

## Chapter 16

# Nonlinear system identification using neural networks: dynamics and instabilities

R. Rico-Martínez<sup>†</sup> and I. G. Kevrekidis

*Department of Chemical Engineering, Princeton University, Princeton, NJ 08544, U.S.A., E-mail: ramiro@arnold.princeton.edu, yannis@arnold.princeton.edu*

<sup>†</sup>*Current Address: Instituto Tecnológico de Celaya, Departamento de Ingeniería Química, Av. Tecnológico e Irrigación, Celaya, Gto., 38010 México.*

K. Krischer

*Fritz-Haber-Institut der Max-Planck-Gesellschaft, Faradayweg 4-6, 14195 Berlin, Germany.*

## 1. Introduction

Many systems of interest in chemical engineering are inherently strongly nonlinear. The often complicated dynamic behavior of such systems can be successfully analyzed, characterized and predicted using Artificial Neural Networks (ANNs) to construct empirical or semi-empirical (hybrid) dynamic models (see e.g. [1, 2, 3]). These models can in turn be used as surrogate models for the control and/or optimization of the processes (see e.g. [4, 5, 6]). In this chapter we will study the properties of some representative neural network identification schemes pertaining to the characterization of nonlinear system dynamics. As we will show, the success of such characterization (and thus of the subsequent application of the model constructed) may be compromised by the choice of the type of model selected.

When constructing ANN models from experimental time-series, a basic requirement is short term prediction accuracy. For many applications (e.g. receding horizon control) the constructed model should provide accurate *long term* predictions as well. A final requirement that one may impose on the performance of an

ANN model is the ability to predict the dependence of the system dynamics on one or more operating parameters or, more specifically, the instabilities and transitions that occur as these operating parameters are varied. The successful characterization of nonlinear systems requires satisfactory performance of the ANN model at all three of these levels.

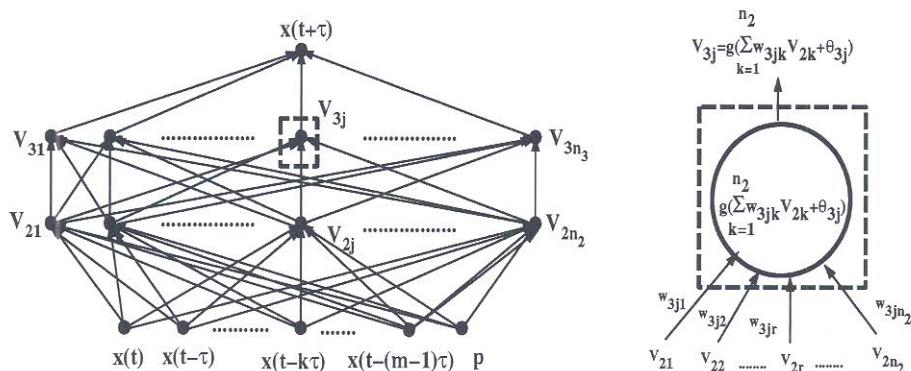
One of the most common approaches towards the identification of linear and nonlinear systems involves the construction of *discrete-time* delay-based mappings. In the ANN context, Lapedes and Farber [7] demonstrated that a class of neural networks (specifically, four-layer feedforward networks) can be used to construct discrete delay-based models with accurate short-term prediction even for systems with (deterministic) chaotic dynamics. By iterating these discrete-time mappings, one obtains predictions of the long-term behavior of the system (its "attractors" in phase space, whether steady, periodic, or more complicated). Comparison of the time-series obtained through such repeated iteration against the true time-series will always show the eventual development of discrepancies. These discrepancies, measured pointwise in time, can be dramatic. In the presence of chaotic dynamics, minute perturbations of the initial conditions can make nearby trajectories diverge exponentially in time. Similar effects are caused by infinitesimal perturbations in the model (e.g. slight fitting errors), and the predicted and real time-series eventually completely disagree when compared point by point. *Nevertheless, the ANN model can still be considered successful.* To see this, one should not consider the pointwise comparison of the predicted and real time-series, but the ultimate *attractors* of the two systems as sets in phase space. If the distance *between the two sets* (real and predicted) is small, measured in the Euclidean sense or some other appropriate norm, we can say that the true long-term dynamics are correctly reproduced by the ANN long-term dynamics.

In this chapter we will adopt this "long-term" approach. We will consistently compare attractors of experimental systems with the attractors of their corresponding ANN models. Qualitative changes (instabilities, bifurcations) of the system attractors will be examined as the operating parameters vary. We will then study how faithful the ANN models are in capturing the nature of these instabilities, as well as their accurate location in parameter space. As we illustrate this view of analyzing and evaluating ANN model predictions, the purpose of this chapter will be to demonstrate some significant inherent shortcomings of *discrete-time* ANN models used in the identification of *continuous-time* systems.

Discrete-time models may be capable of predicting long-term attractors which, as sets in phase space, are located "close" to the true attractors. However, as we will see, the *nature* of the attracting sets, and more importantly, their instabilities and transitions as predicted by "traditional" discrete dynamical models, are *inherently different* from the attractors and transitions of continuous-time systems. In other words, the proximity of the long-term predictions to the experimentally observed attractors *is not enough* for a complete characterization of nonlinear system dynamics. We will demonstrate that certain generic features of discrete-time models, which are absent in continuous-time systems (and models), may lead to

the loss of predictive capabilities of typical discrete-time ANNs. Alternatively, one may attempt to capture the dynamical behavior of experimental observations through the construction of *continuous-time* models (i.e. sets of Ordinary Differential Equations, ODEs). This allows a more consistent characterization of the system dynamical behavior.

In the following sections we start with a description of the methodologies for the construction of ANN-based discrete- (time delay) and continuous-time models from experimental observations. We then illustrate these procedures using experimental data from two systems that exhibit unsteady dynamic behavior. We confine our illustrations to the occurrence of simple periodic behavior for both cases. The experimental data studied involve spatio-temporal oscillations observed during the oxidation of CO on Pt(110) single crystals, and the oscillatory behavior of the electrode potential during the electrochemical oxidation of H<sub>2</sub> on a Pt wire in HClO<sub>4</sub> solutions.



**Figure 1.** Schematic of the four-layer feedforward ANN architecture for the construction of discrete-time dynamical models as delay-based mappings.

The formula to obtain the output of the neurons is shown on the right.

## 2. Discrete-Time Modeling

Throughout this chapter we assume that the dynamical behavior observed experimentally can be captured in a model of the form

$$\dot{\underline{X}} = \underline{G}(\underline{X}; \underline{p}) \quad (1)$$

where  $\underline{X}$  is the vector of states of the system,  $\underline{X} \in \mathcal{R}^n$  ( $n$  is the dimension of the system phase space), and  $\underline{p}$  is the vector of parameters,  $\underline{p} \in \mathcal{R}^p$  ( $p$  is the dimension of the system parameter space). Typically, experimental data from the system come in the form of several sequences of observations ( $\{\underline{x}(t_i; \underline{p})\}, i = 1, \dots, N$ ), where  $\underline{x}$  is a subset of the full vector of states  $\underline{X}$  (it can be only one state, a few, or all of

them). The parameters may be physical parameters (such as kinetic constants), operating parameters (such as flow rates or inlet stream temperatures) as well as control-related parameters (controller settings, such as gains). Without loss of generality, we will consider that each experimental time series has been obtained at fixed parameter values.

A dynamical model of the system, for our purposes, is a formula (or computer subroutine) which, given the value(s) of the observed state(s) of the system, possibly a short history of their measurements and the parameter settings, is capable of predicting the value(s) of the same observed state(s) at a future time. This can be done either through a *discrete-time* mapping of the form

$$\underline{x}(t + \tau) = \underline{F}(\underline{x}(t), \underline{x}(t - \tau), \dots, \underline{x}(t - (m - 1)\tau); \underline{p}) \quad (2)$$

where  $\tau$  is the time interval (delay) between observations, or through a *continuous-time* system of differential equations

$$\dot{\underline{x}} = \underline{f}(\underline{x}; \underline{p}). \quad (3)$$

The map can be evaluated once, or the ODEs numerically integrated for one time step, to provide *short term* predictions. If these predictions are “fed” again as initial conditions to the models, and the map is iterated (or the ODE integrated) *ad infinitum* we obtain predictions of the *long term* system dynamics.

The most common approach to the construction of empirical models from experimental time-series involves the construction of time delay-based mappings of the form of (2). This approach is motivated by the well-known ARMA models in process control and identification [8], and by attractor reconstruction in phase space for nonlinear systems [9, 10]. For a deterministic system, the geometrical concept of attractor reconstruction with a finite number of time delays implies that the future state of a system can be obtained as a (nonlinear) function of the measurement of the present value of a single state variable and an appropriate number of previous (delayed) measurements of the same variable. The number of delayed measurements required to construct this type of mappings, the embedding dimension, is a function of the complexity of the data sequences studied and can be associated with the *dimension* of the attractors of the system to be modeled.

Figure 1 schematically depicts the four-layer feedforward ANN architecture originally proposed by Lapedes and Farber for the identification of nonlinear systems. The figure shows the ANN input and output assignments for the construction of a mapping of the form of (2). On the right we show how to obtain the output of a neuron. The activation function  $g(\cdot)$  is nonlinear for the hidden layers, usually with sigmoidal shape ( $g(Z) = \frac{1}{2}(1 + \tanh(Z))$ ) and for the input and output layers it is just the identity function. These types of ANN models have been shown to exhibit good short-term prediction accuracy and, as we mentioned earlier, may give predictions of the attractors (long-term solutions) very close to the “correct”

ones in the reconstructed phase space. Nevertheless, their discrete-time character severely constrains their applicability for the characterization of the bifurcations (transitions) of nonlinear systems, as we will illustrate below.

Since the pioneering work of Lapedes and Farber, other ANN architectures (such as recurrent networks) have been proposed for the identification of nonlinear systems through the construction of time-delayed mappings (see e.g. [11, 12, 13]). Most of them, however, retain the discrete-time character of the approximation. For this reason, we concentrate our discussion of discrete-time ANN models to the representative four-layer feedforward architecture described here.

### 3. Continuous-Time Modeling

An alternative approach to the construction of discrete-time delay-based models from experimental time-series of continuous systems involves the construction of *continuous-time models* (sets of Ordinary Differential Equations, ODEs).

The task of constructing a continuous-time model capturing the dynamics contained in the time-series is equivalent to approximating the right-hand side of such an ODE set; i.e. we seek a continuous-time model of the form

$$\dot{\underline{y}} = \underline{f}(\underline{y}; \underline{p}) \quad (4)$$

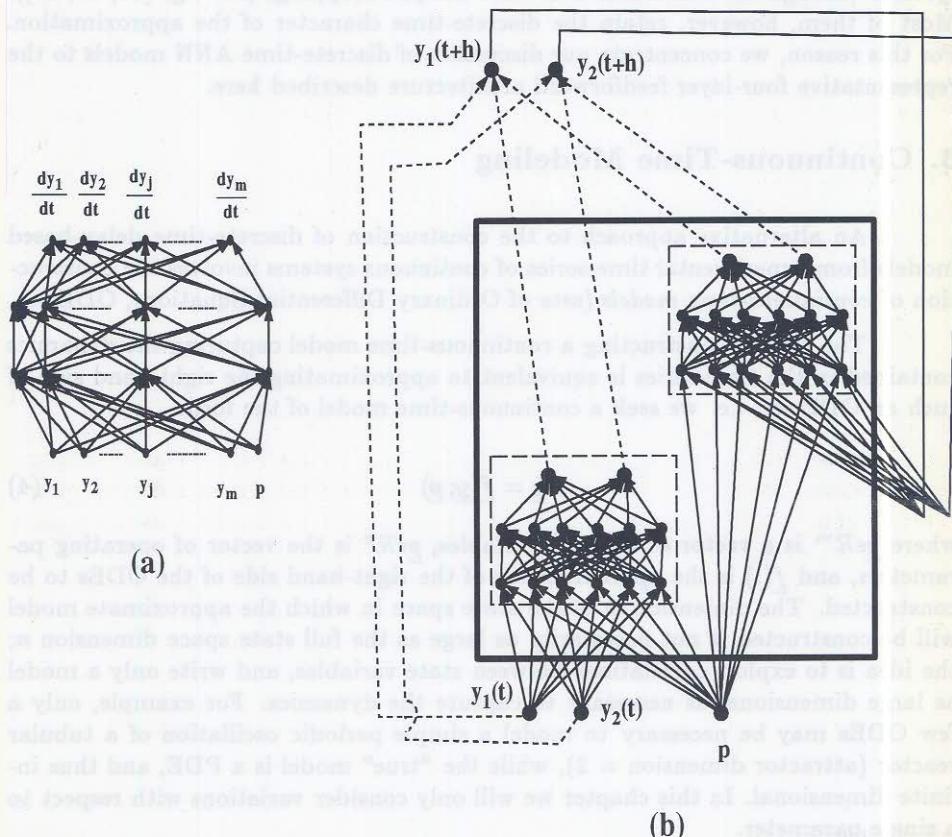
where  $\underline{y} \in \mathcal{R}^m$  is a vector of  $m$  state variables,  $\underline{p} \in \mathcal{R}^p$  is the vector of operating parameters, and  $\underline{f}(.)$  is the approximation of the right-hand side of the ODEs to be constructed. The dimension  $m$  of the state space in which the approximate model will be constructed is not necessarily as large as the full state space dimension  $n$ ; the idea is to exploit correlations between state variables, and write only a model as large dimensional as necessary to capture the dynamics. For example, only a few ODEs may be necessary to model a simple periodic oscillation of a tubular reactor (attractor dimension = 2), while the “true” model is a PDE, and thus infinite dimensional. In this chapter we will only consider variations with respect to a single parameter.

Thus our efforts are directed towards approximating the right-hand side of a system of the form of (4), using experimental measurements of the state variables only (i.e., without numerical evaluation of time derivatives). We can consider the sequence of measurements ( $\{\underline{x}(t_i; \underline{p})\}$ ,  $i = 1, \dots, N$ ) as the result of numerically integrating an unknown system of  $n$  ODEs for the appropriate time-step. The model sought can be constructed by “embedding” the training of a four-layer feedforward ANN (approximating the right-hand-side of the ODEs) in a numerical integrator scheme [14, 15, 16, 17]. We chose the four-layer feedforward architecture as our basic building block because of its capabilities in approximating non-trivial functions and its relative simplicity.

The inputs to the four-layer ANN are the values of a vector of state variables at the current time as well as the values of the operating parameter settings. Its

introduces an additional state variable  $\dot{y}$  which is used to determine the final output. This approach is useful for nonautonomous but not stiff differential equations since it does not need stability. We can see another application if (with some) constraints on the right-hand side of the ODEs, such as  $y_i(t+h) = y_i(t) + \frac{h}{2} \{f_i[y(t); p] + f_i[y(t+h); p]\}$ ,  $i=1, n$ .

$$y_i(t+h) = y_i(t) + \frac{h}{2} \{f_i[y(t); p] + f_i[y(t+h); p]\} \quad i=1, n$$



**Figure 2.** (a) The four-layer feedforward architecture for the construction of continuous-time models. The inputs are a vector of state variables  $y$  and the operating parameter settings; the outputs are the prediction of the right-hand side of the set of ODEs capturing the dynamics of the system. (b) The training of the ANN in (a) is achieved by embedding it in a numerical integrator template such as the implicit integrator (trapezoidal rule) shown. The resulting composite network will be feedforward for explicit integrators and recurrent (Elman type) for implicit integrators.

outputs are the approximation of the right-hand side of the set of ODEs at the current time, as it is depicted in Figure 2(a). Figure 2(b) shows the *composite* network resulting from embedding this four-layer feedforward ANN in a simple implicit integrator scheme. The composite network is only used for training of the building block network. Once training is completed, the building block network can be used by *any* numerical integrator to obtain predictions of the behavior of the system at any required future time (and, for that matter, past times also).

The composite networks can be constructed using as their underlying structure either explicit or implicit integrators. In the case of explicit integrator templates, the resulting network will be feedforward, but there will be recurrent connections (see Figure 2(b)) for implicit integrator templates. Consider, as an example, the trapezoidal rule implicit integrator used to generate the composite network of Figure 2(b): the predicted values of the vector of state variables at the end of the next time interval  $\underline{y}(t + h)$  is given by

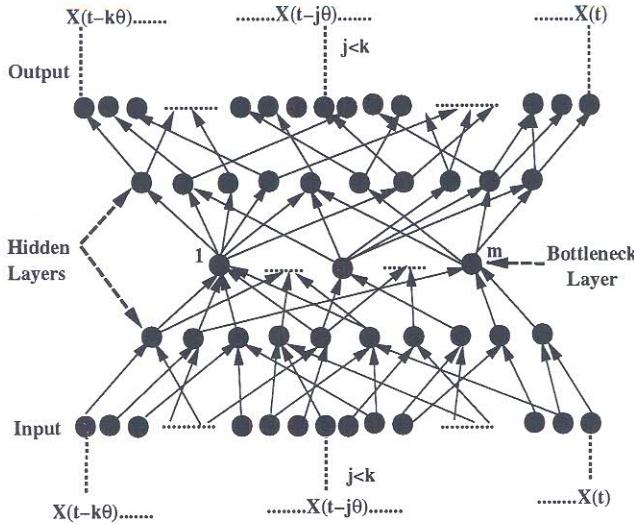
$$\underline{y}(t + h) = \underline{y}(t) + \frac{h}{2} [\underline{f}(\underline{y}(t); \underline{p}) + \underline{f}(\underline{y}(t + h); \underline{p})] \quad (5)$$

Because of the implicit character of the numerical integrator, the prediction of  $\underline{y}(t + h)$  depends on itself. This is the origin of the recurrent connections in the composite network. Note also that in (5) the function  $\underline{f}(.)$  will be given by the four-layer building block network. Learning rules for the parameters of the network can be obtained by posing the training of the composite network as a least squares problem between the experimentally observed time-series values and the values predicted by the composite network (Eq. 5). Further details about the derivation of the training rules, implementation of the algorithms, etc., can be found in references [14, 15, 16, 17, 18].

### 3.1. Preprocessing: NLPC Neural Network

In many instances, in the process of constructing an empirical model from experimental observations, the only data available are in the form of time-series of a single state variable (or a very small set of state variables). Under these circumstances, the procedure outlined above for the construction of continuous-time models cannot be applied directly. First, one must preprocess the data in order to “reconstruct phase space” with a sufficiently high dimension (i.e. obtain the vector of states to be used as outputs and targets of the composite network discussed above). Numerical differentiation can be used to produce additional independent variables from the history of measured ones (in analogy with the use of time delayed measurements). It is, however, notoriously sensitive to noise and measurement error.

An alternative procedure we have adopted for phase space reconstruction in the case of continuous systems, uses the well-known Principal Components Analysis (PCA). Broomhead and King [19] proposed the use of the linear PCA, a standard



**Figure 3.** Schematic of a nonlinear principal component (NLPC) neural network architecture.

technique in signal processing, to reconstruct the phase space from experimental time-series. In our work, we make use of a related PCA technique based on a neural network platform: the so-called “Nonlinear Principal Component Analysis” (NLPCA) [20, 21, 22, 23, 24].

Figure 3 schematically depicts the ANN architecture used for the *simultaneous* NLPCA. Kramer [20] also describes a sequential procedure for training NLPCA networks. The nonlinear principal component (NLPC) neural network we use is a five-layer feedforward network. The intermediate or *bottleneck* layer, the input layer and the output layer are all composed of neurons with linear activation functions (i.e.  $g(Z) = Z$ ). The layers above and below the bottleneck layer are composed of neurons with nonlinear (sigmoidal) activation functions. These hidden layers serve as encoder/decoder of the time-series segments used to train the NLPC network. The network is trained to approximate its inputs (identity mapping).

Following Broomhead and King’s procedure for the linear case, segments of the time-series sequence are used as input vectors to the NLPC network ( $\{x(t_i; p)\}$ ,  $i = 1, \dots, k$ ). The time interval between elements of the time-series sequence is denoted by  $\theta$  in Figure 3. The training set is constructed by sliding forward a time-window along the time-series. The time interval separating two consecutive training vectors will become the time step of the integrator ( $h$ ) in the construction of a continuous-time model. Upon convergence, the outputs of the neurons in the bottleneck layer will become the NLPCs we seek (i.e. will be used as input and target values of the composite network for the construction of continuous-time models). Examples of the application of the NLPC procedure for the reconstruction

of the phase space behavior of nonlinear systems, in addition to the ones that will be presented below, can be found in references [14, 15, 17]. We only mention in passing here that when we invert the time-series from NLPC space to real space, a consistency problem arises for data in the overlapping region of two successive time-series windows [14, 25]. This consistency issue is a subject of current research.



**Figure 4.** Sequence of images collected during CO oxidation on Pt(110) single crystals.

#### 4. Case Study I: CO Oxidation on Pt.

Chemically reacting systems are inherently nonlinear. As a result of this nonlinearity, they often operate in unsteady and/or spatially nonuniform conditions. Spatiotemporally varying systems are, in principle, characterized by an infinite number of “degrees of freedom” and their modeling requires partial differential equations. In many cases, however, the long-term dynamics of these systems may be characterized as being “low-dimensional”; in the context of the work described here, this term means that the dynamical behavior of such systems can be qualitatively described by a relatively small number of degrees of freedom (a small set of ODEs). For these reasons, spatially varying reactions offer an excellent test-bed for the neural network based methods of characterizing the long-term dynamics of continuous-time nonlinear systems.

The experimental data used in our first example come from the heterogeneous catalytic oxidation of CO on Pt(110) single crystals. This system is known to exhibit a rich variety of oscillatory behavior in the overall reaction rate (see e.g. [26] and references therein). These temporal oscillations are frequently accompanied by spatial pattern formation [27].

The experimental data were collected at the Fritz-Haber-Institut of the Max-Planck-Gesellschaft, Berlin, Germany. Figure 4 presents a sequence of four images showing a particular type of spatio-temporal pattern observed during CO oxidation. The images were collected using a photoemission electron microscope (PEEM) and represent an approximately  $300 \times 300 \mu\text{m}^2$  section of the Pt(110) crystal. The temperature of the system was kept at 550 K and the partial pressures of oxygen and carbon monoxide were  $4.1 \times 10^{-4}$  and  $1.75 \times 10^{-4}$  mbar respectively. The PEEM monitors changes in the work function ( $\phi$ ) of the catalyst surface. Adsorption of the reactants leads to an increase in  $\phi$ ; however, the chemisorption of O<sub>2</sub> has an effect that is nearly twice as large as that of CO on  $\phi$ . In the arbitrary gray-scale of Figure 4 the dark stripe-like patterns of images 1, 2 and 4 represent regions mainly covered with CO, while the broader dark regions surrounding the stripes are regions mainly covered with oxygen atoms. The narrow gray bands bordering the stripes in these images represent regions with medium coverage of both chemical species. In image 3 the well-defined narrow stripes, observed in the other images, do not appear, indicating the absence of regions predominantly covered by CO. The time interval between the first and subsequent images is:  $\frac{6}{30}$ ,  $\frac{11}{30}$  and  $\frac{23}{30}$  seconds. The O<sub>2</sub>-covered areas convert continuously into regions where CO is predominantly covered and *vice versa*. After half a cycle, image 4 is shifted with respect to image 1 by half a spatial wavelength indicating a spatio-temporal symmetry of the dynamics.

In previous work [28] we constructed a discrete-time ANN-based model capturing these spatio-temporal oscillations. In what follows, we proceed to construct a continuous-time model and to compare the properties of both approximations. The digitized images are first processed using the *proper orthogonal decomposition* (POD or Karhunen-Loeve expansion) to identify important coherent spatial structures of the system. This procedure allows us to effectively decouple the spatial and the temporal evolution of the phenomena, achieving a significant data reduction.

For a set of spatially varying data (an ensemble of M instantaneous “snapshots”) we can write the state variables as deviations from their ensemble average ( $\langle V(x) \rangle$ )

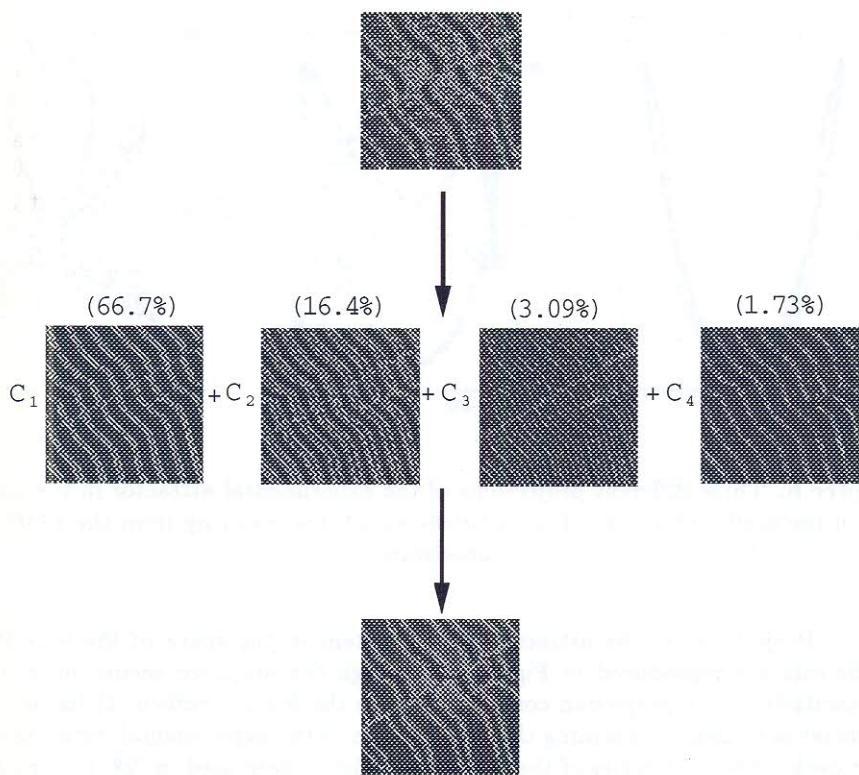
$$V(x, t) = \langle V(x) \rangle + v(x, t) \quad (6)$$

Then the state of the system at any instant can be represented in terms of the normalized eigenvectors ( $\phi_i$ , also called *coherent structures*) of the two-point correlation matrix formed with the ensemble of “snapshots”

$$u(x, t) = \sum_{j=1}^M C_j \phi_j(x) \quad (7)$$

The coefficients  $C_i$  are obtained by projecting the images onto the complete orthonormal set formed by the eigenvectors  $\phi_i$

$$C_i = (\phi_i(x), u(x, t)) \quad (8)$$

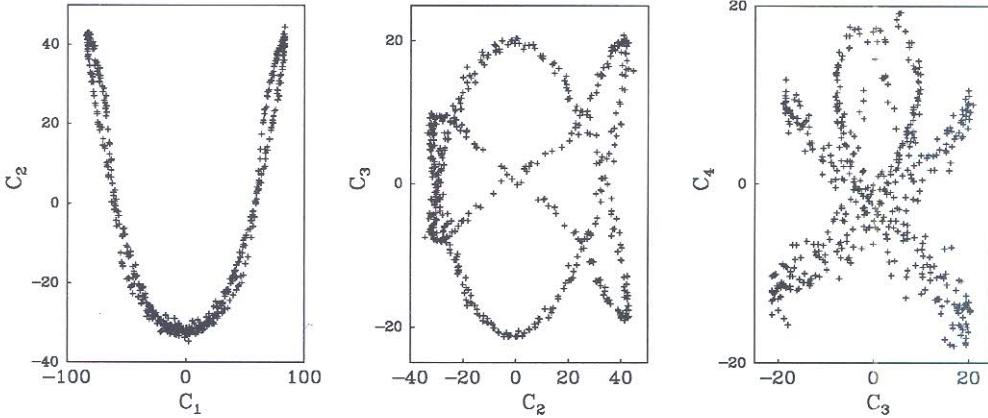


**Figure 5.** Reconstruction of the images of the CO oxidation on Pt using the POD procedure. The four coherent structures chosen are the ones with the highest occurrence in the ensemble of data; their contribution to the data, normalized on a percent basis, is indicated.

The probability of occurrence in the data of each one of these eigenvectors is quantified by the corresponding eigenvalue. The reduction of the data is achieved by neglecting the contribution of the eigenvectors with very small eigenvalue, an action that can be regarded as a filtering procedure. Further details and discussion about this procedure can be found in [28, 29] and references therein.

For the phenomena studied here, a total of 500 images were used for the POD procedure. Each image was digitized to a resolution of  $256 \times 256$  values or pixels. Figure 5 schematically shows the results of the reduction procedure. Each image is expressed as the linear combination of the four most “energetic” coherent structures. The relative contribution of each coherent structure is indicated in the figure. The reconstructed image (at the bottom of Figure 5) exhibits a remarkable

agreement with its experimental counterpart. This indicates that the dynamics of this system indeed “lives” in a low-dimensional subspace of the 65,536-dimensional phase space of the original digitized images.

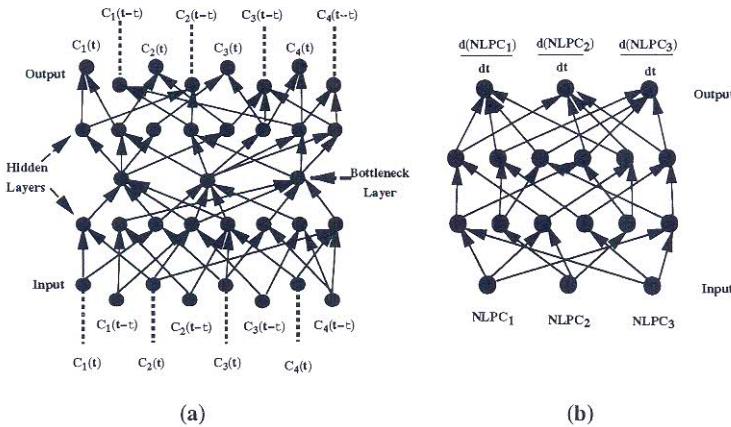


**Figure 6.** Three different projections of the experimental attractor in the space of the coefficients of the four coherent structures resulting from the POD procedure.

Projections of the attractor of the system in the space of the four POD coefficients are reproduced in Figure 6. Though the attractor seems quite noisy (particularly in the projection corresponding to the fourth coefficient) its periodic character is evident, confirming that the motion of the experimental data lies on a limit cycle. The time-series of these four coefficients were used in [28] to construct a discrete-time model capturing the temporal evolution of the dynamics contained in the data. Such a discrete model was constructed using a four-layer feedforward ANN with a configuration of the type discussed earlier in the chapter. The ANN consisted of eight inputs, four outputs and ten neurons in each of the two nonlinear hidden layers. The outputs of the network were the predictions of the four POD coefficients at a future time ( $t + \tau$ ), and its inputs were the current value of the same four coefficients (at time  $t$ ) and at a previous time ( $t - \tau$ ). The time delay ( $\tau$ ) was chosen to be 0.1 s (about one tenth of the period of  $C_1$ ). We will address the predictive capabilities of this discrete model later in this section, after describing the steps for the construction of its continuous counterpart.

The POD procedure allowed us to construct a discrete-time model with *eight* degrees of freedom; the low-dimensionality of the data (a limit cycle is a one-dimensional closed curve), based on Takens' version of Whitney's embedding theorem, indicates that 3 degrees of freedom should provide a successful embedding space for the data. For this reason, and in order to construct a continuous-time model, we chose to further pre-process the data.

The additional preprocessing step is achieved using the NLPC procedure

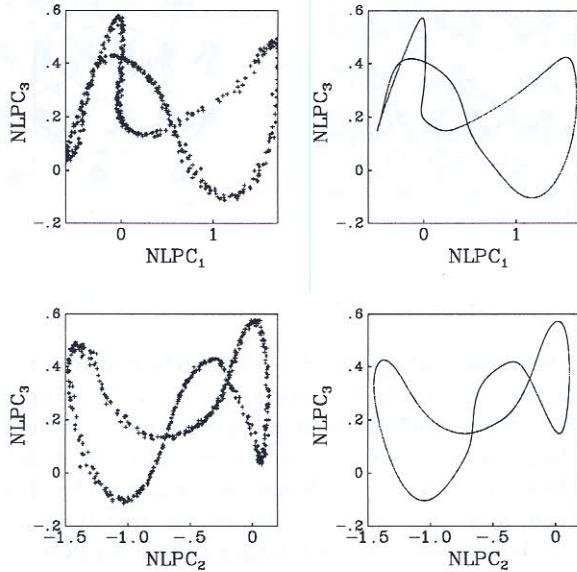


**Figure 7.** (a) Schematic of the NLPC ANN architecture for the additional preprocessing of the data of the CO oxidation on Pt. The inputs (and target values) are the four POD coefficients at a current and a previous time interval. (b) Four-layer feedforward ANN for the continuous-time modeling of the CO oxidation data. The inputs of the ANN are the values of the three NLPC states previously obtained, and its outputs constitute the prediction of the right-hand side of a set of three ODEs on these NLPCs.

described in a previous section. The NLPC procedure can be applied directly to the digitized images; however, their size (vectors with 65536 elements) is well beyond the practical applications of this technique. For this reason we make use of the POD coefficients obtained before. In this manner, we take advantage of the compression of the data already achieved.

Since an eight-element vector, formed with the four POD coefficients at the current time and a previous time, was needed to construct the discrete approximation, we use as inputs (and target values) of the NLPC network the same structure. The NLPC is schematically depicted in Figure 7(a). The bottleneck layer consists of three neurons whose outputs will become, upon convergence, the three NLPCs we seek. Although the nature of the experimental motion suggests a two-dimensional state as the minimal representation of the dynamics, our attempts to reduce the data to such a minimal representation were unsuccessful (when only two neurons are used in the bottleneck layer of the NLPC network the resulting attractor exhibits self-intersections). Training of the three-bottleneck-neuron NLPC network was achieved using a conjugate gradient algorithm on 500 data points (sets of POD coefficients). Convergence was achieved after approximately 2000 neuron parameter updates. Figure 8 shows on the left two projections of the experimental attractor in the NLPC space. The scattering of the data has been reduced significantly. On the left are two different projections of the experimental attractor (the POD coefficients are processed through the NLPC network). Note the smoother

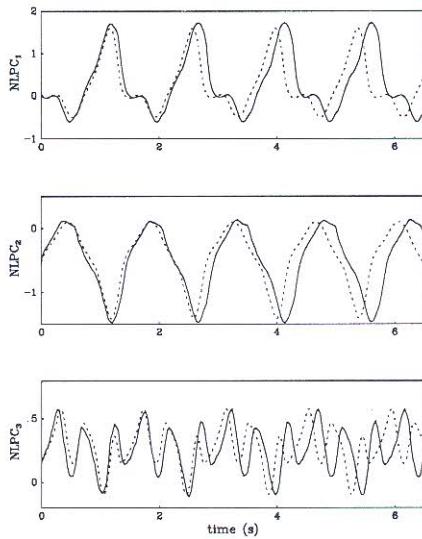
appearance of the experimental attractor compared to the attractor in the POD coefficient space. On the right are the same projections of the long-term prediction from the ODE network.



**Figure 8.** Comparison of the experimental and predicted attractors in NLPC space for the continuous-time model.

The three NLPC time-series are used as the state variables to construct the continuous-time model sought. Figure 7(b) schematically depicts the configuration of the four-layer feedforward ANN used. This four-layer ANN is embedded in an explicit (Runge-Kutta fourth order) integrator for its training. The training set consisted of 450 of the 500 available data points (the remaining data points are used for validation purposes) already processed through the NLPC network. Training of the composite network was performed using a conjugate gradient algorithm. Convergence was achieved after approximately 5000 network parameter updates. Further training reduced the error of the 50 point validation set only marginally.

The resulting ODE network correctly predicts a periodic attractor. A comparison of the experimental and predicted attractors in NLPC space is presented in Figure 8. The continuous-model exhibits satisfactory agreement in the shape and amplitude of the predicted attractor with respect to the real attractor. It is important to note that the plotted trajectory constitutes the long-term behavior of the system: a single initial point was fed to the ODE network to start the integration. The trajectory plotted was obtained after neglecting the first 1000 points resulting from this integration to eliminate the effect of the choice of the initial point. The short-term prediction is similarly accurate, as can be seen in Figure 9. The amplitudes of the oscillations are very well reproduced and the prediction of the

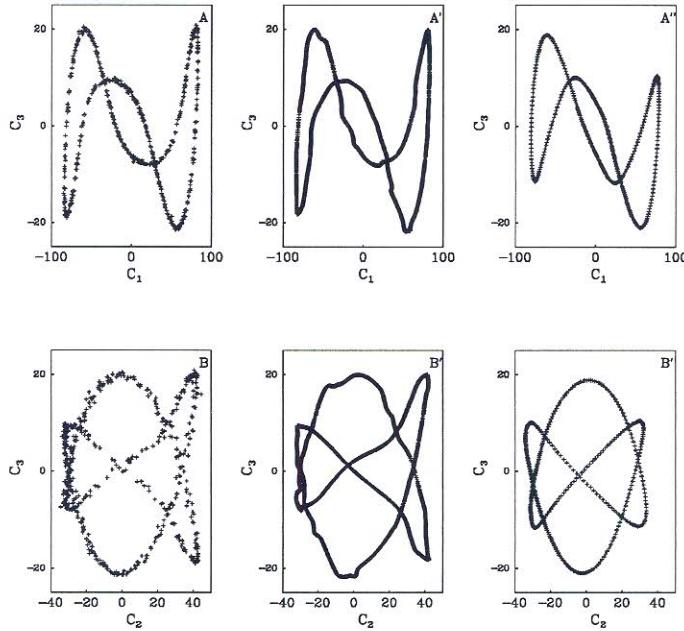


**Figure 9.** Comparison of the three NLPC time-series with the corresponding prediction of the ODE network. The dashed line is the prediction of the ODE network, and the solid line is the experimental data NLPC coefficients (POD coefficients processed through the NLPC network).



**Figure 10.** Comparison of the experimental images and the continuous-time model prediction. The long-term predictions of the ODE-model are translated to real space (first processing through the upper part of the NLPC network and then reconstruction using the POD coherent structures) and compared side by side with the corresponding experimentally recorded images.

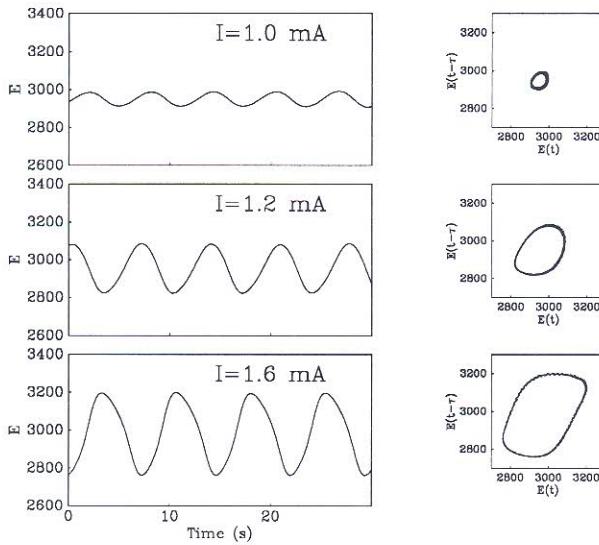
frequency of the oscillations is slightly inaccurate. Figure 10 compares the experimental snapshots against the prediction of the continuous model. All significant features of the images are well reproduced.



**Figure 11.** Comparison of two projections of the attractor on the POD coefficient space.

It is interesting to compare the predictions of the continuous-time model with its discrete-time counterpart. Figure 11 shows two different projections on the space of POD coefficients of the experimental attractor along with the discrete and continuous-time model prediction of the attractor of the system. First column (A,B) shows the experimental attractor after applying the POD procedure. The second column (A',B') shows the long-term prediction of the discrete-time model, and the third column (A'',B'') the prediction of the attractor given by the set of ODEs (ANN approximation based on the NLPC time-series) converted back to POD coefficient space using the upper (decoder) part of the NLPC network. Although the discrete-time model captures with excellent accuracy the amplitude and shape of the attractor, the predicted attractor exhibits several “bumps” against the very smooth, though not as accurate, prediction of the continuous-time model. The origin of the bumps in the prediction of the time-delay model is precisely its discrete character. Even though for this example such discrete character does not seem to affect the predictive qualities of the model constructed, it becomes of particular relevance during the characterization of the long-term behavior of systems as the operating parameters are varied (transitions and bifurcations), as

we will illustrate in the following section.



**Figure 12.** Experimental time-series from the electrochemical oxidation of hydrogen at a platinum anode under galvanostatic conditions in  $\text{HClO}_4$  solution along with the corresponding attractors reconstructed using a  $1.25\text{s}$  time-delay.

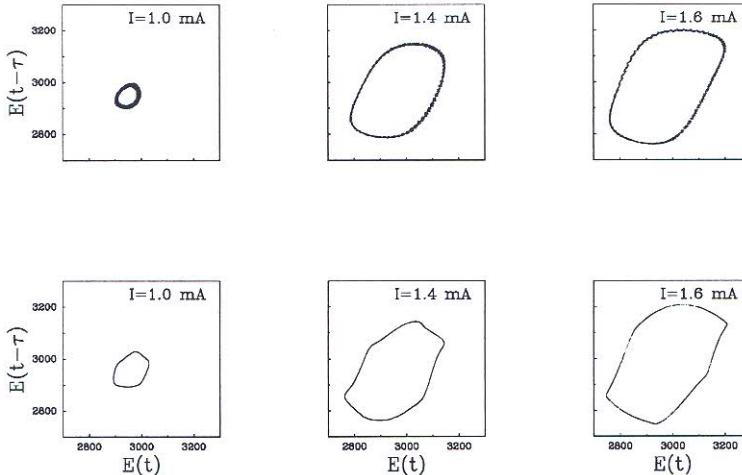
## 5. Case Study II: Electrochemical Oxidation of $\text{H}_2$

In our first case study we were concerned with the construction of an empirical model capable of capturing the long-term behavior of a system at a single parameter setting. The problem of characterizing the dynamical behavior of a nonlinear system usually involves a sequence of time-series obtained at several distinct parameter settings. Our concern is, then, not only to reproduce a single attractor, but rather a sequence of attractors and their instabilities and transitions to each other as the operating conditions are varied.

Our second case study involves a sequence of time-series collected during experiments on the electrochemical oxidation of hydrogen. The experiments were performed with a conventional cell using a platinum wire (total surface area approximately  $20 \text{ cm}^2$ ) as the working electrode, a Pt rod as the counter-electrode, and a  $\text{Hg}/\text{Hg}_2\text{SO}_4$  standard electrode as the reference electrode. The electrolyte consisted of  $1 \text{ M HClO}_4$  (containing  $10^{-6} \text{ M Cl}^-$ , which turned out to be essential for the oscillatory behavior) to which small concentrations of  $\text{Cu}^{2+}$  were added ( $4 \times 10^{-5} \text{ M}$  for the results presented here) and it was saturated with  $\text{H}_2$ . The system was operated under galvanostatic conditions and time-series of the potential of the working electrode were collected at different values of the applied current.

Further details about the experimental setup and the preparation of the electrodes can be found in references [30, 31].

Figure 12 shows a sequence of three time-series collected from the experiment along with the corresponding reconstructed attractors using a time-delay ( $\tau$ , the delay, is chosen to be 1.25 s, approximately one fifth of the period of the oscillation). Each time-series was collected at a different value of the applied current, noted in the figure. The variable plotted ( $E$ ) is a scaled version of the potential measured versus the normal hydrogen electrode (NHE) ( $e(mV)=0.1E+100$ ). A total of seven experimental time-series are available in the range of 0.975 mA to 1.6 mA, all exhibiting oscillatory behavior with increasing amplitude as the applied current is increased. The system undergoes a Hopf bifurcation (transition from a stable steady state to periodic behavior) at approximately 0.95 mA. The characterization goals in this simple illustration involve the detection of this Hopf bifurcation and the reproduction of the growth of the amplitude of the observed limit cycle. In what follows we describe the construction of empirical neural network models using both types of modeling approaches, discrete and continuous, and their results.



**Figure 13.** Comparison of experimental (top) and predicted (bottom) attractors for three different parameter values. The experimental attractor recorded at  $I=1.4$  mA was not included in the training set.

### 5.1. Discrete-Time Model

Following the procedure described previously, we attempt to construct a discrete-time empirical model of the form

$$E(t + \tau) = F(E(t), E(t - \tau); I) \quad (9)$$

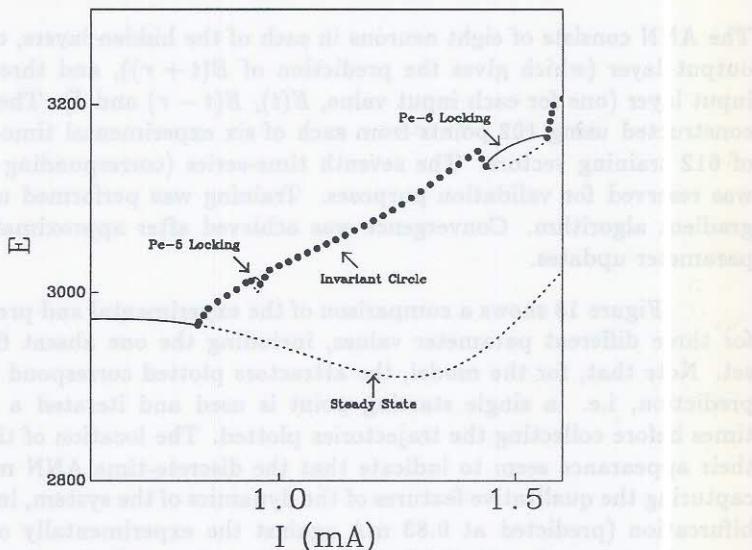
where the function  $F(\cdot)$  will be approximated by a four-layer feedforward ANN.

The ANN consists of eight neurons in each of the hidden layers, one neuron in the output layer (which gives the prediction of  $E(t + \tau)$ ), and three neurons in the input layer (one for each input value,  $E(t)$ ,  $E(t - \tau)$  and  $I$ ). The training set was constructed using 102 points from each of six experimental time-series for a total of 612 training vectors. The seventh time-series (corresponding to  $I = 1.4$  mA) was reserved for validation purposes. Training was performed using a conjugate gradient algorithm. Convergence was achieved after approximately 1500 neuron parameter updates.

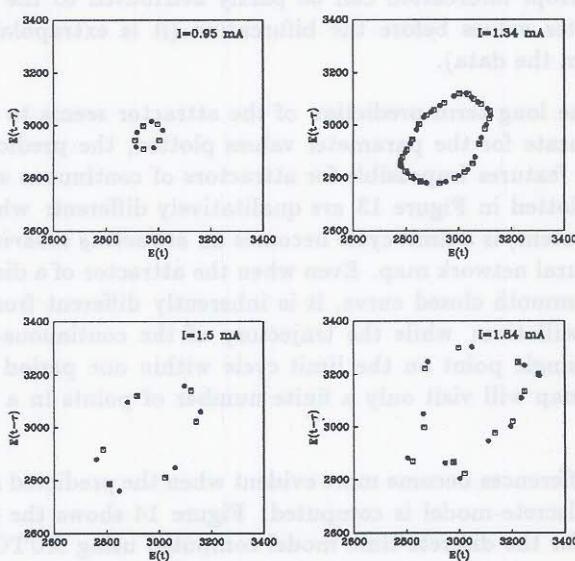
Figure 13 shows a comparison of the experimental and predicted attractors for three different parameter values, including the one absent from the training set. Note that, for the model, the attractors plotted correspond to the long-term prediction, i.e. a single starting point is used and iterated a large number of times before collecting the trajectories plotted. The location of the attractors and their appearance seem to indicate that the discrete-time ANN model succeeds in capturing the qualitative features of the dynamics of the system, including the Hopf bifurcation (predicted at 0.83 mA against the experimentally observed value of approximately 0.95 mA). While the point-by-point prediction deteriorates, and the experimental and predicted time-series do not match for long times, the *attractors* of the two dynamical systems (experiment and model) as invariant sets in phase space lie close to each other for infinite time. The relatively poor prediction of the location of the Hopf bifurcation can be partly attributed to the lack of training data at parameter values before the bifurcation (it is extrapolated rather than interpolated from the data).

While the long term prediction of the attractor seems to be qualitatively correct and accurate for the parameter values plotted, the predicted periodic attractor develops features impossible for attractors of continuous systems. Indeed, the attractors plotted in Figure 13 are qualitatively different: what for the ODE (experimental system) is a limit cycle becomes an *attracting invariant circle* for the discrete-time neural network map. Even when the attractor of a discrete map looks like a perfectly smooth closed curve, it is inherently different from a continuous-time periodic oscillation: while the trajectory of the continuous-time oscillation will visit every single point on the limit cycle within one period  $T$ , a trajectory of the discrete map will visit only a finite number of points in a time interval of length  $T$ .

These differences become more evident when the predicted scenario of transitions for the discrete-model is computed. Figure 14 shows the predicted bifurcation diagram for the discrete-time model computed using AUTO [32]. A couple of the most prominent “phase lockings” predicted by the model on the invariant circle are noted in the diagram. The branch of stable steady states (solid line) undergoes a Hopf bifurcation, at  $I=0.83$  mA, becoming unstable (dashed line) and giving rise to a branch of invariant circles (filled circles). A phase locked state appears when the period of the natural oscillation approaches a rational multiple of the time-delay used to construct the map; in these cases, the attractor is a finite number of points in phase space instead of a closed curve. The invariant circle

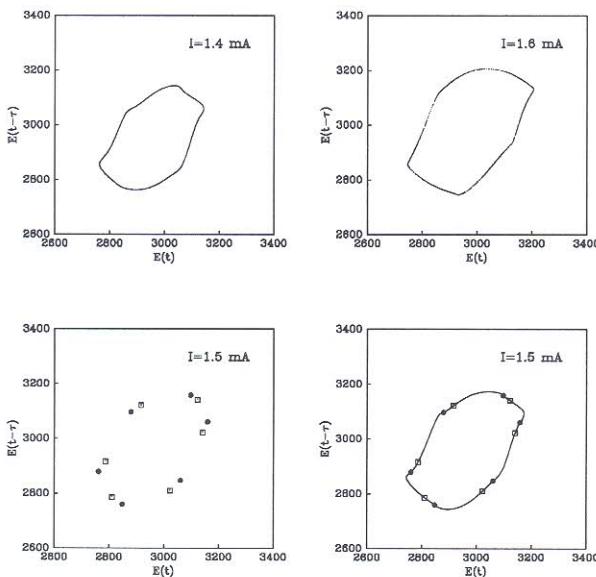


**Figure 14.** The bifurcation diagram predicted by the discrete-time model for the experimental data shown in Figure 12.



**Figure 15.** A sample of the phase locked states predicted by the discrete-time neural network model. The nodes are indicated by the filled circles and the saddles by the empty squares.

gives way to phase locked states at a saddle-node bifurcation: in addition to the set of attracting periodic points (nodes) a set of unstable periodic points (saddles) is born at the bifurcation with the same periodicity as the stable set. These saddle-node bifurcations may be considered "spurious" in the sense that they cannot be associated with any transition of the long-term behavior of the continuous-time experimental system. The branches of periodic points persist for a finite range of the parameter value (the applied current) before closing and disappearing, leaving again the invariant circle as the attractor of the system. A large number of such lockings develop on the invariant circle for the parameter range of interest. Figure 15 shows a sample of such phase locked states predicted for the discrete-time model at different values of the applied current. At  $I=0.95$  mA the attractor of the system is a set of five periodic points (stable period-5 points). A stable period-23 solution is observed at  $I=1.34$  mA. A period-6 solution is observed at  $I=1.5$  mA and a period-11 is observed at  $I=1.84$  mA.



**Figure 16.** The predicted phase locked invariant circle, constructed using the unstable manifolds of the saddle solutions for a period-5 locking observed at  $I=1.5$  mA (bottom row).

While phase locked, the invariant circle can be constructed using the *unstable manifolds* of the saddle solutions. The saddles are characterized by one stable (attracting) and one unstable (repelling) direction. In their vicinity, trajectories "escape" along the unstable direction, which can be approximated by the eigenvector of the map linearization corresponding to the unstable eigenvalue. The unstable manifolds are approximated by initializing a number of trajectories locally on the unstable eigenvector, and recording their approach to the stable set of five periodic

points (phase-locked period-5 solution). Unless such a procedure is followed, one ultimately “sees” only the attractor: the stable phase locked period-5 solution.

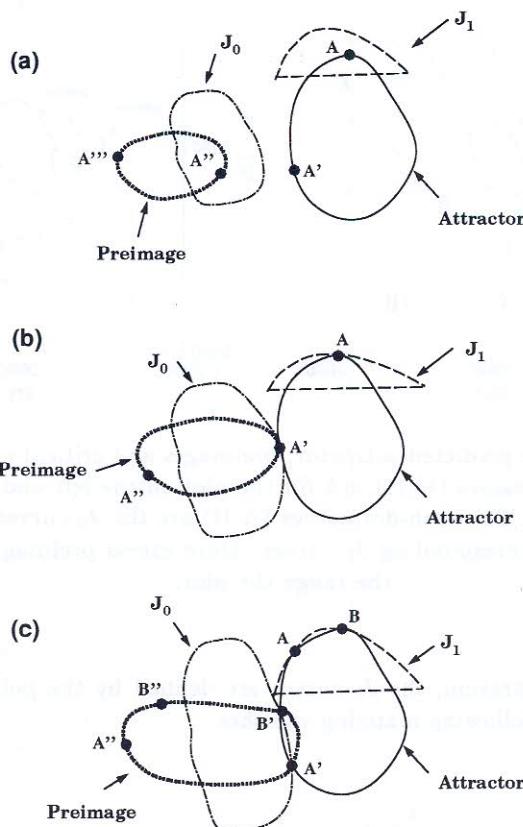
Figure 16 shows the period-5 fixed point attractor and the construction of the invariant circle using the unstable manifolds of the period-5 saddles at  $I = 1.5$  mA. Two nearby invariant circles (before and after the locking) are shown in the top row for comparison. Note that the shape and location of the phase locked invariant circle closely resembles the shape and location of attracting invariant circles at nearby parameter values. The prediction of locked states (and the associated spurious bifurcations) is just one of the features of discrete-time models that make them inappropriate for characterizing the dynamical behavior of continuous-time systems. As we will illustrate below, these discrete maps can also become *noninvertible*. The interaction of noninvertibility and the phase locked states may cause deterioration of the predictive capabilities of the map, adding even more unphysical features to the long-term predictions of the discrete-time model.

## 5.2. Noninvertibility in discrete-time ANN models

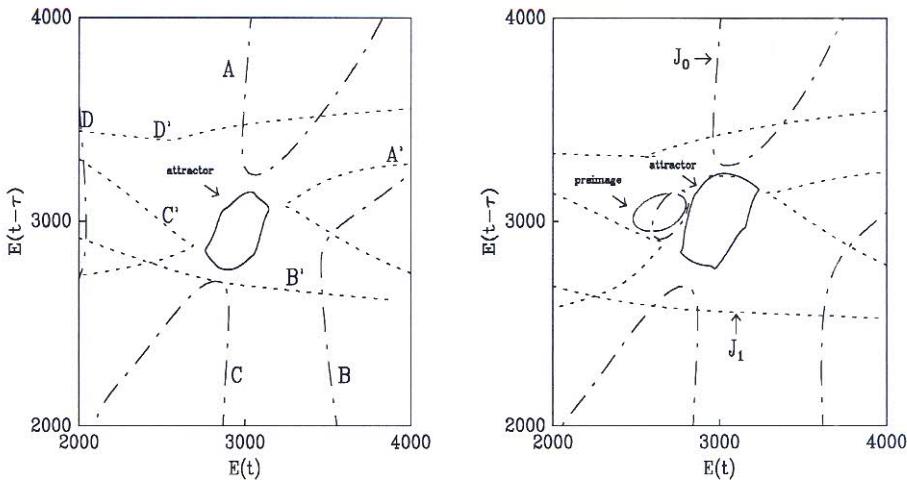
For continuous systems every phase space point possesses a unique preimage backwards in time, i.e. it *comes* from a single possible previous state. This is equivalent to integrating a set of (smooth) ODEs backwards for the appropriate time interval. Discrete dynamical systems, on the other hand, do not necessarily have to be uniquely invertible. The discrete-time ANNs we typically construct, due to their nonlinear activation functions, are *noninvertible* (non-uniquely invertible backward in time). In other words, given  $E(t+\tau)$ ,  $E(t)$  and  $I$ , we may find more than one preimage (i.e., more than one value of  $E(t-\tau)$  satisfying the expression that defines the neural network (Eq. 9)). The noninvertible features of neural network maps play an important role in the prediction of impossible structures and transitions of the attractors in phase space. As a consequence, the onset of noninvertible behavior can be used to determine the range of validity of the predictions of neural network maps [33, 34].

In principle, any neural network model allowing multiple preimages is *inconsistent* (globally in phase space) with a continuous-time dynamical system. However, if only one of the multiple preimages lies in the vicinity of some physically relevant region of the phase space (e.g., if only one preimage of a concentration variable is positive), one still can consider the network predictions to be valid *in a restricted region in phase space*. An invertible mapping is characterized by a nonsingular linearization (the determinant of its Jacobian does not become zero in phase space). The onset of noninvertible behavior is then signaled by the presence, in phase space, of curves defined by the vanishing of the determinant of the linearization of the (forward-time) mapping. These curves (denoted by  $J_0$ ) along with their  $n$ -th iterates ( $J_n$ ) play a crucial role in generating noninvertible dynamical features; such features are impossible for continuous-time systems described by ODEs which can be uniquely “inverted”, that is, integrated backward in time in all of phase space.

**Onset of Noninvertibility:**



**Figure 17.** Schematic of the onset of noninvertibility in the predictions of a neural network model. (a) Portions of the attractor lie in the “interior” of a  $J_1$  curve, thus having multiple preimages. The “excess” ( $A''$  and  $A'''$ ) preimages (in addition to the regular preimage  $A'$  on the attractor) lie on the nearby preimage curve. Note that only points in the interior of the  $J_1$  curve have excess-preimages on the preimage curve shown. (b) An excess preimage crosses the attractor at a higher parameter value; the crossing occurs *on* a  $J_0$  curve. The iterate ( $A$ ) of the point of crossing ( $A'$ ) lies at the tangency of the  $J_1$  curve with the attractor; its other excess preimage is denoted by  $A''$ . (c) A noninvertible attractor crossing one of its excess preimages twice on a  $J_0$  curve. At the iterates ( $A$  and  $B$ ) of the points of crossing ( $A'$  and  $B'$ ), the attractor is tangent to the corresponding  $J_1$  curve. Following these tangencies, the attractor shape becomes gradually distorted, eventually generating cusp-like features at higher parameter values.



**Figure 18.** The predicted attractor, preimages and critical curves for two different parameter values ( $I=1.4$  mA for the plot on the left and  $1.68$  mA for the plot on the right). The dash-dot curves ( $J_0$ ) are the  $J_0$  curves and the short dash ( $J_1$ ) the corresponding  $J_1$  curves. More excess preimages exist outside the range the plot.

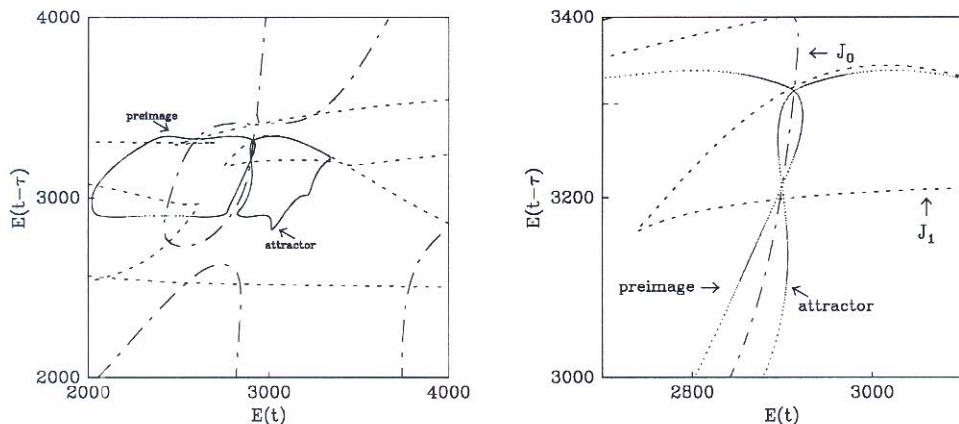
For our illustration, the  $J_0$  curves are defined by the points at which the determinant of the following mapping vanishes

$$\begin{aligned} X_{n+1} &= Y_n \\ Y_{n+1} &= F(Y_n, X_n; I) \end{aligned} \quad (10)$$

where  $X_n \equiv E(t_0 + n\tau)$ ,  $Y_n = X_{n+1} \equiv E(t_0 + (n+1)\tau)$  and  $F(\cdot)$  represents the ANN approximation (note that  $X_0 = E(t_0)$ ,  $Y_0 = X_1 = E(t_0 + \tau)$  are required to initialize the iteration of the map). When iterating the neural network mapping forward in time, the phase space is “folded” along the  $J_0$  curves and mapped “into” itself (with  $J_0$ ’s mapped on  $J_1$ ’s). As a result of this “forward folding”, some regions of phase space have more preimages than others; the boundaries separating regions with different numbers of preimages are defined by the  $J_1$  curves (the first iterates of the  $J_0$  curves). Figure 17 schematically depicts a sequence of events leading to a “truly” noninvertible attractor for our system. In Figure 17(a) noninvertibility is, in principle, present, since some points on the attractor (the closed curve on the right) have more than one preimage. Only one of these preimages, however, is *on the attractor itself*. This means that, if we restrict ourselves to a finite neighborhood of the attractor, the forward-in-time mapping can be *uniquely* inverted. The onset of “true” or “observable” noninvertibility, that is, one that involves the attractor itself, is the result of the interaction of the attractor (and necessarily its excess preimages)

with the  $J_0$  curve (and necessarily its forward iterates) over a parameter range. For our discrete-time model such events occur at parameter values just outside the range used in training. Figure 18 shows the predicted attractor, preimages and numerically calculated critical curves ( $J_0$ 's and  $J_1$ 's) for two different parameter values. For the parameter range used in training (0.975-1.6 mA) the attractor is free of noninvertible features. The close proximity of an excess preimage to the attractor, shown for the phase space plot at  $I=1.68$  mA, suggests that the noninvertible features of the map may, for nearby parameter values, begin to affect the quality of its predictions.

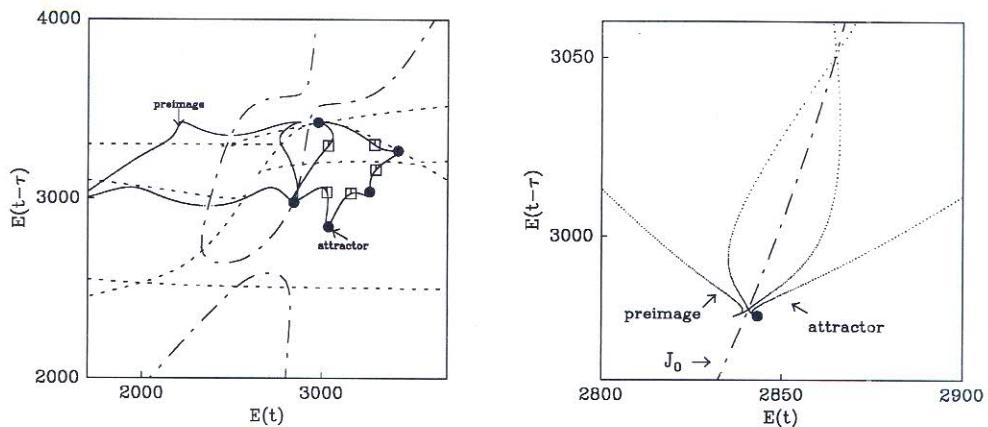
As the parameter value is increased further (see Figure 19) one excess preimage intersects the attractor. This crossing occurs on the nearby  $J_0$  curve. In Figure 19 the attractor, its excess preimages and the critical curves are plotted for  $I=1.92$  mA. These noninvertible phenomena result in a deterioration of the round shape of the attractor. Notice also the presence of several sharp sections on the attractor in Figure 19.



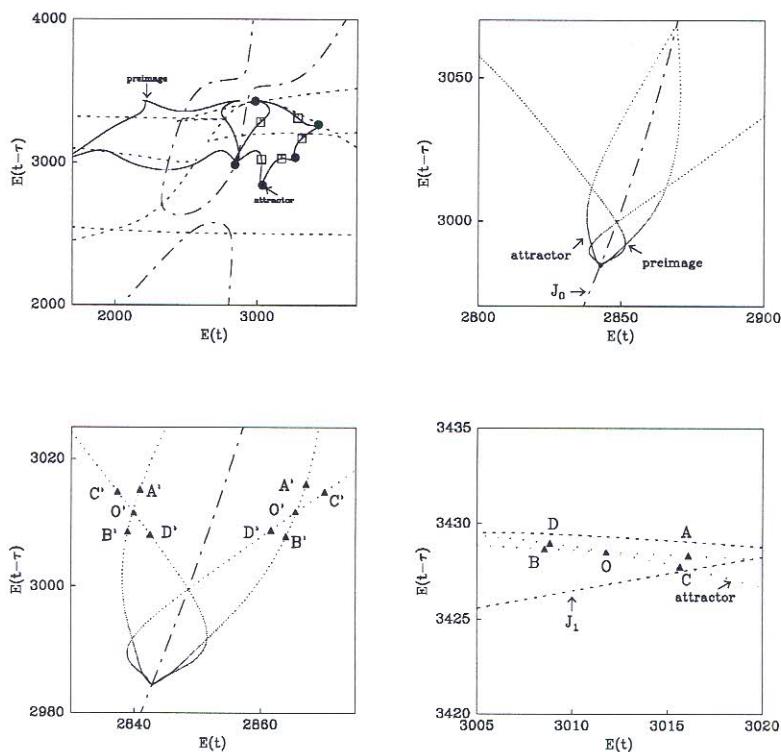
**Figure 19.** Crossings of an excess preimage and the attractor as it interacts with a  $J_0$  curve at  $I=1.92$  mA.

The interaction between noninvertibility and the phase locking of the invariant circle will be responsible for a more dramatic "unphysical" feature of the map predictions: *self-intersections of the attractor*. Figure 20 shows the phase-locked invariant circle, constructed using the unstable manifolds of the period-5 saddles, along with the nearby preimage and critical curves observed at  $I=2.08$  mA. Note that the invariant circle is phase-locked: a pair of period-5 solutions has appeared on it. The period-5 nodes are indicated in the figure by the filled circles and the period-5 saddles by the empty squares. The blowup on the right shows more clearly the interaction of the invariant circle with the  $J_0$  curve: the intersections with its excess preimages still occur on  $J_0$ .

Although "pointy" features on the attractor are more evident, the attractor



**Figure 20.** The predicted attractor, its excess preimages and the critical curves at  $I=2.08$  mA.



**Figure 21.** A self-intersecting invariant circle predicted by the discrete-time neural network model at  $I=2.09$  mA.

is free of self-intersections as can be better appreciated in the blowup on the right of Figure 20. As the parameter value is slightly increased to  $I=2.09$  mA, the invariant circle (constructed using the unstable manifolds of the period-5 saddles) develops self-intersections. Figure 21 shows the self-intersecting invariant circle at this parameter value (2.09 mA). The “braided” structure is the result of having points on the invariant circle *with multiple preimages on it* (such as the point O in Figure 21, and its two preimages O'). The (phase-locked) invariant circle, its excess preimages and the critical curves in the “relevant” region of phase space are shown in the upper left corner of the figure. The pictures in the upper right and bottom left corners show blowups in the vicinity of one of the period-5 nodes. The self-intersections become evident in these blowups. There exists points on the invariant circle (O) that possess two distinct preimages on it (O'). These multiple preimages are the origin of the braided, self-intersecting structure.

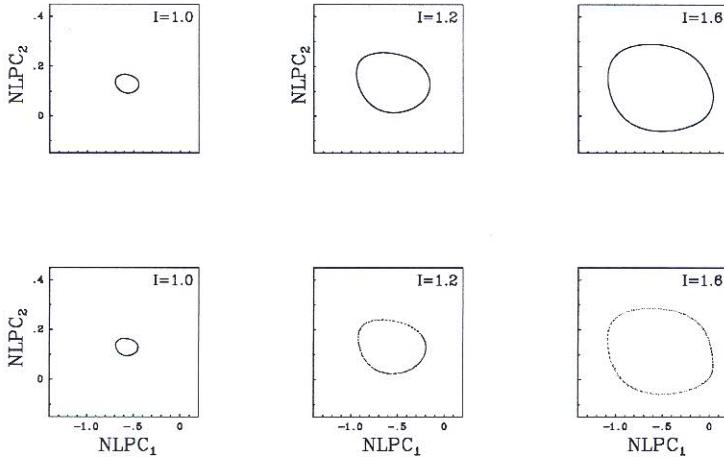
We conclude by reiterating that discrete-time ANN models may not be the best alternative for the characterization of certain dynamical features of continuous-time nonlinear systems. Because of their discrete nature, these models may predict attractors and transitions impossible for continuous-time systems. The illustrations presented here are intended to serve as an indication of these pathologies; the reader should keep in mind that the predictions of incorrect scenarios of transitions may be of far greater complexity than the relatively simple illustrations described in this chapter [33, 34].

### 5.3. Continuous-Time Model

Consider the characterization of the same set of time-series (from the electrochemical oxidation of hydrogen) using now the continuous-time modeling approach previously described. The first step consists of reconstructing a phase space for the system using NLPCs. The NLPC neural network used has twenty neurons in the input and output layers, two neurons in the bottleneck layer and six neurons in each of the two nonlinear hidden layers. Since the motion of the system involves simple periodic behavior only, the *minimal* dimension of the phase space consistent with the reconstruction of such behavior is two. For this reason we use two neurons in the bottleneck layer (recall that the outputs of these neurons will give, upon convergence, the time-series of the NLPC states).

The twenty elements of each training vector are selected to span a time interval of approximately the same length as the time-delay used in the discrete-time model previously discussed (i.e. 1.25 seconds). The time interval between elements within each vector is 0.0625 seconds. The training vectors are constructed using the same segments of the same six experimental time-series used to construct the discrete-time approximation. The first element of each training vector was displaced forward by 0.1825 seconds with respect to the first element of the previous vector, giving an overlap of seventeen measurements with it. The time step of the integrator template on which we base the training of the ODE network (four layer feedforward ANN, predicting the right-hand side of the set of two ODEs on the

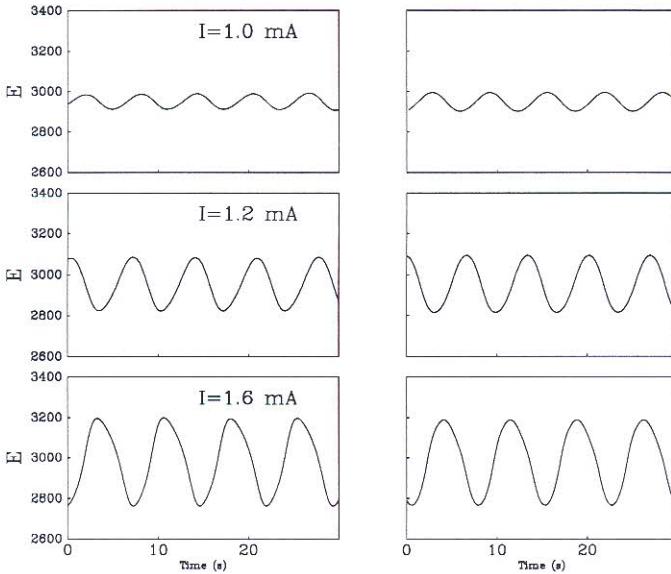
NLPC states) is then precisely 0.1825 seconds. As before, a seventh time-series is reserved for validation purposes.



**Figure 22.** Comparison of the continuous-time model prediction (top) and experimental (bottom) attractors in NLPC space.

The training of the NLPC network was performed using a conjugate gradient algorithm. Convergence was achieved after approximately 2500 neuron parameter updates. In the lower part of Figure 22 the reconstructed attractor in the NLPC space is plotted for three different parameter values. The time-series of the two NLPC states are used to construct the continuous-time model. The experimental attractors were obtained after processing the experimental time-series through the lower (encoder) part of the NLPC network. The continuous model predictions are long-term predictions: a single initial point was used to generate the trajectory plotted (neglecting the initial transient).

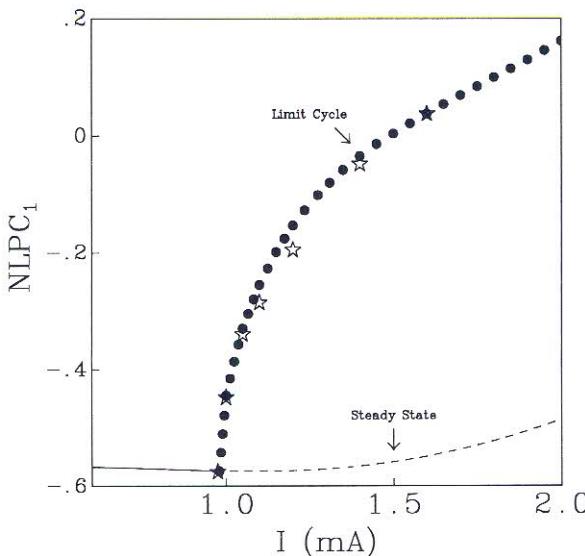
As we mentioned in a previous section, a four-layer feedforward ANN, whose inputs are the NLPC states and operating parameter value, and whose outputs are the prediction of the right-hand side of a set of ODEs on the NLPC states, is embedded in a numerical integrator template. The four-layer ANN has three inputs (the two NLPCs and the value of the current), two outputs (the prediction of the right-hand side of the two ODEs on the two NLPC states) and six neurons in each of the two nonlinear hidden layers. The numerical integrator chosen to generate the composite network used for training was the implicit trapezoidal rule used in Figure 2(b). Training was performed using conjugate gradient algorithm in a Pineda-type recurrent network training procedure [14, 17, 35]. Frequent restarts were required because of approximations inherent in the evolution rule used to evaluate recurrent network output and derivatives. Convergence was achieved after approximately 4000 neuron parameter updates. Figure 22 shows a comparison of the (reconstructed) experimental and predicted attractors in NLPC space for



**Figure 23.** Comparison of the experimental (left) and predicted (right) time-series for three different parameter values.

three different operating parameter values. The plotted trajectories represent the long-term prediction of the model: they were obtained using a single initial starting point for the integration and discarding the first 1000 points. Once the continuous-time model has been constructed, the four layer ANN can be coupled with *any* numerical integrator using *any* time step for the integration. The final integrations for this case were performed using a robust multi-step integrator with error control [36].

The continuous-time ANN model also succeeds in predicting the period of the oscillations, as evident in Figure 23 where segments of the experimental and predicted time-series are plotted for three different parameter values. These again constitute long-term predictions. The model is successful in predicting the variation of the period of the oscillations with the operating parameter. Figure 24 shows the bifurcation diagram for the continuous-time model constructed using AUTO [32]. Solid lines represent stable steady states, dashed lines represent unstable steady states, filled circles represent limit cycles (the maximum amplitude of the oscillation at the corresponding parameter value), and the stars are the experimental measurements. The predicted bifurcation scenario is also consistent with the continuous nature of the phenomena studied: no spurious transitions of impossible attractors are predicted. The location of the Hopf bifurcation is accurately predicted at  $I=0.9795$  mA (compare against the experimentally interpolated value of 0.95 mA).



**Figure 24.** The continuous-time model prediction of the bifurcation scenario for the experiments of the electrochemical oxidation of hydrogen.

## 6. Summary and Conclusions

We have taken a “long-term dynamics” approach in discussing the spatio-temporal behavior of nonlinear systems, its dependence on operating parameters, and how this affects the construction of neural network dynamic models for their identification. In particular, we discussed some of the problems that traditional, discrete-time delay-based ANN models exhibit when used to characterize continuous-time systems described by nonlinear coupled ordinary differential equations. The presence of phase locking phenomena, as well as the noninvertible features of discrete delay-based ANN maps, results in the prediction of spurious transitions and impossible (for continuous-time systems) attractors. We demonstrated the use of alternative ANN based methods for the construction of continuous-time models (sets of ODEs). The capabilities of this approach were illustrated using experimental time-series from two chemically reacting systems exhibiting oscillatory behavior.

Discrete-time delay-based models are currently widely used as the basis for developing control schemes. The availability of good such models may greatly enhance the effectiveness of control schemes and ease the characterization of the response of the system under control. In this context, the pathologies of ANN maps studied here acquire a new relevance in determining the range of validity of discrete-time model predictions. We have previously suggested that the onset of noninvertibility may provide such a bound on the validity of model predictions

[33, 34].

Current research directions, along the lines of the work described here include:

- the parallel implementation of the algorithms for continuous-time model identification (both in SIMD and MIMD architectures, see [37]), and
- the extension of the continuous-time ANN modeling to the construction of gray box (hybrid) models [18] in which the knowledge derived from first principles is combined with a neural network to construct a dynamical model of the system.

The examples presented suggest that ANN-based continuous-time models of continuous-time systems are a viable alternative in analyzing, characterizing and predicting complex nonlinear dynamical behavior.

## 7. Acknowledgements

We are indebted to Prof. Dr. G. Ertl and Dr. H.-H. Rotermund of the Fritz-Haber-Institut der MPG, Berlin, Germany, for providing us with the data for the CO oxidation on Pt example. We are also indebted to Dr. A. S. Lapedes and R. M. Farber of Los Alamos National Laboratory and Prof. J. L. Hudson of the U. of Virginia for collaboration and discussions over the last few years. This work was partially supported by ARPA/ONR, NSF and the Exxon Education Foundation. RRM acknowledges the support of the Mexican Ministry of Public Education (Sistema Nacional de Investigadores) and KK the support of the Deutsche Forschungsgemeinschaft.

## References

- [1] D. C. Psichogios and L. H. Ungar, A hybrid neural network-first principles approach to process modeling. *AICHE J.*, vol. 38, pp. 1499-1511 (1992).
- [2] Y. You and M. Nikolaou, Dynamic process modeling with recurrent neural networks. *AICHE J.*, vol. 39, pp. 1654-1667 (1993).
- [3] I. G. Kevrekidis, R. Rico-Martínez, R. E. Ecke, R. M. Farber, and A. S. Lapedes, Global bifurcations in Rayleigh-Bénard convection: Experiments, empirical maps and numerical bifurcation analysis. *Physica D*, vol. 51, pp. 342-362 (1994).
- [4] E. Hernández and Y. Arkun, Study of the control-relevant properties of back-propagation neural network models of nonlinear dynamical systems. *Computers chem. Engng.*, vol. 16, pp. 227-240 (1992).

- [5] E. P. Nahas, M. A. Henson, and D. E. Seborg, Nonlinear internal model control strategy for neural network models. *Computers chem. Engng.*, vol. 16, pp. 1039-1057 (1992).
- [6] B. E. Ydstie, Forecasting and control using adaptive connectionist networks. *Computers chem. Engng.*, vol. 14, pp. 583-599 (1990).
- [7] A. S. Lapedes and R. M. Farber, Nonlinear signal processing using neural networks: prediction and system modeling. *Los Alamos Report LA-UR 87-2662* (1987).
- [8] G. C. Goodwin and K. S. Sin, *Adaptive Filtering, Prediction and Control*. Prentice Hall, Englewood Cliffs, USA (1984).
- [9] F. Takens, *Detecting strange attractors in turbulence*. In: "Dynamical Systems and Turbulence", D. A. Rand and L. S. Young (eds.), Lect. Notes in Math., Springer, Heidelberg, Germany (1981).
- [10] N. H. Packard, J. P. Crutchfield, J. D. Farmer and R. S. Shaw, Geometry from a time series. *Phys. Rev. Letters*, vol. 45, pp. 712-716 (1980).
- [11] F.-S. Tsung and G. W. Cottrell, Learning temporal signals in phase space. *International Symposium on Nonlinear Theory and Applications*, Hawaii, USA, preprint (1993).
- [12] H.-T. Su and T. J. McAvoy, Identification of chemical processes using recurrent networks. *Proc. 1991 American Control Conference*, pp. 2314-2319 (1991).
- [13] A. S. Weigend and N. A. Gershenfeld, *Time Series Prediction: Forecasting the Future and Understanding the Past*. Addison-Wesley (1993).
- [14] R. Rico-Martínez, *Neural Networks for the Characterization of Nonlinear Deterministic Systems*. Ph. D. Thesis, Princeton University, Dept. of Chemical Engineering (1994).
- [15] R. Rico-Martínez, K. Krischer, I. G. Kevrekidis, M. C. Kub, and J. L. Hudson, Discrete- vs. continuous-time nonlinear signal processing of Cu electrode dissolution data. *Chem. Eng. Comm.*, vol. 118, pp. 25-48 (1992).
- [16] R. Rico-Martínez and I. G. Kevrekidis, Continuous-time modeling of nonlinear systems: A neural network approach. *Proc. 1993 IEEE Int. Conf. on Neural Networks*, vol. III, pp. 1522-1525 (1993).
- [17] R. Rico-Martínez, I. G. Kevrekidis, M. C. Kub, and J. L. Hudson, Discrete- vs. continuous-time nonlinear signal processing: Attractors, transitions and parallel implementation issues. *Proc. 1993 American Control Conference*, vol. 2, pp. 1475-1479 (1993).

- [18] R. Rico-Martínez, J. S. Anderson and I. G. Kevrekidis, Continuous-time nonlinear signal processing: A neural network based approach for gray box identification. *Proc. 1994 IEEE Workshop on Neural Networks for Signal Processing*, Ermioni, Greece, in press (1994).
- [19] D. S. Broomhead and G. P. King, Extracting qualitative dynamics from experimental data. *Physica D*, vol. 20, pp. 217-236 (1986).
- [20] M. A. Kramer, Nonlinear principal component analysis using autoassociative neural networks. *AICHE J.*, vol. 37, pp. 233-243 (1991).
- [21] S. Usui, S. Nakauchi and M. Nakano, Reconstruction of Munsell color space by a five-layered neural network. *Proc. 1990 IEEE Int. Joint Conf. in Neural Networks*, vol. 2, pp. 515-520 (1990).
- [22] E. Saund, Dimensionality reduction using connectionist networks. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 305-314 (1989).
- [23] M. Kirby and R. Miranda, Nonlinear reduction of high-dimensional dynamical systems via neural networks. *Phys. Rev. Letters*, vol. 72, pp. 1822-1825 (1994).
- [24] D. DeMers, Dimensionality reduction for non-linear time series. SPIE. Vol. 1766, *Neural and Stochastic Methods in Image and Signal Processing*, pp. 200-211 (1992).
- [25] E. J. Kostelich and J. A. Yorke, Noise reduction: Finding the simplest dynamical system consistent with the data. *Physica D*, vol. 41, pp. 183-196 (1990).
- [26] G. Ertl, Oscillatory catalytic reactions at single-crystal surfaces. *Adv. Catalysis*, vol. 37, pp. 213-277 (1990).
- [27] S. Jakubith, H. H. Rotermund, W. Engel, S. A. von Oertzen and G. Ertl, Spatiotemporal concentration patterns in a surface reaction: Propagating and standing waves, rotating spirals and turbulence. *Phys. Rev. Letters.*, vol. 65, pp. 3013-3016 (1990).
- [28] K. Krischer, R. Rico-Martínez, I. G. Kevrekidis, H. H. Rotermund, G. Ertl and J. L. Hudson, Model identification of a spatiotemporally varying catalytic reaction. *AICHE J.*, vol. 39, pp. 89-98 (1993).
- [29] G. Berkooz, P. Holmes, and J. L. Lumle, The proper orthogonal decomposition in the analysis of turbulent flows. *Annu. Re. Fluid Mech.*, vol. 25, pp. 539-575 (1993).
- [30] K. Krischer, *Nichtlineare Dynamik zweier Grenzflächenreaktionen – Kinetische Oszillationen, Bifurkationen und Deterministisches Chaos*. Ph. D. Thesis, Free University Berlin, Germany (1990).

- [31] K. Krischer, M. Lübke, W. Wolf, M. Eiswirth and G. Ertl, Chaos and interior crisis in an electrochemical reaction. *Ber. Bunsenges. Phys. Chem.*, vol. 95, pp. 820-823 (1991).
- [32] E. J. Doedel and J. P. Kérévez, AUTO: A program for continuation and bifurcation problems in ordinary differential equations. *Applied Mathematics Report, California Institute of Technology*, USA (1986).
- [33] R. Rico-Martínez, I. G. Kevrekidis and R. A. Adomaitis, Noninvertibility in neural networks. *Proc. 1993 IEEE Int. Conf. on Neural Networks*, vol. I, pp. 382-386 (1993).
- [34] R. Rico-Martínez, I. G. Kevrekidis and R. A. Adomaitis, Noninvertible dynamics in neural network models. *Proc. 28th IEEE Annual Conf. on Information Sciences and Systems*, Princeton University, Princeton, USA, in press (1994).
- [35] F. J. Pineda, Generalization of back-propagation to recurrent neural networks. *Phys. Rev. Letters*, vol. 59, pp. 2229-2232 (1987).
- [36] J. R. Leis and M. A. Kramer, ODESSA - An ordinary differential equation solver with explicit simultaneous sensitivity analysis. *ACM Trans. Math. Software*, vol. 14, pp. 61-67 (1988).
- [37] R. M. Farber, A. S. Lapedes, R. Rico-Martínez and I. G. Kevrekidis, Identification of continuous-time dynamical systems: Neural network based algorithms and parallel implementation. *Proc. Sixth SIAM Conf. on Parallel Processing for Scientific Computing*, R. F. Sincovec, D. E. Keyes, M. R. Leuze, L. R. Petzold and D. A. Reed (eds), SIAM Publications, vol. I, pp. 287-291 (1993).