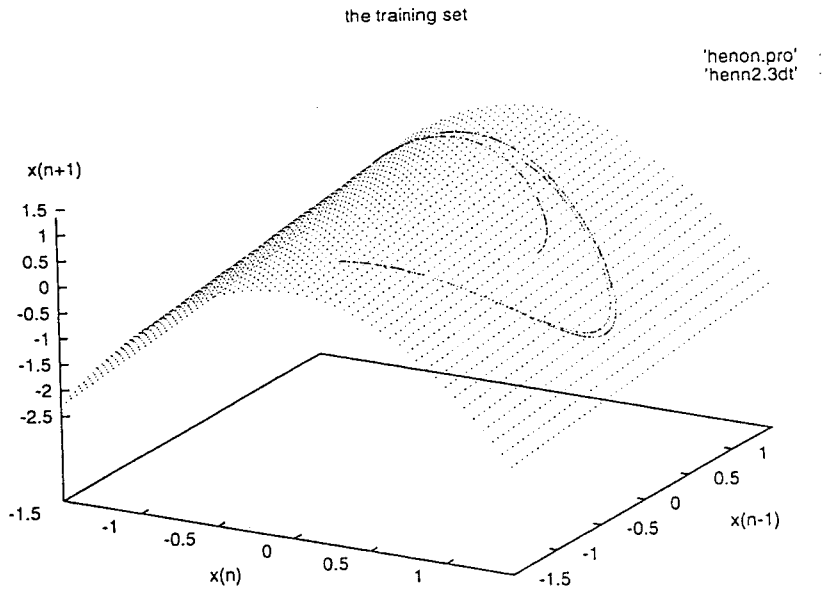

Figure 3.11: Three-dimensional return map of the time series. The surface is given by Eq. (3.50) with $p_1 = 1.3$ and $p_2 = 0.25$.

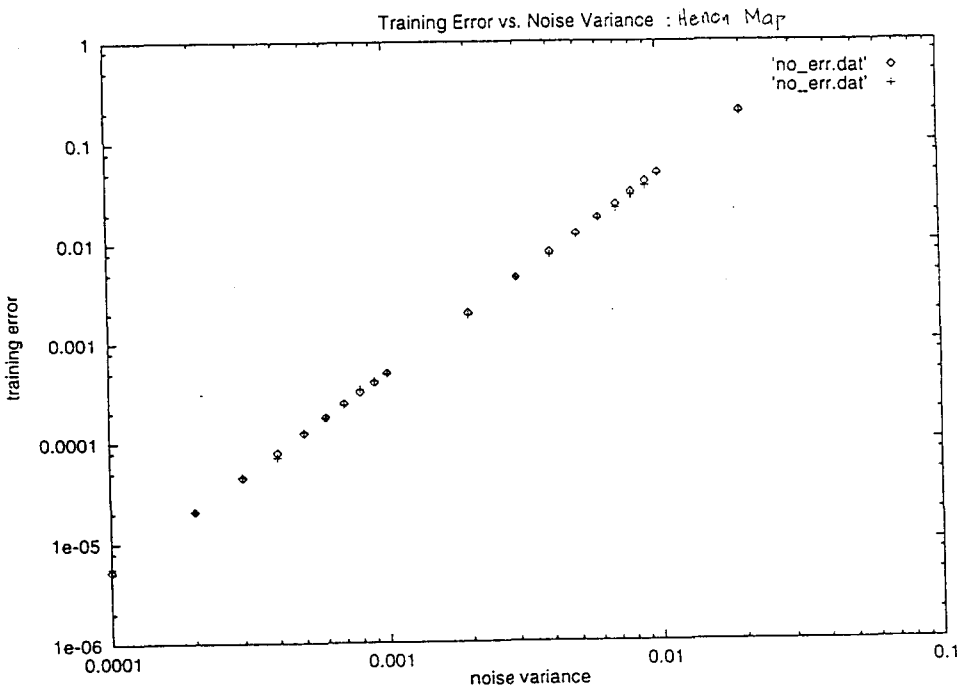

Figure 3.12: The Hénon map. Training error as a function of noise variance. As the variance increases, the convergence of the error function decreases corresponding to increasing error.

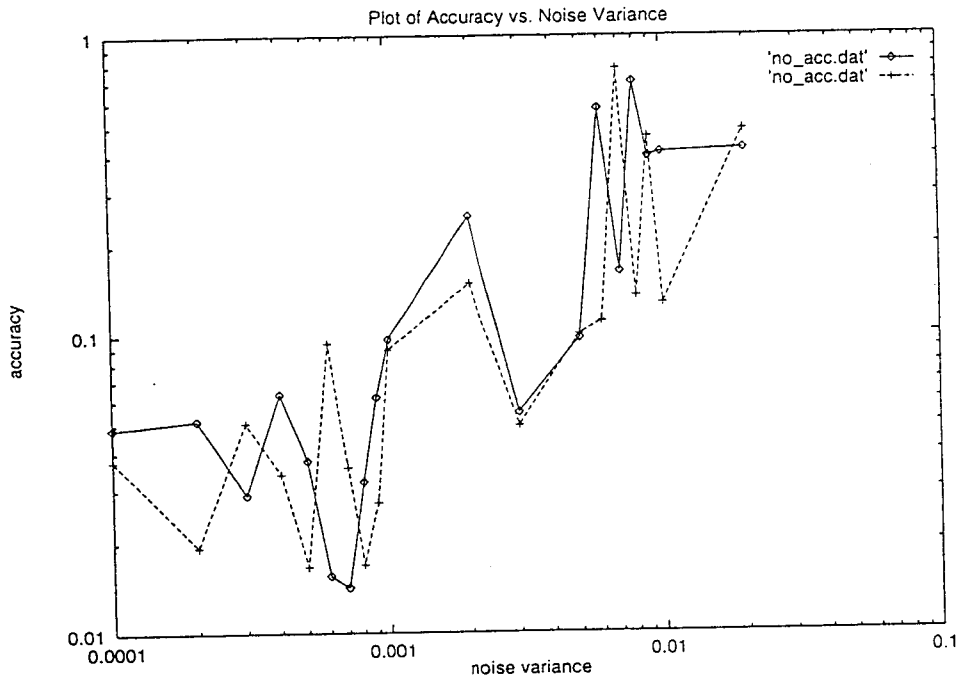

Figure 3.13: The accuracy, β, as a function of the noise variance.

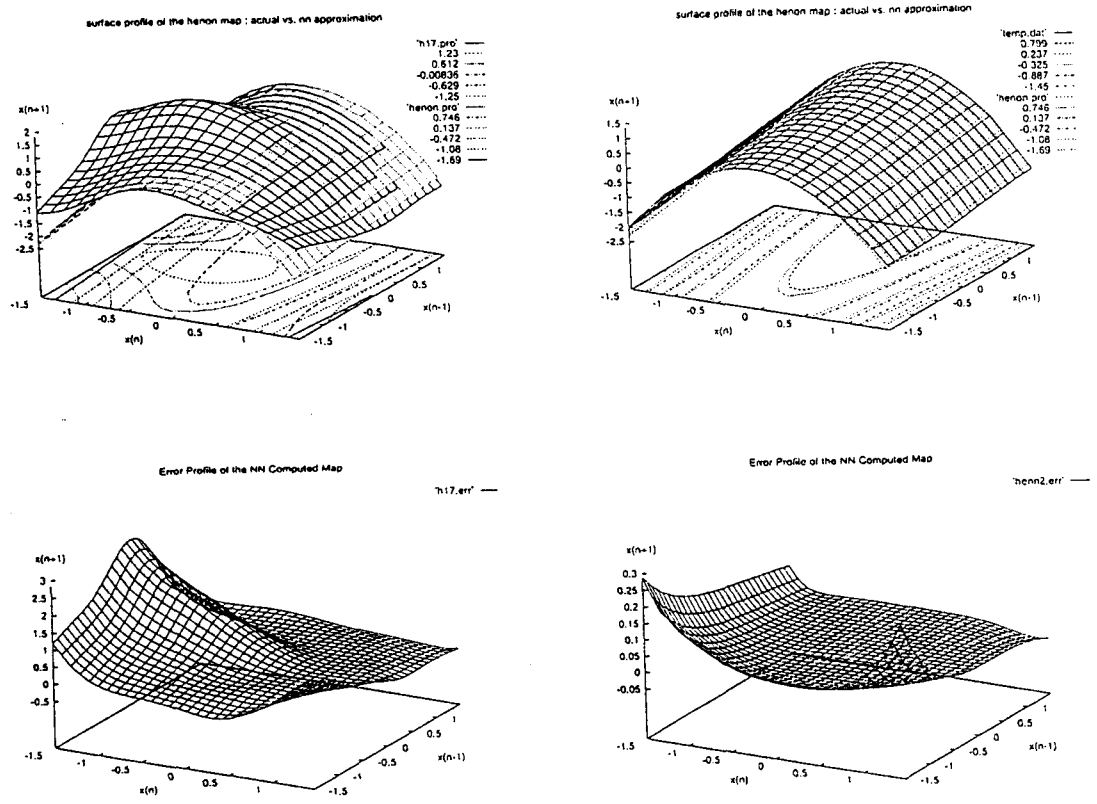

Figure 3.14: Top: Comparison of the NN learned map vs. the actual map. Bottom: Difference between the two surfaces.

3.4.1 Orthogonal polynomials

Suppose that a time series S_N of length N is given. The problem is to find a map with parameters $\mathbf{a}: X \to G(X;\mathbf{a})$, which evolves data points $Y_n \to Y_{n+1}$. Following [1, 22, 34], $G(X;\mathbf{a})$ can be approximated by polynomials $W_m(X)$ which are orthogonal with respect to the natural invariant measure $\rho(X)$ associated to the time series S_N, that is,

$$\int d^d X \rho(X) W_i(X) W_j(X) = N_i \delta_{ij} \tag{3.51}$$

and which can be determined by the conventional Gram-Schmidt procedure starting from

$$W_0(X) = 1. \tag{3.52}$$

From this, $G(X;\mathbf{a})$ is approximated to the Mth order as

$$G(X;\mathbf{a}) = \sum_{m=0}^{M} a_m W_m(X). \tag{3.53}$$

The expansion coefficients (or parameters) are determined by

$$\begin{aligned} a_m &= \int d^d X G(X;\mathbf{a}) W_m(X) \rho(X) \\ &= \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} G(Y_n;\mathbf{a}) W_m(Y_n) \\ &= \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} Y_{n+1} W_m(Y_n) \end{aligned} \tag{3.54}$$

by applying the ergodic theorem.

From Eq. (3.54), one can see the advantages of the approach. It eliminates the problem of multiparameter optimization since the coefficients are readily computed from the time series. Moreover, the estimates of the coefficients can be easily corrected when new data points are available. But the best aspect of this method which makes it more promising in actual application is its robustness to noise since computation of distances in the reconstructed phase-space is unnecessary.

3.4.2 Nonlinear autoregressive models

From here on, only the component of $G(X;\mathbf{a})$ that gives

$$y_n = g(Y_{n-1};\mathbf{a}) \tag{3.55}$$

is considered. To approximate y_n, a nonlinear autoregressive (NAR) model is used

$$\begin{aligned} y_n^{\text{calc}} &= g(Y_{n-1};\mathbf{a}) + \epsilon_n \\ &= a_0 + a_1 y_{n-1} + \cdots + a_d y_{n-d} + a_{d+1} y_{n-1}^2 + a_{d+2} y_{n-1} y_{n-2} \\ &\quad + \cdots + a_M y_{n-d}^k + \epsilon_n \qquad (3.56) \\ &= \sum_{m=0}^{M} a_m z_m(Y_{n-1}) + \epsilon_n \qquad (3.57) \end{aligned}$$

where $Y_{n-1} = (y_{n-1}, y_{n-2}, \cdots, y_{n-d})$ represents a vector in the reconstructed d-dimensional state space (embedding space), the functional basis $\{z_m(X)\}$ is composed of all the distinct combinations of coordinates up to degree k, d is the delay which also corresponds to the dimension of the embedding space, ϵ_n represents the random forcing of the system, and $M+1 = (k+d)!/(d!k!)$ determines the number of coefficients to be computed.

One of the main difficulties of using Eq. (3.56) for function approximation is that the number of parameters to be fitted grows very fast as soon as the phase-space dimension d and the degree of the polynomial k becomes greater than one. Thus, multiparameter optimization becomes unmanageable for large d and k. Moreover, there is the problem of determining which terms to include in the model. Model-selection criteria such as the Akaike's information criterion (AIC)[6] that are valid for linear AR models may not apply to NAR models.

To circumvent these difficulties, auxiliary orthogonal functions are employed in estimating the model parameters. To do this, the right-hand side of Eq. (3.57) is rearranged into a sum of terms which are mutually orthogonal over the given time series:

$$g(X;\mathbf{a}) = \sum_{m=0}^{M} b_m w_m(X) \qquad (3.58)$$

where $w_m(X)$ are orthogonal functions constructed from the $z_m(X)$ using the Gram-Schmidt orthogonalization procedure

$$w_m(X) = z_m(X) - \sum_{r=0}^{m-1} \alpha_{mr} w_r(X) \qquad (3.59)$$

and satisfy Eq. (3.51). The α_{mr} can be computed using

$$\alpha_{mr} = \frac{1}{N_r} \int d^d X \rho(X) z_m(X) w_r(X), \qquad (3.60)$$

$$N_r = \int d^d X \rho(X) [w_r(X)]^2. \qquad (3.61)$$

The expansion coefficients, b_m, are computed using the orthogonality of $w_m(X)$ and is given by

$$b_m = \frac{1}{N_m} \int d^d X g(X;\mathbf{a}) w_m(X) \rho(X). \qquad (3.62)$$

This procedure achieves a least-squares fit by minimizing the mean square error (e_{MSE})

$$e_{\text{MSE}} = \sum_{n=0}^{N} \left[y_{n+1} - \sum_{m=0}^{M} b_m w_m(Y_n) \right]^2 \tag{3.63}$$

which can be rewritten in the form

$$e_{\text{MSE}} = \sum_{n=0}^{N} y_{n+1}^2 - \sum_{m=0}^{M} b_m^2 \sum_{n=0}^{N} [w_m(Y_n)]^2. \tag{3.64}$$

From the above, the inclusion of a given orthogonal function $w_m(X)$ will reduce the e_{MSE} by an amount equal to

$$Q(m) = b_m^2 \sum_{n=0}^{N} [w_m(Y_n)]^2. \tag{3.65}$$

This leads to a systematic way of selecting the terms $a_m z_m(X)$ in Eq. (3.57). Terms with higher $Q(m)$ values can reduce the error significantly and must be included in the model. On the other hand, terms with lower $Q(m)$ values can be neglected since they will not contribute in reducing the e_{MSE}.

Another advantage of this formulation is that the explicit creation of the orthogonal functions $w_m(X)$ in Eq. (3.59) is unnecessary. This can be done using Korenberg's [52] algorithm with a slight modification. The iterative procedure, as derived in Appendix A, to compute the significant quantities, α_{mr}, b_m, and N_m, is summarized by the following relations:

$$\alpha_{mr} = \frac{\beta_{mr}}{N_r} - \sum_{j=0}^{r-1} \alpha_{mj}\alpha_{rj}\frac{N_j}{N_r}, \tag{3.66}$$

$$N_m = \beta_{mm} - \sum_{r=0}^{m-1} \alpha_{mr}^2 N_r, \tag{3.67}$$

$$b_m = \frac{\gamma_m}{N_m} - \sum_{r=0}^{m-1} \alpha_{mr} b_r \frac{N_r}{N_m}, \tag{3.68}$$

where $m = 0, \cdots, M$, $r = 0, \cdots, m-1$, and

$$\beta_{mr} = \frac{1}{N+1} \sum_{n=0}^{N} z_m(Y_n) z_r(Y_n), \tag{3.69}$$

$$\gamma_m = \frac{1}{N+1} \sum_{n=0}^{N} y_{n+1} z_m(Y_n). \tag{3.70}$$

Equations (3.69) and (3.70) can be computed independently from the given time series. This reduces the amount of computation required in obtaining the model parameters.

The original model parameters a_m can now be determined from the computed b_m and α_{mr} using the following relations:

$$a_m = \sum_{i=m}^{M} b_i V_i, \tag{3.71}$$

where $V_m = 1$ and

$$V_i = -\sum_{r=m}^{i-1} \alpha_{ir} V_r \tag{3.72}$$

for $i = m+1, \cdots, M$. Moreover, $Q(m)$ can be easily evaluated from b_m and N_m, that is,

$$Q(m) = b_m^2 N_m. \tag{3.73}$$

3.4.3 Model selection

From the results of the previous section, one can see that $Q(m)$, which quantifies the amount a term $z_m(X)$ can contribute in reducing the overall mean square error, plays an important role in model selection. In this respect, terms which do not significantly reduce the error function (small $Q(m)$ values) are not included in the optimal model. This criterion can be quantified by defining a threshold below which the contribution of $z_m(X)$ becomes insignificant. In [23], a 95% confidence interval (assuming a normal distribution) was used to determine significant terms. For example, before a term $z_m(X)$ is selected, it must satisfy

$$\left[\frac{Q(M)}{\sum_{n=0}^{N} y_n^2 - \sum_{m=0}^{M-1} Q(m)}\right] > \frac{2}{\sqrt{N}}. \tag{3.74}$$

Another criterion is given by the error reduction ratio (e_{ERR}) [4] defined by

$$e_{\mathrm{ERR}_m} = \frac{Q(m)}{\sum_{n=0}^{N} y_n^2}, \tag{3.75}$$

where $m = 0, \cdots, M$. The terms are then ordered according to their e_{ERR} values. An optimal number of terms are then selected by means of information criteria such as Akaike's information criterion, among others.

In the following, a criterion based on the multistep prediction performance of the resulting model is described. The multistep prediction error is defined as

$$e(P) = \frac{1}{N+1} \sum_{n=0}^{N} \sum_{j=1}^{P} [y_{n+j} - g^j(Y_n; \mathbf{a})]^2, \tag{3.76}$$

where P is the number of steps to predict forward in time. The basic algorithm is as follows:

1. Initialize the values of d and k, the delay and degree of Eq. (3.56) respectively.

2. Add each term one by one to the model. For each time, evaluate the multistep prediction error of the resulting model.

3. The number of terms that gives the minimum of $e(P)$ for higher values of P will be included in the final model.

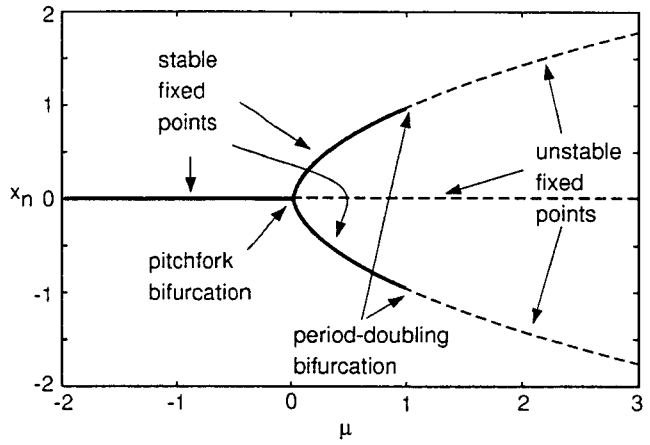

Figure 3.15: Bifurcation diagram of the cubic map.

4. Recompute the coefficients **a** including only the selected terms.

The above list outlines the basic procedure in selecting the optimal model. This can still be improved in a number of ways. For example, in step 2, one can rank first all the candidate terms according to their $Q(m)$ values before adding to the model starting from the most significant to the least. Another way is adding the terms as they appear in Eq. (3.56) and then removing those with insignificant $Q(m)$ values. In this work, both approaches to determine the optimal model are employed. The following examples will illustrate the orthogonal approach.

Example 1: the cubic map

An example of a one-parameter, one-dimensional forced map is given by

$$x_{n+1} = x_n + \mu x_n - x_n^3 + \kappa\epsilon_n \tag{3.77}$$

where ϵ_n represents the random forcing of the system distributed uniformly between -1 to 1 and κ represents the amount of forcing. When $\kappa = 0$, the behavior of the system as a function of the parameter μ is quite simple. The system has fixed points at $x_o = 0$ for all values of μ and $x_\pm = \pm\sqrt{\mu}$ for $\mu \geq 0$. For $\mu < 0$, the origin is stable and globally attracting. A pitchfork bifurcation occurs at $\mu = 0$ and the origin becomes unstable. This also gives rise to the two stable fixed points $x_\pm$. This is summarized in Fig. 3.15.

Ten time series, S_k, with 1000 data points for each parameter values $\mu_k = 2-4(k-1)/10$, $k = 1, \cdots, 10$ and for $\kappa = 0$, were generated. These are shown in Fig. 3.16. Observe that for most of the parameter values, the time series are almost constant except during transient. The few

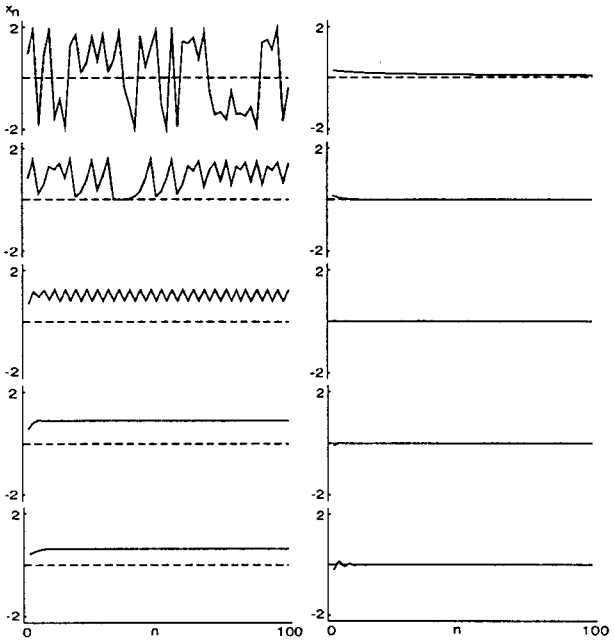

Figure 3.16: Time series for $\kappa = 0.0$. The first 100 data points of S_1 to S_{10} (from top to bottom, left to right).

exceptions are S_1, S_2 and S_3 corresponding to $\mu = 2.0$, $\mu = 1.6$, $\mu = 1.2$, respectively. In S_1, a switching from one fixed point, x_+, to the other, x_-, can be observed. On the other hand, S_3 exhibits a periodic solution. The rest of the time series converges to the stable fixed point x_+ for $\mu \geq 0$ or x_o for $\mu < 0$.

Equation (3.56) with $k = 3$ and $d = 3$, totalling to 20 model parameters to estimate, is used as the starting model. More terms are included in modeling to exhibit the robustness of the method described in subsection 3.4.2 in obtaining only the significant terms as compared to that of the traditional least-squares approach. The estimated values are summarized in Table 3.1 for some of the terms in the model. Terms not shown have parameter values equal to zero.

From the table, several observations are worth discussing. The first observation comes from S_1, S_2, and S_{10}. In this case, the algorithm correctly determines the terms in the original system which are x_{n-1} and x_{n-1}^3. Moreover, it also obtains the accurate values of the associated parameters. This is due to the fact that the time series contain enough information of the structure of the original system as can be seen in S_1 and S_2. Next is the case of S_4 and S_5, wherein the model gives a constant function equal to the value of the fixed point of the map at this particular parameter value. The last case comes from S_7, S_8, and S_9. Here, the algorithm

accurately determines the true value of the parameter associated with the linear part, x_{n-1}, though the cubic term is zero. This is because the values of the series are very close to zero and thus only the linear part is significant. It is suspected that the algorithm determines the linear part instead of the constant term because of the presence of transients which are more significant compared to the long term value of the series. This is the opposite of the second case where the long term values are more significant than the transient. This is more pronounced in S_{10} where the algorithm detected the cubic term in spite of the fact that most of the data in the series are constants with values close to zero. These results illustrate the reliability of the algorithm in determining the best model for a given time series.

Next, the case when the forcing is not zero, in particular, when $\kappa = 0.2$ was considered. Ten time series were generated using the same parameter values as those of the unforced case. These time series are shown in Fig. 3.17. S_1 and S_{10} are not shown since both time series diverged to infinity with this amount of noise. The estimated parameters of the model are summarized in Table 3.2. An interesting difference between Table 3.1 and Table 3.2 is seen in S_3 and S_4. In the case of the forced system, the terms x_{n-1} and x_{n-1}^3 are now determined correctly. Moreover, the estimated parameter values of the model are very close to the true values. This is a case where the presence of noise helps reveal the underlying structure of the given system as can be seen in Fig. 3.17, S_2 and S_3 in particular.

From the analysis, it is seen that there are three competing models, namely:

$$y_n^{\text{calc}} = \alpha y_{n-1} - y_{n-1}^3 \tag{3.78}$$

modeling S_2, S_3, and S_4;

$$y_n^{\text{calc}} = \alpha y_{n-1} \tag{3.79}$$

for $S_k, k = 6, \cdots, 9$; and

$$y_n^{\text{calc}} = \alpha + \beta y_{n-1} \tag{3.80}$$

which models S_5. Since all time series come from the same dynamical system, one of the above models should be able to describe all the given time series. By inspection, it is obvious that Eq. (3.78) represents the most probable model to use. This is because the other models can not take into account the behavior of the other time series, for e.g. the behavior of S_2. With this, the parameter values were recomputed using Eq. (3.78) for $S_k, k = 5, \cdots, 9$. The new parameter values are shown in Table 3.3. The behavior of this model with α as the parameter is shown in Fig. 3.18.

		1.0	x_{n-1}	x_{n-3}^2	x_{n-1}^3	$x_{n-1}^2 x_{n-3}$
S_1	T	0.0	3.0	0.0	-1.0	0.0
	E	0.0	3.0	0.0	-1.0	0.0
S_2	T	0.0	2.6	0.0	-1.0	0.0
	E	0.0	2.6	0.0	-1.0	0.0
S_3	T	0.0	2.2	0.0	-1.0	0.0
	E	1.41	0.0	0.0	0.0	-0.31
S_4	T	0.0	1.8	0.0	-1.0	0.0
	E	0.89	0.0	0.0	0.0	0.0
S_5	T	0.0	1.4	0.0	-1.0	0.0
	E	0.63	0.0	0.0	0.0	0.0
S_6	T	0.0	1.0	0.0	-1.0	0.0
	E	0.0	1.02	-0.22	0.0	0.0
S_7	T	0.0	0.6	0.0	-1.0	0.0
	E	0.0	0.6	0.0	0.0	0.0
S_8	T	0.0	0.2	0.0	-1.0	0.0
	E	0.0	0.2	0.0	0.0	0.0
S_9	T	0.0	-0.2	0.0	-1.0	0.0
	E	0.0	-0.2	0.0	0.0	0.0
S_{10}	T	0.0	-0.6	0.0	-1.0	0.0
	E	0.0	-0.6	0.0	-1.0	0.0

Table 3.1: Cubic map. Estimated parameter values compared to the true values for $\kappa = 0$. T = True value, E = Estimated value.

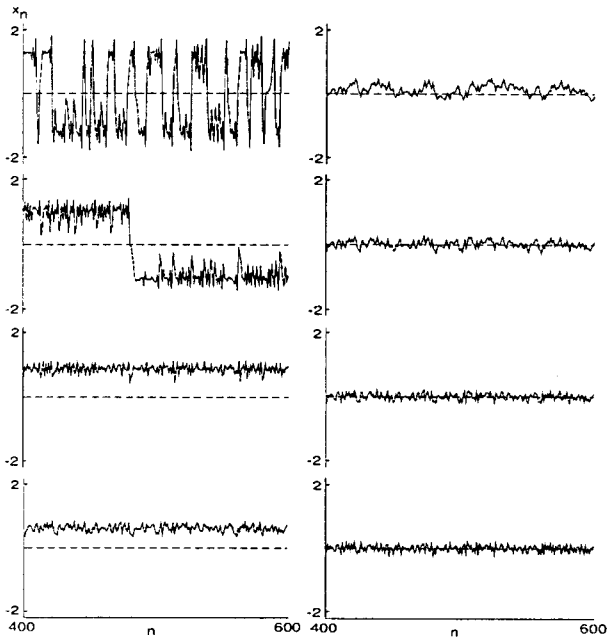

Figure 3.17: Time series for $\kappa = 0.2$. Data points from the 400th to 600th iteration of S_2 to S_9 (from top to bottom, left to right).

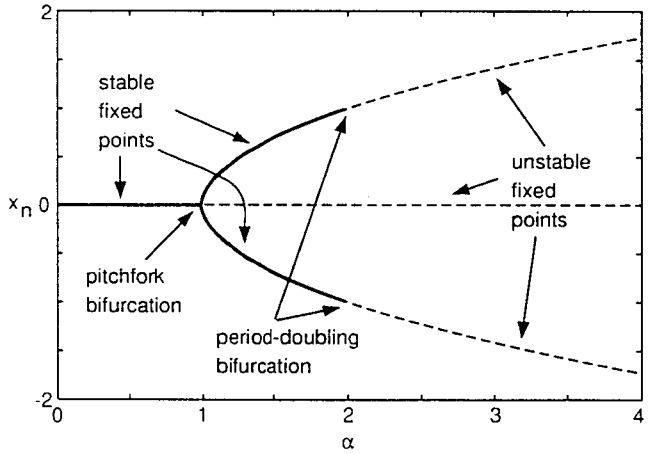

Figure 3.18: Bifurcation diagram of Eq. (3.78) with α as the bifurcation parameter.

		1.0	x_{n-1}	x_{n-3}^2	x_{n-1}^3	$x_{n-1}^2 x_{n-3}$
S_1	T	0.0	3.0	0.0	-1.0	0.0
	E	–	–	–	–	–
S_2	T	0.0	2.6	0.0	-1.0	0.0
	E	0.0	2.6	0.0	-1.0	0.0
S_3	T	0.0	2.2	0.0	-1.0	0.0
	E	0.0	2.18	0.0	-1.0	0.0
S_4	T	0.0	1.8	0.0	-1.0	0.0
	E	0.03	1.77	0.0	-1.0	0.0
S_5	T	0.0	1.4	0.0	-1.0	0.0
	E	0.45	0.25	0.0	0.0	0.0
S_6	T	0.0	1.0	0.0	-1.0	0.0
	E	0.0	0.87	0.0	0.0	0.0
S_7	T	0.0	0.6	0.0	-1.0	0.0
	E	0.0	0.5	0.0	0.0	0.0
S_8	T	0.0	0.2	0.0	-1.0	0.0
	E	0.0	0.0	0.0	0.0	0.0
S_9	T	0.0	-0.2	0.0	-1.0	0.0
	E	0.0	-0.28	0.0	0.0	0.0
S_{10}	T	0.0	-0.6	0.0	-1.0	0.0
	E	–	–	–	–	–

Table 3.2: Cubic map. Estimated parameter values compared to the true values for $\kappa = 0.20$ ($\Theta_{TH} = 0.05$). T = True values, E = Estimated values, – = parameter values cannot be estimated because the time series diverges with this amount of noise.

	S_5	S_6	S_7	S_8	S_9
True	1.4	1.0	0.6	0.2	-0.2
α	1.4	0.99	0.54	0.14	-0.25

Table 3.3: Cubic map. Estimated values of α in Eq. (3.78).

Example 2: the Hénon map

In this example, a two-parameter, two-dimensional map given by the Hénon map is considered. The map is given by

$$x_n = 1.0 - p_1 x_{n-1}^2 + p_2 x_{n-2} + \kappa \epsilon_n \tag{3.81}$$

where ϵ_n represents the random forcing of the system distributed uniformly between -1 to 1 and κ represents the amount of forcing. Ten time series were obtained for ten different values of (p_1, p_2) given by $\{0.12\sin[2\pi(i-1)/10] + 1.346,\ 0.12\cos[2\pi(i-1)/10] + 0.2\}, i = 1, \cdots, 10$. Equation (3.56) is used to model each time series with $k = 2$ and $d = 5$. Again, a model with more terms is used. Table 3.4 shows the estimated model parameters for the case where $\kappa = 0$. Observe how accurately (up to the third decimal place) the algorithm estimated the parameter values of each terms included in the model. Terms not shown have zero parameter values.

When the forcing is not zero, in particular for $\kappa = 0.1$, four out of the ten time series diverged. Increasing the amount of noise further will cause all of the time series to diverge. For the six nondiverging time series, the estimated parameter values are shown in Table 3.5. In spite of the presence of random forcing, the correct terms (1.0, x_{n-2} and x_{n-1}^2) and their associated parameter values were still obtained. A working model can be obtained using the table and it is given by

$$x_n = a_0 + a_1 x_{n-2} + a_2 x_{n-1}^2, \tag{3.82}$$

which is exactly the Hénon map.

3.5 Testing for Nonlinearities

In this section[2], nonlinear autoregressive models (subsection 3.4.2) are used to detect the presence of nonlinearities in time series. The data are the recorded electric organ discharges (EOD) of resting *Gymnotus carapo* specimens. See Appendix C for the experimental preparations.

The EOD consists of brief electric pulses separated by comparatively longer intervals of silence (for review see [12]). In general, the variability of the inter-EOD interval depends on behavioral conditions. For example, in *Gymnotus carapo*, the variability increases due to shortening of some of the intervals i) in response to sensory stimulation, i.e. the novelty response, ii) after pharmacological treatments with serotonergic agents [20], iii) during the interaction between

[2]The results presented in this section also appear in Capurro A, Longtin A, Bagarinao E, Sato S, Macadar O, and Pakdaman K, Variability of the electric organ discharge interval duration in resting *Gymnotus carapo*, submitted for publication.

		1.0	x_{n-1}	x_{n-2}	x_{n-3}	x_{n-1}^2
TS_1	T	1.0	0.0	0.320	0.0	-1.346
	E	1.0	0.0	0.320	0.0	-1.346
TS_2	T	1.0	0.0	0.297	0.0	-1.416
	E	1.0	0.0	0.297	0.0	-1.416
TS_3	T	1.0	0.0	0.237	0.0	-1.460
	E	1.0	0.0	0.237	0.0	-1.460
TS_4	T	1.0	0.0	0.163	0.0	-1.460
	E	1.0	0.0	0.163	0.0	-1.460
TS_5	T	1.0	0.0	0.103	0.0	-1.416
	E	1.0	0.0	0.103	0.0	-1.416
TS_6	T	1.0	0.0	0.080	0.0	-1.346
	E	1.0	0.0	0.080	0.0	-1.346
TS_7	T	1.0	0.0	0.103	0.0	-1.275
	E	1.0	0.0	0.103	0.0	-1.275
TS_8	T	1.0	0.0	0.163	0.0	-1.232
	E	1.0	0.0	0.163	0.0	-1.232
TS_9	T	1.0	0.0	0.237	0.0	-1.232
	E	1.0	0.0	0.237	0.0	-1.232
TS_{10}	T	1.0	0.0	0.297	0.0	-1.275
	E	1.0	0.0	0.297	0.0	-1.275

Table 3.4: Hénon map. Estimated parameter values compared to the true values for $\kappa = 0.0$. T = True values, E = Estimated values.

		1.0	x_{n-1}	x_{n-2}	x_{n-3}	x_{n-1}^2
TS_1	T	1.0	0.0	0.320	0.0	-1.346
	E	–	–	–	–	–
TS_2	T	1.0	0.0	0.297	0.0	-1.416
	E	–	–	–	–	–
TS_3	T	1.0	0.0	0.237	0.0	-1.460
	E	–	–	–	–	–
TS_4	T	1.0	0.0	0.163	0.0	-1.460
	E	–	–	–	–	–
TS_5	T	1.0	0.0	0.103	0.0	-1.416
	E	1.0	0.0	0.103	0.0	-1.416
TS_6	T	1.0	0.0	0.080	0.0	-1.346
	E	1.003	0.0	0.080	0.0	-1.353
TS_7	T	1.0	0.0	0.103	0.0	-1.275
	E	1.002	0.0	0.101	0.0	-1.278
TS_8	T	1.0	0.0	0.163	0.0	-1.232
	E	1.002	0.0	0.165	0.0	-1.237
TS_9	T	1.0	0.0	0.237	0.0	-1.232
	E	1.001	0.0	0.237	0.0	-1.234
TS_{10}	T	1.0	0.0	0.297	0.0	-1.275
	E	1.0	0.0	0.298	0.0	-1.276

Table 3.5: Hénon map. Estimated parameter values compared to the true values for $\kappa = 0.1$ ($\Theta_{TH} = 0.05$). T = True values, E = Estimated values, – = parameter values cannot be estimated because the time series diverges with this amount of noise.

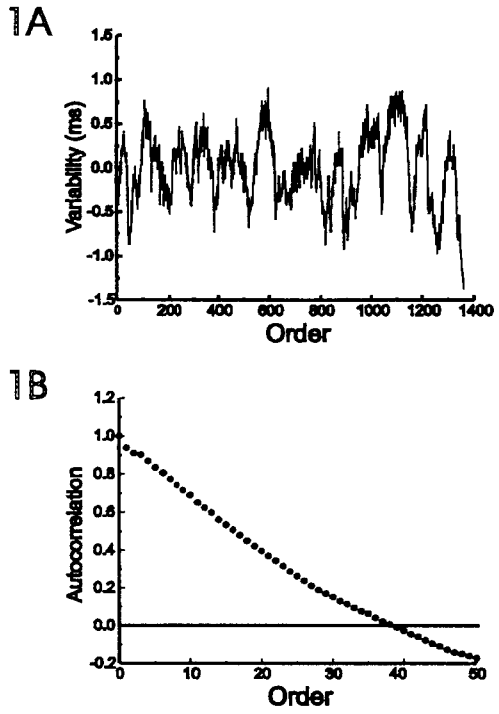

Figure 3.19: (a) Time series of the inter EOD intervals of a resting *Gymnotus carapo*, 40 seconds of recording of fish 1. Abscissa: Interval order. Ordinate: Noise amplitude (msec.). (b) Autocorrelation function of the time series shown in (a). Abscissa: Interval order. Ordinate: Autocorrelation (dimensionless).

two fish [21], iv) in the active moments of the circadian cycle [15], v) during the reproductive social behavior [41] and vi) during the escape response [29].

Figure 3.19 shows an example of the sequence of inter-EOD intervals and their autocorrelation function. These intervals display some variability around their mean value. The purpose of the present section is to determine the nature of these fluctuations: whether the fluctuations in the inter-EOD intervals have a deterministic component or whether they are predominantly stochastic. This is achieved by using an approach based upon the comparison of best fitted linear and nonlinear ARs, in order to test for the presence of nonlinearities. The technique is particularly effective for time series heavily contaminated with noise, and which, furthermore, directly compares the performance of linear AR models with nonlinear ones. Following this analysis, a statistical model for the sequence of inter-EOD intervals is developed.

The basic idea is to compare the best linear model ($k = 1$) and nonlinear model ($k > 1$) for each data set [11]. The best linear model is obtained by searching for d which gives the first

fish	σ_{AR}	σ_{NAR}	U_{calc}	$U_{0.05}$
fish1	0.133	0.125	1.766	1.645
fish2	0.115	0.113	0.665	1.645
fish3	0.192	0.185	0.876	1.645
fish4	0.289	0.283	0.428	1.645
fish5	0.171	0.144	3.438	1.645

Table 3.6: Comparison of the best AR and the best NAR model. σ_{AR} is the standard deviation of the residuals of the AR model, σ_{NAR} is the standard deviation of the residuals of the best NAR model, U_{calc} represents the estimated squared ranks statistics, and $U_{0.05}$ is the value of U at 0.05 significance level.

minimum of Akaike's information criterion with $k = 1$. Repeating the same procedure for values of $k > 1$ yields the best nonlinear model. For each data set and each value of k and d, the AR and NAR coefficients a_i were estimated using the method presented in the previous section. For each of the resulting models, the residuals were computed, and the standard deviations of each of the series of residuals were estimated. The presence of nonlinear determinism is established when the best nonlinear model is more predictive than the best linear model, i.e., σ_{NAR} is significantly smaller than σ_{AR}, where σ_{NAR} and σ_{AR} are the standard deviations of the residuals of the best NAR and best AR, respectively. Practically, the competing models were compared using the nonparametric squared ranks test for variances[25].

The results are summarized in Table 3.6 which compares the calculated test statistics (U-value) with the theoretical value at 5% significance level. The U-values indicate that the nonlinear models are not significantly better than the linear ones for fish 2, 3, and 4. Surprisingly, for fish 1 and 5, the U-test indicates that the NAR is a better one step predictor than the AR. This seems to contradict the results of previous studies. However, in the following, it will be shown that the reason for this apparent discrepancy is not the presence of nonlinearities that went undetected with the previous methods but rather, it is due to the fact that the assumption about the noise in the AR models is not appropriate.

In order to establish this, the data is modeled by a linear AR subject to a combination of both Gaussian and shot noise. The model is of the form:

$$y_n = a_0 + \sum_{r=1}^{d} a_r y_{n-r} + \sum_{r=1}^{k} b_r P_n^{T_r} + \sigma W_n \tag{3.83}$$

where $P_n^{T_r}$ represents an *innovational outlier* (IO) equal to 1 if $n = T_r$ and 0 if $n \neq T_r$, k is the

total number of outliers in the series, and b_r is the amplitude of the rth outlier. To fit this model to the data, i.e., to estimate the parameters, the scheme of Box, et al. [16] is used. To detect the timing of the outliers, the model is first estimated assuming no outliers are present. The residuals are computed and the maximum of their absolute values is obtained. If this maximum is larger than three standard deviations, then it is an outlier and the time of occurrence is recorded. A new series of residuals is then obtained by setting the maximum value to zero and the standard deviation is re-estimated. This procedure is repeated until the maximum value of the residual series is less than three times the standard deviation. After all the outliers are detected, the coefficients in Eq. (3.83) are reestimated using the fast Korenberg algorithm. A revised set of residuals is obtained using this new set of coefficients. The same iterative procedure described above is applied to revised residual series until no new outliers are detected.

The resulting AR+IO models were compared to the best AR models as well as the best NAR models. Comparisons based on the one step prediction performance of the models were performed. The nonparametric squared ranks test for variances is used to test whether the best NAR models are performing significantly better than the AR+IO models. Furthermore, the residuals for the best fit AR and AR+IO were tested whether they form a sequence of independent Gaussian random variables. To this end, the first one hundred autocorrelation coefficients of the residuals were estimated and the Ljung and Box statistics[56] was used to test for the lack-of-fit of the model. The standard deviations of the best fit ARs were compared to that of the AR+IO models. Tables 3.7 and 3.8 summarize the results. The U-values indicate that the NARs are not significantly better than the AR+IOs. In fact, the performance of the AR+IO model as one step predictor is significantly better than the best NAR model for the case of fish 5 which has the largest number of outliers. Furthermore, the residual time series of the ARs with IOs are completely free of outliers as compared to that of the AR alone. These result in a significant decrease of the standard deviations of the residuals. Furthermore, for the AR+IO, all lack-of-fit tests of the autocorrelation of the residuals were satisfactory. For the standard ARs, this was not the case for fish 3.

These results show that the AR+IOs perform better or as well as the best NARs and the best ARs. Furthermore, the residuals of the AR+IOs, in contrast with some of those of the best ARs and best NARs, successfully pass all the tests pertaining to their independence and Gaussian distribution. This is illustrated in Fig. 3.20 which shows the residuals of the best fit AR, NAR, and AR+IO models for fish 5. The AR coefficients of the final models are given in Table 3.9.

fish	σ_{AR+IO}	σ_{NAR}	U_{calc}	$U_{0.05}$
fish1	0.124	0.125	0.780	1.645
fish2	0.106	0.113	-0.711	1.645
fish3	0.180	0.185	-0.092	1.645
fish4	0.262	0.283	-1.024	1.645
fish5	0.132	0.144	-0.242	1.645

Table 3.7: Comparison of the AR+IO and the best NAR model. σ_{AR+IO} is the standard deviation of the residuals of the AR+IO model, σ_{NAR} is the standard deviation of the residuals of the best NAR model, U_{calc} represents the squared ranks statistics, and $U_{0.05}$ is the value of U at 0.05 significance level.

fish	model	N	p	SD	Q	Q_{crit}	DSD[%]	NO
fish1	AR	2055	5	0.133	87.847	129.97	-	16
	AR+IO	2055	5	0.124	93.571	129.97	7.1	0
fish2	AR	1940	6	0.115	98.759	128.80	-	21
	AR+IO	1940	6	0.106	97.021	128.80	7.8	0
fish3	AR	1875	6	0.192	133.123	128.80	-	13
	AR+IO	1875	7	0.180	126.647	127.63	6.2	0
fish4	AR	1775	3	0.289	100.931	132.31	-	21
	AR+IO	1775	3	0.262	98.139	132.31	9.3	0
fish5	AR	1587	4	0.171	86.409	131.14	-	27
	AR+IO	1587	7	0.132	100.678	127.63	22.8	0

Table 3.8: Comparison of the AR and the AR+IO. N is the number of residuals used in the analysis, SD is the standard deviation of the residuals of the model, Q is the estimated Ljung and Box statistics, $Q_{crit} = \chi^2_{100-p;\,\alpha=0.01}$ gives the value of Q at 0.01 significance level, DSD is the decrease in the standard deviation computed as DSD = $(\mathrm{SD}_{AR} - \mathrm{SD}_{AR+IO})/\mathrm{SD}_{AR}$ where SD_{AR} is the standard deviation of the residuals of the AR model and SD_{AR+IO} is the standard deviation of the AR+IO model, and NO is the number of outliers present in the residual series.

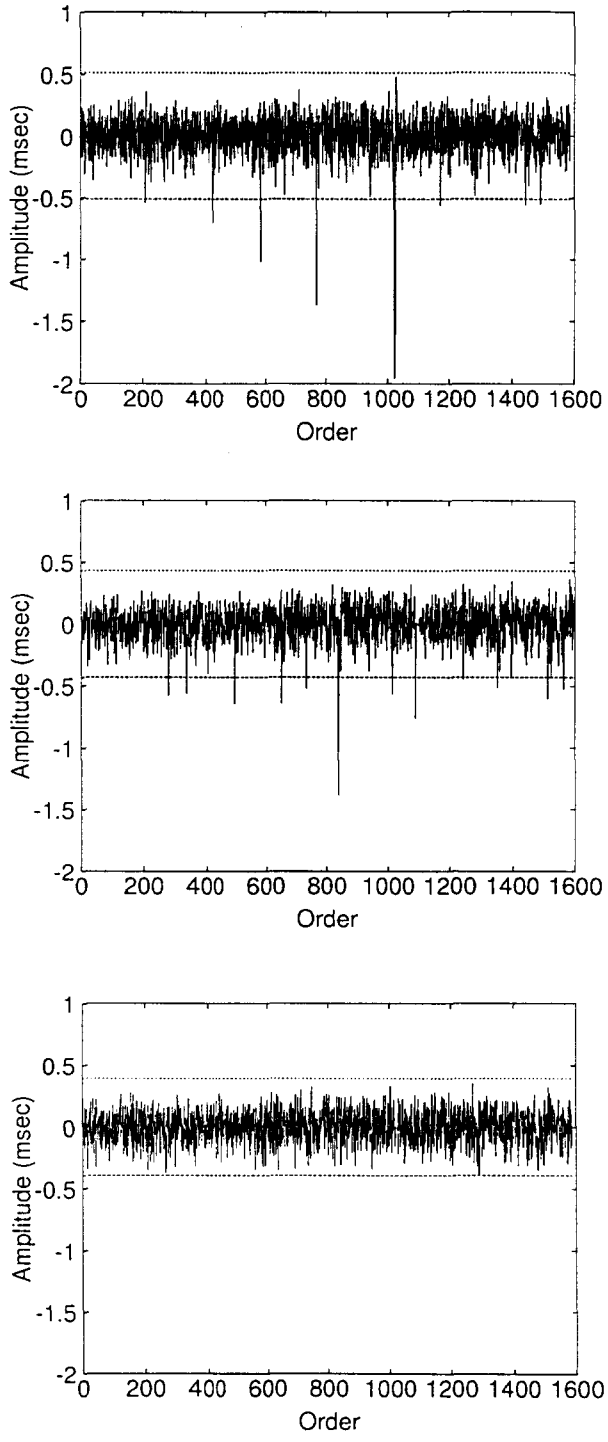

Figure 3.20: Residuals of fish 5 models: AR (top), NAR (middle), and AR+IO (bottom). Abscissa is interval order and ordinates is residual (msec).

fish	a_0	a_1	a_2	a_3	a_4	a_5	a_6	a_7
1	0.000942	0.671543	0.203362	0.237763	-0.036813	-0.076328		
2	0.005311	0.491533	0.314534	0.283100	-0.052195	-0.008962	-0.048943	
3	0.007252	0.503950	0.287591	0.249927	0.015180	0.024365	-0.071875	-0.031984
4	0.012477	0.777074	0.025463	0.176697				
5	0.013912	0.784023	0.089700	0.047697	0.009994	0.090963	0.012050	-0.070122

Table 3.9: AR coefficients of the final models.

Chapter 4

Bifurcation Reconstruction

4.1 Introduction

Analysis of experimentally measured time series has been used to gain insights into the underlying physical processes, to do prediction, as well as to determine invariants associated to the dynamics such as Lyapunov exponents, correlation dimension, among others. When time series measured at different values of the system parameters are given, additional information about the system's behavior becomes available. This extra information can be exploited to reveal the different bifurcations the system undergoes as the parameters are changed, as well as to uncover behaviors of the system which may be present but not readily observed. The goal of bifurcation diagram reconstruction is to address this problem by obtaining a BD qualitatively similar to that of the given system using time series measured at a finite number of parameter values. It is only recently that the BD reconstruction problem has received considerable attention. This is brought about by the development of new algorithms for estimating predictor functions at fixed parameter values and the increasing need to characterize the different behaviors of systems with unknown dynamics using observations. The different methods to solve this problem will be discussed in the remaining chapters of this work.

4.2 The Reconstruction Problem

Suppose that K time series $S_k = \{y_t^k,\ 0 \leq t \leq \tau\}$ of length N are measured at different parameter values of an unknown dynamical system. The point of interest is to know, using S_k, the qualitative behavior of the system as a function of the parameter set $\mathbf{c}$. The basis of reconstruction is provided by Takens' theorem[79] which states that it is possible to reconstruct

the dynamics of an unknown system using its time series output. Accordingly, for a given time series corresponding to a particular parameter value $\mathbf{c}$, one can find a predictor function $f(\cdot\,;\mathbf{c})$ such that

$$Y_{t+\Delta} = f(Y_t;\mathbf{c}) \tag{4.1}$$

where $f : \mathbb{R}^d \times \mathbb{R}^k \to \mathbb{R}^d$, $\mathbf{c} \in \mathbb{R}^k$, k is the number of system parameters, $Y_t = \{y_t, \cdots, y_{t-(d-1)\Delta}\}$ represents an observation from 0 to τ, Δ is the time delay and d is the embedding dimension.

The prediction problem as defined in Eq. (4.1) deals with time series corresponding to a single parameter value. This does not take into account the changes in the behavior of the system as the parameter $\mathbf{c}$ is varied. For example, in some neuronal models such as the FitzHugh-Nagumo equation, the system can behave as either excitable or oscillating depending upon the value of the parameters involved[83].

The BD reconstruction problem using time series is an extension of the prediction problem. This extended problem does not only aim to reconstruct the dynamics of the system for fixed parameter values but also to interpolate the changes in the behavior of the system when some of its parameters are changed and to estimate the bifurcation points where these changes take place. More precisely, the problem is defined as follows: it is assumed that

1. the equation describing the dynamical system and the dependence of the system parameters are unknown;

2. the predictor function $f(\cdot;\ \mathbf{c})$ is also unknown and needs to be approximated;

3. a finite number of time series S_k, $k = 1, \cdots, K$ at different parameter values $\mathbf{c}_1, \cdots, \mathbf{c}_K$ are available; and

4. the values of $\mathbf{c}_1, \cdots, \mathbf{c}_K$ may or may not be known.

Thus, the time series are the only source of information regarding the dynamics of the system. Under these assumptions, it is desired to reconstruct the bifurcation diagram of the given system. The term *reconstruction* is used here in the sense described in [81, 82], that is, finding a suitable parametrized function which exhibits the same bifurcation sequence as the original system.

4.3 The Projection Region

The key idea that makes possible the analysis of bifurcation structure from time series is the concept of *projection region*. The projection region is defined as the region in the parameter

space of the model with similar bifurcation structure as the system. Thus for parameter values within this region, the dynamics of the model is therefore the same as that of the given system. The bifurcation structure of the system can now be analyzed using that of the model on the projection region. The problem therefore is to determine the projection region using the available time series.

To understand this, suppose that $f(X;\mathbf{c})$ represents the dynamics of the given system and $f : \mathbb{R}^d \times \mathbb{R}^k \rightarrow \mathbb{R}^d$, where d represents the dimension of X and k the number of system parameters. Suppose further that there exists a family of functions $g(X;\mathbf{w})$, parametrized by $\mathbf{w} \in \mathbb{R}^p$, such that the behavior of $g(X;\mathbf{w})$ in some region of its parameter space, denoted as Ω, is similar to that of $f(X;\mathbf{c})$ for some region in its parameter space denoted as Γ. The subspace Ω is referred to as the projection region of Γ. Furthermore, it can be assumed that there exists a function $\mathbf{h} : \Gamma \rightarrow \Omega$ such that if $\mathbf{c}_i \in \Gamma$ and $\mathbf{w}_i \in \Omega$ then $\mathbf{w}_i = \mathbf{h}(\mathbf{c}_i)$ and $g(X;\mathbf{w}_i)$ behaves similarly as $f(X;\mathbf{c}_i)$. Note that $\mathbf{h}(\cdot)$ determines the projection region Ω and depends on the chosen family of functions $g(X;\mathbf{w})$. When $\mathbf{h}(\cdot)$ is linear, then the projection region is referred to as a *linear manifold* in the parameter space of the model. Otherwise when $\mathbf{h}(\cdot)$ is nonlinear, the resulting projection region is a *nonlinear manifold.*

In most applications, however, only a finite number of time series S_i, $i = 1, \cdots, K$, corresponding to a finite number of parameter values $\mathbf{c}_i$, are available. Moreover, the function $g(X;\mathbf{w})$ is predetermined. Thus, the problem is to find $\mathbf{w}_1, \ldots, \mathbf{w}_K$ such that $g(X;\mathbf{w}_i)$ approximates $f(X;\mathbf{c}_i)$ for the ith time series. In this case, the projection region is the region where the set $\{\mathbf{w}_i\}$ is located. In other words, the projection region is the region in the parameter space of $g(X;\mathbf{w})$ defined by the set of points $\{\mathbf{w}_i\}$ corresponding to all possible time series that can be generated by varying the system parameters $\mathbf{c}$ within Γ. The problem is to determine Ω for the chosen $g(X;\mathbf{w})$ using the finite set of points $\{\mathbf{w}^i\}$ computed from the given time series. This problem will be addressed in the succeeding chapters.

To compare the bifurcation structures on Γ and Ω, some criteria are needed to determine their similarities. These criteria are defined as follows: Suppose that for any system parameter $\mathbf{c} \in \Gamma$ there exists a $\mathbf{w} = \mathbf{h}(\mathbf{c}) \in \Omega$ such that $\| f(X;\mathbf{c}) - g_\varepsilon(X;\mathbf{w}) \| < \varepsilon$ for some small ε, for X in some compact set, and for some appropriate norm, then the bifurcation diagram of $g_\varepsilon(X;\mathbf{w})$ is similar to that of $f(X;\mathbf{c})$ if the following conditions hold:

1. *Let us assume that X^* is a fixed point of $f(X;\mathbf{c})$, for a given $\mathbf{c}$, then there is a fixed point $\tilde{X}^*$ of $g(X;\mathbf{w})$ such that $\tilde{X}^* = X^* + \mathcal{O}(\varepsilon)$, except possibly for a parameter set of measure $\mathcal{O}(\varepsilon)$. Furthermore, X^* and $\tilde{X}^*$ have the same stability.*

2. *To make sure that* $g(X;\mathbf{w})$ *does not introduce spurious stable equilibria, we need also to assume that for all* $\mathbf{c}$*, except possibly on a set of measure* $\mathcal{O}(\varepsilon)$*,* $f(X;\mathbf{c})$ *and* $g(X;\mathbf{w})$ *have the same number of stable equilibria in* $\| X \| < M$.

3. *Finally, we need* $g(X;\mathbf{w})$ *to reproduce the various dynamics displayed by* $f(X;\mathbf{c})$*, so that we assure that if* $f(X;\mathbf{c})$ *undergoes a bifurcation at* $\mathbf{c}^*$*, then* $g(X;\mathbf{w})$ *undergoes a similar bifurcation at* $\tilde{\mathbf{w}}^* = \mathbf{h}(\mathbf{c}^*) + \mathcal{O}(\varepsilon)$.

Although in general, (1) - (3) are only useful when the said quantities (bifurcation points and fixed points of the original system) are known, these criteria are significant in clarifying the meaning of similar bifurcation diagrams. In the following section, a general reconstruction algorithm is introduced. This acts as the template algorithm where the algorithms in the remaining chapters are based.

4.4 A Reconstruction Template

The general approach to reconstruct bifurcation diagrams can be summarized into four major steps. The first step is to choose a parametrized function, say $g(X;\mathbf{w})$ with $\mathbf{w} \in \mathbb{R}^p$, that best approximates the predictor function $f(X;\mathbf{c})$. Like any other modeling problems, finding an appropriate $g(X;\mathbf{w})$ is always a difficult task especially with the extra requirement that $g(X;\mathbf{w})$ should also model other time series in the set $\{S_1, \cdots, S_K\}$. In this work, a three-layer fully-connected feedforward neural network and nonlinear autoregressive models are employed. The former is used for noiseless case while the latter deals with time series corrupted with noise.

The use of neural networks is motivated by the fact that neural networks are universal approximators [44], in the sense that they can approximate any continuous mapping (section 3.3). Moreover, neural networks can have an unlimited number of parameters by simply changing the number of units in the hidden layer without affecting the number of input and output variables. This allows more degrees of freedom in obtaining better function approximation without affecting the number of input variables which is dictated by the embedding dimension. On the other hand, the NAR model has been applied effectively in obtaining predictor functions for a number of systems (maps and flows), detecting nonlinearities in noisy time series (observation and dynamical noise), estimation of dynamical invariants, among others.

With this, the approximating function $g(\cdot\,;\mathbf{w})$ for the NN-case takes the form

$$y_{n+1} = g(y_n, \cdots, y_{n-(d-1)}; \mathbf{w})$$

$$= \sum_{j=1}^{s} w_j^0 q(\sum_{i=1}^{d} w_i^j y_{n-(i-1)} + y_b\theta_j) \tag{4.2}$$

where d is the embedding dimension, $\mathbf{w} \in \mathbb{R}^p$, $p = (d+2)s$, represents the weight vector with the interconnection weights (w_j^0, w_i^j, and θ_j) between neurons as its elements, s is the number of hidden units, y_b is the bias neuron, and $q(\cdot)$ is the activation function of the hidden layer neurons. The output neuron has a linear activation function. For the NAR-case, the approximating function $g(\cdot\,;\mathbf{a})$ takes the form

$$\begin{aligned} y_n &= g(y_{n-1}, \cdots, y_{n-d}; \mathbf{a}) \\ &= \sum_{m=0}^{M} a_m z_m(n) + \epsilon_n \end{aligned} \tag{4.3}$$

where $\{z_m(n)\}$ is composed of all the distinct combinations of the embedding space coordinates $(y_{n-1}, y_{n-2}, \cdots, y_{n-d})$ up to degree k [Eq. (3.56)].

The second step is to "project" each time series in the set $\{S_1, \cdots, S_K\}$ into the parameter space of the approximating function $g(\cdot\,;\mathbf{w})$. This means finding a $\mathbf{w}_i$ that minimizes $\parallel y_{n+1}^i - g(y_n^i, \cdots, y_{n-(d-1)}^i; \mathbf{w}) \parallel$ for some specified norm. Denote this $\mathbf{w}_i$ by $\mathbf{w}_{\min}^i$. Here, an error function of the form

$$e(\mathbf{w}) = \sum_{i=d}^{N} [y_i - g(y_{i-1}, \cdots, y_{i-d}; \mathbf{w})]^2, \tag{4.4}$$

where N represents the number of data points in each time series, is used. For K time series, K corresponding $\mathbf{w}_{\min}^i$s, each representing a point in the parameter space of the approximating function, will be obtained. These points can then be employed to obtain the projection region. This is the third step of the reconstruction algorithm. The success of the reconstruction depends on how well the projection region is approximated by $\{\mathbf{w}_{\min}^i\}$. One way to do this is to approximate the projection region by a linear manifold and obtain basis vectors from $\{\mathbf{w}_{\min}^i\}$ using principal component analysis (PCA). This will be discussed in the next section.

The final step is to take the bifurcation structure of the approximating function in the projection region as the reconstruction of the bifurcation structure of the given system. One can employ the methods described in section 2.2.

4.5 Linear Manifold Approximation

A key observation by Tokunaga, et al.[82] and Tokuda, et al.[81] was that if the covariance matrix of the estimated parameter values are formed and the eigenvalues of this matrix is computed, the number of significant eigenvalues corresponds to the number of system parameters. Moreover,

the bifurcation structure on the space span by the eigenvectors associated to the significant eigenvalues is qualitatively similar to that of the original system. Thus using PCA, they were able to reconstruct the bifurcation diagram of an unknown system from time series data.

This finding can be explained using the concept of projection region. In fact, this is the case when the projection region is a linear manifold in the parameter space of the model. In order to obtain an expression for this linear manifold, principal component analysis was employed. The dimension of the subspace was determined from the number of significant eigenvalues of the covariance matrix of $\{\mathbf{w}^i_{\min}\}$ and the subspace itself was spanned in terms of the eigenvectors associated with these eigenvalues.

Specifically, the sample covariance matrix of the p-dimensional data vectors $\{\mathbf{w}^i_{\min}\}$, $i = 1, \cdots, K$ given by

$$\mathbf{C} = \sum_{i=1}^{K} \delta\mathbf{w}_i \delta\mathbf{w}_i^T, \tag{4.5}$$

where $\delta\mathbf{w}_i = \mathbf{w}^i_{\min} - \bar{\mathbf{w}}$ and $\bar{\mathbf{w}}$ represents the mean, is formed. The eigenvalues of this matrix is then computed and the eigenvectors obtained. The number of nonzero eigenvalues corresponds to the number of system parameters. Moreover, the space spanned by the eigenvectors associated with the q significant eigenvalues defines a linear subspace which can then be taken as a linear approximation of the projection region. With this, any point on the projection region can be estimated using

$$\mathbf{w}_{\mathrm{PR}}(\alpha) = \sum_{i=1}^{q} \alpha_i \phi_i + \bar{\mathbf{w}} \tag{4.6}$$

where the ϕ_i's are the dominant eigenvectors, which can be taken as the new basis vectors, and the α_is are the expansion coefficients, which can be taken as the effective parameters. A qualitatively similar BD is then reconstructed with the expansion coefficients α_is as the new bifurcation parameters.

In general, the problem of obtaining the projection region is identified as a problem of interpolating $\mathbf{h}(\cdot)$ using the computed $\mathbf{w}^i_{\min}$s. The linear manifold approximation means a linear approximation of $\mathbf{h}(\cdot)$, that is,

$$\mathbf{h}(\mathbf{c}) = \mathbf{u}_o + \mathbf{T}\mathbf{c} + \mathbf{e} \tag{4.7}$$

where $\mathbf{T}$ is a $p \times k$ matrix and $\mathbf{e}$ represents the error term. PCA can then be used to estimate the matrix $\mathbf{T}$ from the $\mathbf{w}^i_{\min}$s and thus, the number of significant eigenvalues of $\mathbf{C}$, which gives the dimension of the projection region, can correspond to the number of bifurcation parameters of the system which is given by the dimension of $\mathbf{c}$.

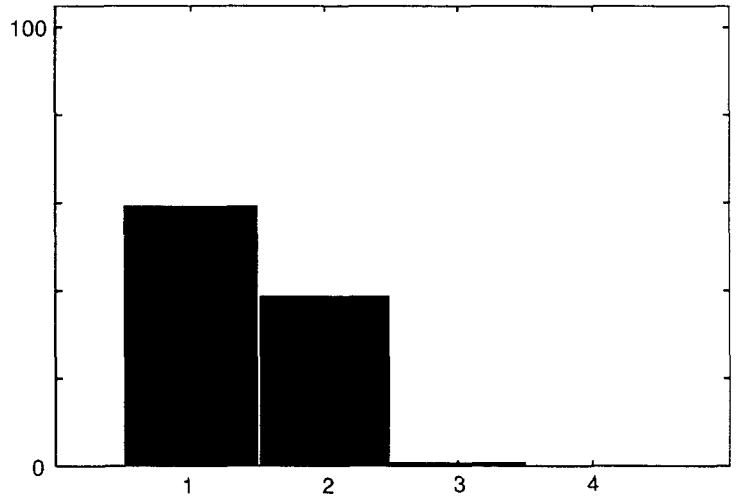

Figure 4.1: Significant eigenvalues of the covariance matrix formed by the weight vectors $\mathbf{w}^i_{\min}$'s obtained after training the network. The y-axis is the relative value in percentage, that is, abs(eigenvalue)/$\sum$(eigenvalues). The number of significant eigenvalues is equal to 2 which corresponds to the number of system parameters.

To illustrate this, consider again the well-known Hénon map:

$$\begin{aligned} x_{n+1} &= g(x_n, x_{n-1}; \Lambda) \\ &= 1 - p_1 x_n^2 + p_2 x_{n-1} \end{aligned} \tag{4.8}$$

where $\Lambda = (p_1, p_2)$. Using Eq. (4.8), ten time series were generated each of length $N = 100$ for parameter values

$$\begin{aligned} p_1 &= 0.135\cos[2\pi(j-1)/10] + 1.346 \\ p_2 &= 0.135\sin[2\pi(j-1)/10] + 0.2 \\ j &= 1, \cdots, 10. \end{aligned} \tag{4.9}$$

A neural network with an input-output function given by Eq. (4.2) was employed in approximating the map. The network parameters were as follows: $d = 2$, $s = 7$, $y_b = 1$, and $q(x) = 1/[1 + \exp(-x)]$.

The network was trained using backpropagation. Minimization of the error function given by Eq. (4.4) was done using the BFGS algorithm[69]. After computing the set of weights $\{\mathbf{w}^i_{\min}\}$, the covariance matrix [Eq. (4.5)] was set up. The significant eigenvalues of this matrix together with their respective eigenvectors were then computed. The eigenvalues are shown in Fig. 4.1.

From the figure, the number of significant eigenvalues is the same as the number of system parameters. This suggests that the dimension of the projection region is equal to the number of bifurcation parameters and thus, establishing the correspondence between the two parameter sets. Any point on the projection region can be expressed in terms of the significant eigenvectors associated with the significant eigenvalues and is given by Eq. (4.6). Using Eq. (4.6), the BD of the approximating network on this region can be constructed by varying the α_is. This BD then serves as the reconstruction of the BD of the original map. See for example [82] for the actual BD reconstruction of the Hénon map using this approach.

Chapter 5

Linear-In-Parameter Maps

5.1 Introduction

In general, the linear manifold approximation presented in the previous chapter is not always valid for any system. It is imperative, therefore, to know 1) when there is an appropriate linear manifold and 2) how it can be systematically located. The purpose of this chapter[3] is to provide partial answers to such questions. More precisely, it will be shown that there is a class of maps, henceforth referred to as linear-in-parameter (LIP) maps, for which a linear manifold of dimension equal to the number of bifurcation parameters and bifurcation structure that is qualitatively similar to that of the map can always be found in the weight space of the approximating network. The problem of recognizing time series generated by this class of maps is also discussed. Finally, an algorithm is proposed to reconstruct BDs of LIP maps using predictor functions obtained by neural networks. This algorithm is flexible so that other classes of predictors, apart from neural networks, can be used in the reconstruction.

5.2 An LIP Map: An Illustration

In this section, the Hénon map example in section 4.5 is used to illustrate the reconstruction of LIP maps from time series. It is supposed that $g(x_n, x_{n-1}; \Lambda)$ is known at three parameter values $\Lambda^1 = (p_1^1, p_2^1)$, $\Lambda^2 = (p_1^2, p_2^2)$, and $\Lambda^3 = (p_1^3, p_2^3)$. Denote these maps as g_1, g_2, and g_3, respectively. With this, Eq. (4.8) takes the form

$$g_1 \;=\; g(x_n, x_{n-1}; \Lambda^1) = 1 - p_1^1 x_n^2 + p_2^1 x_{n-1} \tag{5.1}$$

[3]The contents of this chapter also appear in Bagarinao, et al., 1999 [8].

$$g_2 = g(x_n, x_{n-1}; \Lambda^2) = 1 - p_1^2 x_n^2 + p_2^2 x_{n-1} \tag{5.2}$$

$$g_3 = g(x_n, x_{n-1}; \Lambda^3) = 1 - p_1^3 x_n^2 + p_2^3 x_{n-1}. \tag{5.3}$$

The goal of the new reconstruction scheme is to approximate Eq. (4.8) using the three functions g_1, g_2, and g_3. To do this, one can take the function

$$g(x_n, x_{n-1}; \alpha, \beta) = g_1 + \alpha(g_1 - g_2) + \beta(g_1 - g_3). \tag{5.4}$$

The new set of parameters (α, β) is related to the old set (p_1, p_2) via the relation

$$p_1 = p_1^1 + \alpha(p_1^1 - p_1^2) + \beta(p_1^1 - p_1^3) \tag{5.5}$$

$$p_2 = p_2^1 + \alpha(p_2^1 - p_2^2) + \beta(p_2^1 - p_2^3) \tag{5.6}$$

From this, the BD of the Hénon map can be reconstructed using Eq. (5.4) by varying α and β. Thus, only three predictor functions are needed to realize the reconstruction. This requirement can be easily met by using the given time series to approximate g_1, g_2, and g_3.

The most compelling question at hand is how to use Eq. (5.4) in finding a linear subspace in the weight space of the approximating network. To answer this, one can rewrite Eqs. (5.1), (5.2), and (5.3) in the form similar to Eq. (4.2), that is,

$$\begin{aligned} \tilde{g}_1 &= \tilde{g}(\cdot; \mathbf{w}^1) \\ &= \sum_{j=1}^{s} w_j^1 q(\sum_{i=1}^{d} w_i^{1,j} x_{n-(i-1)} + x_b \theta_j^1) \end{aligned} \tag{5.7}$$

$$\begin{aligned} \tilde{g}_2 &= \tilde{g}(\cdot; \mathbf{w}^2) \\ &= \sum_{j=1}^{s} w_j^2 q(\sum_{i=1}^{d} w_i^{2,j} x_{n-(i-1)} + x_b \theta_j^2) \end{aligned} \tag{5.8}$$

$$\begin{aligned} \tilde{g}_3 &= \tilde{g}(\cdot; \mathbf{w}^3) \\ &= \sum_{j=1}^{s} w_j^3 q(\sum_{i=1}^{d} w_i^{3,j} x_{n-(i-1)} + x_b \theta_j^3) \end{aligned} \tag{5.9}$$

where $\tilde{g}_1$, $\tilde{g}_2$, and $\tilde{g}_3$ are the network approximations of g_1, g_2, and g_3, respectively. In terms of the $\tilde{g}_i$'s, Eq. (5.4) can be written as follows:

$$g(x_n, x_{n-1}; \alpha, \beta) \approx (1 + \alpha + \beta)\tilde{g}_1 - \alpha\tilde{g}_2 - \beta\tilde{g}_3. \tag{5.10}$$

Equation 5.10 can be interpreted as a composite network made up of the three networks $\tilde{g}_1$, $\tilde{g}_2$, and $\tilde{g}_3$. See Fig. 5.1(A). Since $(1 + \alpha + \beta)$, α, and β are multiplicative factors of the $\tilde{g}_1$, $\tilde{g}_2$, and $\tilde{g}_3$, respectively, their variation only affects the hidden-to-output (outer) weights of

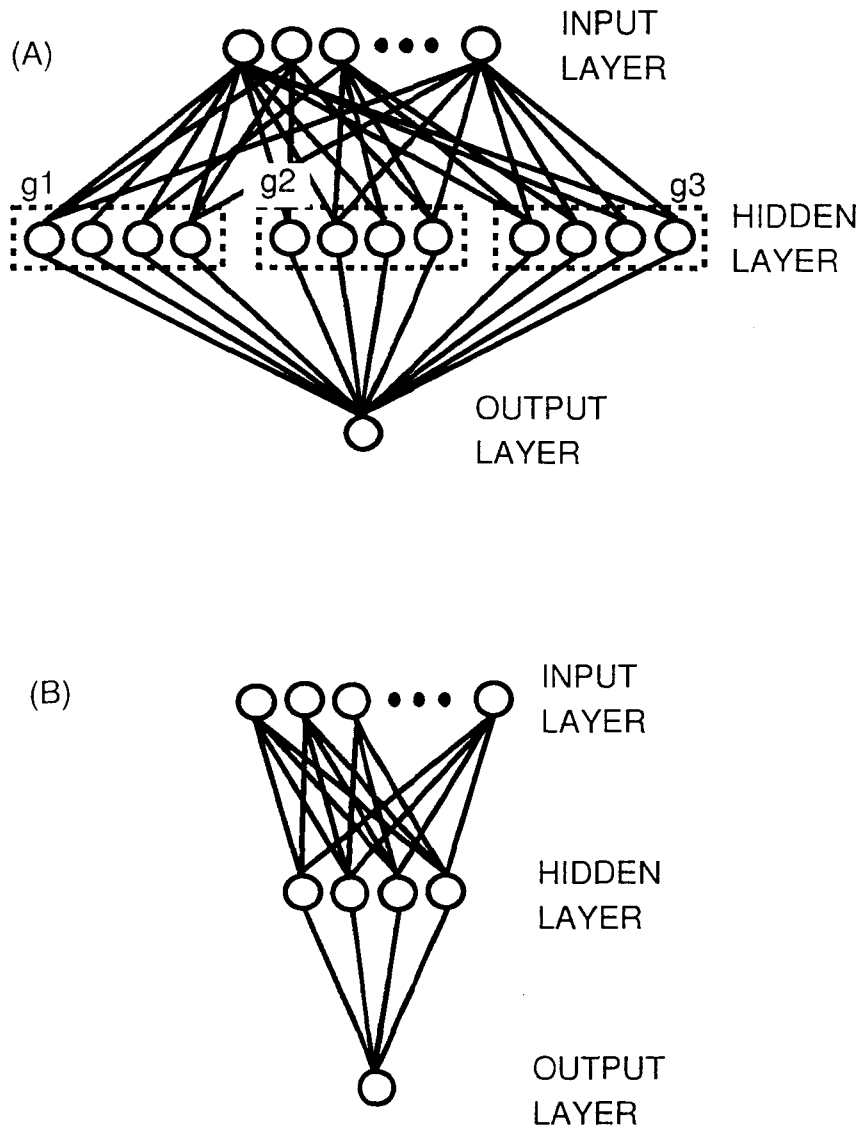

Figure 5.1: (A) The composite network made up of the three predictor functions $\tilde{g}_1$, $\tilde{g}_2$, and $\tilde{g}_3$. The inner weights are unchanged while the outer weights are given by Eq. (5.11). (B) When the inner weights of the three networks are the same, the composite network can be trimmed to make the number of hidden units equal to that of the predictor functions. For this case, the outer weights are given by Eq. (5.12).

the network while the input-to-hidden (inner) weights remain unchanged. From Eq. (5.10), the outer weights of the composite network can be written as follows:

$$\mathbf{w}_c^o = (1 + \alpha + \beta)\mathbf{w}^{o,1} \oplus (-\alpha)\mathbf{w}^{o,2} \oplus (-\beta)\mathbf{w}^{o,3} \tag{5.11}$$

where $\mathbf{w}_c^o \in \mathbb{R}^{3s}$ represents the outer weights of the composite network, $\mathbf{w}^{o,i} \in \mathbb{R}^s$ represents the outer weights of $\tilde{g}_{i=1,2,3}$ and $\oplus$ denotes a direct sum.

Equation (5.11) defines a linear subspace in the space of the outer weights of the composite network. This subspace is generated by varying the parameters α and β. The dimension is equal to the number of parameters (two for this case) and the bifurcation structure of the network on this subspace is qualitatively the same as that of the Hénon map since the composite network approximates Eq. (5.4). Equation (5.11) thus locates the desired linear manifold without using PCA. All it requires are the outer weights of the three networks $\tilde{g}_1$, $\tilde{g}_2$, and $\tilde{g}_3$.

Further simplification of the above results can be attained by assuming that the inner weights of all three networks are equal. This assumption is justifiable when one thinks of the $\tilde{g}_i$s as composed of subnetworks each approximating the functions $H_1(x_n, x_{n-1}) = x_n^2$, $H_2(x_n, x_{n-1}) = x_{n-1}$, and $H_3(x_n, x_{n-1}) = 1$ of the Hénon map. Following the same line of reasoning as above, each p_i being multiplicative factors of the $H_i(\cdot)$ can affect only the outer weights of the $\tilde{g}_i$'s. Thus, with an appropriate choice of the number of hidden units and training algorithm, it is possible to keep the inner weights equal for all the trained networks. In the simulation, a grouping of hidden units was observed for the case of the Hénon map. Four hidden units collectively approximate the function x_n^2, two hidden units do not contribute to the input-output function (outer weights are zero), and the remaining one hidden unit approximates the rest of the terms. This is shown in Fig. 5.2.

With this assumption, the composite network can be trimmed so that the number of hidden units will be the same as that of the $\tilde{g}_i$'s, that is, equal to s [see Fig. 5.1(B)]. The linear subspace can still be located in the outer weights' space. This subspace is spanned by the vectors $(\mathbf{w}^{o,1} - \mathbf{w}^{o,2})$ and $(\mathbf{w}^{o,1} - \mathbf{w}^{o,3})$. Any point on this subspace is given by

$$\mathbf{w}_T^o = \mathbf{w}^{o,1} + \alpha(\mathbf{w}^{o,1} - \mathbf{w}^{o,2}) + \beta(\mathbf{w}^{o,1} - \mathbf{w}^{o,3}) \tag{5.12}$$

where $\mathbf{w}_T^o \in \mathbb{R}^s$ represents the outer weights of the trimmed network and $\mathbf{w}^{o,i} \in \mathbb{R}^s$ represents the outer weights of $\tilde{g}_{i=1,2,3}$.

Equation (5.11) or (5.12) defines the desired linear subspace in terms of the computed outer weight values of the predictor functions. It guarantees the existence of this subspace in the weight space of the network for the case of the Hénon map.

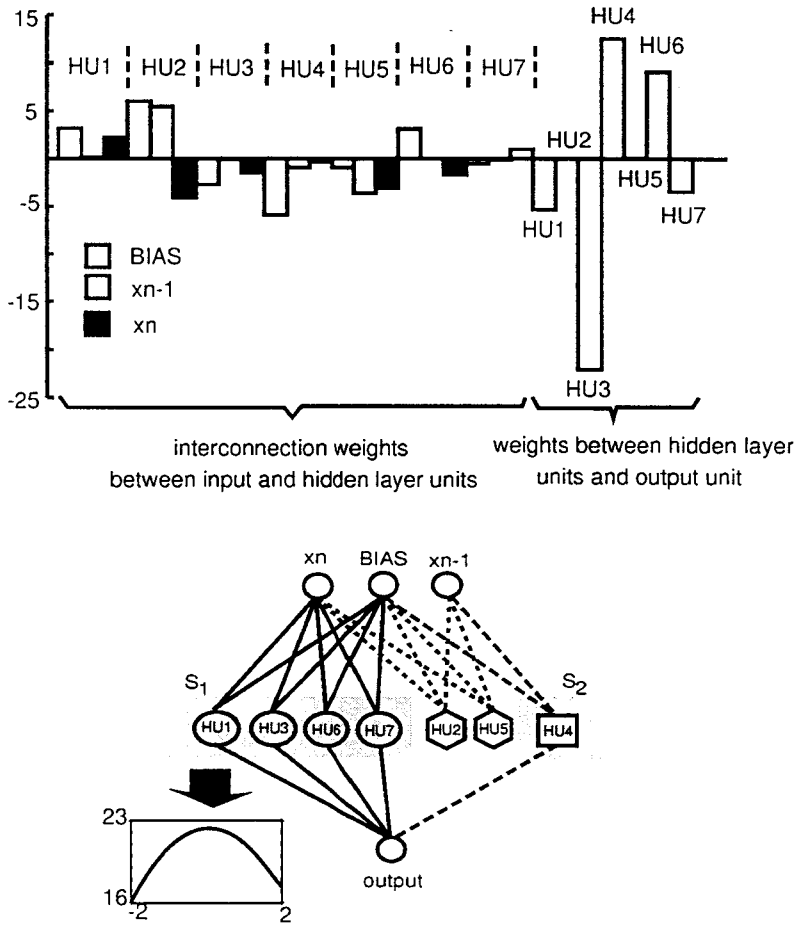

Figure 5.2: The estimated weight values of the trained network. Observed how the weights are distributed to attain the different functions in the original Hénon map.

Figure 5.3: Reconstructed bifurcation diagram of the Hénon map on the linear subspace defined by Eq. (5.11) in the parameter region $p_1 \in [1.15, 1.548]$ and $p_2 \in [0.0, 0.398]$. See Table 5.1 for the color coding.

To reconstruct the bifurcation diagram, three predictor functions were used. The functions were approximated by neural networks and corresponded to parameter values $\Lambda^1 = (1.388, 0.328)$, $\Lambda^2 = (1.211, 0.2)$, and $\Lambda^3 = (1.388, 0.072)$. Equation (5.11) is used in the reconstruction. The parameters α and β are varied from $[a_1, a_2]$ and $[b_1, b_2]$, respectively, such that $p_1 \in [1.15, 1.548]$ and $p_2 \in [0.0, 0.398]$. The reconstructed BD using this scheme is shown in Fig. 5.3. Table 5.1 summarizes the color code used in the figure. From the figure, it is evident that the new scheme captures the different dynamics of the Hénon map (See [82] for the BD, both original and reconstructed using the PCA-based approach discussed in the previous chapter, of the Hénon map on the same parameter region).

5.3 Definition and BD Reconstruction

The reason why the reconstruction using Eq. (5.4) works for the case of the Hénon map is that the Hénon map belongs to a large class of maps called LIP maps. An LIP map is a map that can be written in the following manner:

$$g(x; \mathbf{\Lambda}) = \sum_{i=1}^{k} \lambda_i H_i(x) + H_{\text{aux}}(x) \tag{5.13}$$

where $g : \mathbb{R}^d \times \mathbb{R}^k \mapsto \mathbb{R}^1$, $x \in \mathbb{R}^d$, $\mathbf{\Lambda} = (\lambda_1, \lambda_2, \cdots, \lambda_k) \in \mathbb{R}^k$ are the parameters, and the functions $H_i(x)$ are linearly independent. The functions $H_i(x)$ are linearly independent if the

COLOR	COMMENTS
blue	period 1 or 8 or 15
red	period 2 or 9 or 16
magenta	period 3 or 10 or 17
cyan	period 4 or 11 or 18
brown	period 5 or 12 or 19
yellow	period 6 or 13 or 20
white	period 7 or 14 or 21
black	period greater than 21
	quasi-periodic region
	chaotic region

Table 5.1: Color code used in Fig. 5.3.

only solution to the equation

$$\sum_{i=1}^{r} \alpha_i H_i(x) = 0 \tag{5.14}$$

is the trivial solution $\alpha_i = 0$ for all values of i.

To reconstruct the BD of an LIP map, it is assumed that at least $k+1$ predictor functions $g_i(x) = g(x; \mathbf{\Lambda}^i)$ for fixed parameter values $\mathbf{\Lambda}^i$, $i = 1, \cdots, k+1$, are known. Two possible cases can arise. The first case is when $H_{\text{aux}}(x)$, the last term in Eq. (5.13), is zero. Under this condition, $g(x; \mathbf{\Lambda})$ can be approximated by the equation

$$g(x, \beta) = \sum_{i=1}^{k} \beta_i g_i(x). \tag{5.15}$$

The second case is when $H_{\text{aux}}(x)$ is not equal to zero. For this case, Eq. (5.15) cannot be used since it will introduce parameters to the term $H_{\text{aux}}(x)$ of each $g_i(x)$ which should not vary with parameters. To remedy this situation, the following equation is employed:

$$g(x, \beta) = g_1(x) + \sum_{i=1}^{k} \beta_i [g_1(x) - g_{i+1}(x)]. \tag{5.16}$$

By taking the difference between two predictor functions before multiplying by the new parameters, the contribution of $H_{\text{aux}}(x)$ in each predictor function is effectively canceled out. This removes the possibility that $H_{\text{aux}}(x)$ is made to vary with the new parameters. Equation (5.4)

is an example of Eq. (5.16). In both equations [Eqs. (5.15) and (5.16)], the β_is serve as the new bifurcation parameters.

When the g_i's are approximated using neural networks, a linear subspace can be found in the space defined by the outer weights of the approximating network. For the composite network, this subspace is given by

$$\mathbf{w}_c^o = \bigoplus_{i=1}^{k} \beta_i \mathbf{w}^{o,i} \tag{5.17}$$

for the case when $H_{\text{aux}}(x)$ is zero and $\mathbf{w}_c^o \in \mathbb{R}^{ks}$ represents the outer weights of the composite network. For the case when $H_{\text{aux}}(x)$ is not zero, $\mathbf{w}_c^o$ has the form

$$\mathbf{w}_c^o = \left(1 + \sum_{i=1}^{k} \beta_i\right) \mathbf{w}^{o,1} \oplus \bigoplus_{i=1}^{k} (-\beta_i) \mathbf{w}^{o,i+1} \tag{5.18}$$

where $\mathbf{w}_c^o \in \mathbb{R}^{(k+1)s}$ represents the outer weights of the composite network, $\mathbf{w}^{o,i} \in \mathbb{R}^s$ represents the outer weights of the known predictor functions, and $\oplus$ denotes direct summation. Similarly for the trimmed network, the outer weight vector is given by

$$\mathbf{w}_T^o = \mathbf{w}^{o,1} + \sum_{i=1}^{k} \beta_i (\mathbf{w}^{o,1} - \mathbf{w}^{o,i+1}) \tag{5.19}$$

for the case when $H_{\text{aux}}(x)$ is not zero, or

$$\mathbf{w}_T^o = \sum_{i=1}^{k} \beta_i \mathbf{w}^{o,i} \tag{5.20}$$

for the case when $H_{\text{aux}}(x)$ is zero. Here, $\mathbf{w}_T^o \in \mathbb{R}^s$ represents the outer weights of the trimmed network, $\mathbf{w}^{o,i}$ represents the outer weights of each predictor functions, and the β's are the new bifurcation parameters.

The preceding four equations defined the desired linear subspace which can be generated by varying the new parameters β. The dimension of this subspace is equal to the number of bifurcation parameters. Moreover, the BD of the network on this subspace is qualitatively the same as the BD of the LIP map. Using either one of these equations, one can reconstruct the BD of the LIP map from the given time series. Two important issues remain to be considered: 1) how to determine the number of bifurcation parameters k from the time series themselves; and 2) how to know if indeed the time series are generated from an LIP map. These issues are explored in the next section.

5.4 Identifying LIP Maps

Suppose that $g(x, \Lambda)$ [Eq. (5.13)] is known at K different values of Λ, say Λ^i, $i = 1, \cdots, K$. Let $g_i(x) = g(x, \Lambda^i)$. The goal is to determine the number of parameters k using $g_i(x)$. To do this, a function $X^{ij}(x)$ is defined such that

$$\begin{aligned} X^{ij}(x) &= g_i(x) - g_j(x) && (5.21)\\ &= \sum_{r=1}^{k} (\lambda_r^i - \lambda_r^j) H_r(x) && (5.22)\\ &= \sum_{r=1}^{k} \alpha_r^{ij} H_r(x) && (5.23) \end{aligned}$$

where $\alpha_r^{ij} = (\lambda_r^i - \lambda_r^j)$. This effectively cancels the $H_{\text{aux}}(x)$ of each $g_i(x)$. A vector $\mathbf{V}_M^{ij}$ is formed by evaluating $X^{ij}(x)$ at M discrete values of x:

$$\begin{aligned} \mathbf{V}_M^{ij} &= [X^{ij}(x_1), \cdots, X^{ij}(x_M)]^T && (5.24)\\ i &= 1, \cdots, K-1 \\ j &= i+1, \cdots, K. \end{aligned}$$

The vector $\mathbf{V}_M^{ij}$ is therefore $X^{ij}(x)$ evaluated at discrete points x_m, $m = 1, \cdots, M$. Similarly, the vector $\mathbf{h}_k$ can be defined as $H_k(x)$ evaluated at the same discrete points x_m, $m = 1, \cdots, M$. From Eq. (5.23), $\mathbf{V}_M^{ij}$ can also be written as

$$\mathbf{V}_M^{ij} = \sum_{r=1}^{k} \alpha_r^{ij} \mathbf{h}_r. \tag{5.25}$$

The vectors $\mathbf{h}_r$'s are in general independent since the $H_r(x)$'s are independent functions [Eq. (5.14)]. These vectors can then be thought of as basis vectors for $\mathbf{V}_M^{ij}$ spanning a k-dimensional subspace of $\mathbb{R}^M$. Therefore, the vectors $\mathbf{V}_M^{ij}$ can only be found on this subspace which can be generated by varying the parameters α_r^{ij}. Thus, if a finite number of vectors $\mathbf{V}_M^{ij}$ are known, PCA can be used to determine the dimension of the space occupied by these vectors. This would yield the important result that the number of significant eigenvalues of the covariance matrix formed by the $\mathbf{V}_M^{ij}$ would be equal to the number of parameters k.

In terms of the eigenvectors associated with the significant eigenvalues of the covariance matrix, $\mathbf{V}_M^{ij}$ can be written as

$$\mathbf{V}_M^{ij} = \bar{\mathbf{V}}_M + \sum_{r=1}^{k} \gamma_r^{ij} \phi_r \tag{5.26}$$

where γ_r^{ij}'s represent the expansion coefficients, $\bar{\mathbf{V}}_M$ is the mean of the $\{\mathbf{V}_M^{ij}\}$ and the ϕ_rs are the eigenvectors. These eigenvectors can be interpreted as an orthonormalization of the independent

vectors $\mathbf{h}_r$. One can also think of the continuous equivalent of ϕ_r denoted as $\Phi_r(x)$. In terms of these functions, Eq. (5.13) can be written as

$$g(x,\mathbf{\Gamma}) = \sum_{r=1}^{k} \gamma_r \Phi_r(x) + \Phi_{\text{aux}}(x) \tag{5.27}$$

where $\mathbf{\Gamma} = (\gamma_1, \cdots, \gamma_l)$ represents some effective parameter. If the $\mathbf{h}_r$'s or the $H_r(x)$'s are themselves mutually orthogonal, an idea of the functional form of the original $H_r(x)$'s can be extracted from the ϕ_r or $\Phi_r(x)$.

With these results, the number of bifurcation parameters, together with the functional form of $g(x;\mathbf{\Gamma})$, can be determined from the time series. This can be done by approximating the $g_i(x)$'s using predictor functions $\tilde{g}_i(x)$ obtained via neural networks or some other approach. Once the $\tilde{g}_i(x)$ are known, the difference between each $\tilde{g}_i(x)$ can be estimated and the difference function $X^{ij}(x)$ [Eq. (5.23)] can be obtained. The set of vectors $\{\mathbf{V}_M^{ij}\}$ is then generated by evaluating $X^{ij}(x)$ at M different values of x. After which, the covariance matrix [Eq. (4.5)] of these vectors is formed and the eigenvalues and their corresponding eigenvectors computed. The number of significant eigenvalues is equal to the number of bifurcation parameters if the time series comes from an LIP map. The associated eigenvectors are related to the original map via Eq. (5.27).

To use these results in identifying time series generated from LIP maps, one can take subsets of the set of vectors $\{\mathbf{V}_M^{ij}\}$. The membership of each subset can be chosen randomly from the original set. For each subset, PCA is applied and the number of significant eigenvalues and the associated eigenvectors are determined. For LIP maps, the result should be the same, that is, the same number of significant eigenvalues and the same eigenvectors should be obtained. Otherwise the map is non-LIP. This will be illustrated in the examples considered (section 5.5).

Finally for the case where the λs in Eq. (5.13) are functions of parameters rather than the parameters themselves, a generalized LIP (GLIP) map can be defined as follows:

$$g(x;\mathbf{\Lambda}) = \sum_{i=1}^{r} f_i(\mathbf{\Lambda}) H_i(x) \tag{5.28}$$

where $\mathbf{\Lambda} = (\lambda_1, \cdots, \lambda_k) \in \mathbb{R}^k$ are the parameters and the $f_i(\mathbf{\Lambda})$s are functions of the parameters and are linearly independent, that is, they satisfy Eq. (5.14). For GLIP maps, the results in the previous section and the current one should still apply for $f_i(\mathbf{\Lambda})$ rather than $\mathbf{\Lambda}$. This means that instead of the number of actual parameters k, the method will determine the number of independent functions $f_i(\mathbf{\Lambda})$, which is equal to r. Also, the reconstructed BD is relative to the parameter $\psi_i[= f_i(\mathbf{\Lambda})]$ rather than to the λ_is. This was illustrated in sections 4.5 and 5.2 using the Hénon

map. Observe that the time series were generated using parameter values $p_1 = r_o \cos(\theta) + c_1$ and $p_2 = r_o \sin(\theta) + c_2$. In the example, θ was varied while keeping the other variables (r_o, c_1, and c_2) fixed to some values in generating the time series. This effectively reduces the number of parameters to one, which is θ. However, the method detected the parameters $p_1 = f_1(\theta)$ and $p_2 = f_2(\theta)$. And since these two parameters were detected, the reconstruction was based on p_1 and p_2 rather than on θ.

5.5 Examples

In this section, the procedure in reconstructing BDs of LIP maps will be illustrated via some examples. In all the examples, a three-layer neural network with input-output function given by Eq. (4.2) was used to approximate the predictor functions. For each of the system, first, the number of parameters is discerned from the significant eigenvalues of the covariance matrix of $\{\mathbf{V}_M^{ij}\}$, then the functional form of the map is determined from the eigenvectors associated with the significant eigenvalues, and finally, the BD is reconstructed.

To get the functional form of the map using Eq. (5.27), one also needs to specify $\Phi_{\text{aux}}(x)$ which can not be obtained directly from the eigenvectors. To do this, first each $\tilde{g}_i(x)$ is evaluated at M values of x (the same values used in obtaining $\mathbf{V}_M^{ij}$) and the vector $\tilde{G}_i = [\tilde{g}_i(x_1), \cdots, \tilde{g}_i(x_M)]^T$ is formed. Then each vector is projected onto the eigenvectors ϕ_k. From this, a discrete equivalent of the function $\Phi_{\text{aux}}(x)$ is given by the equation

$$\phi_{\text{aux}} = \tilde{G}_i - \sum_{r=1}^{k} \gamma_r^i \phi_r \tag{5.29}$$

where ϕ_{aux} is the discretized equivalent of $\Phi_{\text{aux}}(x)$ and γ_r^i is the projection of the vector $\tilde{G}_i$ onto the rth eigenvector ϕ_r. This equation is the discrete equivalent of Eq. (5.27).

The Hénon map

An example of a LIP map is the Hénon map. It is given by the following equations:

$$\begin{aligned} x_{n+1} &= 1 - p_1 x_n^2 + y_n \\ y_{n+1} &= p_2 x_n \end{aligned} \tag{5.30}$$

with

$$H_1(x_n, x_{n-1}) = x_n^2$$

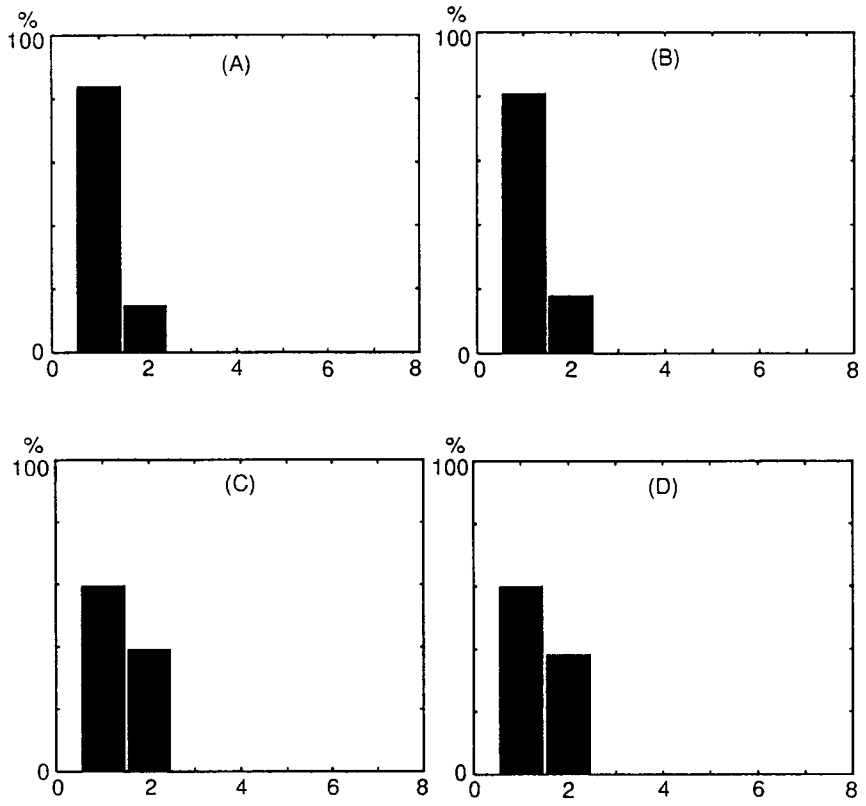

Figure 5.4: Significant eigenvalues of the covariance matrix form by the vector V_M^{ij}. Same unit (in percentage) as in Fig. 4.1. (A) The first group with five predictor functions $\tilde{g}_1$ to $\tilde{g}_5$. (B) The second group with $\tilde{g}_6$ to $\tilde{g}_{10}$. (C) The third group with $\tilde{g}_i$, i even. (D) The ten predictor functions. In all cases, there are two significant eigenvalues indicating that the time series were generated from an LIP map.

$$\begin{aligned} f_1(p_1, p_2) &= p_1 \\ H_2(x_n, x_{n-1}) &= x_{n-1} \\ f_2(p_1, p_2) &= p_2 \\ H_{\text{aux}}(x_n, x_{n-1}) &= 1 \end{aligned}$$

The ten predictor functions obtained in section 4.5 were used in the analysis. From these predictor functions, 4 groups were formed: the first group was composed of the predictor functions $\tilde{g}_1$ to $\tilde{g}_5$; the second was from $\tilde{g}_6$ to $\tilde{g}_{10}$; the third group was composed of $\tilde{g}_i$, i even; and the last group was composed of the entire ten predictor functions. For each group, the function $X^{ij}(x)$ was evaluated on a square grid given by $x_n = -2.5 + 5i/20$, $i = 1, \cdots, 20$ and $x_{n-1} = -2.5 + 5j/20$, $j = 1, \cdots, 20$ to generate the set of vectors $\{\mathbf{V}_{M=400}^{ij}\}$. The eigenvalues, and their associated eigenvectors, of the covariance matrix of $\{\mathbf{V}_{400}^{ij}\}$ were then computed. Fig-

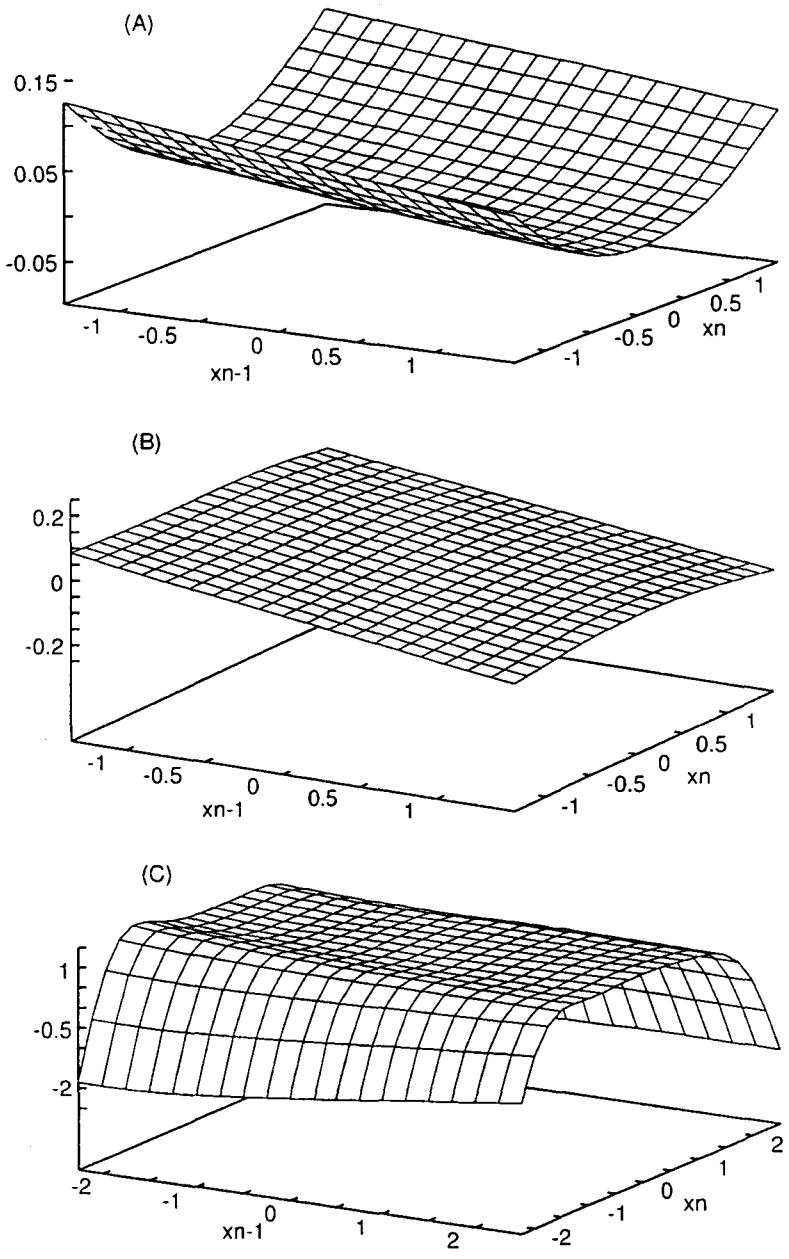

Figure 5.5: The eigenvectors of the Henon map. (A) The eigenvector associated with the first significant eigenvalue. This is parabolic in x_n. (B) The eigenvector associated with the second significant eigenvalue. This is linear in x_{n-1}. (C) $\Phi_{\text{aux}}(x)$. The surface is almost flat and approximately equal to one except near the edges where it drops sharply.

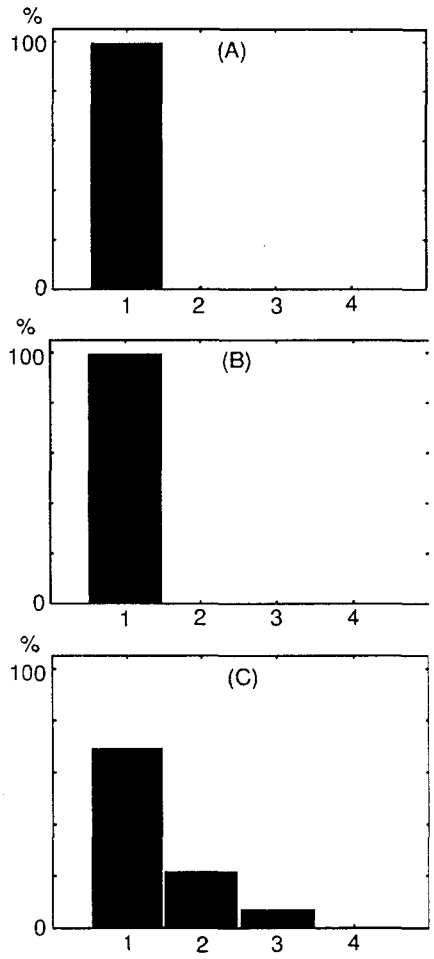

Figure 5.6: Significant eigenvalues of the covariance matrix form by the vectors $\{V_M^{ij}\}$. Same unit (in percentage) as in Fig. 4.1. (A) Logistic map. (B) Cubic map. (C) Delayed logistic map. The number of significant eigenvalues is equal to the number of system parameters for all the systems considered.

ure 5.4 shows the significant eigenvalues to be equal to two for all groups (A)-(D). Since this number is the same for all groups, it can be concluded that the time series were generated from an LIP map. The normalized eigenvectors are shown in Fig. 5.5. Figure 5.5(A) is the first principal eigenvector which corresponds to the parabolic term of the map. In Fig. 5.5(B), the surface is almost planar corresponding to the term x_{n-1} and in (C), the $\Phi_{\text{aux}}(x)$-surface is close to one except near the boundaries. Thus, all the terms in the Hénon map are clearly identified. The reconstructed BD is shown in Fig. 5.3.

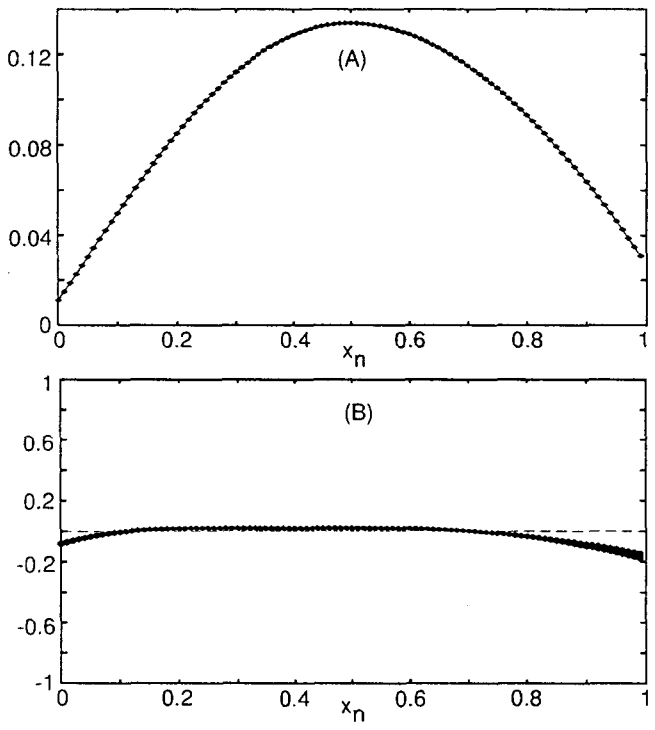

Figure 5.7: The logistic map. (A) The eigenvector associated with the only significant eigenvalue of the covariance matrix of $\{V_M^{ij}\}$. The curve is parabolic. (B) $\Phi_{\text{aux}}(x)$ is zero for $x \in [0.1, 0.75]$, consistent with Eq. (5.31).

The logistic map

The logistic map

$$x_{n+1} = 4\mu x_n(1 - x_n) \tag{5.31}$$

is an LIP map with

$$\begin{aligned} H_1(x_n) &= 4x_n(1 - x_n) \\ f_1(\mu) &= \mu \\ H_{\text{aux}}(x_n) &= 0. \end{aligned}$$

Ten time series of length $T = 100$ were generated for the parameter values $\mu_j = 1 - 5(j - 1)/10$, $j = 1, \cdots, 10$. The network parameters used in approximating Eq. (5.31) were as follows: $d = 1$, $s = 6$, $y_b = 1$, and $q(x)$ is sigmoidal. The BFGS algorithm was used to minimize the error function.

After computing the $\tilde{g}_i$s, $X^{ij}(x_n)$, $i = 1, \cdots, 10$, $j = i + 1$, was evaluated at different values of x_n given by $x_n = (i - 1)/100$, $i = 1, \cdots, 101$, then the vector $\mathbf{V}_{101}^{ij}$ was formed, and the principal components computed. Figure 5.6(A) shows one significant eigenvalue corresponding to the only bifurcation parameter of the logistic map. A plot of the normalized eigenvector is shown in Fig. 5.7(A). The eigenvector does exhibit the parabolic form of the logistic map. Moreover, $\phi_{\text{aux}}(x_n)$ is almost zero for $x \in [0.1, 0.75]$, as shown in Fig. 5.7(B). This is consistent with Eq. (5.31). The BD of the logistic map was reconstructed using Eq. (5.15) since $\phi_{\text{aux}}(x)$ is zero. For this case, the equation has the form $\tilde{g}(x_n; \beta) = \beta \tilde{g}_1(x_n)$. Figure 5.8(A) shows the reconstructed BD for $\beta \in [0, 1]$. Evidently, the reconstructed BD exhibits the period doubling route to chaos behavior, among others, of the logistic map.

The cubic map

Another example of an LIP map is the cubic map given by

$$x_{n+1} = x_n + \mu x_n - x_n^3 \tag{5.32}$$

with

$$\begin{aligned} H_1(x_n) &= x_n \\ f_1(\mu) &= \mu \\ H_{\text{aux}}(x_n) &= x_n - x_n^3 \end{aligned}$$

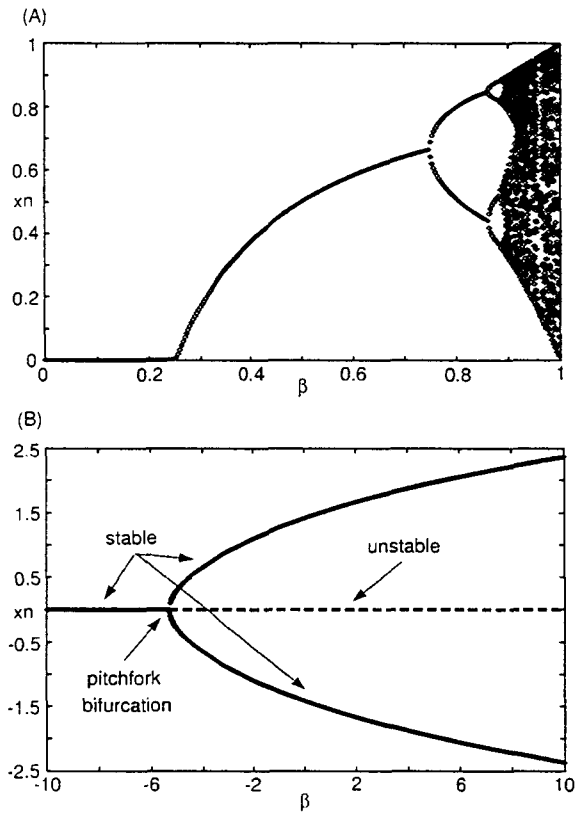

Figure 5.8: Reconstructed BD of (A) the logistic map and (B) the cubic map using the proposed algorithm.

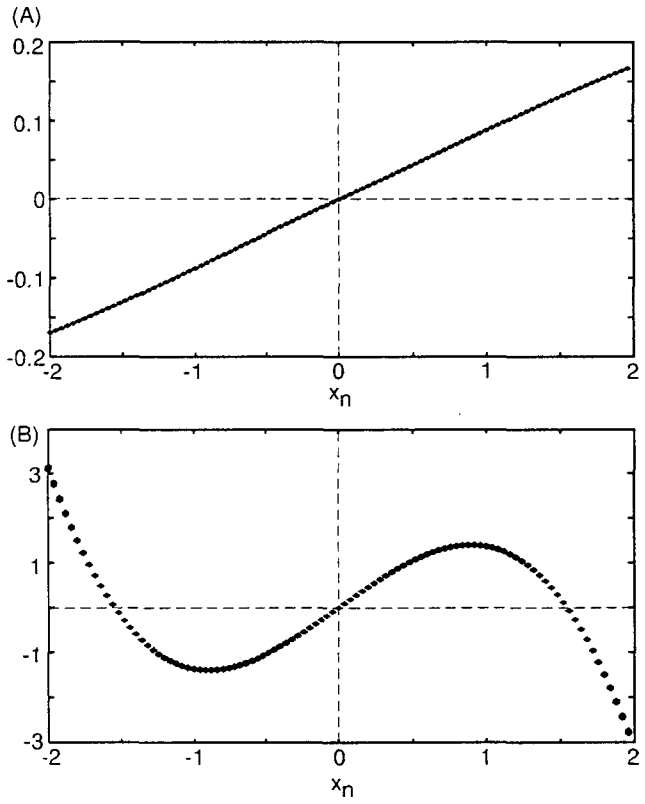

Figure 5.9: The cubic map. (A) The eigenvector associated with the only significant eigenvalue of the covariance matrix of $\{V_M^{ij}\}$. This is the linear term in Eq. (5.32). (B) The $\Phi_{\text{aux}}(x)$ of the ten predictor functions $\tilde{g}_i$. All $\Phi_{\text{aux}}(x)$ are the same.

For this example, ten time series with 100 data points each were generated for the parameter values $\mu_j = 2-4(j-1)/10$, $j = 1, \cdots, 10$. The interesting feature of this example is the pitchfork bifurcation at $\mu = 0$. The trained network has the following parameters: $d = 1$, $s = 5$, $y_b = 0$, and $q(x) = \tanh(x)$. For this system, $X^{ij}(x)$ was evaluated at $x = -2 + 4(i-1)/100$, $i = 1, \cdots, 101$; thus, $M = 101$. Figure 5.6(B) shows one significant eigenvalue which is consistent with the expected result. The normalized eigenvector and $\Phi_{\text{aux}}(x)$ are shown in Fig. 5.9. The eigenvector is linear as expected and $\Phi_{\text{aux}}(x)$ is cubic, which is the same for all the predictor functions. In this case, $\Phi_{\text{aux}}(x) \approx 2.4x - x^3$ is not exactly equal to $H_{\text{aux}}(x) = x - x^3$. The BD was reconstructed using the equation $\tilde{g}(x_n;\beta) = \tilde{g}_1(x_n) + \beta[\tilde{g}_1(x_n) - \tilde{g}_2(x_n)]$. The result is shown in Fig. 5.8(B) for $\beta \in [-10, 10]$. The reconstructed BD exhibits the pitchfork bifurcation of the original map.

The delayed logistic map

Another example of an LIP map studied in [82] is the delayed logistic map given by

$$\begin{aligned} x_{n+1} &= p_1(1-x_n)x_n + p_2 y_n + p_3(1-y_n)x_n \\ y_{n+1} &= x_n \end{aligned} \tag{5.33}$$

with

$$\begin{aligned} H_1(x_n, x_{n-1}) &= (1-x_n)x_n \\ f_1(p_1, p_2, p_3) &= p_1 \\ H_2(x_n, x_{n-1}) &= x_{n-1} \\ f_2(p_1, p_2, p_3) &= p_2 \\ H_3(x_n, x_{n-1}) &= (1-x_{n-1})x_n \\ f_3(p_1, p_2, p_3) &= p_3 \\ H_{\text{aux}}(p_1, p_2, p_3) &= 0 \end{aligned}$$

The same parameters as in [82] were used in deriving the results here. These parameter values are given by

$$\begin{aligned} p_1 &= 1.0 \\ p_2 &= 0.11\sin(\pi(j-1)/5) + 0.224 \\ p_3 &= 0.08\cos(\pi(j-1)/5) + 1.85 \end{aligned}$$

$$\text{for } j = 1, \cdots, 11$$

$$\begin{aligned} p_1 &= 1.25 \\ p_2 &= 0.11\sin(\pi(j-12)/5) + 0.224 \\ p_3 &= 0.08\cos(\pi(j-12)/5) + 1.85 \\ &\text{for } j = 12, \cdots, 22 \end{aligned}$$

Here, $X^{ij}(x)$ was evaluated on a square area given by $x_n = 1.5i/20$, $i = 1, \cdots, 20$ and $x_{n-1} = 1.5j/20$, $j = 1, \cdots, 20$; thus $M = 400$. The significant eigenvalues of the covariance matrix of $\{\mathbf{V}_{400}^{ij}\}$ are shown in Fig. 5.6(C). There are three significant eigenvalues corresponding to the three bifurcation parameters (p_1, p_2, p_3). Moreover, the normalized eigenvectors capture the form of the different terms in the original map as can be seen in Fig. 5.10. Figure 5.10(A) shows the first eigenvector which is planar corresponding to the term (x_{n-1}), (B) shows the second eigenvector representing the parabolic term $(1 - x_n)x_n$, and (C) shows the third eigenvector for the term $(1 - x_{n-1})x_n$. The surface of $\Phi_{\text{aux}}(x)$ shown in Fig. 5.10(D) is flat and close to zero. The reconstructed BD is not shown here.

The Lorenz equations

As a final example, a continuous system given by the Lorenz equation is considered. The differential equation has the form

$$\begin{aligned} \frac{dx}{dt} &= \sigma(y - x), \\ \frac{dy}{dt} &= rx - y - xz, \\ \frac{dz}{dt} &= xy - bz, \end{aligned} \tag{5.34}$$

where σ, b and r are the parameters. Qualitatively different dynamics can be observed from this system by changing the parameters across critical values. A detailed description of the bifurcation structure of the Lorenz equations with respect to the parameter r will be presented in section 6.4. What is of interest in this example is the pitchfork bifurcation at $r = 1$ which gives rise to two stable equilibrium points and makes the origin unstable. The bifurcation diagram with respect to r is shown in Fig. 6.8.

The motivation of including this example is to examine the general applicability of the proposed approach. In this example, 40 time series were generated for the parameter values

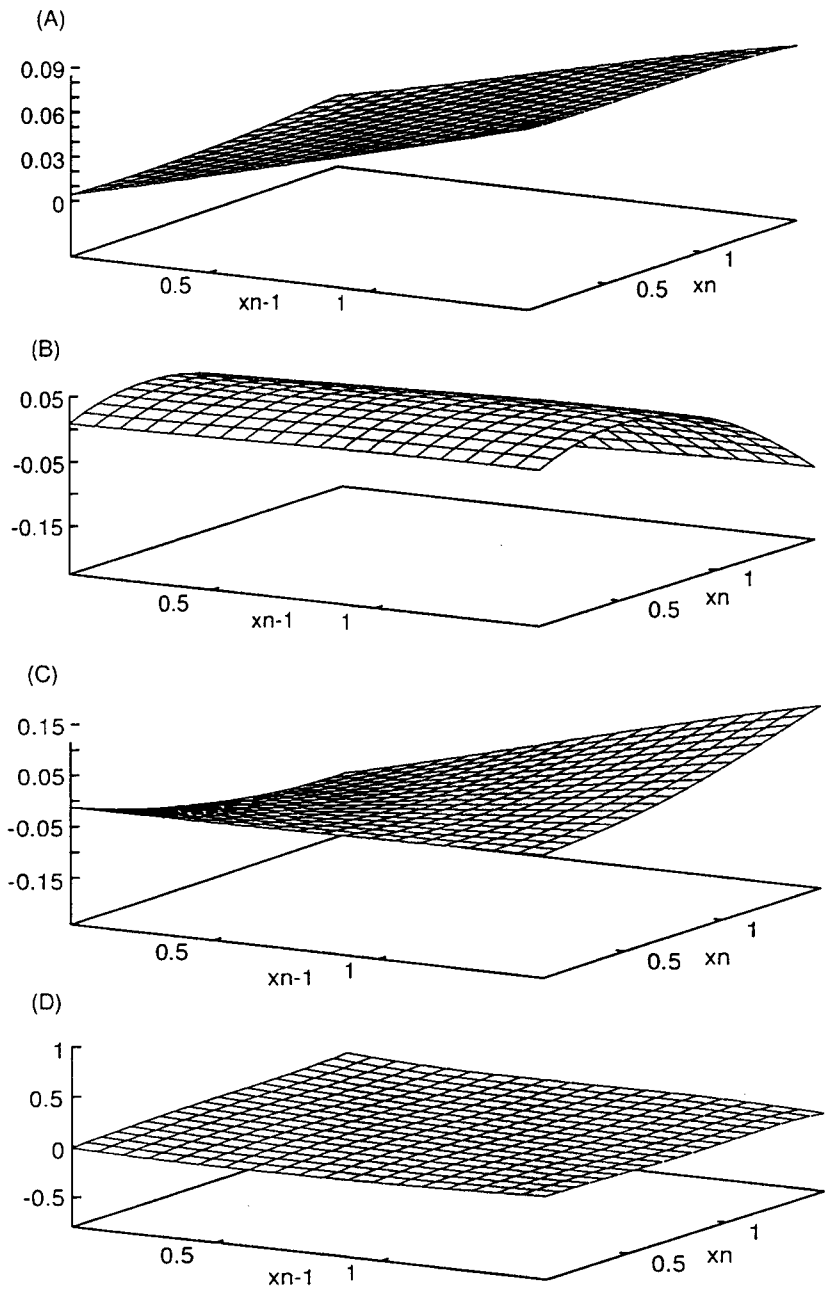

Figure 5.10: The normalized eigenvectors of the delayed logistic/coupled logistic map. (A) The first principal eigenvector. This eigenvector represents the term x_{n-1} in Eq. (5.33). (B) The eigenvector associated with the second significant eigenvalue. It is parabolic in x_n and thus represents the term $(1-x_n)x_n$. (C) The third eigenvector has a functional form similar to that of the third term $(1-x_{n-1})x_n$. (D) Φ_{aux} is approximately zero which is consistent with Eq. (5.33).

$r_i = 0.5i$, $i = 1, \cdots, 40$, $\sigma = 10$, and $b = 8/3$. The 4th-order Runge-Kutta method was used in integrating Eqs. (5.34) with a time step set to 0.001. The time delay Δ computed using mutual information is 0.1[1] and the embedding dimension d is equal to 3. The network parameters were as follows: $s = 5$, $y_b = 0$, and $q(x) = \tanh(0.5x)$.

The first task is to check if the obtained predictor functions belong to a family of LIP maps. To do this, the predictor functions $\{\tilde{g}_i\}$ were divided into six subsets: the first subset was composed of the first five predictor functions $\{\tilde{g}_1, \cdots, \tilde{g}_5\}$; the second was composed of the first ten $\{\tilde{g}_1, \cdots, \tilde{g}_{10}\}$; the third had 20 predictor functions $\{\tilde{g}_1, \cdots, \tilde{g}_{20}\}$; the fourth, 30 predictor functions $\{\tilde{g}_1, \cdots, \tilde{g}_{30}\}$; the fifth, the entire 40 predictor functions $\{\tilde{g}_1, \cdots, \tilde{g}_{40}\}$; and the last, 8 predictor functions $\{\tilde{g}_i,\ i = 5, 10, 15, 20, 25, 30, 35, 40\}$. Note that the range of r increases from the first group to the fifth group, that is, r ranges from 0.5 to 2.5 in the first group and 0.5 to 20 in the fifth group.

For each set, the vectors $\mathbf{V}^{ij}_{M=125}$s were generated and the significant eigenvalues of the covariance matrix formed by these vectors were computed. Figure 5.11 summarizes the result. From the figure, as the range of r increases from (A) to (E), the number of significant eigenvalues also increases. For example, for $0.5 \leq r \leq 2.5$ (the first group), there is only one significant eigenvalue [Fig. 5.11(A)]. While for $0.5 \leq r \leq 5$ (the second group), the number of significant eigenvalues becomes two [Fig. 5.11(B)] and increases further for wider range of r [Fig. 5.11(C-E)]. This is different from the results obtained with the Hénon map. This clearly indicates that the time series were generated by a non-LIP map.

This can be seen by considering the discretization of Eqs. (5.34):

$$\begin{aligned} x_{n+1} &= x_n + [\sigma(y_n - x_n)]\Delta t \\ y_{n+1} &= y_n + (rx_n - y_n - x_n z_n)\Delta t \\ z_{n+1} &= z_n + (x_n y_n - b z_n)\Delta t \end{aligned}$$

or $w_{n+1} = f(w_n; \Lambda)$ where $w_n = (x_n, y_n, z_n) \in \mathbb{R}^3$ represents the discretized state of the system, $f : \mathbb{R}^3 \times \mathbb{R}^3 \mapsto \mathbb{R}^3$ is a map that brings w_n to w_{n+1}, $\Lambda = (r, \sigma, b) \in \mathbb{R}^3$ are the parameters, and Δt is the discretization time. The above equation is linear-in-parameter for the given discretization time Δt. However, the observation time (= 0.1) is not equal to the discretization time (= 0.001) such that the effective predictor function $g(\cdot)$ is a composition of $f(\cdot)$, that is, $g(w_n; \Lambda) = f(\cdots(f(w_n; \Lambda)))$. This causes $g(\cdot)$ to have higher order terms in Λ making it a GLIP rather than a LIP. This suggests that the number of significant eigenvalues corresponds to the number of significant $f_i(\Lambda)$'s rather than Λ.

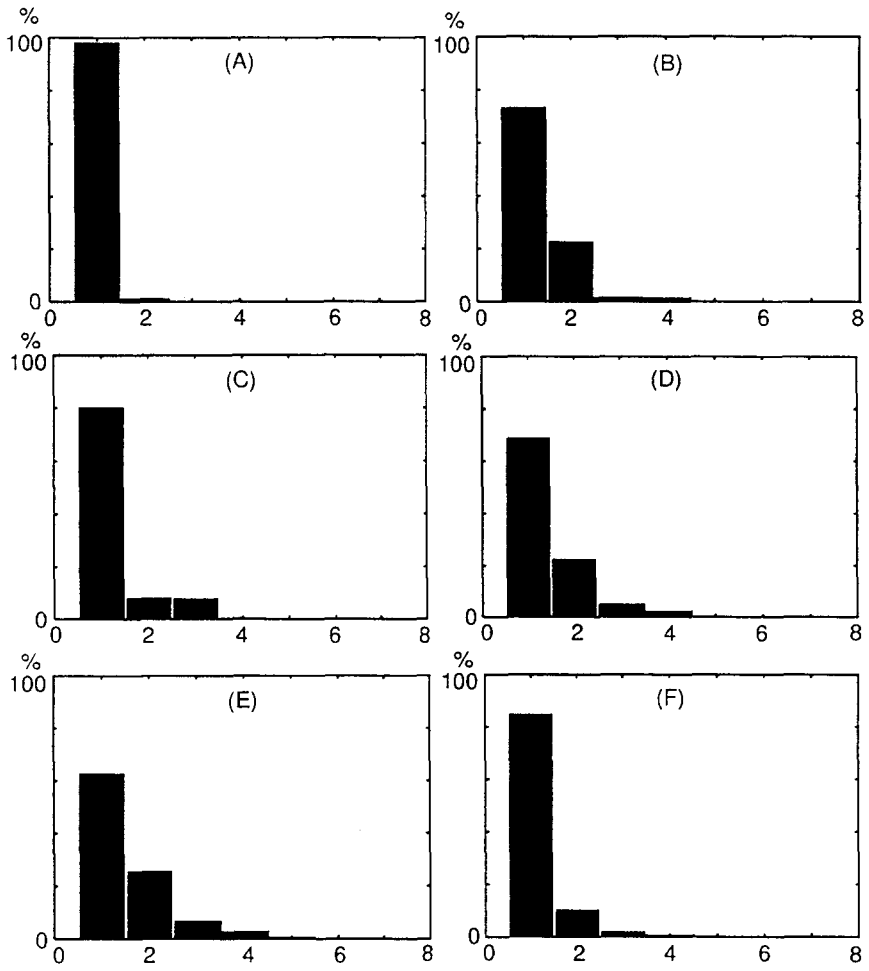

Figure 5.11: The significant eigenvalues (%). (A) The first group composed of predictor functions $\{\tilde{g}_1, \cdots, \tilde{g}_5\}$. (B) The second group with $\{\tilde{g}_1, \cdots, \tilde{g}_{10}\}$. (C) The third group with $\{\tilde{g}_1, \cdots, \tilde{g}_{20}\}$. (D) The fourth group with $\{\tilde{g}_1, \cdots, \tilde{g}_{30}\}$. (E) The fifth group with $\{\tilde{g}_1, \cdots, \tilde{g}_{40}\}$. (F) The last group with $\{\tilde{g}_i,\ i = 5, 10, 15, 20, 25, 30, 35, 40\}$.

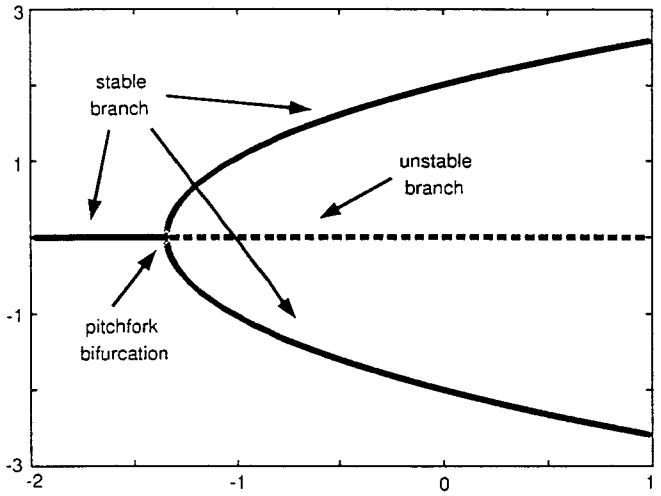

Figure 5.12: Reconstructed bifurcation diagram of the Lorenz equation around $r = 1$ where a pitchfork bifurcation occurs. The new parameter β ranges from $[-2, 1]$.

However, a closer investigation of the figure reveals that for $0 \leq r \leq 2.5$, the obtained map resembles an LIP map. This is because the number of significant eigenvalue is the same as the number of parameter - that is, one. To examine this further, the BD of the map was reconstructed within this parameter region using the proposed approach. Equation (5.16) was used in the reconstruction. In particular, the fixed points of the equation

$$\tilde{g}(x;\beta) = \tilde{g}_5(x) + \beta[\tilde{g}_5(x) - \tilde{g}_3(x)] \tag{5.35}$$

for $\beta \in [-2, 1]$ and $x = (x_n, x_{n-1}, x_{n-2})$ was computed. The reconstructed BD is shown in Fig. 5.12. The pitchfork bifurcation of the Lorenz equation at $r = 1$ is reconstructed. In spite of the fact that the system is not LIP, the result suggests that the method is still applicable as long as reconstruction is confined to small parameter regions. This is evident when one considers the Taylor expansion of a general map $g(x;r)$ around some bifurcation point r_o, that is,

$$g(x; r_o + \Delta r) \approx g(x; r_o) + \Delta r \frac{\partial g(x; r_o)}{\partial r} + \frac{(\Delta r)^2}{2} \frac{\partial^2 g(x; r_o)}{\partial r^2} + \cdots \tag{5.36}$$

which resembles a GLIP map with parameter Δr.

To summarize, this subsection shows how to detect time series generated by non-LIP maps. Furthermore, the general applicability of the proposed algorithm is exhibited for the case of the Lorenz system. As long as the parameters considered are confined to a small region, then the above approximation still holds for non-LIP maps.

5.6 Discussion

For dynamical systems that can be written in the form given by Eq. (5.13), it is shown that a correspondence exists between the number of parameters and the number of significant eigenvalues of the covariance matrix formed by either the weight vectors $\mathbf{w}^i_{\text{min}}$ or the vectors $\mathbf{V}^{ij}_M$. For the former, this means that in the weight space of the approximating network, a linear manifold can be found such that on this manifold the behavior of the network is qualitatively similar to that of the given LIP map. With this, the reconstructed BD of the original map is given by the BD of the approximating network on this linear manifold. For the latter, this correspondence can be used to recognize if the given time series are generated from an LIP map.

The definition of the difference function $X^{ij}(x)$ and the associated vector $\mathbf{V}^{ij}_M$ leads to a scheme that can be employed to determine the number of bifurcation parameters from the predictor functions. It is also shown numerically that the number of parameters as well as the functional form of the unknown map can be determined by this scheme and thus, the unknown map can be completely specified. In line with these ideas, a new reconstruction algorithm is presented. This algorithm can be summarized as follows: The first step is to obtain predictor functions using the given time series. Next, the number of bifurcation parameters are determined using the significant eigenvalues of the covariance matrix formed by the vectors $\mathbf{V}^{ij}_M$. Once the number of parameters k are known, the map is approximated using $k+1$ predictor functions and Eq. (5.15) or (5.16). Finally, the reconstruction follows by varying the parameters β_i in either of the two equations.

The new approach is different from the one presented in [7, 81, 82] in the following aspects: First, the reconstruction is based not on the parameter space of the approximating predictor functions but rather on the predictor functions themselves, as given in Eqs. (5.15) and (5.16). Second, the method requires only few predictor functions (approximately the same as the number of bifurcation parameters) to reconstruct the BD, whereas the previous approach requires more $\mathbf{w}^i_{\text{min}}$s to obtain a good projection region. Thus, the number of time series required to reconstruct the BD of the given system is reduced. However, to determine accurately the number of bifurcation parameters, more time series may be needed.

The result in the reconstruction of the pitchfork bifurcation of the Lorenz equation exemplifies the general applicability of the new approach. As long as the parameter values are not too wide apart, the LIP-approximation still holds. This is due to the fact that even though the original map is not linear-in-parameter, it can be written in a form similar to Eq. (5.13) by

using Taylor expansion.

The results in [81] showed that the BD of a given system can be reconstructed even if the time series were corrupted with observation noise. This implies that predictor functions which approximate the given system for fixed parameter values can still be obtained. Since the current approach is based on the predictor functions themselves, it follows that the proposed algorithm should still work in the presence of observation noise. For the case of dynamical noise, finding the appropriate predictor functions for the noisy time series becomes an important issue because such perturbations can cause drastic changes in the dynamics of the system. The use of neural networks in this case may be difficult since neural networks do not incorporate dynamical noise in its learning scheme. Other predictor functions such as nonlinear autoregressive models (section 3.4) may prove useful in this case. Once the appropriate (robust to noise) predictor functions are obtained, the reconstruction follows immediately from Eqs. (5.15) and (5.16). This problem will be addressed in more details in the following chapters.

Chapter 6

Nonlinear Methods

6.1 Introduction

In the previous chapters, the reconstruction is based on the approximation that in the parameter space of the predictor function there exists a linear manifold on which the bifurcation structure is similar to that of the original. Under this approximation, the number of bifurcation parameters corresponds to the dimension of this linear manifold. Basis vectors of the manifold can then be obtained from a known set of discrete points using principal component analysis.

In this chapter[4], the feasibility of reconstruction is explored when the linear manifold approximation of the previous method is insufficient, thus entailing a more general approach. Moreover, under this condition, the correspondence between the number of bifurcation parameters and the number of basis vectors obtained by PCA is lost. As a consequence, the number of bifurcation parameters is undetermined. It is therefore natural to start with a one-parameter BD reconstruction. In this regard, a generalized one-parameter BD reconstruction algorithm using time series is considered. Possible extension to higher dimensional reconstruction is also discussed in the later part of the chapter.

The algorithm is tested using time series from systems exhibiting stable limit cycles and stable equilibrium points. Surprisingly, these systems require a more general approach for the reconstruction of their BDs. The following section will cover the algorithm which is based on principal curves as approximated by piecewise linear curves. Numerical results on the FitzHugh-Nagumo equations and the Lorenz equations are presented to illustrate the effectiveness of the approach.

[4]The contents of this chapter also appear in Bagarinao, et al., (1998) [7].

6.2 Principal Curves

The problem of obtaining the projection region from the computed $\{\mathbf{w}^i_{\min}\}$ is related to the problem of finding lower dimensional manifolds (the projection region) in a high dimensional space (model's parameter space) which can be solved using several well-established approaches. In this section, the case for $k = 1$ (only one bifurcation parameter) is considered.

A natural generalization of Eq. (4.7) is to consider $\mathbf{h}(\cdot)$ as a nonlinear function of $\mathbf{c}$. With this, $\mathbf{h}(\cdot)$ defines a one-dimensional curve, say $\mathcal{C}$, in the parameter space of the predictor function. The problem is to obtain an approximation $\mathcal{C}_{\text{app}}$ to this curve using the computed $\mathbf{w}^i_{\min}$s. Letting λ represents the curvilinear abscissa along $\mathcal{C}_{\text{app}}$, then for each $\mathbf{w} \in \mathcal{C}_{\text{app}}$ the parameter $\mathbf{w}$ can be expressed as a function of λ, that is, $\mathbf{w} = \mathbf{h}(\lambda)$ where $\mathbf{h} : \mathbb{R}^1 \to \mathcal{C}_{\text{app}}(\subset \mathbb{R}^p)$ is nonlinear. The BD of the predictor function along $\mathcal{C}_{\text{app}}$ with λ as the effective bifurcation parameter is then taken as the reconstruction of the BD of the given system.

The curvilinear curve $\mathcal{C}_{\text{app}}$ is approximated using the *principal curve* of the data points $\{\mathbf{w}^i_{\min}\}$. Principal curves as defined in [40] are smooth curves that are self-consistent for a distribution or data set. This means that the average of all the data that project onto a point on the curve coincides with the same point on the curve. For data sets, the curve $\mathcal{C}_{\text{app}}$ is represented by K discrete points $\{\mathbf{h}_i | \mathbf{h}_i = \mathbf{h}(\lambda_i)\}$ joined by a line from $\mathbf{h}_i$ to $\mathbf{h}_{i+1}$ in increasing order of λ_i. Clearly, the shape of $\mathcal{C}_{\text{app}}$ depends only on the order, not on the actual values of the λ_is. Assuming that the $\mathbf{h}_i$'s are sorted in increasing order of λ and using the arc-length parameterization such as in [40], one can assign $\lambda_1 = 0$ and λ_i equal to the arc length along the curve from $\mathbf{h}_1$ to $\mathbf{h}_i$. In the simulation, the principal curve is approximated by piecewise linear curves

$$\mathbf{w}(\lambda) = \mathbf{w}^{i-1}_{\min} + \frac{\mathbf{w}^i_{\min} - \mathbf{w}^{i-1}_{\min}}{\lambda_i - \lambda_{i-1}}(\lambda - \lambda_{i-1}) \tag{6.1}$$

where $i = 2, \cdots, K$.

The algorithm to compute the principal curve as described in [40] is composed of two basic steps; namely, the projection step and the conditional-expectation step. As an initialization, the principal curve at the zeroth iteration, denoted as $\mathbf{h}^0$, is given by the first linear principal component of the given data set $\{\mathbf{x}^i\}$. Each of the $\{\mathbf{x}^i\}$ is then projected onto $\mathbf{h}^0$, the projection being the point on $\mathbf{h}^0$ closest to $\{\mathbf{x}^i\}$. Denote this as $\mathbf{h}^0_i$. This gives an ordering of the $\{\mathbf{x}^i\}$ along $\mathbf{h}^0$. The curvilinear abscissa λ for each $\mathbf{h}^0_i$ can now be defined as follows. The endmost $\mathbf{h}^0_i$, denoted as $\mathbf{h}^*$, is set to have a value of $\lambda_i = 0$ and all other values of λ_i is defined as the distance of $\mathbf{h}^0_i$ from $\mathbf{h}^*$ along $\mathbf{h}^0$. Thus, for each $\{\mathbf{x}^i\}$ is an associated λ_i and the curve is represented by

N number of $(\lambda_i, \mathbf{h}_i^0)$ joined up linearly in increasing order of λ to form a polygon. This serves as the projection step.

To get the next approximation of the principal curve, that is, $\mathbf{h}_i^1$, all the points $\mathbf{x}^k$ in the sample, whose associated λ_k is close to λ_i, are averaged. This average value is then taken as the estimate of $\mathbf{h}_i^1$. This is the conditional-expectation step. For local averaging methods, see for example spline smoothers[76] and locally weighted running-line smoother[24], among other methods. Once all the $\mathbf{h}_i^1$s are obtained, new estimates of the λ_is associated to these new $\mathbf{h}_i^1$s are computed. To get the new estimate, define d_{ik} as the distance between $\mathbf{x}^i$ and its closest point on the line segment joining each pair $(\mathbf{h}_k^1, \mathbf{h}_{k+1}^1)$. Then set λ_i to correspond to the smallest value of d_{ik}, that is, $\lambda_i = \min d_{ik}$. Corresponding to each λ_i is an interpolated $\mathbf{h}_i$. Using these values to represent the curve, replace λ_i by the arc length from $\mathbf{h}^*$ to $\mathbf{h}_i$. Finally, the procedure is iterated until the relative change in $\mathbf{h}_i^j$ is below some threshold. Refer to [40] for a more complete description of the algorithm. The complete algorithm is summarized below.

Initialization: Set $\mathbf{h}^{(0)}(\lambda) = \bar{\mathbf{x}} + \mathbf{a}\lambda$, where $\mathbf{a}$ is the first linear principal component of the given data set $\{\mathbf{x}\}$. Set $\lambda^{(0)}(\mathbf{x}) = \lambda_{\mathbf{h}^{(0)}}(\mathbf{x})$.

Repeat: Over the counter j

1. Conditional-expectation step.
2. Projection step.
3. Evaluate the change in $\mathbf{h}^j$.

Until: The change in $\mathbf{h}^j$ is below some threshold.

6.3 Example 1: The FitzHugh-Nagumo Equations

In this section, the reconstruction algorithm using principal curve is applied to time series of the FitzHugh-Nagumo (FHN) equation [32, 59], which is given by

$$\begin{aligned} \epsilon \frac{dv}{dt} &= -v(v-0.5)(v-1.0) - w + I_{\text{ext}} \\ \frac{dw}{dt} &= v - w - b. \end{aligned} \tag{6.2}$$

This is a simple mathematical model which mimics to some extent the excitability of a neuron. For some range of I_{ext}, the model behaves as an excitable membrane. Whereas for some other parameter values, it behaves as an oscillating membrane defining a stable limit cycle. This gives

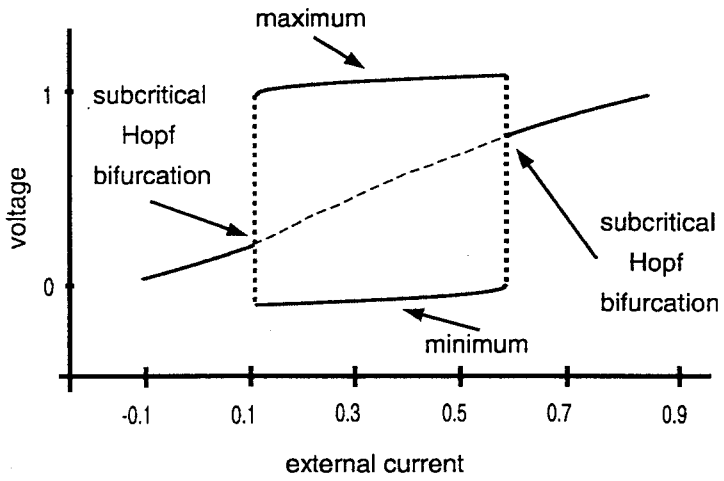

Figure 6.1: Bifurcation diagram of the FHN-system. It is characterized by two stable fixed point regions, a stable limit cycle region, and two subcritical Hopf bifurcations.

a good example of a simple system which can either have a stable fixed point or a stable limit cycle depending on the value of the parameter[83].

The bifurcation diagram of Eq. (6.2) is shown in Fig. 6.1. In this figure, the bifurcation parameter is the applied external current I_{ext} and the other parameters are fixed ($\epsilon = 0.005$, $b = 0.15$). As I_{ext} is increased, the stable equilibrium point becomes unstable and a subcritical Hopf bifurcation occurs at $I_{\text{ext}} \simeq 0.112$ giving rise to the stable limit cycle. As the current is increased further, another subcritical Hopf bifurcation takes place at $I_{\text{ext}} \simeq 0.588$ and the stable limit cycle disappears.

The FHN-system is used to illustrate the applicability of the proposed algorithm since it has a well-defined sequence of bifurcation points and a bifurcation structure that can be analyzed in detail. Moreover, it also presents some fundamental issues and difficulties to the reconstruction problem which are not obvious if the system is chaotic. Take for example the case with the stable equilibrium points of the FHN equations. For this case, the corresponding time series is a constant in time and therefore, does not contain any information of the system dynamics at all except during transient. It is therefore inferable that finding a suitable $\tilde{g}(\cdot)$ would be easier in this case than the case of chaotic time series. Unexpectedly, this ease in obtaining the predictor function does not translate to easier reconstruction of the bifurcation diagram.

The first task is examined if the simple BD of FHN can be reconstructed by using the PCA-based algorithm, that is, by a linear approximation of $\mathbf{h}(\cdot)$ [see Eq. (4.7)]. To do this, 65 time series of the variable v were generated by integrating Eq. (6.2) using fourth-order Runge-Kutta

I_{ext}	Pred. Error
0.00	0.00
0.10	8.23×10^{-12}
0.20	9.49×10^{-05}
0.30	2.10×10^{-06}
0.40	9.90×10^{-07}
0.50	1.69×10^{-06}
0.60	1.14×10^{-08}
0.70	2.30×10^{-13}

Table 6.1: One step prediction error of the obtained predictors for some values of I_{ext}.

method with the time step set to 0.001 and initial value ($v = 0.0, w = 0.0$). The corresponding parameter values were given by $I^i_{\text{ext}} = 0.011(i-1)$, $i = 1, \cdots, 65$. To account for all the transient behavior which varies from one parameter to another and for uniformity, the number of data points per time series is set to $N = 1000$ with sampling interval $\Delta = 0.2$. The false nearest neighbor algorithm (FNN) is used to determine the embedding dimension d, and this value was found to be equal to 2. The computation of the $\{\mathbf{w}^i_{\text{min}}\}$ started with the time series having the most complex behavior. The initial weights were set randomly between $[-1.0, 1.0]$, and the error was minimized using the BFGS algorithm. For the remaining time series, the computed $\mathbf{w}_{\text{min}}$ of the preceding time series was used as the initial weight value. The activation function employed for the hidden units was the usual sigmoidal function given by $q(x) = 1/[1+\exp(-x)]$. The output of the bias neuron was set to 1 ($y_b = 1.0$). Different numbers of time series (e.g. $K = 5, 10, 19, 37$, and 65) were used in reconstructing the BD of FHN. Here, the case for $N = 65$ and $s = 13$ (the number of hidden units) represents the best possible reconstruction using the method based on PCA. Table 6.3 shows the value of the error function, Eq. (4.4), for some representative values of I_{ext}. As can be observed, the prediction error for each time series is really small. This confirms the observation that finding a suitable predictor function for each of the time series of the FHN is relatively trivial.

The PCA results suggest that the obtained $\{\mathbf{w}^i_{\text{min}}\}$ is nonlinear in I_{ext}. This is evident in Fig. 6.2 which shows the significant eigenvalues of the sample covariance matrix of $\{\mathbf{w}^i_{\text{min}}\}$. From the figure, the number of significant eigenvalues is roughly equal to 8 representing 95.75%

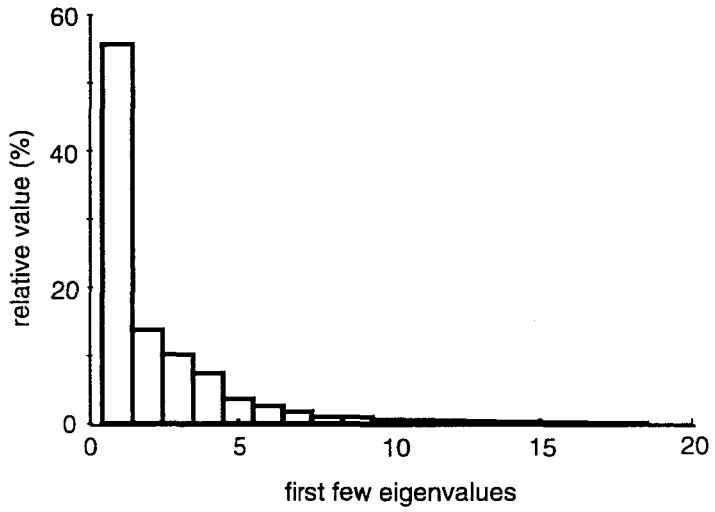

Figure 6.2: Significant eigenvalues of the sample covariance matrix for the 65 weight vectors obtained after training the network.

of the total eigenvalues. Obviously, this is much greater than the expected one corresponding to I_{ext}. Note that for other values of K, the same observation holds; that is, the number of significant eigenvalues is always greater than one. The nonlinearity can be further observed by taking the projection [i.e. the α_is in Eq. (4.6)] of each $\mathbf{w}^i_{\text{min}}$ onto the associated eigenvectors of the sample covariance matrix and plotting the computed values against the original parameter. The resulting curve should be roughly linear when the $\mathbf{w}^i_{\text{min}}$s lie on a linear manifold. A linear curve thus indicates the validity of the linear manifold approximation. Figure 6.3 shows the computed $\alpha_{1,2,3}$s of the obtained $\mathbf{w}^i_{\text{min}}$s for the first three eigenvectors. From the figure, the resulting curve is not linear. This indicates that $\mathbf{h}(I_{\text{ext}})$ is nonlinear in I_{ext}.

Figure 6.4 shows the two dimensional BD of $\tilde{g}(\cdot\,;\mathbf{w})$ in the vicinity of the projection region. The x- and y-axis correspond to the axes spanned by the first and second eigenvectors [ϕ_1 and ϕ_2 in Eq. (4.6)], respectively. The scattered points indicate the projection of the $\mathbf{w}^i_{\text{min}}$s onto this principal plane. PS and SFP correspond to regions of (pseudo-) periodic solutions and stable fixed point solutions, respectively. The arrows indicate the $\mathbf{w}^i_{\text{min}}$s associated with the constant time series (stable equilibrium of the system). The projection of these points into the PS region in the figure is an artifact since only two out of the eight significant eigenvectors are used. Observe the parabolic-like distribution of the scattered points in the vicinity of the projection region, another manifestation of the nonlinearity of $\mathbf{h}(I_{ext})$.

Finally for the PCA-based algorithm, a one-dimensional BD was constructed using the first

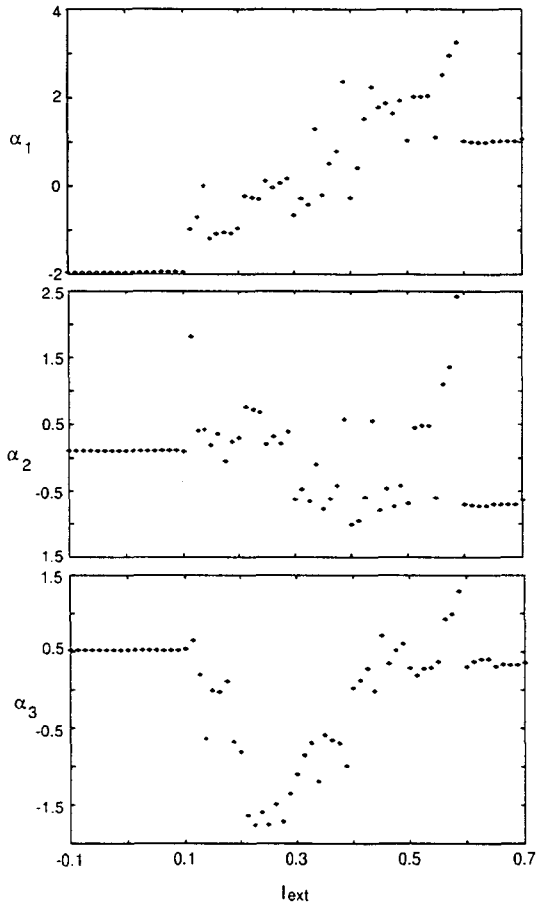

Figure 6.3: Projections of the set $\{\mathbf{w}^i_{\min}\}$ onto the eigenvectors of the first three significant eigenvalues of the sample covariance matrix.

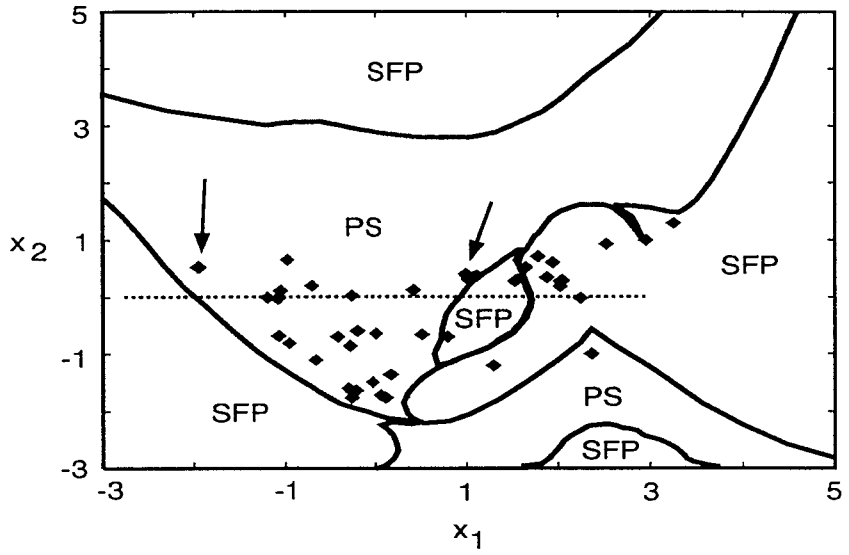

Figure 6.4: The local bifurcation structure of $\tilde{g}(\cdot\,;\mathbf{w})$ relative to the first two eigenvectors of the sample covariance matrix. The x_1 - axis is spanned by the first eigenvector while the x_2 - axis is spanned by the second eigenvector. PS denotes region of (pseudo -) periodic solutions and SFP denotes region of stable fixed point solutions. The scattered points are the projections of the time series onto this principal plane.

principal axis (shown as dashed line in Fig. 6.4). The choice of the first principal axis is dictated by the fact that the projection of $\{\mathbf{w}^i_{\min}\}$ onto this axis has maximum variance. The fixed points of $\tilde{g}(\cdot\,;\mathbf{w})$ were computed for a given $\mathbf{w}$ by finding the zeros of

$$z(y) = y - \tilde{g}(y, \cdots, y; \mathbf{w}). \tag{6.3}$$

To evaluate the local stability of the fixed points, the characteristic equation associated with $\tilde{g}(\cdot\,;\mathbf{w})$ is defined as a polynomial of degree d (embedding dimension) with coefficients equal to the partial derivatives of $\tilde{g}(\cdot\,;\mathbf{w})$ evaluated at the fixed point, that is,

$$P(\lambda) = \sum_{i=0}^{d} a_i \lambda^{d-i} \tag{6.4}$$

where $a_0 = -1$ and $a_i = \partial\tilde{g}(y, \cdots, y\,;\mathbf{w})/\partial y_{t-i\Delta}$ evaluated at the fixed points (see section 2.2).

Figure 6.5 shows the reconstructed BD. It can be seen that the reconstructed diagram has qualitatively similar bifurcations as compared to the one depicted in Fig. 6.1 for $x_1 \in [-4.0, 3.0]$. In particular, the fixed point region, followed by the limit cycle region, then by another fixed point region, is preserved in the reconstruction. Furthermore, the two Hopf bifurcations bounding the stable limit cycles of the original system are also present. However, the reconstructed

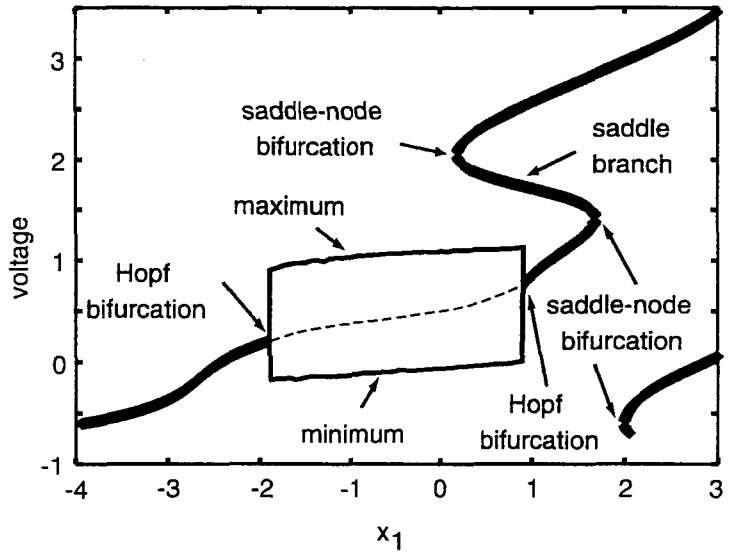

Figure 6.5: Reconstructed bifurcation diagram relative to the first principal axis. The reconstruction preserves the two fixed point regions, the limit cycle region and the two Hopf bifurcations. However, it also exhibits extra fixed points and bifurcations which are not present in the original system. Note that x_1 does not directly correspond to I_{ext}.

BD also exhibits extra fixed points such as the saddle and upper stable node branches and extra bifurcations such as the three saddle-node bifurcations.

After establishing the need for an alternative approach, the algorithm presented in the previous section was employed to reconstruct the BD of the FHN-system. Twenty time series were generated using the same conditions mentioned previously for parameter values $I^i_{\text{ext}} = -0.1 + 0.05(i-1)$, $i = 1, \cdots, 20$. In this case, 20 time series were used, instead of the 60 employed, in order to have a rough comparison of the data requirements between the two algorithms. To compute $\{\mathbf{w}^i_{\text{min}}\}$, the same procedure mentioned earlier was applied. Different number of hidden units ($s = 7, 13, 19, 25, 31$) were used in training the network.

The results indicated that the fidelity of the reconstructed BD depends on the number of hidden units, that is, the more hidden units, the better the reconstruction. Figure 6.6 shows the reconstruction for $s = 7$ while Fig. 6.7 shows the reconstruction for $s = 31$. The difference is obvious in the two reconstructed BDs. This result is consistent with the fact that the more hidden units the network has, the better its approximation capability. The results also suggest that the projection region should be a *connected* region in the parameter space to avoid unwanted behavior such as the intervening stable equilibrium region within the limit cycle region of the

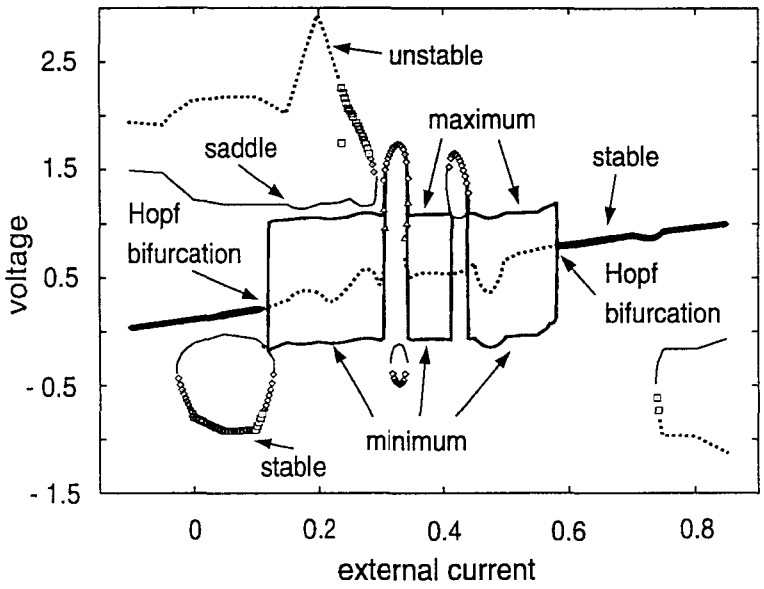

Figure 6.6: Piecewise linear reconstruction with $s = 7$.

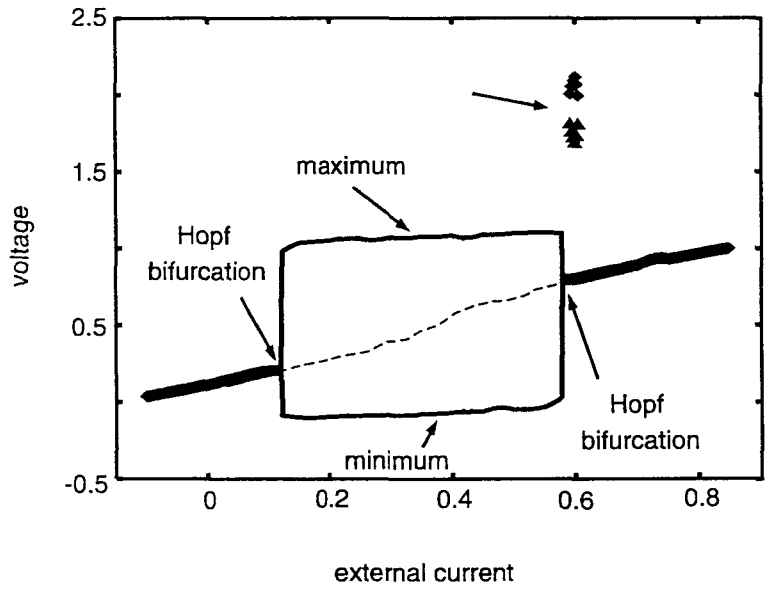

Figure 6.7: Piecewise linear reconstruction with $s = 31$. All the features of the original FHN bifurcation diagram are preserved. The arrow indicates some extraneous fixed points of the obtained map.

reconstructed BD as shown in Fig. 6.6.

Figure 6.7 shows the reconstructed BD for $s = 31$ and $I_{\text{ext}} \in [-0.1, 0.85]$. The fixed points and their associated stability were obtained using the methods presented in section 2.2. This time, a better reconstruction of the BD of FHN is obtained. From the figure, the locations of the fixed points correspond well with the locations of the fixed points of the FHN-system. The two Hopf bifurcations of the FHN-system also appear in the reconstructed BD occurring at $I_{ext} \simeq 0.118$ (0.112 for FHN) and $I_{ext} = 0.581$ (0.588 for FHN), respectively. The stable limit cycle region is preserved. The number of fixed points for a given parameter value is the same except in the narrow interval indicated by an arrow in the figure.

6.4 Example 2: The Lorenz Equations

The other system used in the simulation is the Lorenz system. This is one of the most well-known systems in the study of chaos and is used to model the unpredictable behavior usually associated with weather phenomenon. The point of interest here is not particularly the chaotic aspect of the system but rather the transition of its behavior from the fixed point region to the chaotic region as the parameters vary. The equations are given below:

$$\begin{aligned} \frac{dx}{dt} &= \sigma(y - x) \\ \frac{dy}{dt} &= rx - y - xz \\ \frac{dz}{dt} &= xy - bz. \end{aligned} \tag{6.5}$$

where σ, b and r are the parameters.

The origin ($x = y = z = 0$) is an equilibrium point for any parameter values. For $b = 8/3$, $\sigma = 10$, and $0 \leq r \leq 1$, the origin is stable and globally attracting. At $r = 1$, a pitchfork bifurcation occurs and the origin becomes unstable. This also gives rise to two more stable equilibrium points given by $C_{\pm} = (\pm\sqrt{b(r-1)}, \pm\sqrt{b(r-1)}, r-1)$ which remain stable until $r \simeq 24.74$ where a subcritical Hopf bifurcation occurs. Beyond this value, all equilibrium points are unstable. Figure 6.8 summarizes the behavior of the Lorenz equations as a function of the parameter r. The presence of a homoclinic orbit at $r \simeq 13.93$ causes some of the trajectories originating from the positive x-axis to converge to C_- and vice versa as shown in the lower middle figure. Furthermore, as r approaches 24.74, some trajectories wander from one stable fixed point to another but still converge to either C_+ or C_- after a long time. For a detailed analysis of the bifurcation of the Lorenz equations, see for example [77].

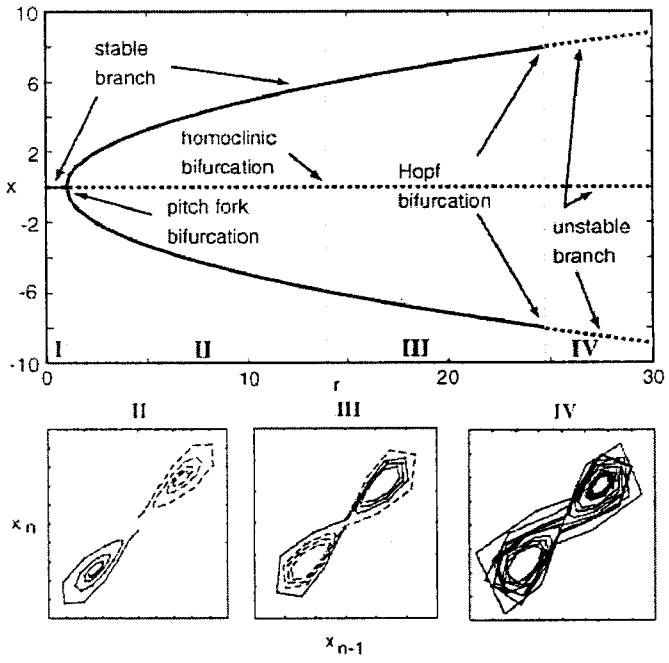

Figure 6.8: Bifurcation diagram of the Lorenz equations with r as the bifurcation parameter. In region I, the origin is stable and globally attracting. In region II, all trajectories move towards either C_+ or C_-. In region III, some of the trajectories in the positive x-axis converges to C_- and vice versa. Region IV is usually associated with the chaotic behavior of the system. Three main bifurcations are present - the pitchfork at $r = 1$, the homoclinic at $r \simeq 13.93$ and the Hopf at $r \simeq 24.74$. The lower figures show some representative trajectories for each region in the embedded space.

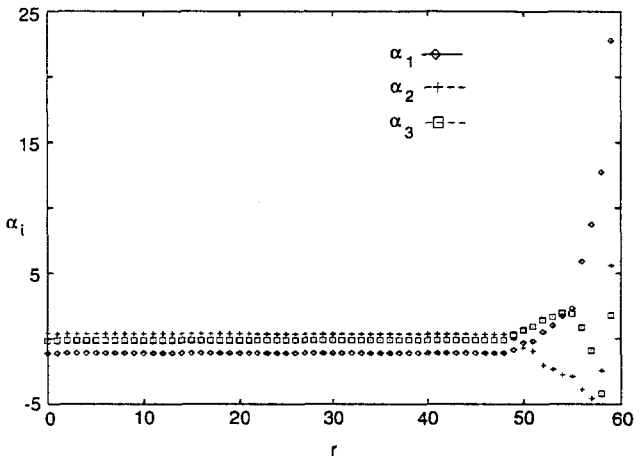

Figure 6.9: Projections of the set $\{\mathbf{w}^i_{\min}\}$ onto the eigenvectors of the first three significant eigenvalues of the sample covariance matrix.

In the simulation, both b and σ (values given above) were fixed. Time series were obtained by integrating Eqs. (6.5) using the 4th-order Runge-Kutta method with time step equal to 0.001 and parameter values $r_i = 0.5 + 0.5(i-1)$, $i = 1, \cdots, 60$. Sixty time series of the variable x were used to make r equally sampled from 0.5 to 30 at 0.5 interval and to have two time series to represent the case where the origin is globally attracting ($r = 0.5, 1.0$). These two time series are necessary for the reconstruction of the pitchfork bifurcation at $r = 1$. Again, each time series was composed of 1000 data points to account for long transient behavior. The time delay Δ, obtained by mutual information, is equal to 0.1 and the embedding dimension, calculated using the FNN, is equal to 3.

To approximate the predictor function, Eq. (4.2) with $s = 100$, $y_b = 0.0$ and $q(x) = [1 - \exp(-2x)]/[1 + \exp(-2x)]$ was employed. The use of hyperbolic tangent as the activation function of the hidden unit neurons was dictated by the symmetry of the original system as observed in the available time series. The $\{\mathbf{w}^i_{\min}\}$ were computed in the same manner used in the FHN case and minimization of Eq. (4.4) was performed using the BFGS algorithm. Again, the PCA results suggest that the $\mathbf{w}^i_{\min}$s are nonlinear in r. This is evident in Fig. 6.9 which shows the plot of the projection $\alpha_{1,2,3}$ of the obtained $\{\mathbf{w}^i_{\min}\}$ onto the first three significant eigenvectors versus the original parameter r. Since the resulting curve is nonlinear in r, then the $\{\mathbf{w}^i_{\min}\}$ cannot lie on a line. It follows that the linear manifold approximation, or equivalently, the use of the PCA-based algorithm would be insufficient. This agrees with the results in the previous chapter. Moreover, this implies that the BD of the Lorenz system for $r \in [0.5, 30]$ is

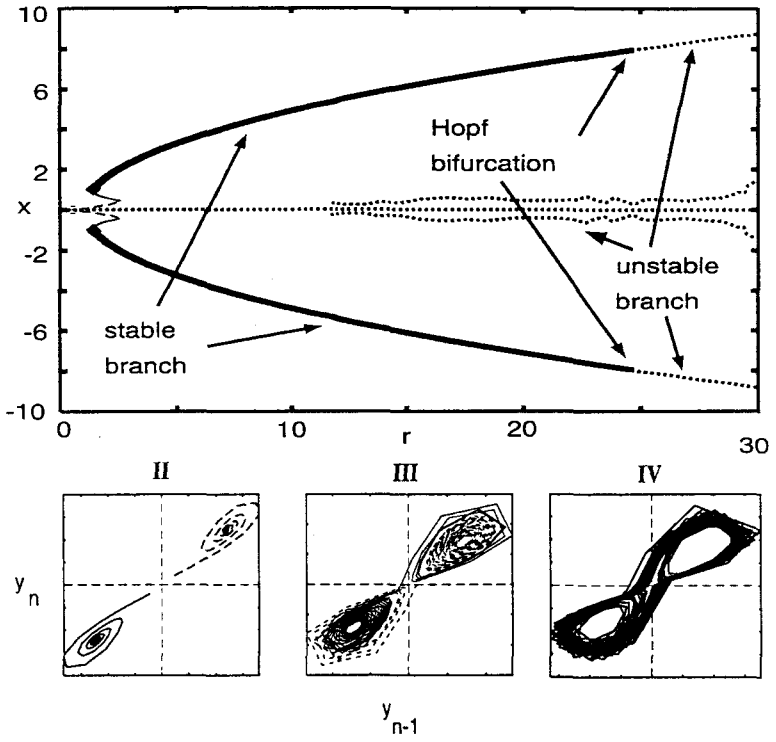

Figure 6.10: Piecewise linear reconstruction of the bifurcation diagram of the Lorenz equations. The fixed points and their corresponding stability are reconstructed accurately except near $r = 1$. The Hopf bifurcation is also preserved. However, it also exhibits extra saddle points near $x = 0$. The lower figures show some representative orbits of the obtained map.

best reconstructed by using the generalized algorithm.

Figure 6.10 shows the reconstructed BD using the proposed algorithm for $r \in [0.5, 30]$. The reconstruction does capture the important features of the original BD. From the figure, it is evident that the fixed points of the reconstructed system are very close to that of the Lorenz system. The associated stability of the fixed points are the same as the stability of that of the original system except in the region close to $r = 1$. Furthermore, the reconstructed system also undergoes Hopf bifurcation in both stable branches at $r \simeq 24.6$ ($r \simeq 24.74$ in the original diagram). Moreover, the symmetry of the location of the fixed points in the BD with respect to $x = 0$ is also preserved.

The reconstructed system also exhibits the same sequence of behavior as the original system with respect to the changes in the bifurcation parameters. To illustrate, representative trajectories of the reconstructed system at different parameter values are shown in the lower figures of Fig. 6.10. The leftmost figure shows representative trajectories in the range of r where orbits

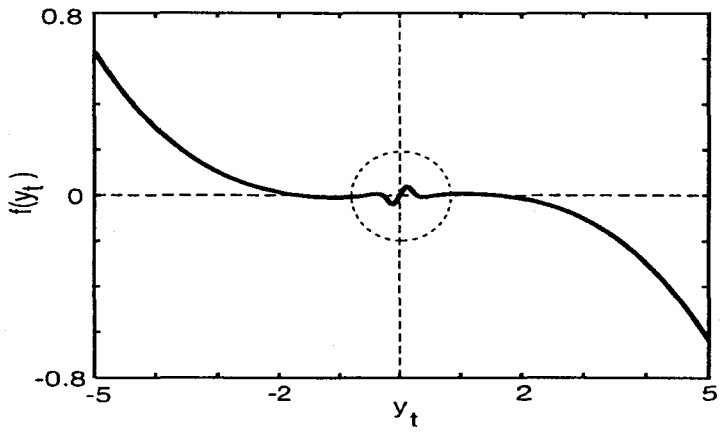

Figure 6.11: The profile of the function $f(y_t)$ for $r = 2$.

starting from the positive side converge to the positive stable fixed point and orbits starting from the negative side converge to the negative stable fixed point. This is region II in the original system. The middle figure shows representative trajectories in the region where orbits starting from the positive side converge to the negative fixed point and vice versa. This is region III in the original system. And lastly, the rightmost figure shows the Lorenz attractor where trajectories wander from one fixed point to the other. This is region IV in the original system.

Finally, one can see that the presence of extra unstable fixed points near $x = 0$ affects the reconstruction of the pitchfork bifurcation at $r = 1$ and the homoclinic orbit at $r \simeq 13.93$. To examine the source of these extra fixed points, the function $z(y)$, Eq. (6.3), was plotted for $y \in [-5.0, 5.0]$. This is shown in Fig. 6.11. From the figure, the curve is almost cubic in y far from $y = 0$; however, it is distorted near $y = 0$. This distortion is evidently the source of the extra fixed points close to $x = 0$ in the reconstructed BD.

6.5 Other Nonlinear Approaches

The use of principal curves in one-dimensional BD reconstruction offers the possibility of extending the algorithm to the two-parameter case. For a two-dimensional BD reconstruction, the algorithm will be based on the idea of principal surfaces [40]. Principal surfaces are defined in the same manner as principal curves in that they are self-consistent for a distribution or data set. This means that each point on a principal surface is the average of all the points that project there. Thus, a two-dimensional parameterized surface $\mathbf{f} : \Lambda \rightarrow \mathbb{R}^p$ for $\Lambda \subseteq \mathbb{R}^2$, where $\mathbf{f}$

is a vector of continuous functions:

$$\mathbf{f}(\mathbf{\Lambda}) = \begin{pmatrix} f_1(\lambda_1,\lambda_2) \\ f_2(\lambda_1,\lambda_2) \\ \vdots \\ f_p(\lambda_1,\lambda_2) \end{pmatrix} \tag{6.6}$$

is a principal surface if it is self-consistent.

The results also offer interesting insights into the different issues regarding bifurcation diagram reconstruction using time series that still need to be addressed in future research. The first of these is the proper choice of the predictor function, which is as stated, the most difficult task for modelers. For the neural network approach, this is related to the question of the appropriate number of hidden units, activation function, and network architecture, among others. From the results, what appears to be more important in the reconstruction is the localization of $\{\mathbf{w}^i_{\min}\}$ in some *connected* region in the parameter space of $\tilde{g}(\cdot\,;\mathbf{w})$.

6.6 Discussion

In this chapter, the need to extend the PCA-based approach (linear manifold approximation) to nonlinear cases especially for systems with stable fixed points and limit cycles has been demonstrated. As the results suggest, even for systems with simple bifurcation structure such as the FHN, the linear approximation may not hold under the given condition. This can manifest in the reconstructed BD as extra bifurcations which cannot be found in the original system. On the other hand, the proposed nonlinear algorithm based on principal curves shows very promising results. Using this approach, a high fidelity reconstruction of the original BDs was achieved. The reconstruction preserves the important features of the bifurcation diagrams of the original systems such as the Hopf bifurcations of the two systems, the limit cycle of the FitzHugh-Nagumo equations, and the symmetry of the fixed points of the Lorenz equations, among others. Furthermore, the reconstructed systems also exhibit the same sequence of behavior as the original systems with respect to the changes of the bifurcation parameters. This is evident in the reconstructed BDs of the FHN system and the Lorenz system. For the FHN system, the sequence is as follows: first, the fixed point region, then the Hopf bifurcation, followed by the limit cycle region, then another Hopf bifurcation, and finally, the other fixed point region. For the Lorenz system, the reconstructed BD contains region II, followed by region III and then by region IV.

The use of principal curves has a number of merits. Aside from being a good nonlinear approximator, principal curves reduce to the first linear principal component when the linear

approximation holds. Thus, the reconstruction is the same in principle as that of the PCA-based algorithm for one-parameter BD reconstruction. For higher dimensional cases, the principal curve-approach is generalized as principal surfaces, nonlinear principal component analysis, bottle-neck neural networks (NNs), among others [26, 40, 45, 53, 73].

In conclusion, the feasibility of reconstructing one-parameter BD using time series has been demonstrated. The reconstruction algorithm works even for the case where the linear manifold approximation of the projection region is insufficient. In the simulation, the reconstructed BD of the local bifurcation structures of the FitzHugh-Nagumo equations and the Lorenz equations preserved the critical bifurcations of the original systems. Moreover, the global behavior of the original systems are also preserved in the reconstructed diagrams.

Chapter 7

NAR-Based Reconstruction

7.1 Introduction

In this chapter[5], the reconstruction of bifurcation diagrams using time series from randomly forced maps will be investigated. Orthogonal polynomials, as discussed in section 3.4, are employed as predictor functions. However, for this formulation to be applicable to the BD reconstruction problem, a slight modification is needed. The needed modification will take into account the fact that the same model will be used for time series at different parameter values.

Given K time series S_k measured at different parameter values of an unknown dynamical system, the problem is to find a map with parameters $\mathbf{a}$: $X \mapsto G(X;\mathbf{a})$, which evolves data points $Y_n \mapsto Y_{n+1}$. Following section 3.3, $G(X;\mathbf{a})$ is approximated by polynomials $W_m(X)$ which are orthogonal with respect to the natural invariant measure $\rho(X)$ associated to each time series S_k. For the kth time series, $G(X;\mathbf{a}^k)$ is approximated to the Mth order as

$$G(X;\mathbf{a}^k) = \sum_{m=0}^{M} a_m^k W_m^k(X), \tag{7.1}$$

where the superscript denotes the time series number. Using the orthogonality property of $W_m(X)$, the expansion coefficients (or parameters) are determined by

$$a_m^k = \frac{1}{N_m^k} \int d^d X \rho_k(X) G(X;\mathbf{a}^k) W_m^k(X), \tag{7.2}$$

$$N_m^k = \int d^d X \rho_k(X) [W_m^k(X)]^2. \tag{7.3}$$

Since each time series may be characterized by different $\rho_k(X)$, the orthogonal polynomials $\{W_m^k(X)\}$ obtained using the Gram-Schmidt procedure may differ from time series to time series.

[5]The materials in this chapter also appear in Bagarinao, et al. (1999) [9] and Bagarinao, et al. (2000) [10].

This makes the reconstruction of $G(X;\mathbf{a})$ from the $G(X;\mathbf{a}^k)$ difficult since each of the $G(X;\mathbf{a}^k)$ is spanned by different basis functions. To circumvent this difficulty, $G(X;\mathbf{a})$ is expressed using the same basis functions. The orthogonal polynomials are then used as auxiliary basis to compute the expansion coefficients. In this way, the advantages of using orthogonal polynomials are retained. At the same time, it would be possible to interpolate the functional dependence of $G(X;\mathbf{a})$ with respect to $\mathbf{a}$ from the estimated $G(X;\mathbf{a}^k)$.

The results of this chapter are obtained by employing nonlinear autoregressive (NAR) models with polynomial terms as predictor functions. The use of this model is motivated by the following: NAR models are particularly effective for modeling noisy time series; their dependence on the parameters, i.e., coefficients of the polynomial, is linear which makes the structure of the model simple; and most importantly, an efficient scheme to compute the model parameters exists. This scheme makes possible the construction of parsimonious models necessary in the BD reconstruction problem.

More precisely, the NAR given by Eq. (3.56) is used as the predictor function $g(X;\mathbf{a})$. The equations are rewritten below:

$$\begin{aligned} y_n^{\text{pred}} &= g(Y_{n-1};\mathbf{a}) + \epsilon_n \\ &= a_0 + a_1 y_{n-1} + \cdots + a_d y_{n-d} + a_{d+1} y_{n-1}^2 + a_{d+2} y_{n-1} y_{n-2} \\ &\quad + \cdots + a_M y_{n-d}^k + \epsilon_n \qquad (7.4) \\ &= \sum_{m=0}^{M} a_m z_m(Y_{n-1}) + \epsilon_n \qquad (7.5) \end{aligned}$$

where $Y_{n-1} = (y_{n-1}, \cdots, y_{n-d})$ represents a vector in the d-dimensional reconstructed state space, the functional basis $\{z_m(X)\}$ is composed of all the distinct combinations of the coordinates up to degree k, $\mathbf{a} = (a_0, \cdots, a_M)$ represents the parameter set, ϵ_n accounts for the random forcing of the system, and $M + 1 = (k + d)!/(d!k!)$ determines the number of coefficients to be computed. The problem is to find $\mathbf{a}^1, \cdots, \mathbf{a}^K$ such that $g(X;\mathbf{a}^i)$ is a predictor function for the ith time series. To do this, the methods discussed in section 3.4 are used to get the optimal number of terms in Eq. (7.5) and to compute the values of their associated coefficients, that is, the set $\{\mathbf{a}^i\}$. Moreover, the predictors of all time series should have the same structure, i.e., the same terms should be present in all of them. This is discussed in the next section.

7.2 Model Selection

The criteria for model selection used in this chapter follows the model selection algorithm described in section 3.3 except for a slight modification. This modification arises due to the requirement that the predictor function for each time series should have similar structures or similar terms. Since it is assumed that all time series come from the same dynamical system, then there should only be one *working model* for all of the time series differing only in their respective parameter values. Once an appropriate working model is chosen, its bifurcation diagram is constructed relative to the model parameter that changes from one time series to another.

To take this into account, let $e_{\mathrm{CE}}(P)$ be the *cumulative multistep prediction error* given by

$$e_{\mathrm{CE}}(P) = \sum_{k=1}^{K} e_k(P) \tag{7.6}$$

where $e_k(P)$ is the multistep prediction error for each time series as defined in Eq. (3.76). The model selection algorithm can now be modified as follows:

1. For each time series:
   (a) Initialize the values of d and k, the delay and degree of Eq. (7.5) respectively.
   (b) Add each term one by one to the model. For each time, evaluate the multistep prediction error of the resulting model.
   (c) Add the multistep error of this time series to the cumulative multistep error.
2. The number of terms that gives the minimum of $e_{\mathrm{CE}}(P)$ for higher values of P will be included in the final model.
3. Recompute the coefficients $\mathbf{a}^k$ for all the time series and include only the selected terms.

Again, this only lists the basic approach to select the appropriate terms in the final model. This can still be enhanced in a number of ways using the suggestions mentioned in subsection 3.4.3

The Rossler equations

To illustrate the above procedure, consider the following set of equations originally suggested by Rossler [71]

$$\begin{aligned} \frac{dx}{dt} &= -(x+z), \\ \frac{dy}{dt} &= x + ay, \\ \frac{dz}{dt} &= bx - cz + xz, \end{aligned} \tag{7.7}$$

where a, b, and c are the system parameters. Twelve time series of the y-component were generated by integrating Eqs. (7.7) using Euler's method with time step $\Delta = 0.001$. The following parameter values were used:

$$\begin{aligned} b &= 0.3, \\ c &= 0.7\sin[2\pi(i-1)/12] + 5.0, \\ a &= 0.03\cos[2\pi(i-1)/12] + 0.33, \end{aligned}$$

where $i = 1, \cdots, 12$.

From these time series, the goal is to obtained an optimal model for the Rossler system. Figure (7.1) shows the different plots of the multistep prediction error for different values of sampling time τ. The delay d and degree k were initially set at 3 and 5, respectively. For higher values of τ [Fig. 7.1(A) and (B)], the multistep prediction error has a minimum after the eighth term was added to the model. Addition of more terms to the model makes it unstable as can be seen in very large prediction errors. On the other hand, for lower values of τ [Fig. 7.1 (C) and (D)], the minimum occurred after the addition of the tenth term. These results indicate two candidate models for the Rossler system depending on the value of the sampling time τ. These are given by

MODEL 1:

$$\begin{aligned} y_n^{\text{calc}} &= a_0 + a_1 y_{n-1} + a_2 y_{n-2} + a_3 y_{n-3} + a_4 y_{n-1}^2 \\ &\quad + a_5 y_{n-1} y_{n-2} + a_6 y_{n-1} y_{n-3} + a_7 y_{n-2}^2 \\ &\quad + a_8 y_{n-2} y_{n-3}, \end{aligned} \tag{7.8}$$

MODEL 2:

$$\begin{aligned} y_n^{\text{calc}} &= a_0 + a_1 y_{n-1} + a_2 y_{n-2} + a_3 y_{n-3} + a_4 y_{n-1}^2 \\ &\quad + a_5 y_{n-1} y_{n-2} + a_6 y_{n-1} y_{n-3} + a_7 y_{n-2}^2 \\ &\quad + a_8 y_{n-2} y_{n-3} + a_9 y_{n-3}^2 + a_{10} y_{n-1}^3, \end{aligned} \tag{7.9}$$

for higher and lower values of τ, respectively.

Figure 7.2 shows sample time series of the above models (left side) and the time series used to estimate the model parameters (right side). For higher values of τ, some of the identified models [e.g. Fig. 7.2(A)] behaved differently as compared to the original system. On the other hand, for smaller values of τ, all the models behaved similarly as the original system. This is illustrated in Fig. 7.2(E) where the time series is similar to that of Fig. 7.2(F).

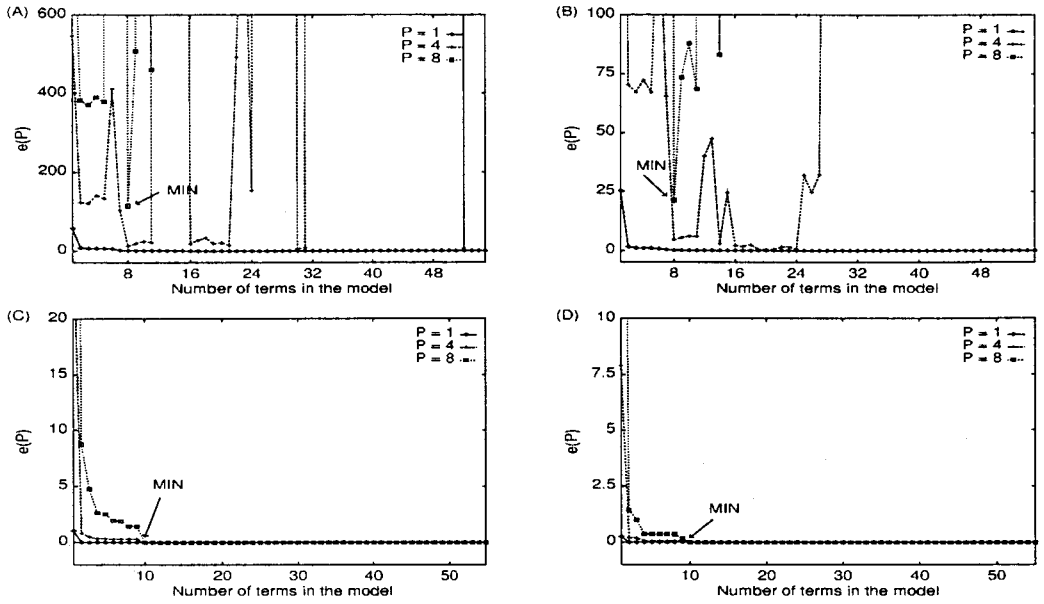

Figure 7.1: Multistep prediction error. (A) $\tau = 0.8$. (B) $\tau = 0.5$. (C) $\tau = 0.1$. (D) $\tau = 0.05$. The x-axis is the number of terms included in the model in the order given by Eq. (3.56). The minimum is indicated by the arrow.

The Lorenz equations

Another familiar set of differential equations is the Lorenz equation given by

$$\begin{aligned} \frac{dx}{dt} &= \sigma(y - x), \\ \frac{dy}{dt} &= \rho x - y - xz, \\ \frac{dz}{dt} &= xy - \beta z, \end{aligned} \tag{7.10}$$

where σ, ρ, and β are the parameters of the system. Time series of the x-variable were generated at different parameter values: $\sigma = 10$, $\beta = 8/3$, and $\rho = 0.5i$, $i = 45, \cdots, 60$. Moreover, time series with dynamical noise of intensity κ added to the x-variable were also generated. The integration of Eq. (7.10) was carried out using Euler's method with time step $\Delta = 0.001$. The sampling time τ was set at 0.1 and three different values of κ were used.

The multistep prediction errors of the generated time series were evaluated with $d = 3$ and $k = 7$ as the initial delay and degree, respectively. The results are summarized in Fig. 7.3. For different values of κ, different number of significant terms were obtained. Thus, three different models were identified for the Lorenz system depending on the value of the noise intensity κ. Figure 7.4 shows the behavior of the identified models in comparison to that of the original

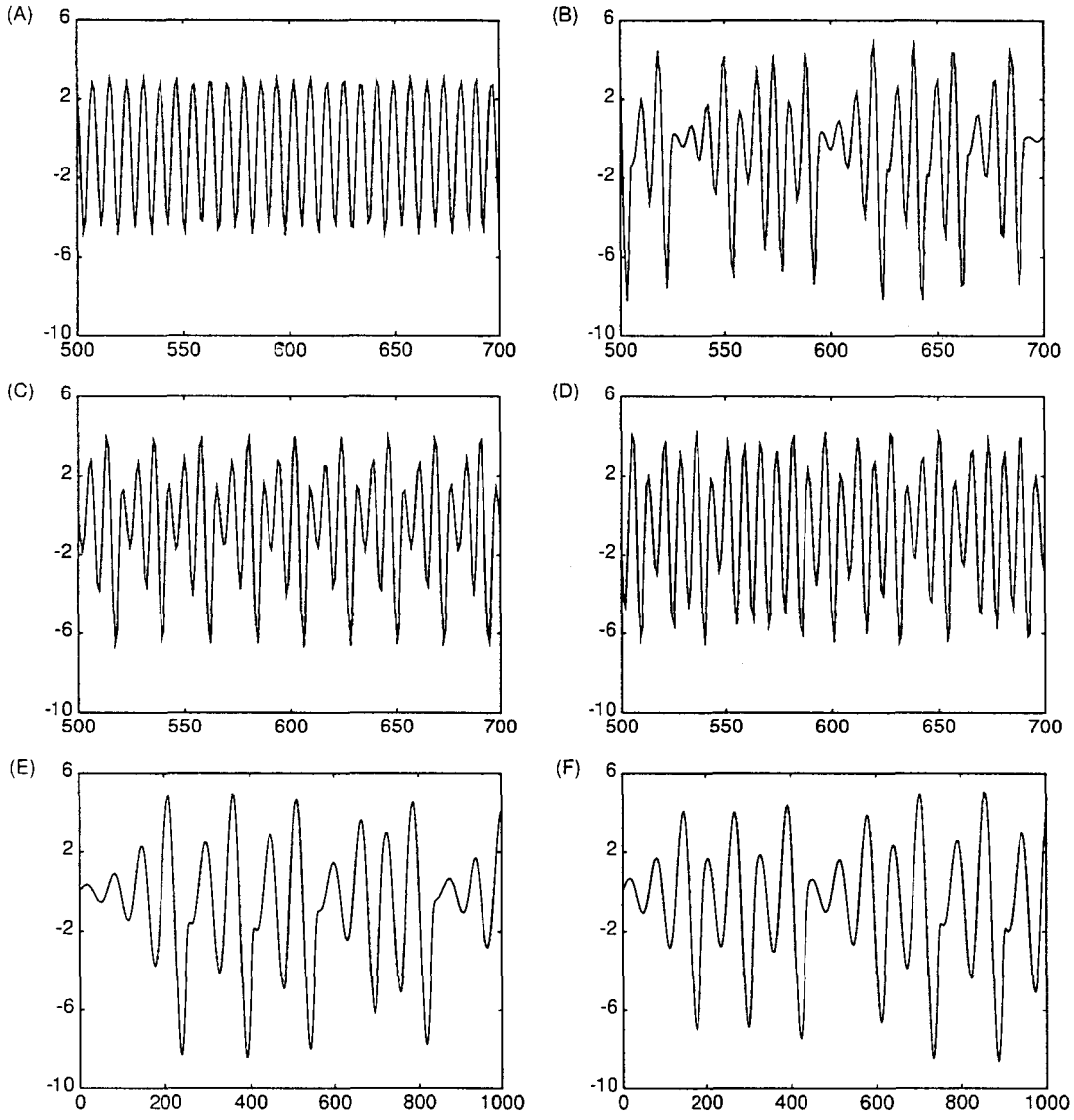

Figure 7.2: Sample time series generated by the obtained models (left side) and the corresponding time series used in estimating the parameter values of the models. (A)-(D) $\tau = 0.8$. (E) and (F) $\tau = 0.1$.

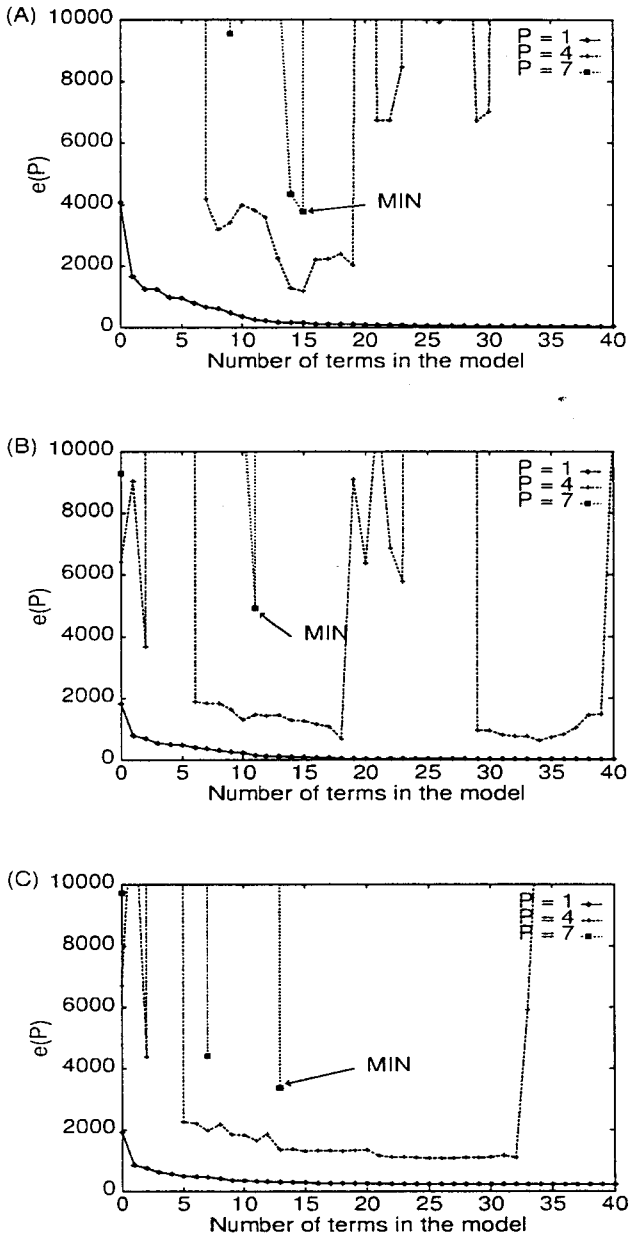

Figure 7.3: Multistep prediction error. (A) $\kappa = 0.0$. (B) $\kappa = 0.01$. (C)$\kappa = 1.0$. The minimum is indicated by the arrow.

system. At the left side are the time series generated from these models, while at the right side are those used in estimating the model parameters. For higher values of noise, time series of the identified models appear periodic rather than chaotic [Fig. 7.4(C)].

7.3 Bifurcation Parameter Identification

In general, the model given by Eq. (7.5) has more parameters than the original system. To determine the bifurcation parameters, the computed coefficients of each predictor function are collected to form the set of parameter vectors $\{\mathbf{a}^i = (a_0^i, \cdots, a_M^i)\}$, $i = 1, \cdots, K$. The set of points $\{\mathbf{a}^i\}$ are then used to locate the projection region.

In many situations, the projection region is well-approximated by a linear subspace of the parameter space of the model (section 4.5). This is the case when dealing with a small parameter region, reconstructing specific bifurcations, or the given system is an LIP map (chapter 5). Under these conditions, PCA provides a computationally efficient method to determine a satisfactory approximation of the projection region. Recall that in the PCA-based approach, the number of significant eigenvalues of the sample covariance matrix gives the dimension of the projection region and hence corresponds to the number of bifurcation parameters of the system. Moreover, the eigenvectors associated to the significant eigenvalues span the required projection region. Thus any point in this region can be expressed as

$$\mathbf{a}_{\mathrm{PR}}(\mu) = \bar{\mathbf{a}} + \sum_{i=1}^{P} \mu_i \mathbf{e}_i \tag{7.11}$$

where the $\mathbf{e}_i$'s are the eigenvectors associated with the P significant eigenvalues, $\mu = (\mu_1, \cdots, \mu_P)$ represent the expansion coefficients, and $\bar{\mathbf{a}}$ represents the mean of $\{\mathbf{a}^i\}$. The BD of the model $(X; \mu) \rightarrow g(X; \mathbf{a}_{\mathrm{PR}}(\mu))$ on the projection region can then be taken as the reconstructed BD with μ as the effective bifurcation parameter.

Going back to the Lorenz system, one can see from Fig. 7.3 that the resulting NAR model has 15, 11, and 13 coefficients for $\kappa = 0.0$, $\kappa = 0.01$, and $\kappa = 1.0$, respectively. The number of coefficients is thus larger compared to the actual number of varying bifurcation parameters which is just ρ. To determine the actual number of system parameters, PCA is employed. To do this, the estimated coefficients are collected, the covariance matrix is formed, and the eigenvalues are then computed. These are shown in Fig. 7.5. From the figure, a good correspondence between the number of system parameters and the number of significant eigenvalues can be observed. This implies that under the conditions mentioned above, the number of system parameters can

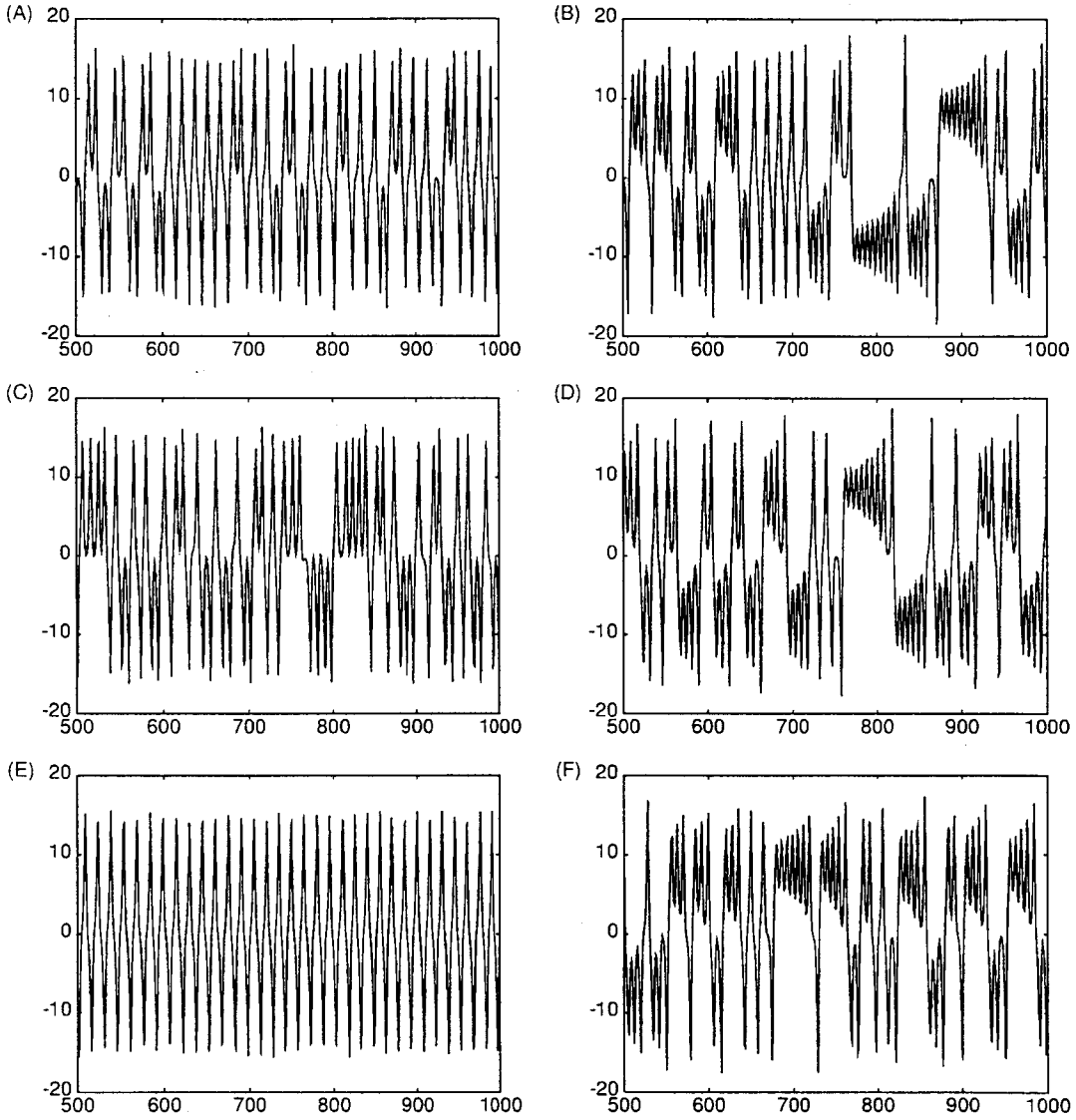

Figure 7.4: Sample time series generated by the obtained models (left side) and the corresponding time series used in estimating the parameter values of the models. (A&B) $\kappa = 0.0$. (C&D) $\kappa = 0.01$. (E&F) $\kappa = 1.0$.

be readily obtained from the number of significant eigenvalues of the covariance matrix of the estimated coefficient values.

7.4 Example 1: The Sine Map

In this section, a map which is nonpolynomial is considered and the applicability of using Eq. (7.5) as a working model is investigated. In particular, the reconstruction of the BD of the sine map with a Gaussian white noise input is studied. The map is given by

$$x_n = \sin(\alpha x_{n-1}) + \beta + \kappa\epsilon_n \tag{7.12}$$

where α and β are the system parameters. The behavior of Eq. (7.12) as a function of the parameter β is characterized by a period-doubling route to chaos followed by period-doubling reversal as can be seen in Fig. 7.6.

Ten time series were generated with 1000 data points for parameter values given by $\beta_j = 0.75 + 0.5j/10$, $j = 1, \cdots, 10$ and $\alpha = 2$. Equation (7.5) with $d = 3$ and $k = 6$ was used as the initial search space. For $\kappa = 0$, the following optimal model was obtained for the ten time series:

$$x_n = a_0 + a_1 x_{n-1} + a_2 x_{n-1}^2 + a_3 x_{n-1}^3 + a_4 x_{n-1}^4 + a_5 x_{n-1}^5 + a_6 x_{n-1}^6 + \epsilon_n. \tag{7.13}$$

To determine which of the seven parameters vary with the time series, the estimated values of the model parameters were analyzed using PCA. The eigenvalues are shown in Fig. 7.7. One eigenvalue is significant indicating that changes in the parameters are oriented along a single direction. The presence of the other nonzero eigenvalue can be attributed to the estimation errors. This is evident in the plot of the projections of the estimated parameter values onto the eigenvectors associated with the two nonzero eigenvalues as shown in Fig. 7.8. The projections onto the first eigenvector clearly follow a linear relation whereas those of the second eigenvector fluctuate around zero.

To reconstruct the BD of the unforced sine map, the BD of Eq. (7.13) is constructed by changing the model parameters along the direction defined by the first eigenvector, that is, $\mathbf{a}_{\mathrm{PR}} = \bar{\mathbf{a}} + \mu_1 \mathbf{e}_1$ where $\bar{\mathbf{a}}$ is the mean of the estimates of the model parameters, $\mathbf{e}_1$ is the first eigenvector and μ_1 serves as the new bifurcation parameter. As shown in Fig. 7.9, the similarity between the two BDs is apparent. This illustrates the applicability of Eq. (7.5) for the BD reconstruction problem even for non-polynomial maps.

Consider now the case which takes into account the presence of random forcing. Three values of the noise amplitude, given by $\kappa = 0.01$, $\kappa = 0.1$ and $\kappa = 0.5$, were investigated. For all cases,

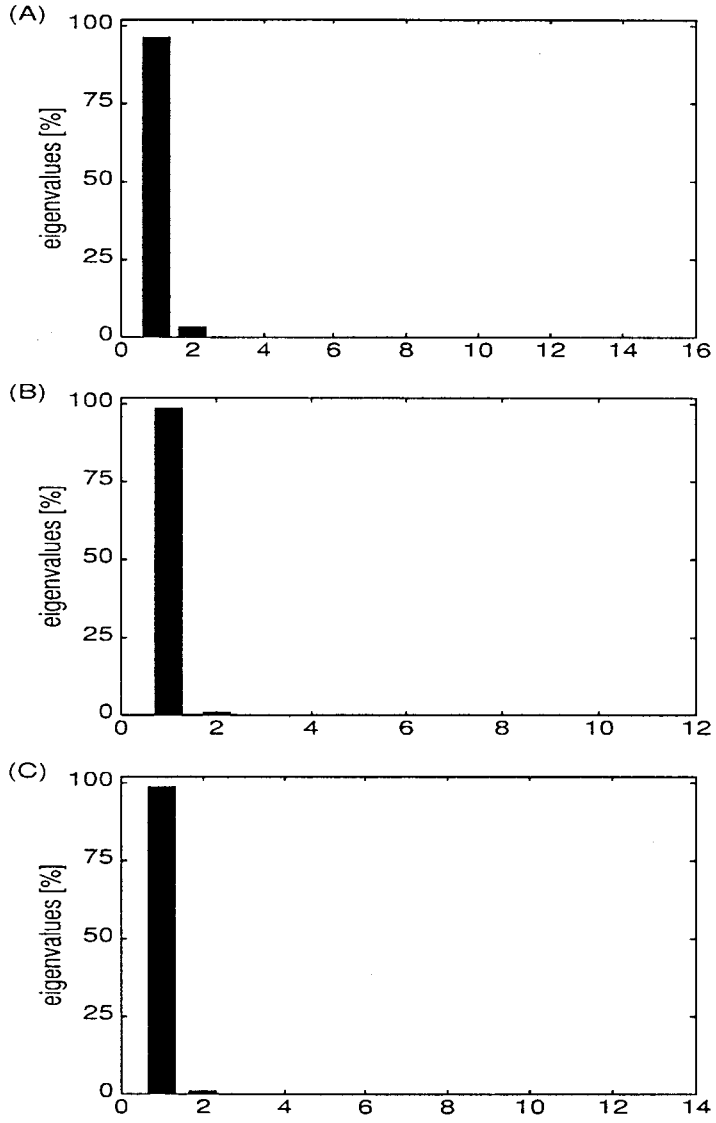

Figure 7.5: Significant eigenvalues [in relative values (eigenvalue)/ $\sum$(eigenvalues)$\times$100] of the covariance matrix formed by the estimated coefficients $\{\mathbf{a}_{\min}\}$. (A) $\kappa = 0.0$, and (B) $\kappa = 0.01$. (C) $\kappa = 1.0$.

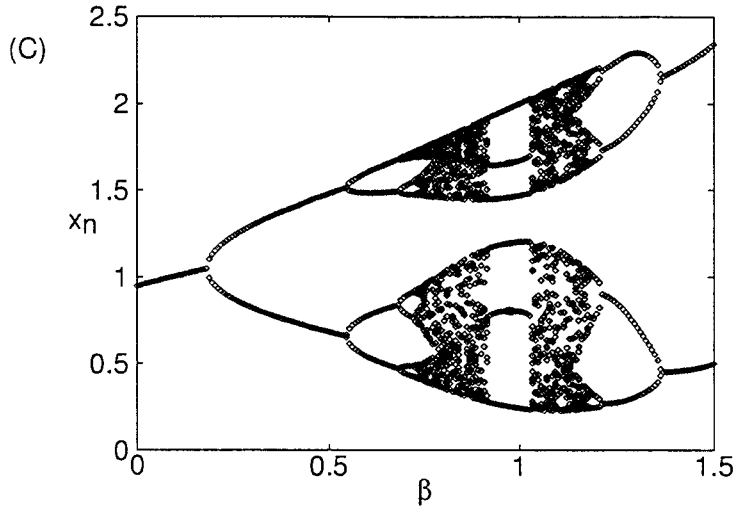

Figure 7.6: One-dimensional bifurcation diagram of the sine map as a function of β ($\alpha = 2$).

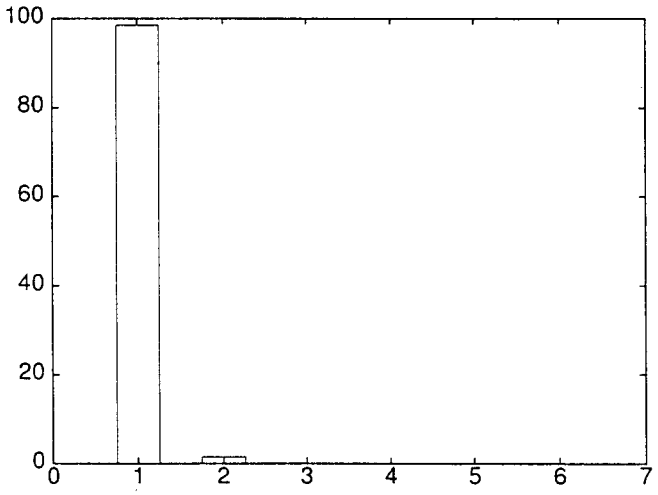

Figure 7.7: Eigenvalue profile (in relative values).

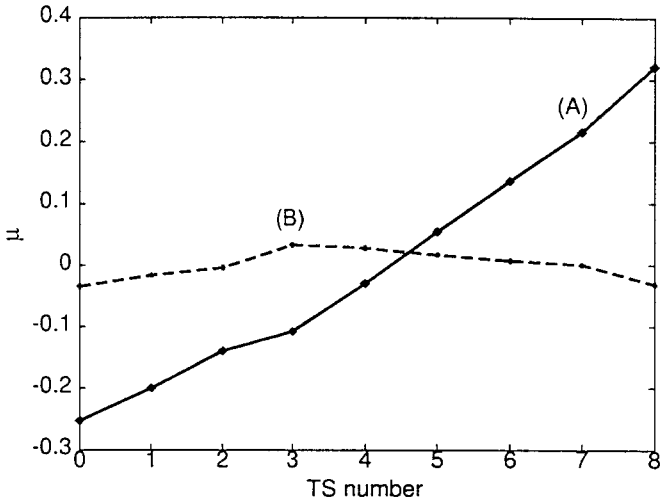

Figure 7.8: (A) Projection onto the first eigenvector. (B) Projection onto the second eigenvector.

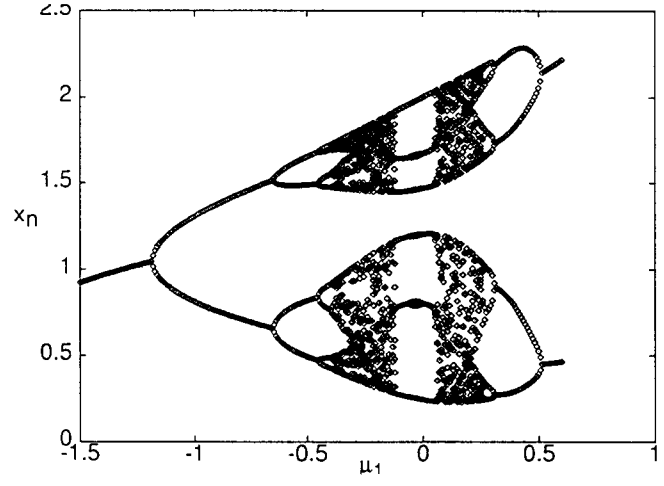

Figure 7.9: Reconstructed BD of the unforced sine map.

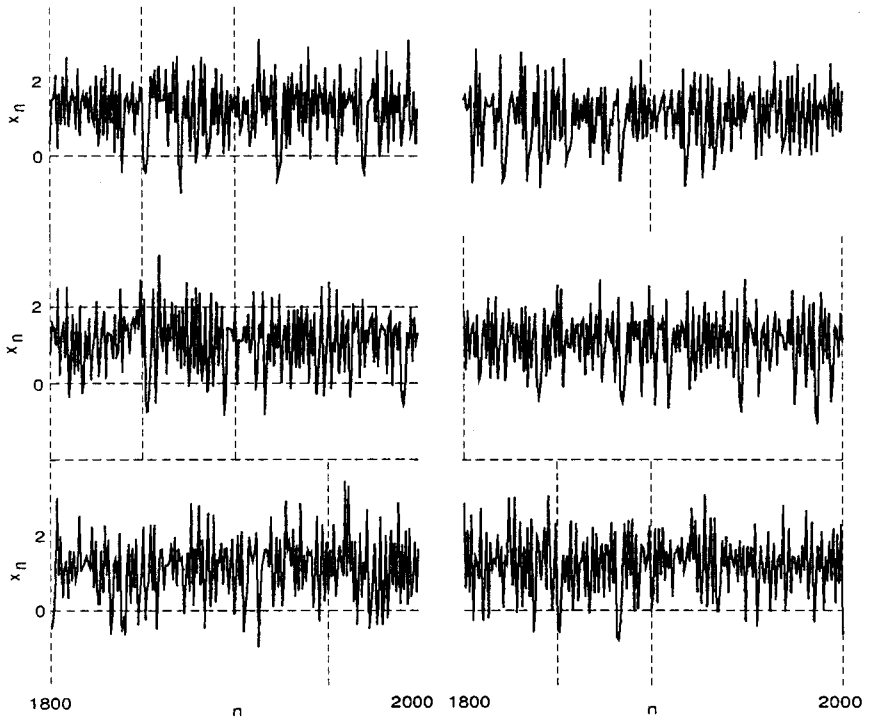

Figure 7.10: Sample time series from the sine map. Abscissas and ordinates are dimensionless.

ten time series with 10000 data points were generated using the same parameter values as that in the unforced case. Some of these are shown in Fig. 7.10. A mere visual examination of the given time series does not reveal any significant difference in the behavior of the system. It is therefore necessary to distinguish the different behaviors by unraveling the putative bifurcations that separate them, as well as to uncover possible behaviors of the system not readily observed in the available time series. This should be done without the explicit knowledge of the system equations and the number of parameters.

Following the same approach as for the case without noise, the initial search space was set with $d = 3$ and $k = 6$. For all noise amplitudes under study, the same model given by Eq. (7.13) was obtained. The PCA analysis of the estimated parameters yielded the same result as that of the unforced case (Fig. 7.7) characterized by two nonzero eigenvalues where one being more significant than the other. The projections of the estimated parameter values onto the eigenvectors associated with the two nonzero eigenvalues also behave similarly as that of the unforced case. Thus in spite of the presence of random forcing, the number of bifurcation parameters in the original map can still be obtained. The bifurcation diagram was reconstructed along the direction defined by the first eigenvector. The resulting BD is shown in Fig. 7.11.

It can be observed that for low noise amplitude, the reconstructed BD closely resembles that of the original BD. For higher noise values, some regions in the original BD disappear in the reconstruction. This is more evident in the period three region which is preserved at low noise forcing [Fig. 7.11 (A)] but completely disappears for higher noise amplitudes [Fig. 7.11 (B & C)].

Similar experiments were conducted for the bifurcation of the sine map with respect to α while keeping β constant and equal to 1. Ten time series for parameter values $\alpha_j = 1.8 + j0.4/10$, $j = 1, \cdots, 10$ were generated. The initial search space was $d = 3$ and $k = 6$ and the obtained model is again given by Eq. (7.13). Figure 7.12 shows the eigenvalues. For $\kappa = 0$, there is only one significant eigenvalue consistent with the number of bifurcation parameter. However, for $\kappa = 0.5$, three significant eigenvalues appear when $N = 1000$. By increasing the number of data points ($N = 10000$ and $N = 20000$), the number of significant eigenvalues decreases as can be seen in Fig. 7.12 (C)-(D). From this, one can conclude that the other eigenvalues are due to estimation errors. This was verified by the plot of the projections of the estimates onto the first eigenvector where a linear relation was observed. The reconstructed BD along the first eigenvector is shown in Fig. 7.13.

Finally, a two-parameter reconstruction of the sine map is considered. The ten time series with varying α were combined with the ten time series with varying β, giving a total of twenty time series. Again the resulting model is given by Eq. (7.13). This model has seven coefficients. Therefore, twenty parameter vectors of order seven from which to estimate the projection region are accessible. The PCA of this set of vectors yields two significant eigenvalues as shown in Fig. 7.14. This suggests that two system parameters were varied when the given time series were generated. This corresponds exactly to the two parameters in the original system. The projection region is now given by Eq. (7.11) with $P = 2$, and $\mathbf{e}_1$ and $\mathbf{e}_2$ are the eigenvectors associated to the two significant eigenvalues. The effective bifurcation parameters are given by μ_1 and μ_2.

To show the correspondence between the two parameter sets, $\{\alpha, \beta\}$ and $\{\mu_1, \mu_2\}$, are plotted in Fig. 7.15 (B). Here, μ_1 and μ_2 are the projections of $\{\mathbf{a}^i\}$ onto the projection region. These projections are computed by taking the inner product between $\{\mathbf{a}^i\}$ and the two eigenvectors $\mathbf{e}_1$ and $\mathbf{e}_2$. The figure shows that the distribution of these points follows that of the original parameters $\{\alpha, \beta\}$. Figure 7.15 also illustrates the effects of noise on the reconstruction process. From a computational point of view, noise generally corrupts the estimation of the model parameters. This would affect the location of the projected points in the parameter space which, in turn,

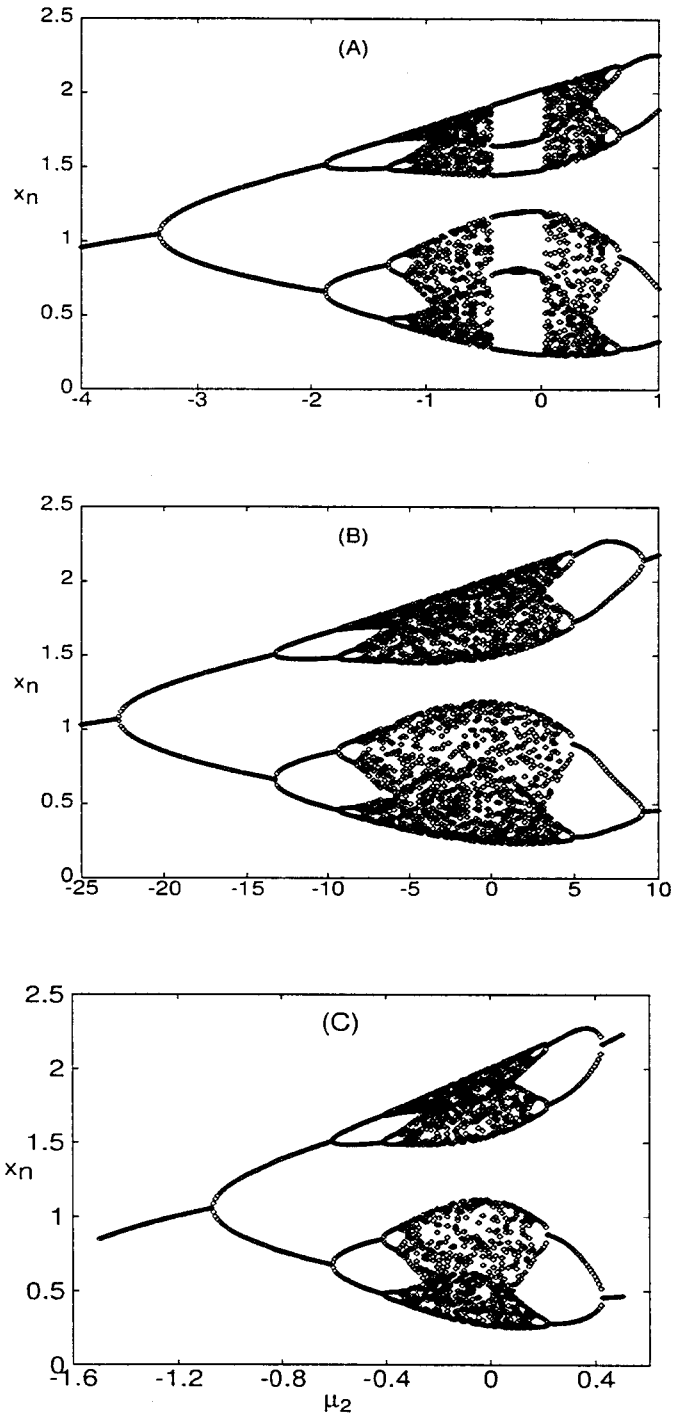

Figure 7.11: Reconstructed BD of the randomly forced sine map. (A) $\kappa = 0.01$. (B) $\kappa = 0.1$. (C) $\kappa = 0.5$.

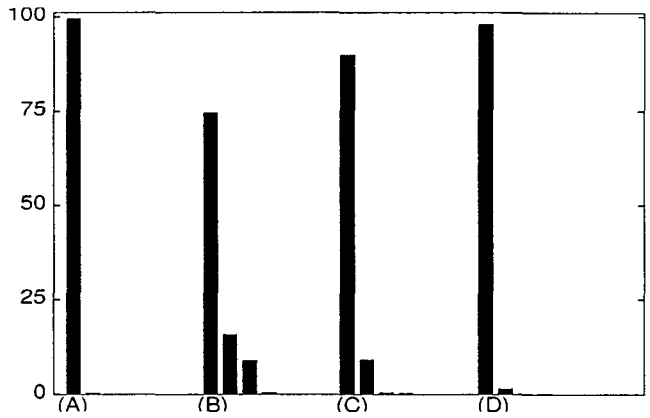

Figure 7.12: Eigenvalues. (A) $\kappa = 0$. (B)$\kappa = 0.5$ and $N = 1000$. (C) $\kappa = 0.5$ and $N = 10000$. (D) $\kappa = 0.5$ and $N = 20000$.

would affect the determination of the projection region. The proposed algorithm is robust to such effects, as illustrated in Fig. 7.15, which shows similar distribution of the two parameter sets in spite of the presence of strong dynamical noise.

7.5 Example 2: The FHN Equations

In this section, the forced FitzHugh-Nagumo (FHN) equation is considered. The set of equations is given by

$$\begin{aligned} \frac{dx}{dt} &= c(y + x - \frac{x^3}{3} + z) + \eta, \\ \frac{dy}{dt} &= \frac{-(x - a + by)}{c}, \end{aligned} \tag{7.14}$$

where η represents the random driving force. The x-variable represents excitability while the y-variable represents refractoriness. z represents the external applied current and a, b, and c are the system parameters. For some range of z, the model behaves as an excitable membrane whereas for some other z values, it behaves as an oscillating membrane defining a stable limit cycle. In the absence of noise, the behavior of Eq. (7.14) as a function of the parameter z is summarized in Fig. 7.16. From the figure, as z is increased, the stable equilibrium point becomes unstable and a subcritical Hopf bifurcation occurs at $z \approx -1.40$ giving rise to the stable limit cycle. As the current is increased further, another subcritical Hopf bifurcation takes place at $z \approx -0.346$ and the stable limit cycle disappears.

Ten time series were generated by integrating Eq. (7.14) using Euler's method with $\Delta t = 0.15$

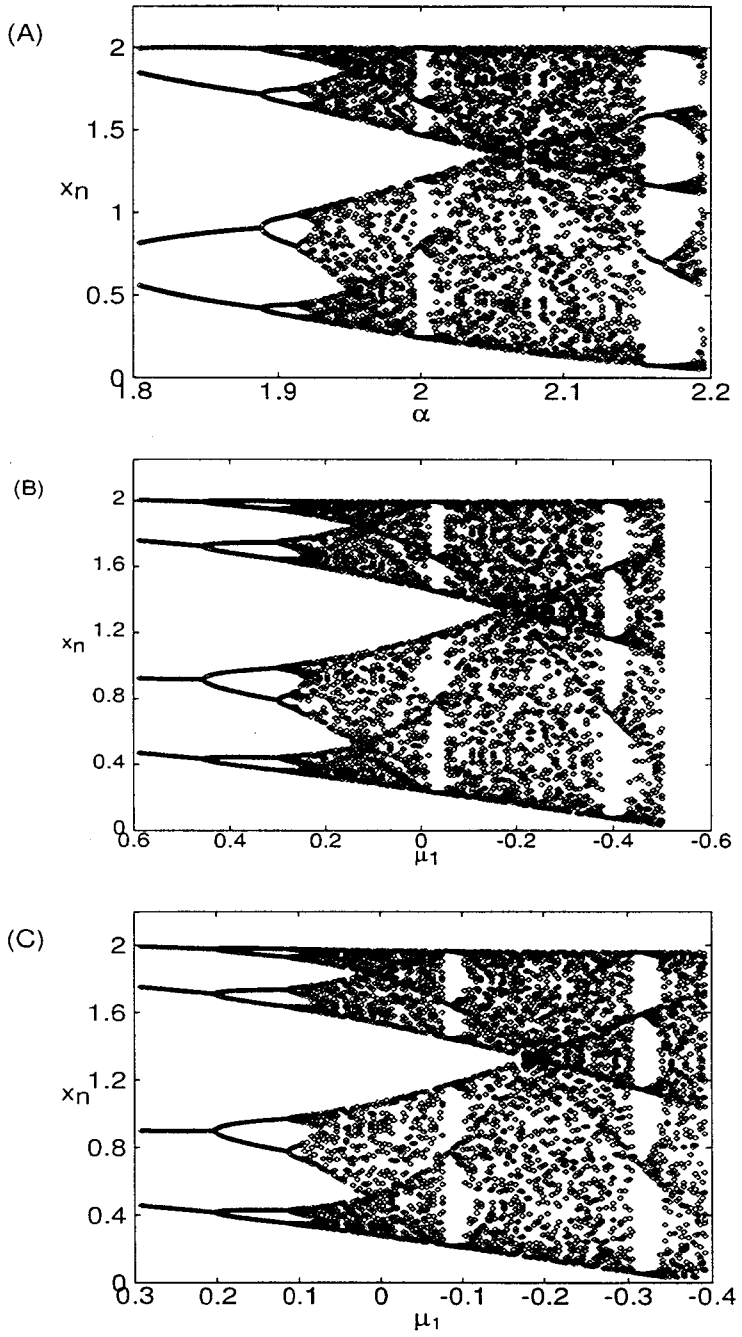

Figure 7.13: Bifurcation diagram of the sine map with respect to α. (A) Original BD. (B) Reconstructed with $\kappa = 0$. (C) Reconstructed with $\kappa = 0.5$.

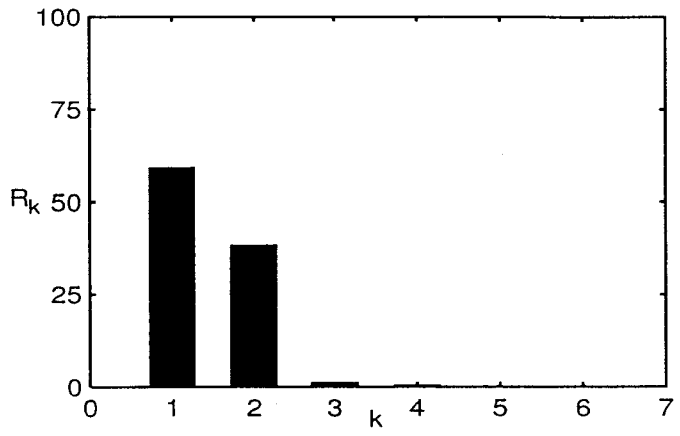

Figure 7.14: Eigenvalues (in decreasing order) expressed in relative units $R_k = (\lambda_k / \sum_{i=1}^{7} \lambda_i) \times 100$ where λ_k represents an eigenvalue and k represents the order. Abscissas and ordinates are dimensionless.

for parameter values $z_i = 0.5 - (i-1)3.0/10.0$, $i = 1, \cdots, 10$. The other parameters were fixed at the following values: $a = 0.7$, $b = 0.8$, and $c = 3.0$. Using Eq. (7.5) with $k = 3$ and $d = 3$ as the initial search space, an optimal model for each time series was obtained. The estimated parameter values are shown in Table 7.1. From the table, one can see a good correspondence between the estimated and the true values. Here the true values were obtained by solving explicitly the terms in the Euler's expansion and computing the associated parameters from the known values of a, b, and c.

To generate time series with system noise, Euler's method was used in differentiating Eq. (7.14). In particular, the following scheme was employed:

$$x_{n+1} = x_n + f(x_n, y_n)\Delta t + \sqrt{\Delta t}\,\kappa\epsilon_n \tag{7.15}$$

$$y_{n+1} = y_n + g(x_n, y_n)\Delta t \tag{7.16}$$

where

$$f(x,y) = c(x + y - \frac{x^3}{3} + z) \tag{7.17}$$

$$g(x,y) = -\frac{x - a + by}{c} \tag{7.18}$$

and ϵ_n is a Gaussian white noise with unit variance. With this, ten time series for $\kappa = 2.0$ where generated using the same parameter values used in the unforced case. These time series are plotted in Fig. 7.17. The same number of terms, as in the case without noise, was used. The estimated model parameter values are shown in Table 7.2.

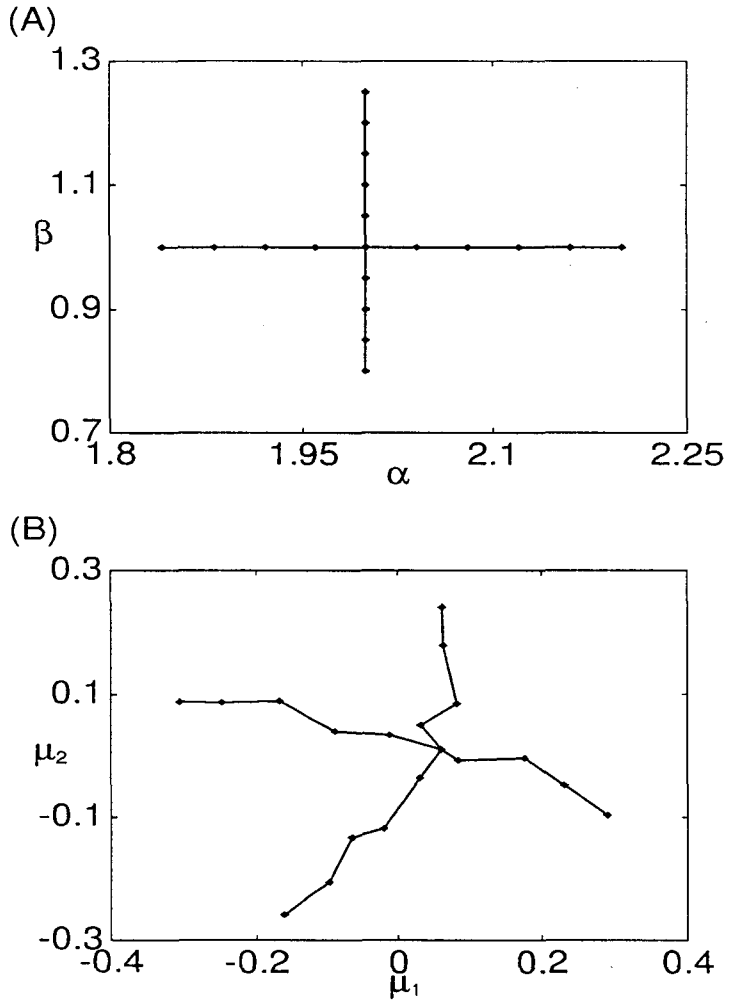

Figure 7.15: (A) Parameter values used in generating the twenty time series. (B) Projections of the computed $\{\mathbf{a}^i\}$ onto the projection region spanned by the eigenvectors associated with the two significant eigenvalues. The x-axis is spanned by the first eigenvector and the y-axis is spanned by the second eigenvector. Units: dimensionless.

TS		1.0	x_{n-1}	x_{n-2}	x_{n-3}	x_{n-1}^2	x_{n-1}^3	x_{n-2}^3
1	T	0.025	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	0.036	2.370	-1.420	0.0	0.039	-0.160	0.144
2	T	0.019	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	0.026	2.384	-1.417	0.0	0.027	-0.157	0.144
3	T	0.014	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	0.018	2.395	-1.416	0.0	0.016	-0.154	0.144
4	T	0.008	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	0.008	2.418	-1.437	0.012	0.0	-0.149	0.146
5	T	0.003	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	0.002	2.409	-1.416	0.0	0.0	-0.149	0.144
6	T	-0.002	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	-0.003	2.412	-1.418	0.0	0.0	-0.150	0.144
7	T	-0.008	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	-0.007	2.413	-1.421	0.002	0.0	-0.150	0.144
8	T	-0.013	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	-0.013	2.412	-1.420	0.002	0.0	-0.150	0.144
9	T	-0.018	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	-0.018	2.413	-1.422	0.003	0.0	-0.150	0.144
10	T	-0.024	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	-0.023	2.413	-1.424	0.004	0.0	-0.150	0.144

Table 7.1: FHN system. Estimated parameter values compared to the true values for $\kappa = 0.0$. T = True values and E = Estimated values.

TS		1.0	x_{n-1}	x_{n-2}	x_{n-3}	x_{n-1}^2	x_{n-1}^3	x_{n-2}^3
1	T	0.025	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	0.029	2.401	-1.416	0.0	0.0	-0.150	0.147
2	T	0.019	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	0.019	2.425	-1.433	0.0	0.0	-0.152	0.148
3	T	0.014	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	0.012	2.418	-1.423	0.0	0.0	-0.151	0.146
4	T	0.008	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	0.008	2.413	-1.419	0.0	0.0	-0.150	0.143
5	T	0.003	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	0.003	2.417	-1.422	0.0	0.0	-0.151	0.146
6	T	-0.002	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	-0.003	2.419	-1.426	0.0	0.0	-0.154	0.150
7	T	-0.008	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	-0.008	2.404	-1.406	0.0	0.0	-0.151	0.147
8	T	-0.013	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	-0.011	2.404	-1.409	0.0	0.0	-0.147	0.141
9	T	-0.018	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	-0.015	2.426	-1.429	0.0	0.0	-0.153	0.147
10	T	-0.024	2.410	-1.410	0.0	0.0	-0.150	0.144
	E	-0.021	2.401	-1.410	0.0	0.0	-0.149	0.145

Table 7.2: FHN system. Estimated parameter values compared to the true values for $\kappa = 2.0$ ($\Theta_{TH} = 0.009$). T = True values and E = Estimated values.

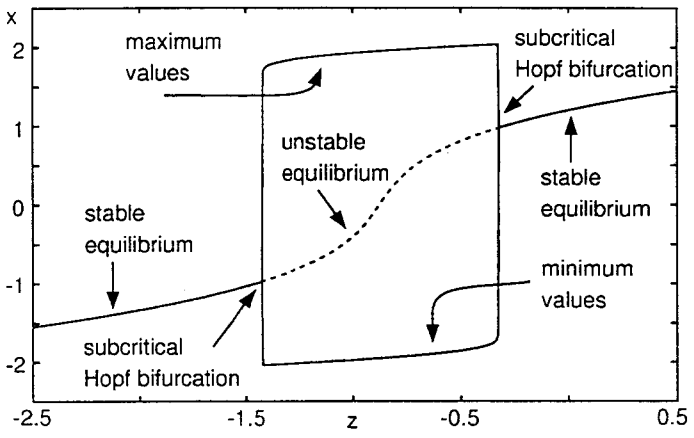

Figure 7.16: Bifurcation diagram of the FitzHugh-Nagumo equation. The other parameters are fixed: $a = 0.7$, $b = 0.8$, and $c = 3.0$.

From the results, it can be seen that all the time series can be modeled with the following NAR equation

$$x_n = a_0 + a_1 x_{n-1} + a_2 x_{n-2} + a_3 x_{n-1}^3 + a_4 x_{n-2}^3 + \epsilon_n \tag{7.19}$$

where a_i, $i = 0, \cdots, 4$ are the model parameters. However, it is noted that for other noise intensities, for example $\kappa = 5.0$, some time series is modeled with a different NAR equation as can be seen from Table 7.3. For this case, there are two competing models that can be deduced from the table. The first model describing S_1 and S_2 is given by

$$x_n = a_0 + a_1 x_{n-1} + a_2 x_{n-3} + a_3 x_{n-1}^3 + a_4 x_{n-1} x_{n-2} x_{n-3} + a_5 x_{n-2} x_{n-3}^2 + \epsilon_n. \tag{7.20}$$

The second model describing the remaining time series is given by Eq. (7.19). In the following, Eq. (7.19) is used as the working model for this system.

To reconstruct the BD of the FHN system, it is necessary to determine which of the five model parameters in Eq. (7.19) represent the actual bifurcation parameters. To do this, PCA was applied to the estimated values of the model parameter in the same manner as the sine map. The eigenvalues of the covariance matrix formed by these estimated values are shown in Fig. 7.18. The figure suggests that there are two significant directions in which the parameters are changed from one time series to the other. This is greater than the actual number of bifurcation parameter, that is, one. In fact, only parameter a_0 is changed from one time series to the other. This is consistent with the results in section 6.2 where the linear manifold approximation (PCA-based reconstruction) is insufficient to get the actual number of bifurcation parameters.

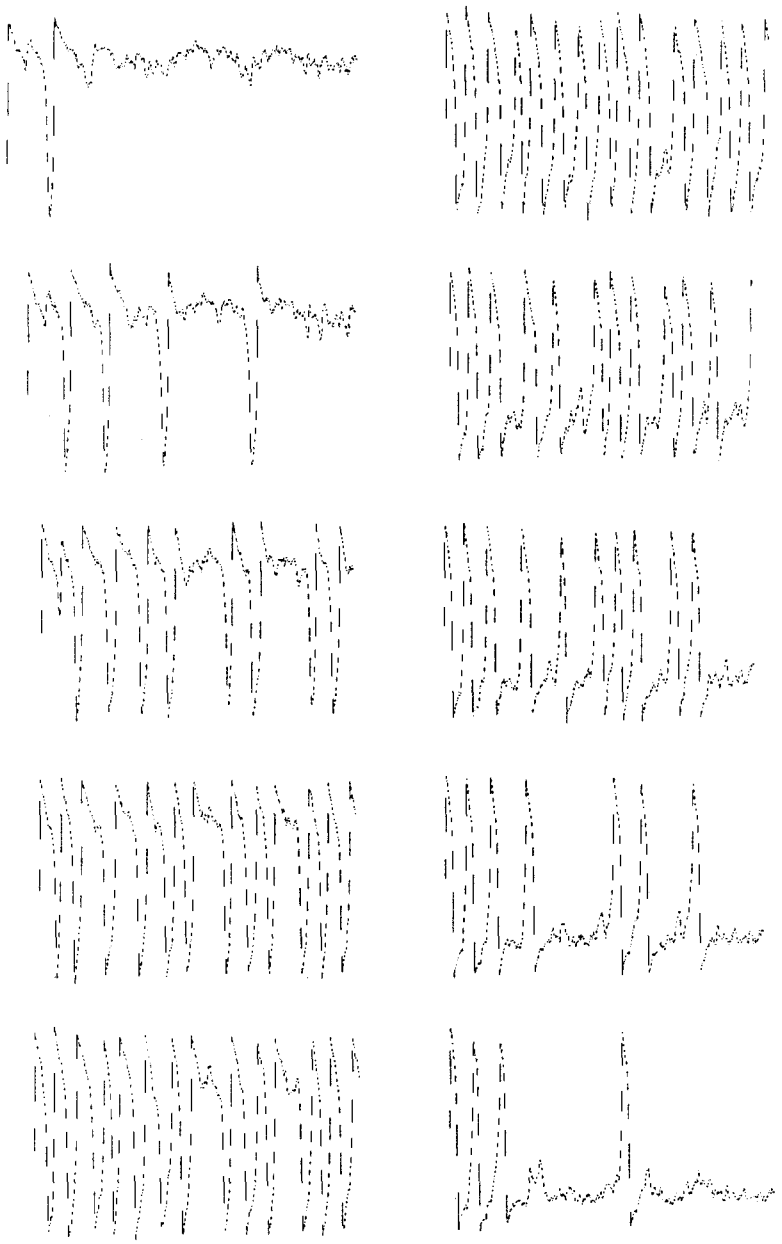

Figure 7.17: Sample time series. First 1000 data points of the time series used in reconstructing the BD of the FHN system. The presence of noise causes spiking events even for parameter values corresponding to the excitable region of the model.

TS		1.0	x_{n-1}	x_{n-2}	x_{n-3}	x_{n-1}^3	$x_{n-1}x_{n-2}x_{n-3}$	x_{n-2}^3	$x_{n-2}x_{n-3}^2$
1	T	0.025	2.410	-1.410	0.0	-0.150	0.0	0.144	0.0
	E	0.029	1.818	0.0	-0.827	-0.141	0.064	0.0	0.072
2	T	0.019	2.410	-1.410	0.0	-0.150	0.0	0.144	0.0
	E	0.024	1.806	0.0	-0.805	-0.141	0.066	0.0	0.067
3	T	0.014	2.410	-1.410	0.0	-0.150	0.0	0.144	0.0
	E	0.013	2.409	-1.409	0.0	-0.151	0.0	0.144	0.0
4	T	0.008	2.410	-1.410	0.0	-0.150	0.0	0.144	0.0
	E	0.011	2.419	-1.425	0.0	-0.151	0.0	0.146	0.0
5	T	0.003	2.410	-1.410	0.0	-0.150	0.0	0.144	0.0
	E	0.003	2.414	-1.418	0.0	-0.152	0.0	0.147	0.0
6	T	-0.002	2.410	-1.410	0.0	-0.150	0.0	0.144	0.0
	E	-0.002	2.408	-1.410	0.0	-0.150	0.0	0.144	0.0
7	T	-0.008	2.410	-1.410	0.0	-0.150	0.0	0.144	0.0
	E	-0.006	2.383	-1.379	0.0	-0.148	0.0	0.140	0.0
8	T	-0.013	2.410	-1.410	0.0	-0.150	0.0	0.144	0.0
	E	-0.008	2.402	-1.401	0.0	-0.152	0.0	0.145	0.0
9	T	-0.018	2.410	-1.410	0.0	-0.150	0.0	0.144	0.0
	E	-0.013	2.425	-1.424	0.0	-0.154	0.0	0.147	0.0
10	T	-0.024	2.410	-1.410	0.0	-0.150	0.0	0.144	0.0
	E	-0.017	2.407	-1.403	0.0	-0.151	0.0	0.143	0.0

Table 7.3: FHN system. Estimated parameter values compared to the true values for $\kappa = 5.0$ ($\Theta_{TH} = 0.009$). T = True values and E = Estimated values.

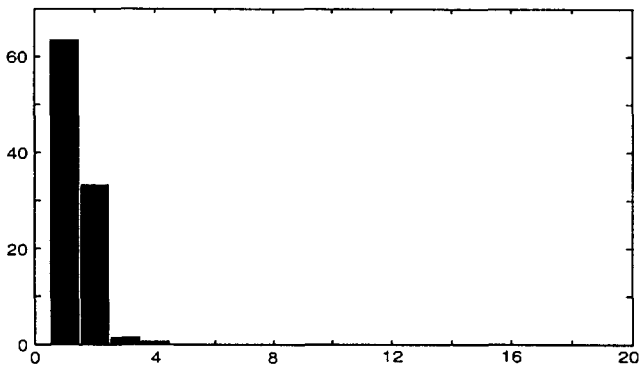

Figure 7.18: Eigenvalues of the covariance matrix form from the estimated parameter values in relative values |eigenvalue|/ $\sum$ eigenvalue $\times$ 100 %.

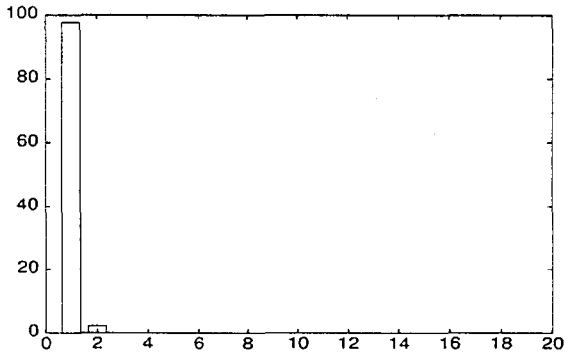

Figure 7.19: Eigenvalues of the covariance matrix form from the estimated parameter values in relative values |eigenvalue|/ $\sum$ eigenvalue $\times$ 100 % and $N = 40000$.

However, by increasing the number of data points in the time series, the relative value of the second eigenvalue decreases as shown in Fig. 7.19. This behavior is similar to the one shown in Fig. 7.12. Assuming a one-parameter reconstruction relative to a_0, the reconstructed BD is shown in Fig. 7.20. The reconstructed BD captures all the important features of the original system.

7.6 Example 3: The *Onchidium* Pacemaker Neuron Model

As a final example, a neuron model that reproduces the discharge pattern of a pacemaker neuron is considered. This pacemaker neuron is localized in the esophageal ganglia of the

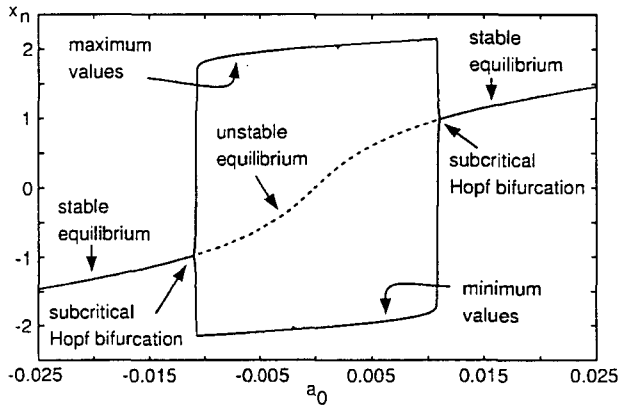

Figure 7.20: Reconstructed BD of the FHN-system as a function of a_0.

marine pulmonate mollusk *Onchidium verruculatum* and the model was proposed by Hayashi and Ishizuka [42]. The 8-variable model is given by

$$\begin{aligned} C_V\frac{dV}{dt} &= g_{Na}m^3h(V-V_{Na}) - g_K n^4(V-V_k) \\ &\quad -g_{Na_s}m_s h_s(V-V_{Na}) \\ &\quad -g_{K_s}n_s(V-V_K) - g_{K_r}n_r(V-V_K) \\ &\quad -g_l(V-V_l) - I_p + I_{ext} \qquad (7.21) \\ \frac{du}{dt} &= \frac{u_\infty(V)-u}{C_f\tau_u(V)}, \quad u = m,h,n \qquad (7.22) \\ \frac{dw}{dt} &= \frac{w_\infty(V)-w}{C_s\tau_w(V)}, \quad w = m_s,h_s,n_s \qquad (7.23) \\ \frac{dn_r}{dt} &= \frac{n_{r\infty}(V)-n_r}{C_r\tau_{n_r}}. \qquad (7.24) \end{aligned}$$

For this case, I_{ext} represents the bifurcation parameter of the system and the other parameters are fixed (see Appendix B). The time series used in the analysis are the maximum values of the membrane potential V. The observed maximum values exhibit period doubling bifurcation followed by the chaotic bursting regime relative to the variation of I_{ext}. The chaotic regime is at the lower bound of the period doubling scheme and the model's firings become irregular except in some I_{ext} windows in which the bursting is periodic with i spikes per period ($i = 5, 7, \cdots$). As the external current is decreased below some critical value, the chaotic bursting is replaced by periodic bursting with three spikes (see Fig. 4a of Maeda, et al. [58]).

Ten time series for $I_{\text{ext}} = -2.25 - 0.02j$, $j = 1, \cdots, 10$ were used in the succeeding analysis. Figure 7.21 shows the multistep prediction error as a function of the number of terms included

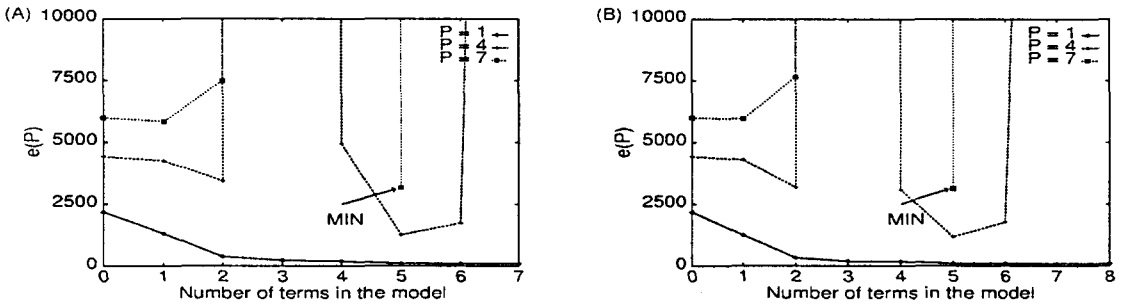

Figure 7.21: Multistep prediction error. (A) Order $k = 7$, and (B) order $k = 8$. The delay $d = 1$. The minimum is indicated by the arrow.

in the model. Different initial values of the delay d and degree k were studied but only the case where $d = 1$, $k = 7$ are shown in [Fig. 7.21 (A)] and $k = 8$ [Fig. 7.21(B)], respectively. The other cases had minimum values occurring at the constant term and diverges when more terms were added. From the figure, five terms are significant. The identified models are:

MODEL 1:

$$y_n^{\text{calc}} = a_0 + a_1 y_{n-1}^7 + a_2 y_{n-1}^6 + a_3 y_{n-1}^5 + a_4 y_{n-1} + a_5 y_{n-1}^4 + \epsilon_n, \tag{7.25}$$

MODEL 2:

$$y_n^{\text{calc}} = a_0 + a_1 y_{n-1}^8 + a_2 y_{n-1}^7 + a_3 y_{n-1}^6 + a_4 y_{n-1} + a_5 y_{n-1}^5 + \epsilon_n. \tag{7.26}$$

Figure 7.22 shows the identified models (solid line) together with the return map (points) of the time series for the chaotic case (left side) and the period three case (right side). The data fitted well with the identified maps. To analyze the bifurcation structure of the original system, the identified models are used. The analysis started with the determination of the projection region associated with the given set of time series. This is done by collecting the estimated coefficients and performing PCA on the resulting parameter vectors.

The significant eigenvalues are shown 7.23. Surprisingly, there is a good agreement between the number of significant eigenvalues of the covariance matrix and the number of system parameters that were changed to generate the time series. Since there is only one significant eigenvalue, the projection region can be determined by the eigenvector associated with this eigenvalue. Any point in the projection region is approximated by

$$\mathbf{a}_{\text{PR}}(\mu) = \mathbf{m} + \mu \mathbf{e}, \tag{7.27}$$

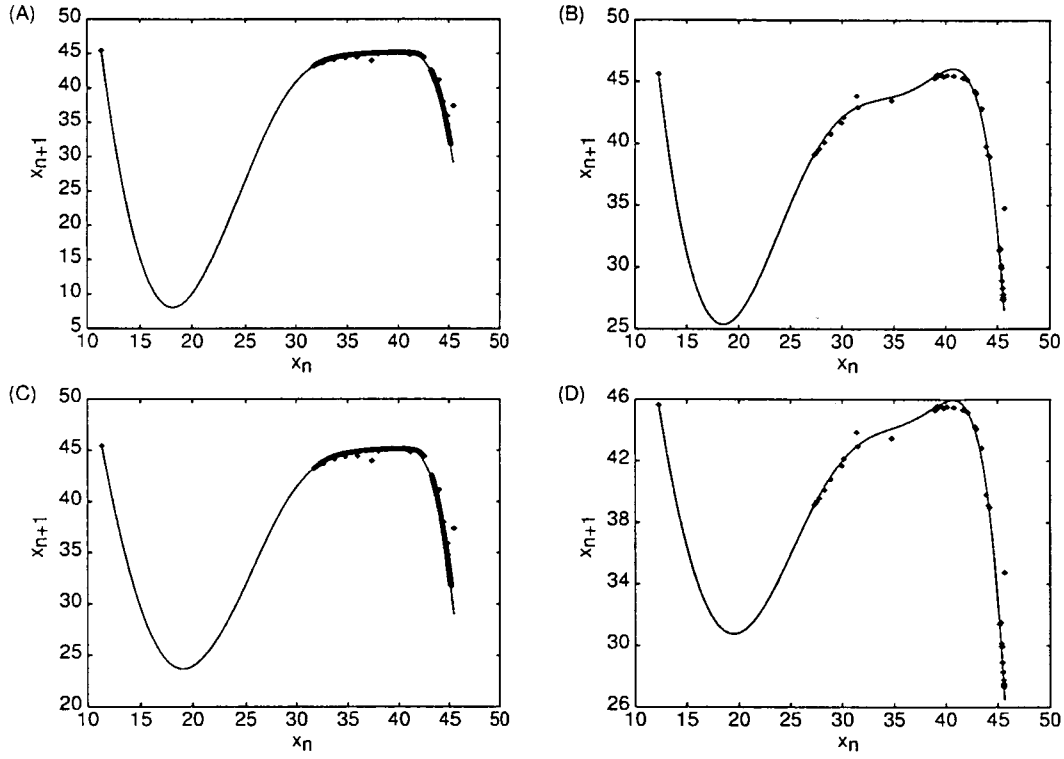

Figure 7.22: Map obtained by the identified model for the given time series. Continuous line: identified map. Discrete points: original data. (A) and (B) MODEL 1. (C) and (D) MODEL 2.

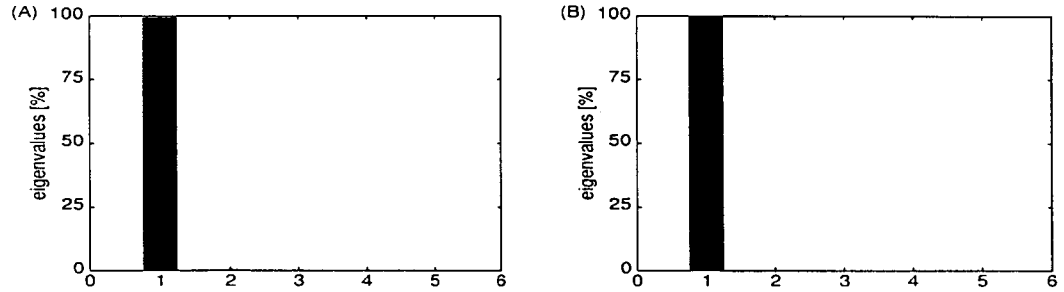

Figure 7.23: Significant eigenvalues (in relative values |eigenvalue|/ $\sum$ eigenvalues $\times$ 100) of the covariance matrix formed by the estimated coefficients $\{\mathbf{a}_{\min}\}$. (A) MODEL 1, and (B) MODEL 2.

where μ serves as the effective bifurcation parameter and $\mathbf{e}$ is the eigenvector of the only significant eigenvalue. To construct the bifurcation diagram, the map is iterated many times, after which the transient behavior is removed from the time series, and the data are plotted as a function of the effective parameter μ. The resulting BDs are shown in Figs. 7.24 (A) and (B). Both BDs show some kind of period-doubling bifurcation route to chaos. Figure 7.24 (B) is closest to the original BD. However, it does not have the period three behavior right after the chaotic regime.

7.7 Discussion

The examples illustrate the different issues in reconstructing bifurcation diagram using time series from dynamical systems driven by noise. One of these issues is the appropriate choice of the approximating function $g(\cdot\,;\mathbf{a})$ that will be used to model the given time series. For the case of using a NAR model such as in Eq. (7.5), it is observed that different models can appear in spite of the fact that all the time series come from the same dynamical system. Since competing models describe individual time series only, some of them may perform badly (for example, in a one-step prediction task) when used with the other time series. This can be used as a criterion in selecting an appropriate model for the unknown dynamical system and is the basis of the criterion discussed in section 7.2 for model selection.

After an appropriate model is obtained, it remains to be determined which among the model parameters represents the actual bifurcation parameters or the system parameters that changed from one time series to another. To resolve this, the PCA of the estimated parameter values was used. The number of significant eigenvalues determines the number of system parameters that changed from one time series to another if the linear manifold approximation holds and the variation of the system parameters are larger than the model parameter estimation error. When the variation are less than the estimation error, PCA will detect the estimation error instead. Also, when the projection region is a nonlinear manifold, the number of significant eigenvalues will exceed the actual number of bifurcation parameters. For a more general derivation, the nonlinear methods discussed in chapter 6 can be applied.

The performance of the algorithm as described in the preceding sections was also demonstrated in several other dynamical systems. The algorithm worked equally well with the Hénon map, the cubic map, and the logistic map. For these systems which are described by polynomial equations, the algorithm determined the correct terms in the polynomial. For continuous

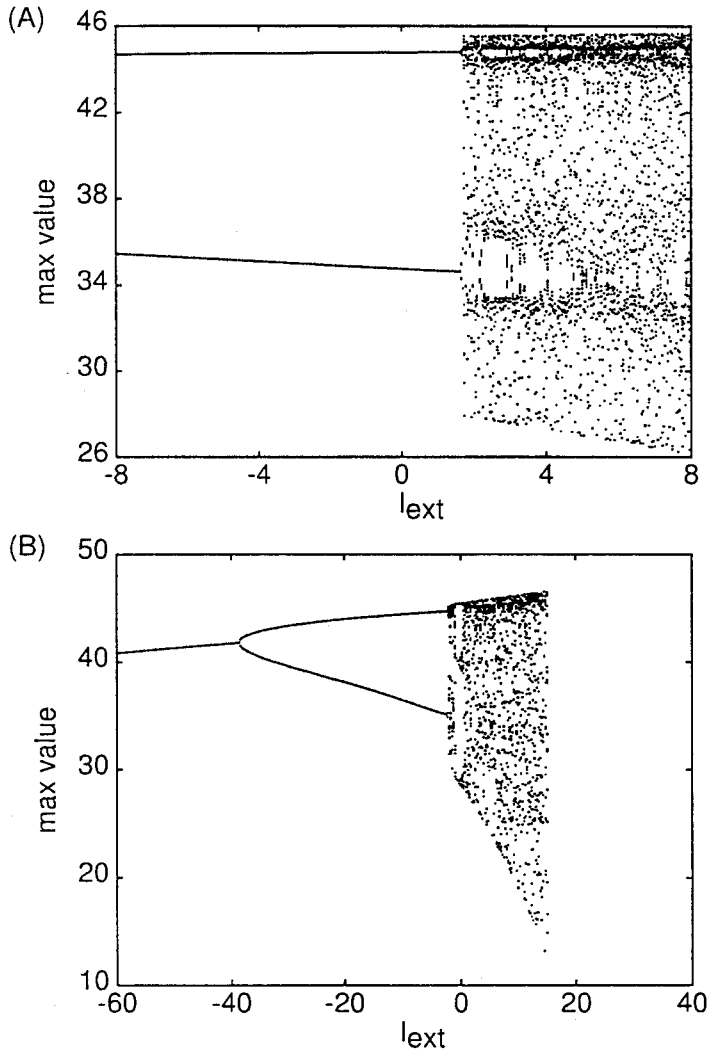

Figure 7.24: Reconstructed bifurcation diagram of the OPN model using the identified models. (A) MODEL 1, and (B) MODEL 2.

systems such as the FHN, the algorithm preserved the different bifurcations of the given system in the reconstructed BD.

The use of NAR models in this reconstruction algorithm is more advantageous than the NN-based approach. The efficacy of the NAR model has been demonstrated in a variety of problems, particularly in the analysis of noisy time series[4, 23, 52]. The NAR model has been applied effectively in obtaining predictor functions for a number of systems (maps and flows), detecting nonlinearities in noisy time series (observation and dynamical noise), estimation of dynamical invariants, among others. Furthermore, NAR models with appropriate number of terms can also capture bifurcation structures as shown in [4]. Aside from polynomials, NAR models can also have other basis functions.

The algorithm is also computationally efficient in a number of ways. In obtaining predictor functions using Korenberg's scheme, the problem of multiparameter optimization is eliminated by employing auxiliary polynomials which are orthogonal with respect to the natural invariant measure of the time series[1, 34]. With this, the parameters are readily obtained from the time series. This scheme also leads to robust-to-noise estimation of the parameters since no distances in the reconstructed state space need to be computed. Moreover, the construction of parsimonious models becomes possible since the contribution of each orthogonal term in reducing the error function can be computed from the time series.

Finally, the effect of noise can be deduced from the results. Noise enhances the detection of the hidden nonlinear structure of the unknown dynamical system. This can be seen in the FHN-system example. In the latter, the addition of noise generates some spikes which cannot be observed in the absence of the random forcing. This positive effect of noise, together with Korenberg's algorithm, makes possible the determination of the nonlinear structure of the unknown system from the observed time series. This in turn improves the obtained model which is critical in the reconstruction process.

In summary, an algorithm in reconstructing BDs from noisy time series is presented. The algorithm consists in finding a parametrized predictor function whose bifurcation structure is qualitatively similar to that of the given system. To account for the effects of noise, NAR models are used as predictor functions. The use of Korenberg's algorithm makes possible the construction of parsimonious models which is advantageous in the reconstruction problem. The algorithm is robust to noise making it more suitable when dealing with noisy time series. Moreover, the algorithm also works well even for a limited number of time series.

Chapter 8

Conclusions

The study of bifurcations from time series was systematically investigated. Several dynamical systems with well-known bifurcation diagrams were included in the simulations. Time series at different parameter values were used to reconstruct the BDs of these systems. The reconstruction assumed that the values, or even the number of parameters, were unknown. The approach used in this study is based on a geometrical interpretation of the reconstruction problem. In this viewpoint, a region in the model's parameter space exhibits the same bifurcation structure as that of the given system. This region is referred to as the projection region. For parameter values within this region, the model's behavior is similar to that of the system. The available time series are then used to determine the location of the projection region in the model's parameter space.

The above interpretation leads to methods of reconstructing bifurcation diagrams from time series. With this, several reconstruction algorithms were formulated depending on the projection region. For linear manifolds, the PCA-based approach is effective in obtaining the BD of the original system. On the other hand, for more general cases (nonlinear manifolds), principal curves can be used for one-dimensional BD reconstruction. In general, any algorithm that solves the BD reconstruction problem will be based on the following steps: 1) choosing the appropriate family of predictor functions; 2) projecting the given time series onto the parameter space; 3) identifying the projection region using the estimated parameter values; and 4) constructing the bifurcation diagram of the predictor function on the projection region. The resulting bifurcation diagram can then be taken as the reconstruction of the bifurcation structure of the unknown system. Each step is associated with some degree of difficulty and the quality of the reconstructed BD depends on the information known *a priori*.

Two classes of predictor functions were chosen for the simulations: neural networks (NN) and nonlinear autoregressive (NAR) models. For the NN-based approach, a three-layer feedforward architecture was employed. The networks were trained using backpropagation and minimization of the error function was done using BFGS algorithm. For the NAR-based approach, the space of candidate terms was set by specifying initial delay d and order k [refer to Eq. (3.56)]. The model coefficients were estimated using the fast orthogonal approach of Korenberg. For both models, the principal components of the estimated parameters determined the projection region when the linear manifold approximation holds. The principal curve approximated the projection region for one-dimensional nonlinear manifold. Several issues appeared in the course of the simulations and these are summarized in the following paragraphs.

The choice of the predictor function. One of the important issues in reconstructing BDs using time series is the appropriate choice of the predictor function $g(\cdot\,;\mathbf{a})$. The success of any reconstruction algorithm depends on the structure of the chosen $g(\cdot\,;\mathbf{a})$. The main requirement for the choice of $g(\cdot\,;\mathbf{a})$ is the presence of projection regions in its parameter space. Under this condition, the use of neural networks in the reconstruction problem comes naturally. Neural networks are nonlinear and known for being universal approximators. Moreover, the number of parameters in the network can be adjusted by simply changing the number of hidden units without affecting the number of input and output units. Thus, in this higher-dimensional weight space, it is safe to assume that there exists a region where the network's bifurcation structure is the same as that of the given system.

Nonlinear autoregressive models are more suitable when dealing with time series corrupted by noise. The coefficients (model parameters) can be computed with the help of auxiliary orthogonal polynomials instead of the traditional least-squares approach through an optimization procedure. This does not only speed up the computation but also makes the approach robust to noise. Moreover, the approach made possible the construction of parsimonious model by including only the terms that reduced the error function significantly. With this, the model's parameter space can be constructed progressively, adding extra dimension only when required.

The choice of predictor functions is not limited to these two models. For instance, radial basis functions may do as well in the reconstruction process. The algorithms described in this study can be applied using other functions as long as appropriate projection regions exist.

The projection region. The problem of obtaining the projection region from the computed model parameters $\{\mathbf{a}^i_{\text{min}}\}$ is related to the problem of finding lower dimensional manifolds in a high dimensional space which can be solved using several well-established approaches. Thus when the projection region is a nonlinear curve in the parameter space of the model, principal curves can be employed to approximate this region. This approach was considered in section 6.2 for one-dimensional BD reconstruction. For higher dimensional cases, this approach is generalized as principal surfaces, nonlinear principal component analysis, bottle-neck neural networks, among others.

In many situations, the projection region is well-approximated by a linear subspace of the model's parameter space. This is the case when dealing with small parameter regions, reconstructing specific bifurcations, or the given system is an LIP map. Under these conditions, PCA provides a computationally efficient method to determine a satisfactory approximate of the projection region. The dimension of the projection region corresponds to the number of system parameters. This can be obtained from the number of significant eigenvalues of the covariance matrix formed by the estimated model parameters. Moreover, the eigenvectors associated to the significant eigenvalues span the required projection region.

There are two possible problems in obtaining the projection region using $\{\mathbf{a}^i_{\text{min}}\}$. First, it is possible that the $\{\mathbf{a}^i_{\text{min}}\}$ will be scattered in the model's parameter space instead of being localized on the desired projection region. This is especially true when several regions in the parameter space have similar bifurcation structures. Several cases were encountered during simulations when dealing with stable fixed point solutions of the dynamics. The time series, in this case, contained no information about the dynamics of the system except during transient. Moreover, several regions in the model's parameter space have fixed points and thus, the obtained $\mathbf{a}^i_{\text{min}}$ did not locate the desired projection region. This situation can be avoided by using the sequential approach, as discussed in chapter 6, in obtaining the $\mathbf{a}^i_{\text{min}}$s. Moreover, one should start with the time series exhibiting the most irregular behavior so that the simpler ones, such as fixed point solutions, will be projected in neighboring regions.

A second possible problem is when the projection region is not compact in the parameter space but rather exists in patches of smaller regions. This situation can occur even though the cost function is minimized for all of the time series in $\{Y^i_\tau\}$. This simply means that the different behaviors of the original system are projected at different locations in the model's parameter space. In this case, the projection region is not appropriate and the model should be changed.

Effects of noise. Another important issue in the reconstruction problem concerns the effects of both observation and system noise. Measurement noise can introduce errors in the estimation of the model parameters. This can be viewed as a displacement of $\{\mathbf{a}^i_{\min}\}$ from their ideal locations by an amount which may depend on the noise intensity. However, the displaced $\{\mathbf{a}^i_{\min}\}$ will still remain in the same bifurcation region as the noiseless case. The effect of this in approximating the projection region can be minimized by preprocessing the data to suppress measurement noise.

For dynamical noise, the situation is quite different. Its presence can cause drastic changes in the values of the associated model parameters especially near bifurcation points. This, in turn, will affect the location of $\{\mathbf{a}^i_{\min}\}$ in the parameter space. The displaced $\{\mathbf{a}^i_{\min}\}$ may be located in other bifurcation regions. This will introduce additional difficulties in obtaining an appropriate projection region using the estimated model parameters. To avoid these difficulties, parameter estimation techniques which are robust to noise should be used. In this study, this is achieved by using Korenberg's fast orthogonal algorithm in estimating the NAR coefficients.

An illustration is shown in Fig. 8.1. A circle in the system's parameter space remains a circle in the projection region in the absence of noise [Fig. 8.1 (B & C)]. In the presence of dynamical noise, the estimated weights for each trained network are scattered in the weight space as can be seen in their projections onto the two principal eigenvectors [Fig. 8.1 (E)]. The projection region is ill-defined for the NN model. On the other hand, the NAR-based approach still obtained the correct projection region as shown in Fig. 8.1 (E).

This study also opens new interesting problems. First, a rigorous investigation of the type of bifurcations that can be reconstructed in the presence of perturbations is necessary. Some types of bifurcations are not persistent when the system is perturbed and their detection from time series may be more difficult. The reconstruction of these bifurcations may require extra information. An insight from the results is that Hopf bifurcations were easily reconstructed as compared to pitchfork and homoclinic bifurcations. A detailed study of this problem can reveal the possible bifurcations the present algorithm can detect from the time series. Second, determining the functional relationship between the two sets of parameters (system and model) is also interesting and worth exploring. In the present case, this is possible when the number of system parameters and their values are known and the linear manifold approximation holds. When these two conditions are met, the model parameters can be expressed as a linear combination of the system parameters and vice versa. Bifurcation points can be predicted and localized in

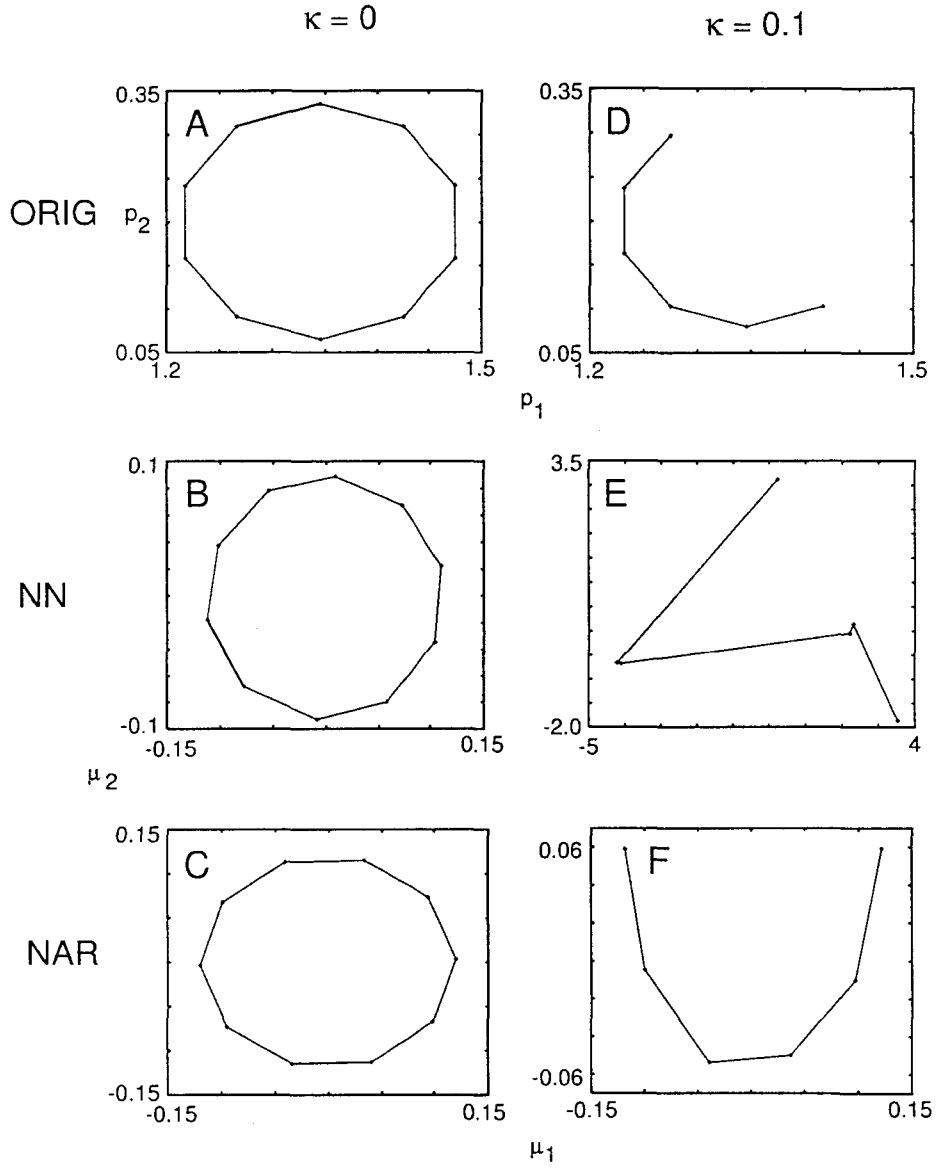

Figure 8.1: Parameter sets used in generating the time series (A & D). Projections of the estimated model parameter values onto the two principal eigenvectors for NN-based (B & E) and NAR-based (C & F).

the parameter space of the system rather than that of the model. Finally, the extension of the current approach to higher dimensional nonlinear cases remains an open problem. The study of this problem is important to address cases when the PCA-based approach is insufficient. The relevant questions are: 1) how to get the dimension of the nonlinear manifold; and 2) how to express this manifold in terms of the discrete points obtained from the time series. The PCA can provide an upper bound for the dimension of the nonlinear manifold. If the dimension is known, the principal curve and the principal surface of the discrete points can represent the required one- and two-dimensional manifold, respectively.

The numerical results of the study were quite remarkable. Even for a limited number of time series (or equivalently, a few parameter values), the BD of the system was reconstructed over a relatively wider parameter region. The reconstructed BD captured the important features of the bifurcation structure of the system. The period doubling of the maximum potential of a pacemaker neuron, the two subcritical Hopf bifurcations of the FitzHugh-Nagumo equations, the period doubling and period doubling reversal of the sine map, the pitchfork bifurcation of the Lorenz equations and the cubic map, and the two-parameter bifurcation structure of the Hénon map, are few examples of the type of bifurcations recovered using the algorithm. The reconstruction algorithm also worked well even when the time series were corrupted by observation or dynamical noise. The success of the simulations promises potential practical applications especially in the analysis of systems where first principle modeling is extremely difficult if not impossible.

With the BD reconstructed, it is easy to know the sequence of bifurcations the system undergoes as the parameters are changed. Unobserved behaviors can be determined by simply examining the reconstructed BD. Critical parameter values become apparent and the system's dependence on its parameters are completely unfolded. All these information are drawn directly from the time series measured at different parameter values, thereby removing the difficult task of constructing differential equations that describe the dynamics of the system. This makes the BD reconstruction algorithm an important tool to uncover the underlying mechanism of the system under study from time series.

Acknowledgment

I would like to acknowledge all the help and encouragement I received from the following persons during the development and towards the completion of this work:

Prof. Shunsuke Sato — for giving me the opportunity to study in his laboratory at Osaka University under the Monbusho scholarship, for his constant supervision in my research, and for his unfailing support during my stay in Japan;

Prof. Toshimitsu Ushio and Prof. Hajime Maeda — for serving as judges for the thesis panel, for their valuable time in reviewing the thesis, and critical evaluation of this work;

Dr. Taishin Nomura — for introducing me to the problem of reconstructing bifurcation diagrams using time series, for being the first reviewer of my papers and spending many hours for discussion in refining each paper; and also, for being a member of the thesis panel;

Dr. Khashayar Pakdaman — for showing me the rigorous way of doing research, for all the precious hours spent reviewing and correcting my papers including this thesis, for all the valuable suggestions in writing styles, and for sharing his infectious vigor for research;

Dr. Alberto Capurro — for the EOD data of the electric fish *Gymnotus carapo*;

Members of the Sato Laboratory — past and present members, for their assistance in my day-to-day life inside and outside the lab, especially for their patience in helping me learn the Japanese language;

My Nanay — for her constant prayers, love, and encouragement;

and last but not the least, Katherine D. Develos — for encouraging me in my graduate study here in Japan, and above all, for serving as my inspiration not only in my research but my whole life as well; this is for you, Meow-dudes.

Appendix A

Recursion Relations

In this section, the recursive relations in estimating α_{mr}, N_m, and b_m are derived. The derivation is based on the orthogonal property Eq. (3.51) and other relations established in Chap. 3, Sec. 3.4.

To compute α_{mr} iteratively, one can use Eq. (3.59) to rewrite Eq. (3.60), that is,

$$\alpha_{mr} = \frac{1}{N_r}\int d^dX\rho(X)z_m(X)w_r(X), \tag{A.1}$$

$$= \frac{1}{N_r}\int d^dX\rho(X)z_m(X)\left[z_r(X)-\sum_{j=0}^{r-1}\alpha_{rj}w_j(X)\right], \tag{A.2}$$

$$= \frac{1}{N_r}\left[\int d^dX\rho(X)z_m(X)z_r(X)-\sum_{j=0}^{r-1}\alpha_{rj}\int d^dX\rho(X)z_m w_j(X)\right], \tag{A.3}$$

$$= \frac{1}{N_r}\int d^dX\rho(X)z_m(X)z_r(X)-\frac{1}{N_r}\sum_{j=0}^{r-1}\alpha_{rj}\alpha_{mj}N_j, \tag{A.4}$$

$$\alpha_{mr} = \frac{\beta_{mr}}{N_r}-\sum_{j=0}^{r-1}\alpha_{mj}\alpha_{rj}\frac{N_j}{N_r}, \tag{A.5}$$

where

$$\beta_{mr} = \int d^dX\rho(X)z_m(X)z_r(X). \tag{A.6}$$

In the same manner, one can compute N_m iteratively by using Eq. (3.59) to rewrite Eq. (3.61) in the form

$$N_m = \int d^dX\rho(X)[w_m(X)]^2. \tag{A.7}$$

$$= \int d^dX\rho(X)\left[z_m(X)-\sum_{r=0}^{m-1}\alpha_{mr}w_r(X)\right]^2, \tag{A.8}$$

$$= \int d^dX\rho(X)\left\{[z_m(X)]^2-2z_m\sum_{r=0}^{m-1}\alpha_{mr}w_r(X)+\left[\sum_{r=0}^{m-1}\alpha_{mr}w_r(X)\right]^2\right\}, \tag{A.9}$$

$$= \int d^dX\rho(X)\left\{[z_m(X)]^2 - 2\sum_{r=0}^{m-1}\alpha_{mr}z_m(X)w_r(X) + \sum_{r=0}^{m-1}\sum_{r'=0}^{m-1}\alpha_{mr}\alpha_{mr'}w_r(X)w_{r'}(X)\right\},$$

$$= \int d^dX\rho(X)[z_m(X)]^2 - 2\sum_{r=0}^{m-1}\alpha_{mr}\int d^dX\rho(X)z_m(X)w_r(X)$$

$$+ \sum_{r=0}^{m-1}\sum_{r'=0}^{m-1}\alpha_{mr}\alpha_{mr'}\int d^dX\rho(X)w_r(X)w_{r'}(X), \tag{A.10}$$

$$= \int d^dX\rho(X)[z_m(X)]^2 - 2\sum_{r=0}^{m-1}\alpha_{mr}\alpha_{mr}N_r + \sum_{r=0}^{m-1}\sum_{r'=0}^{m-1}\alpha_{mr}\alpha_{mr'}N_r\delta_{rr'}, \tag{A.11}$$

$$= \int d^dX\rho(X)[z_m(X)]^2 - \sum_{r=0}^{m-1}\alpha_{mr}^2N_r, \tag{A.12}$$

$$N_m = \beta_{mm} - \sum_{r=0}^{m-1}\alpha_{mr}^2N_r, \tag{A.13}$$

where Eqs. (3.60) and (3.51) are used in deriving line (A.11) from line (A.10).

Finally, to get the equation for b_m, Eqs. (3.62) and (3.59) are used; that is,

$$b_m = \frac{1}{N_m}\int d^dX\rho(X)g(X;\mathbf{a})w_m(X), \tag{A.14}$$

$$= \frac{1}{N_m}\left\{\int d^dX\rho(X)g(X;\mathbf{a})\left[z_m(X) - \sum_{r=0}^{m-1}\alpha_{mr}w_r(X)\right]\right\}, \tag{A.15}$$

$$= \frac{1}{N_m}\int d^dX\rho(X)g(X;\mathbf{a})z_m(X) - \frac{1}{N_m}\sum_{r=0}^{m-1}\alpha_{mr}\int d^dX\rho(X)g(X;\mathbf{a})w_r(X), \tag{A.16}$$

$$= \frac{1}{N_m}\int d^dX\rho(X)g(X;\mathbf{a})z_m(X) - \frac{1}{N_m}\sum_{r=0}^{m-1}\alpha_{mr}b_rN_r, \tag{A.17}$$

$$b_m = \frac{\gamma_m}{N_m} - \sum_{r=0}^{m-1}\alpha_{mr}b_r\frac{N_r}{N_m}, \tag{A.18}$$

where

$$\gamma_m = \int d^dX\rho(X)g(X;\mathbf{a})z_m(X). \tag{A.19}$$

To summarize, the following iterative equations are obtained:

$$\alpha_{mr} = \frac{\beta_{mr}}{N_r} - \sum_{j=0}^{r-1}\alpha_{mj}\alpha_{rj}\frac{N_j}{N_r}, \tag{A.20}$$

$$N_m = \beta_{mm} - \sum_{r=0}^{m-1}\alpha_{mr}^2N_r, \tag{A.21}$$

$$b_m = \frac{\gamma_m}{N_m} - \sum_{r=0}^{m-1}\alpha_{mr}b_r\frac{N_r}{N_m}, \tag{A.22}$$

where

$$\beta_{mr} = \int d^dX\rho(X)z_m(X)z_r(X), \tag{A.23}$$

$$\gamma_m = \int d^dX\rho(X)g(X;\mathbf{a})z_m(X). \tag{A.24}$$

The above integrals can be evaluated by applying the ergodic theorem:

$$\beta_{mr} = \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N} z_m(Y_n) z_r(Y_n), \tag{A.25}$$

$$\begin{aligned} \gamma_m &= \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N} g(Y_n; \mathbf{a}) z_m(Y_n), \\ &= \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N} y_{n+1} z_m(Y_n). \end{aligned} \tag{A.26}$$

Appendix B

Activation/Inactivation Functions

$$
\begin{aligned}
N_\infty(V) &= \alpha_N(V)/[\alpha_N(V) + \beta_N(V)] \\
\tau_N(V) &= 1/[\alpha_N(V) + \beta_N(V)]
\end{aligned}
$$

where $N = m,\ h,\ n,\ m_s,\ h_s,\ n_s,$ and n_r. Rate constants are as follows:

$$
\begin{aligned}
\alpha_m(V) &= 0.1(20+V)/[1-\exp\{-(20+V)/10\}] \\
\alpha_h(V) &= 0.07\exp\{-(V+45)/20\} \\
\alpha_n(V) &= 0.01(20+V)/[1-\exp\{-(20+V)/10\}] \\
\alpha_{m_s}(V) &= 0.1(26+V)/[1-\exp\{-(26+V)/10\}] \\
\alpha_{h_s}(V) &= 0.07\exp\{-(V+51)/20\} \\
\alpha_{n_s}(V) &= 0.01(50+V)/[1-\exp\{-(50+V)/10\}] \\
\alpha_{n_r}(V) &= 0.01(100+V)/[\exp\{(100+V)/10\}-1] \\
\beta_m(V) &= 4\exp\{-(V+45)/18\} \\
\beta_h(V) &= 1/[1+\exp\{-(15+V)/10\}] \\
\beta_n(V) &= 0.125\exp\{-(V+30)/80\} \\
\beta_{m_s}(V) &= 4\exp\{-(V+51)/18\} \\
\beta_{h_s}(V) &= 1/[1+\exp\{-(21+V)/10\}] \\
\beta_{n_s}(V) &= 0.125\exp\{-(V+60)/80\} \\
\beta_{n_r}(V) &= 0.125\exp\{-(V+90)/80\}]
\end{aligned}
$$

Maximum conductances values: $g_{Na} = 60[\mu S]$, $g_K = 10[\mu S]$, $g_{Na_s} = 1.40[\mu S]$, $g_{K_s} = 0.18[\mu S]$ $g_{K_r} = 0.20[\mu S]$, and $g_l = 0.063[\mu S]$. Values of equilibrium potentials of channels: $V_{Na} = 50[mV]$

$V_K = -70[mV]$, and $V_l = -70[mV]$. Values of the membrane capacitance: $C_V = 20[nF]$. Values of dimensionless constants: $C_f = 10$, $C_s = 100$, and $C_r = 800$.

Appendix C

Experimental Preparation

Five specimens of *Gymnotus carapo*, gathered from Laguna del Sauce a lake in Departamento de Maldonado in Southeastern Uruguay, of a length ranging from 7 to 10 cm were selected for the experiments. The fish were placed in separate tanks. Each tank contained a plastic tube open at both ends, with carbon electrodes to record the EOD. The tube was the preferred position of the fish during most of the light phase (which is the resting phase) of the circadian cycle, so that no constraint was necessary to make the animal enter into the tube or remain in there. All experimental procedures were performed during the light phase of the circadian cycle, when the variability in the duration of the intervals is lowest. The voltage of the EOD was digitally recorded at equal time intervals of 0.05 milliseconds (i.e. sample frequency of 20 kHz) and saved as ASCII files with the value of time in the first column and voltage in the second column. A C program calculated the EOD discharging times and the intervals between successive discharges from the digital file of the EOD recording. This program wrote the discharge times and the EOD interval that ended in that discharging time in two columns of an ASCII file. We checked that the appreciation in the value of the discharge times was the same as the sample interval of the voltage vs time file that was directly recorded from the fish.

Since novel stimuli (such as weak mechanical vibrations or electric fields) as well as small movements of the fish induce transient frequency increases in the EOD (e.g. Capurro et al., 1994, and references therein), we controlled visually that the fish was at rest during the recordings i.e. even small macroscopic movements of the fins were absent. Furthermore, the recordings were done in a silent room with no changes in the lighting level and using an anti-vibration table. Digital recordings contaminated by electric fields due to external sources of induced current were discarded.

Bibliography

[1] Abarbanel HDI, Brown R, Sidorowich JJ, and Tsimring L (1993) The analysis of observed chaotic data in physical systems. Rev. Mod. Phys. 65: 1331–1392.

[2] Acosta FMA (1995) Radial basis function and related models: An overview. Signal Processing 45: 37–58.

[3] Admoatis RA, Farber RM, Hudson JL, Kevrekidis IG, Kube M, and Lapedes AS, Application of neural nets to system identification and bifurcation analysis of real world experimental data, In: *Neural Networks: Biological Computers or Electronic Brains* (Springer Verlag, Paris, 1990) 87–97.

[4] Aguirre L and Billings SA (1995) Retrieving dynamical invariants from chaotic data using NARMAX models. Int. J. Bifurcation and Chaos 5: 449–474.

[5] Aguirre L and Mendes E (1996) Nonlinear polynomial models: Structure, term clusters and fixed points. Int. J. Bifurcation and Chaos 6: 279–294.

[6] Akaike H (1974) A new look at the statistical model identification. IEEE Trans. Automat. Control. AC-19: 716–723.

[7] Bagarinao E, Nomura T, Pakdaman K, and Sato S (1998) Generalized one-parameter bifurcation diagram reconstruction using time series. Physica D 124: 258–270.

[8] Bagarinao E, Pakdaman K, Nomura T, and Sato S (1999) Time series-based bifurcation diagram reconstruction. Physica D 130: 211–231.

[9] Bagarinao E, Pakdaman K, Nomura T, and Sato S (1999) Reconstructing bifurcation diagrams from noisy time series using nonlinear autoregressive models. Phys. Rev. E 60: 1073–1076.

[10] Bagarinao E, Pakdaman K, Nomura T, and Sato S. Reconstructing bifurcation diagrams of dynamical systems using measured time series. To appear in a special issue of Methods of Information in Medicine.

[11] Barahona M and Poon C (1996) Detection of nonlinear dynamics in short, noisy time series. Nature (London) 381: 215–217.

[12] Bastian J (1986) Electrolocation: behavior, anatomy and physiology, In: Bullock TH and Heiligenberg W (Eds), *Electroreception* (Wiley, New York, 1986) 577–612.

[13] Berliner LM (1991) Likelihood and Bayesan prediction of chaotic systems. J. Amer. Stat. Soc. 86: 938–952.

[14] Bishop C (1994) Neural networks and their applications. Rev. Sci. Instrum. 65: 1803–1832.

[15] Black-Cleworth P (1970) The role of electrical discharges in the non-reproductive social behavior of *Gymnotus carapo* (Gymnotidae, Pisces). Anim. Behav. Monogr. 31: 1–77.

[16] Box GEP, Jenkins GM, and Reinsel GC (1994) *Time series analysis forecasting and control*, Prentice-Hall, Inc. Englewood Cliffs, NJ.

[17] Broomhead DS and King GP (1986) Extracting qualitative dynamics from experimental data. Physica D 20: 217–236.

[18] Broomhead B and Lowe D (1988) Multi-variable functional interpolation and adaptive networks. Complex Systems 2: 269–303.

[19] Brown R, Bryant P, and Abarbanel HDI (1991) Computing the Lyapunov spectrum of a dynamical system from an observed time series. Phys. Rev. A 43: 2787–2806.

[20] Capurro A, Reyes-Parada M, Ardanaz JL, Silveira R, and Macadar O (1994) Serotonergic control of electric organ discharge in *Gymnotus carapo*: Role of 5-HT2A/2c receptor subtypes. Comp. Biochem. Physiol A 109A: 583–591.

[21] Capurro A, Pakdaman K, Perrone R, and Macadar O (1999) Analysis of the jamming avoidance response in the electric fish *Gymnotus carapo*. Biol. Cybern. 80: 269–283.

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

[23] Chon KH, Kanters JK, and Holstein-Rathlou NH (1997) Detection of chaotic determinism in time series from randomly forced maps. Physica D 99: 471–486.

[24] Cleveland WS (1979) Robust locally weighted regression and smoothing scatterplots. J. Am. Stat. Assoc. 74: 829–836.

[25] Conover WJ (1980) *Practical nonparametric statistics*, John-Wiley and Sons, Inc., New York.

[26] Dong D and McAvoy TJ (1996) Nonlinear principal component analysis – Based on principal curves and neural networks. Computers Chem. Engg 20: 65–78.

[27] Doob J (1953) *Stochastic Processes*, Wiley, New York.

[28] Eckmann JP, Kamphorst SO, Ruelle D, and Ciliberto S (1986) Lyapunov exponents from a time series. Phys. Rev. A 34: 4971–4979.

[29] Falconi A, Lorenzo D, Curti S, Morales F, and Borde M (1997) Mauthner cell-evoked synaptic actions on pacemaker medullary neurons of a weakly electric fish. J. Comp. Physiol. A 181: 143–151.

[30] Farmer JD, Ott E, and York JE (1983) The dimension of chaotic attractors. Physica D 7: 153.

[31] Farmer JD and Sidorowich JJ (1987) Predicting chaotic time series. Phys. Rev. Lett. 59: 845–848.

[32] FitzHugh R (1961) Impulses and physiological states in theoretical models of nerve membrane. Biophys. J. 1: 445–466.

[33] Frost PA and Kailath T (1971) An innovations approach to least-squares estimation – Part III: Nonlinear estimation in white Gaussian noise. IEEE Trans. Automatic Control AC-16: 217–226.

[34] Giona M, Lentini F, and Cimagalli V (1991) Functional reconstruction and local prediction of chaotic time series. Phys. Rev. A 44: 3496–3502.

[35] Gilmore R (1998) Topological analysis of chaotic dynamical systems. Rev. Mod. Phys 70: 1455–1529.

[36] Grassberger P and Procaccia I (1983) Characterization of strange attractors. Phys. Rev. Lett 50: 346–349.

[37] Grassberger P and Procaccia I (1983) Measuring the strangeness of strange attractors. Physica D 9: 189–208.

[38] Grassberger P, Schreiber T, and Schaffrath C (1991) Nonlinear time sequence analysis. Int. J. Bifurcation and Chaos 1: 521–547.

[39] Guckenheimer J and Holmes P (1983) *Nonlinear Oscillations, Dynamical Systems, and Bifurcation of Vector Fields*, Springer-Verlag, New York.

[40] Hastie T and Stuetzle W (1989) Principal curves. J. Am. Stat. Assoc. 84: 502–516.

[41] Hagedorn M (1986) The ecology, courtship, and mating of Gymnotiform Fish, In T. H. Bullock and W. Heiligenberg (Eds), *Electroreception* (Wiley, New York, 1986) 497–526.

[42] Hayashi H and Ishizuka S (1992) Chaotic nature of bursting discharge in the *Onchidium* pacemaker neuron. J. Theor. Biol. 156: 269–291.

[43] Hénon M (1976) A two dimensional mapping with a strange attractor. Commun. Math. Phys. 50: 69.

[44] Hornik K, Stinchcombe M, and White H (1989) Multilayer feedforward networks are universal approximators. Neural Networks 2: 359–366.

[45] Irie B and Kawato M (1991) Acquisition of internal representation by multilayered perceptrons. Elec. and Comm. in Japan 74: 1173–1178.

[46] Jaegar L and Kantz H (1997) Effective deterministic models for chaotic dynamics perturbed by noise. Phys. Rev. E 55: 5234–5247.

[47] Jazwinski A (1970) *Stochastic Processes and Filtering Theory,* Academic Press, New York.

[48] Kaplan DT and Glass L (1992) A direct test for determinism in a time series. Phys. Rev. Lett. 68: 427–430.

[49] Karantonis A and Pagitsas M (1997) Constructing normal forms from experimental observations and time series analysis. Int. J. Bifurcation and Chaos 7: 107–127.

[50] Kennel MB, Brown R, and Abarbanel HDI (1992) Determining embedding dimension for phase space reconstruction using the method of false nearest neighbors. Phys. Rev. A 45: 3403–3411.

[51] Kennel MB and Isabelle S (1992) Method to distinguish possible chaos from colored noise and to determine embedding parameters. Phys. Rev. A 46: 3111–3118.

[52] Korenberg MJ (1988) Identifying nonlinear difference equation and functional expansion representations: the fast orthogonal algorithm. Ann. Biomed. Eng. 16: 123–142.

[53] Kramer M (1991) Nonlinear principal component analysis using autoassociative neural networks. AIChE Journal 37: 233–243.

[54] Lapedes A and Farber R (1988) How neural networks work (Tech. Rep. LA-UR-88-418). Los Alamos, NM: Los Alamos National Library.

[55] Linsay PS (1991) An efficient method of forecasting chaotic time series using linear interpolation. Phys. Lett. A 153: 353–356.

[56] Ljung GM and Box GEP (1978) On a measurement of lack of fit in time series models. Biometrika 65: 297–303.

[57] Lorenz E (1963) Deterministic non-periodic flow. J. Atmos. Sci. 20: 130–141.

[58] Maeda Y, Pakdaman K, Nomura T, and Sato S (1998) Reduction of a model for an *Onchidium* pacemaker neuron. Biol. Cybern. 78: 265–276.

[59] Nagumo J, Arimoto J, and Yoshizawa S (1962) An active pulse transmission line stimulating nerve axon. Proc. IRE 50: 2061–2070.

[60] Neveu J (1965) *Mathematical Foundations of the Calculus of Probability*, Holden - Day, San Francisco.

[61] Ott E, Sauer T, and York J (1994) *Coping with chaos*, John Wiley and Sons, Inc., New York.

[62] Ozaki T (1981) Non-linear threshold autoregressive models for non-linear random vibrations. J. Appl. Prob. 18: 443–451.

[63] Ozaki T. Nonlinear time series models and dynamical systems. In E. Hannan, et al. (Eds.) *Handbook of Statistics* (Amsterdam/New York/London:North-Holland) 25–83.

[64] Ozaki T, Jimenez J, and Haggan-Ozaki V (1996) The role of the likelihood function in the estimation of chaos models. Tech. Rep. 1996/11, Univ. of Manchester/ U.M.I.S.T. Manchester Center for Statistical Science.

[65] Packard NH, Crutchfield JP, Farmer JD, and Shaw RS (1980) Geometry from a time series. Phys. Rev. Lett. 45: 712–716.

[66] Pikovsky AS (1986) Radio Eng. Electron. Phys. 9: 81.

[67] Poggio T and Girosi F (1990) Networks for approximation and learning. Proc. IEEE 78: 1481–1495.

[68] Poon C and Merrill C (1997) Decrease of cardiac chaos in congestive heart failure. Nature (London) 389: 492–495.

[69] Press WH, Vettering WT, Teukolsky SA, and Plannery BP (1992) *Numerical Recipe in C: the art of scientific computing* 2nd Ed. Cambridge University Press.

[70] Rapp PE, Albano AM, Schmah TI, and Farwell LA (1993) Filtered noise can mimic low-dimensional chaotic attractors. Phys. Rev. E 47: 2289–2297.

[71] Rossler OE (1976) An equation for continuous chaos. Phys. Lett. A 57: 397–398.

[72] Sano M and Sawada Y (1985) Measurement of the Lyapunov spectrum from chaotic time series. Phys. Rev. Lett. 55: 1082–1085.

[73] Scholkopf B, Smola A, and Muller K (1998) Nonlinear component analysis as a kernel eigenvalue problem. Neural Computation 10: 1299–1319.

[74] Shoji I and Ozaki T (1994) Estimation for a continuous time stochastic process: A new local linearization approach. Res. Memo. No. 524, The Institute of Statistical Mathematics, Tokyo.

[75] Shoji I and Ozaki T (1995) Comparative study of estimation methods for a continuous time stochastic process. Res. Memo. No. 561, The Institute of Statistical Mathematics, Tokyo.

[76] Silverman BW (1985) Some aspects of spline smoothing approaches to non-parametric regression curve fitting. J. Roy. Stat. Soc. B 47: 1–52.

[77] Sparrow C (1990) *The Lorenz equations: Bifurcation, Chaos, and Strange Attractors*, Springer-Verlag, New York.

[78] Sugihara G and May RM (1990) Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series. Nature (London) 344: 734–741.

[79] Takens F (1980) Detecting strange attractor in turbulence, in Dynamical Systems and Turbulence, Warwick edited by D. Rand and L.S. Young, Lecture Notes in Mathematics No. 898 (Springer, Berlin, 1981) 366–381.

[80] Theiler J (1990) Estimating fractal dimensions. J. Opt. Soc. Am. 7: 1055.

[81] Tokuda I, Kajiwara S, Tokunaga R, and Matsumoto T (1996) Recognizing chaotic time-waveforms in terms of a parametrized family of nonlinear predictors. Physica D 95: 380–395.

[82] Tokunaga R, Kajiwara S, and Matsumoto T (1994) Reconstructing bifurcation diagrams only from time-waveforms. Physica D 79: 348–360.

[83] Troy W (1976) Bifurcation phenomena in FitzHugh's nerve conduction equation. J. Math. Anal. App. 54: 678–690.

[84] Wiggins S (1990) *Introduction to Applied Nonlinear Dynamical Systems and Chaos*, Springer-Verlag, New York.

[85] Wolf A, Swift JB, Swinney HL, and Vastano JA (1985) Determining Lyapunov exponents from a time series. Physica D 16: 285–317.

List of Publications

International journals

1. E. Bagarinao, K. Pakdaman, T. Nomura, and S. Sato, "Reconstructing bifurcation diagrams of dynamical systems using measured time series," To appear in a special issue of Methods of Information in Medicine.

2. E. Bagarinao, K. Pakdaman, T. Nomura, and S. Sato, "Time series-based bifurcation diagram reconstruction," Physica D 130 (1999) 211-231.

3. E. Bagarinao, K. Pakdaman, T. Nomura, and S. Sato, "Reconstructing bifurcation diagrams from noisy time series using nonlinear autoregressive models," Phys. Rev. E 60 (1999) 1073-1076.

4. E. Bagarinao, T. Nomura, K. Pakdaman, and S. Sato, "Generalized one-parameter bifurcation diagram reconstruction using time series," Physica D 124 (1998) 258-270.

5. E. Bagarinao and C. Saloma, "Frequency analysis with the Hopfield encoding neurons," Phys. Rev. E 54 (1996) 5516-5521.

6. C. Saloma and E. Bagarinao, "Profiling the mirror surface from the Fraunhofer amplitude distribution that can be generated under coherent illumination," J. Modern Optics 42 (1995) 1005-1021.

International conference/workshops

1. E. Bagarinao, K. Pakdaman, T. Nomura, and S. Sato, "Reconstructing bifurcation diagrams of dynamical systems using measured time series," The 3rd International Workshop on Biosignal Interpretation 1999, Chicago, USA, June 12–14, 1999.

2. E. Bagarinao, T. Nomura, K. Pakdaman, and S. Sato, "Learning dynamical systems by neural networks," International Conference on Neural Information Processing (ICONIP98), Kita-Kyushu, Japan, October 1998.

Research meetings/workshops

1. E. Bagarinao, T. Nomura, and S. Sato, "Time-series-based bifurcation analysis," in *Mathematical Analysis of Random Phenomena on Science and Engineering - Theories and Applications*, The Institute of Statistical Mathematics Cooperative Research Report 123, pp. 14–23, December 1999.

2. E. Bagarinao, K. Pakdaman, T. Nomura, and S. Sato, "A study on reconstructing bifurcation diagrams using time series from randomly forced maps," in *Mathematical Analysis of Random Phenomena on Science and Engineering - Theories and Applications*, The Institute of Statistical Mathematics Cooperative Research Report 114, pp. 46–55, December 1998.

3. E. Bagarinao, T. Nomura, K. Pakdaman, and S. Sato, "Interpolation of the system behavior from a finite number of time series," in *Analysis of Random Phenomena in Engineering, Science and Medical Science - Modeling and Application*, The Institute of Statistical Mathematics Cooperative Research Report 106, pp. 37–46, January 1998.

4. E. Bagarinao, T. Nomura, K. Pakdaman, and S. Sato, "Some notes on constructing bifurcation diagrams from time series," Nonlinear theory and applications (NOLTA) workshop, pp. 30–33, October 1997.