MODELING, IDENTIFICATION AND CONTROL, 1994, VOL. 15, NO. 4, 225–243
doi:10.4173/mic.1994.4.2

Chaotic time series

Part II. System identification and prediction

B. LILLEKJENDLIE†, D. KUGIUMTZIS‡
and N. CHRISTOPHERSEN‡

Keywords: *Nonlinear systems, chaos, prediction, time series, forecasting*

This paper is the second in a series of two, and describes the current state of the art in modeling and prediction of chaotic time series.

Sampled data from deterministic non-linear systems may look stochastic when analysed with linear methods. However, the deterministic structure may be uncovered and non-linear models constructed that allow improved prediction. We give the background for such methods from a geometrical point of view, and briefly describe the following types of methods: global polynomials, multi-layer perceptrons and semi-local methods including radial basis functions. Some illustrative examples from known chaotic systems are presented, emphasizing the increase in prediction error with time. We compare some of the algorithms with respect to prediction accuracy and storage requirements, and list applications of these methods to real data from widely different areas.

1. Introduction

The first paper in this series (Kugiumtzis *et al.* 1994) discussed the phenomenon of chaotic behaviour, that is the fact that seemingly stochastic time series can be generated from low dimensional deterministic systems. Chaotic systems are characterized by features such as strange attractors and positive Lyapunov exponents, which, when estimated from real data, are used to identify chaos. From this starting point, the focus of the present paper is system identification and prediction; identification is also called the 'inverse problem' in dynamical systems theory. Even under ideal conditions, a chaotic system shows an apparently random behaviour, but may still be identified using techniques from non-linear *deterministic* system identification. In some sense, a seemingly stochastic problem has a deterministic solution. However, in practice this is only partially true since the chaotic signal will often be corrupted by noise. But even with perfect reconstruction of the dynamical equations in the noise free case, only short term predictions are possible due to the extreme sensitivity of chaotic systems to uncertainties in initial conditions. This is because a chaotic system, while globally constrained to a finite region of state space, is locally unstable everywhere.

Prediction of chaotic time series is still a relatively new research topic, dating back to 1987. So far, we have identified more than 50 published articles in the field, with a steady growth. There may well be work that we are not aware of, and we apologise for any such omissions. Another review article, shorter than this paper, is West and Mackey (1992).

Received 10 December 1993.
† SWTEF-SI Pb. 124, Blindern, N-0314 Oslo, Norway.
‡ Department of Informatics, University of Oslo, Pb. 1080 Blindern, N-0316 Oslo, Norway.

Non-linear system identification and prediction at large is a diverse field with a plethora of potentially useful methods originating from different scientific disciplines. A broad set of these methods has been applied to chaotic systems and a survey in this area, which is the topic of the present paper, provides a useful comparison of different techniques. Hopefully, this will contribute both to spur the application of some of these methods to other areas of non-linear identification and prediction as well as providing useful feedback to the study of chaotic systems. As noted in our first paper, one should also realise that even if chaos *per se* is not of interest, some knowledge of this phenomenon is highly useful in many non-linear studies. This is because chaotic behaviour is a pervasive non-linear phenomenon, and many systems may therefore turn chaotic in parts of the parameter space.

To limit the length of this article, we have chosen to omit system identification in the fields of fractals. Work like the Collage theorem by Barnsley (Barnsley 1988) can be viewed as identification of system dynamics in the complex plane with applications mainly to data compression, whereas we focus on more conventional time series prediction.

In the background section below, the notation as well as some basic mathematical little theory has yet evolved in this area. For some initial work, see Grassberger *et al.* (1991) and Lichtenberg and Lieberman (1992).

In the background section below, the notation as well as same basic mathematical concepts are given together with simple examples. The fundamental ideas behind each major approximation method class are then treated in sections covering global polynomials, multi-layer neural network perceptrons, local polynomials, and semi-local methods, including radial basis functions. Here we given brief references to some applications. In section 6 we discuss and compare the advantages of each method through tables and figures compiled from various sources.

2. Notation and mathematical background

For chaotic systems, delay coordinates are commonly used to reconstruct a state space. We consider coordinates derived from scalar time series, but there are no principal difficulties in applying the same theory to multivariate observations. A general treatment of the so called embedding technique is given in Kugiumtzis *et al.* (1994). For simplicity, in this article we let the delay time τ denote the fixed interval between observations (the sampling interval in the case of a continuous system), and consider the discrete scalar time series: $x_k = x(k\tau)$, where k is an integer. The delay state vector at time $t = k\tau$ is defined as

$$\boldsymbol{x}_k = [x_k, x_{k-1}, \ldots, x_{k-(m-1)}]^{\mathrm{T}}, \tag{1}$$

where m is the embedding dimension and T denotes the transpose. Note that the first element of the vector $\boldsymbol{x}_k$ is the sample value at time $k\tau$.

There are two equivalent ways of expressing the map from time $k\tau$ to $(k+1)\tau$,

$$\boldsymbol{x}_{x+1} = \boldsymbol{f}(\boldsymbol{x}_k) \quad \boldsymbol{f}: \mathscr{R}^m \mapsto \mathscr{R}^m, \tag{2}$$

$$x_{k+1} = f(\boldsymbol{x}_k) \quad f: \mathscr{R}^m \mapsto \mathscr{R}, \tag{3}$$

where the vector field $\boldsymbol{f}$ is related to scalar field f as $\boldsymbol{f}(\boldsymbol{x}_k) = [f(\boldsymbol{x}_k), x_k, \ldots, x_{k-(m-2)}]^{\mathrm{T}}$. For a chaotic system, we basically only know that f is non-linear. However, we will assume that f is at least continuous and also continuously differentiable if needed. It is useful to

note that geometrically equation (3) defines an m dimensional surface (manifold) in $\mathscr{R}^{m+1}$. By embedding space we mean $\mathscr{R}^m$, and we denote the space $\mathscr{R}^{m+1}$ where the surface exists as the graph space.

If the time series stem from a chaotic system in its asymptotic state on a strange attractor of dimension d, Takens' theorem (Takens 1981), and its extension (Sauer *et al.* 1991), states that $m \geq 2[d]+1$ components in the delay coordinate vector are sufficient to reconstruct the attractor for almost all dynamical systems, where $[d]$ denotes the smallest integer that is larger than d. This implies that the vectors $\boldsymbol{x}_k$ all lie on the finite attractor in $\mathscr{R}^m$ and that the observation pairs $(x_k, \boldsymbol{x}_{k-1}), (x_{k-1}, \boldsymbol{x}_{k-2}), \ldots, (x_2, \boldsymbol{x}_1)$ lie on the manifold generated by f in $\mathscr{R}^{m+1}$. The identification problem amounts to constructing an approximation $\hat{f}$ to f given the observation pairs. This is a well known approach to system identification as outlined for instance in Ljung (1991). The problem of approximating a manifold in $\mathscr{R}^{m+1}$ given points on, or near to its surface in the case of noise, is a central problem in numerical approximation theory as well as in statistical non-linear regression. Through Takens' theorem, identification of chaotic systems is put on a firm mathematical footing and is shown to be a non-linear identification problem. In practice, the embedding dimension m may be estimated by different methods and values less than $2[d]+1$ may be feasible (Kugiumtzis 1994). Choosing m too large is not a problem in principle, but will certainly lead to a higher computational burden than necessary, and probably a less accurate predictor.

As a simple example, assume the well known logistic map $x_{k+1} = 4x_k(1-x_k)$. A series of observations from this map looks like noise, and the autocorrelation of the data is as for white noise. Linear techniques will therefore be of no use in predicting such a time series, but it is clear from the map itself than $m=1$ will suffice to embed the attractor. In this case it is rather simple to build an approximation $\hat{f}$ to f from the observation pairs $(x_k, x_{k-1}), (x_{k-1}, x_{k-2}), \ldots, (x_2, x_1)$ as illustrated in Fig. 1 (*a*). Here 20 pairs are plotted and the underlying shape of the one-dimensional graph generated by f in $\mathscr{R}^2$ is clearly seen. If $m=2$ was chosen, the result would be points on a one-dimensional curve in $\mathscr{R}^3$. As a second example, 200 observation pairs from the Hénon map, which may be expressed as $x_{k+1} = 1 - 1{\cdot}4\,x_k^2 + 0{\cdot}3x_{k-1}$ (Henon 1976), are shown for $m=2$ in Fig. 1 (*b*). The domain of variation in embedding space is the attractor which is recognized in the figure as lying on the surface in $\mathscr{R}^3$ generated by $f(x, y) = 1 - 1{\cdot}4x^2 + 0{\cdot}3y$.

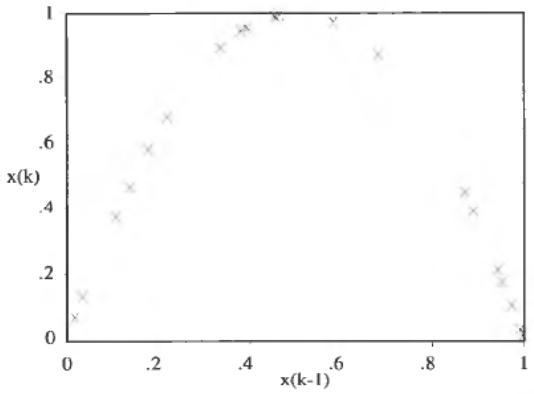
To assess $\hat{f}$, the normalized prediction error e over a set of samples with N elements is used:

$$e = \sigma_\delta / \sigma_x, \tag{4}$$

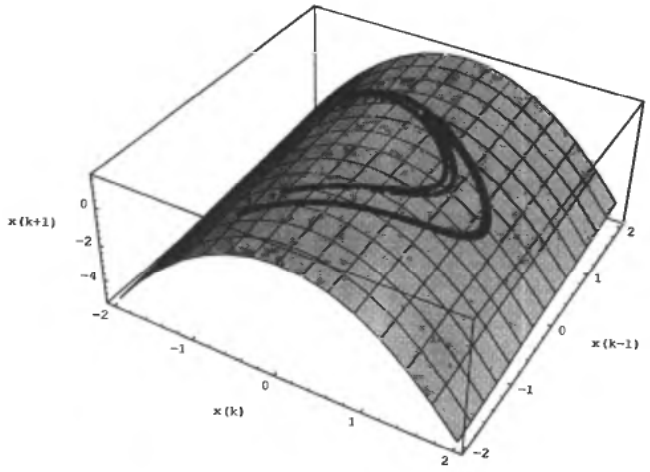
where σ_δ is the root mean square prediction error given by $\sigma_\delta^2 = \frac{1}{N}\sum_{i=1}^{N}(x_i - \hat{f}(\boldsymbol{x}_{i-1}))^2$, and σ_x is the sample standard deviation, $\sigma_x^2 = \frac{1}{N}\sum_{i=1}^{N}(x-\bar{x})^2$, where $\bar{x}$ denotes the average of the x values. If $e \approx 0$, the prediction is almost perfect, whereas an e value equal to 1 is equivalent to using the average as the predictor.

We will mainly think of identification in batch mode, where the model is built to minimize the sum of the root mean square error over all samples in a training set. Good statistical practice dictates that the prediction error e of the estimated model should not be computed from samples used to construct $\hat{f}$, but over a separate test set. When few observation pairs are available, the standard technique is cross validation (Stone 1977).

On-line applications are feasible when the methods described here are used to continuously update $\hat{f}$; think of this as an analogue to a Kalman filter doing on-line



(*a*)



(*b*)

Figure 1. (*a*) Samples from the logistic map time series plotted in the two dimensional delay coordinate space show the form of the system function $4x(1-x)$. (*b*) The Henon attractor embedded in 3-d space.

model parameter estimation. A non-linear model estimated off-line can also be used in the predicting step of an on-line extended Kalman filter.

Maps $\hat{f}$ approximating f in (3) are one-step predictors. If it is desirable to predict more steps ahead in spite of the escalating uncertainty, say $r>1$ steps, one can repeat the one-step prediction $\hat{f}$ r-times. Alternatively one may estimate the r-th iterate $x_{k+r}=f^{(r)}(\mathbf{x}_k)=f(f(\cdots f(\mathbf{x}_k)))$ directly. In Farmer and Sidorowich (1987) and Casdagli (1989) it is argued that iterated predictors outperform direct ones. Intuitively, when the prediction horizon r increases, the function $f^{(r)}$ gets very complex and hard to approximate, which is illustrated in Fig. 2 showing the logistic map together with its 2nd and 6th iterates. Direct approximation of f over only a few time steps quickly becomes intractable because of the wild oscillations occurring.

In Abarbanel, Brown and Kadtke (1989, 1990) is it shown that minimizing the prediction error σ_δ often leads to a $\hat{f}$ which does not reproduce dynamic invariants in the original data like Lyapunov exponents and the density of points on the attractor. They suggest that the performance criterion should also include a fit to these invariants,

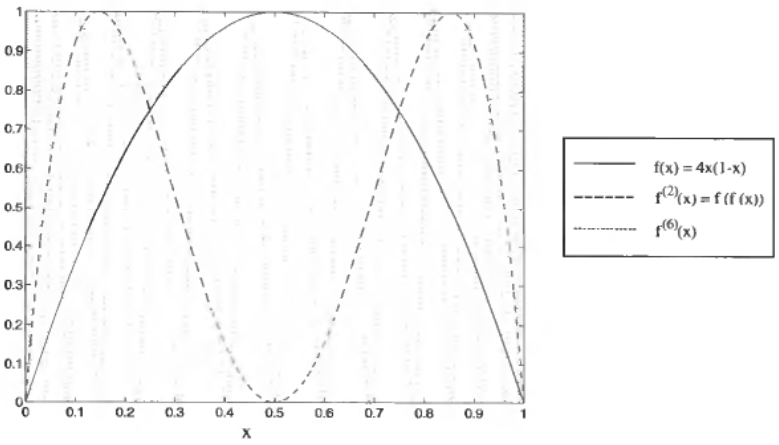


Figure 2. The iterates $f^{(r)}$ of a process get more complex as r increases.

and achieve this by not only predicting x_{k+1} as $\hat{f}(\boldsymbol{x}_k)$, but as a linear combination of the L first iterates $\hat{f}(\boldsymbol{x}_k), \hat{f}(\hat{\boldsymbol{f}}(\boldsymbol{x}_{k-1})), \ldots, \hat{f}(\hat{\boldsymbol{f}}(\cdots(\hat{\boldsymbol{f}}(\boldsymbol{x}_{k-L}))$. Since high iterates defy approximation, L should be low. A system identified this way shows slightly larger prediction errors, but will reproduce the general behaviour of the underlying system better.

An important property of chaotic systems facilitates the approximation task. A sufficiently long time series produces a sequence of vectors $\boldsymbol{x}_k$ that is approximately dense on the attractor, meaning that no new $\boldsymbol{x}_k$ will be far from those already observed. Thus, borrowing a term from the identification field, a chaotic system is in some sense 'persistently excited'. Usually, 'persistently excited' refers to an input which has a rich enough frequency content to guarantee correct parameter estimation. The analogy in chaotic systems is that given enough samples, the system has been so close to all possible points in state space that an accurate approximation can be guaranteed.

Further, a low-dimensional attractor occupies only a fraction of the higher dimensional embedding space, which can significantly reduce storage requirements in approximation problems.

3. Global approximation methods

3.1. *Global polynomials*

An obvious approximation $\hat{f}$ to f (or $\hat{\boldsymbol{f}}$), is a polynomial in the m delay coordinate variables of degree p, set by the user. Polynomials can be written

$$\hat{f}(\boldsymbol{x}_k) = \sum_i w_i \phi_i(\boldsymbol{x}_k), \tag{5}$$

where w_i are parameters and the basis functions ϕ_i are powers and cross products of the components in $\boldsymbol{x}_k$. Small adjustments to the weights w_i will cause the map to change almost everywhere, and therefore this method is classified as global.

Since the parameters enter linearly, they can be fitted to the data using standard least squares techniques involving the normal equations—in practice often done by singular value decomposition (Press *et al.* 1988). A potential advantage is that the parameters may be estimated recursively (and therefore in real time as measurements accumulate), using a Kalman-filter-like algorithm (Söderstrom 1989). Giona *et al.* (1991) showed an alternative and efficient way of estimating the parameters using

orthonormal polynomials and assuming ergodicity. In that case the multivariable parameter estimation problem can be reduced to simple computations involving sums of powers of the variables in the delay coordinate vector.

The simplest first order polynomial approximator is the well known Auto Regressive (AR) model (Priestley 1981):

$$\hat{f}(\mathbf{x}_k)=\sum_{i=0}^{m-1} w_i x_{k-i}+w_m, \tag{6}$$

which geometrically amounts to fitting an m dimensional hyperplane to the data in $\mathscr{R}^{m+1}$, and is thus a global linear model. This is an unsuitable model for predicting chaotic time series, but some authors have used AR models as benchmarks with which to compare non-linear techniques. Among the work trying global polynomials for predicting chaotic time series, we mention Lapedes and Farber (1987), Farmer and Sidorowich (1987), Casdagli (1989, 1991), Pawelzik and Schuster (1991), and Giona *et al.* (1991). We refer to section 6 for a discussion of the quality of such global polynomial approximants.

A disadvantage of polynomials is that the number of independent parameters equals $\binom{m+p}{m}$, which gets intractably large as p increases. Many independent parameters also increase the risks of overfitting noisy time series and higher order polynomials may show strong oscillations between samples. In addition, some scalar fields are not well approximated by polynomials and in such cases rational functions may be used. The reason is basically that rational functions may have poles. If the underlying function has poles, even in the complex extension, these poles may ruin real valued approximations by plain polynomials. Rational functions were tried in Casdagli (1991).

3.2. *Multi-layer perceptron neural networks*

Another class of global methods which have been applied to chaotic time series is multi-layer perceptron (MLP) neural networks. These have an elaborate structure with sigmoid shaped basis functions like for example $\phi(x)=\tanh(x)$ or $\phi(x)=1/(1+e^{-x})$, and are probably the most commonly used neural networks. The building blocks in a neural network are the 'nodes', which is just one basis function with some preprocessing of the input, typically an inner product. As a simple example, an MLP net with two input variables, three hidden nodes and one output node defines a function from $\mathscr{R}^2$ to $\mathscr{R}$, as illustrated in Fig. 3. Such a net can be written

$$\hat{f}(\mathbf{x}_k)=\phi\left(\sum_{i=0}^{2} w_i\phi(w_{0,i}x_k+w_{1,i}x_{k-1})\right). \tag{7}$$

The w_i and $w_{j,i}$ are real valued parameters, denoted weights in neural net terminology. A full net may have any number of layers and any number of nodes in each layer. In contrast to global polynomials, the weights in MLPs do not enter linearly, so iterative parameter estimation is required. Deriving the values of the weights in the net is in most cases done by back propagation, which is a steepest descent search (Rumelhart and McLelland 1986). As with polynomials, such MLP nets may approximate any smooth function $f:\mathscr{R}^m\mapsto\mathscr{R}$ to any degree of accuracy, given enough sigmoid functions with accompanying weights. A standard proof is found in Cybenko (1989).

In Lapedes and Farber (1987), MLP nets were applied for the first time to predict chaotic time series, namely data obtained from the Mackey–Glass equation (Mackey and Glass 1977). Among other work, we mention Elsner (1992) who applied a standard

MLP to the Lorenz attractor (Lorenz 1963), and extended this work in Elsner and Tsonis (1992) to cover data from a controlled fluid dynamics experiment as well as estimates of sea surface temperatures. Cadden (1991) predicted corporate bankruptcy, Lowe and Webb (1991) did prediction on van der Pol's oscillator and the thalamic neuron. We would also like to mention the work of Weigend *et al.* (1990, 1992) and Welstead (1991).

Typical disadvantages of standard MLP nets are the long parameter estimation time and potential local minima. To improve this, Wolpert and Miall (1990) tested a variation of the back propagation parameter estimation algorithm with a momentum term to speed up convergence, and simulated annealing (Kirkpatrick *et al.* 1983) to avoid local minima. They analysed various data sets, including the logistic map and real data from two biological systems. A similar approach is found in Rögnvaldsson (1993). Another successful approach to fast estimation algorithms, is that of Deppish (Deppish *et al.* 1991), which devised a hierarchical way of structuring and estimating the weights in a sigmoid MLP. They tested the system behaviour on the logistic map and the Rössler system (Rössler 1976), and reported an improvement in parameter optimization time of approximately three orders of magnitude.

4. Local methods

One of the main disadvantages of global methods is that a new sample pair $(x_k, \boldsymbol{x}_{k-1})$ may change $\hat{f}$ everywhere. Local interpolation overcomes this drawback by utilising only a limited number, say s, of neighbouring samples. There are two major classes of local methods, those applying neighbour samples directly in the prediction, and those fitting a function locally to the neighbours basing the prediction on the estimated function.

The simplest way to predict x_{k+1} from neighbour samples, is to identify the nearest neighbour to $\boldsymbol{x}_k$ in the embedding space $\mathscr{R}^m$. We denote the nearest neighbour to $\boldsymbol{x}_k$ by $\boldsymbol{x}_{k(1)}$, and the next sample $x_{k(1)+1} = f(\boldsymbol{x}_{k(1)})$ is then known from the time series, and can be used as the predictor. This was suggested by Lorenz (1969), and is equivalent to building a look-up table of previous state mappings. In terms of the original time series, one finds the segment of length m that is 'most similar' to $x_k, x_{k-1}, \ldots, x_{k-(m-1)}$ and then uses the sample following that segment to predict x_{k+1}, in other words $\hat{f}(\boldsymbol{x}_k) = x_{k(1)+1}$. This is also termed the 'analogue method'. In Kennel and Isabelle (1992), the method is used on a number of simulated data sets to distinguish chaos from coloured noise.

An improvement is to take the s nearest neighbours and use the average of their state mappings as the predictor. Another variant was suggested in Sugihara and May (1990). They selected $s = m+1$ (not necessarily closest) neighbours to form the smallest m-dimensional simplex circumscribing $\boldsymbol{x}_k$ in $\mathscr{R}^m$. $\hat{f}(\boldsymbol{x}_k)$ is then computed as a weighted sum of the mapped simplex corners. Besides synthetic data, they experimented with time series from measles, chicken pox and diatom populations (plankton). In Cortini, Cilento and Rullo (1991) and Cortini and Barton (1993), the method of Sugihara and May was used to predict vertical ground movements of an active caldera on Italy. In Linsay (1991), an alternative formulation of the same algorithm was tested on driven semiconductors and the Lorenz attractor. Yet another variation is found in Mees (1992) which applies Voronoi tessellation methods from computational geometry (Preparata and Shamos 1985) to build linear patches (tiles) in the $m+1$ dimensional graph space.

A common mathematical formulation for all these methods is

$$\hat{f}(\boldsymbol{x}_k) = \sum_{i=1}^{s} f(\boldsymbol{x}_{k(i)})\phi(\|\boldsymbol{x}_k - \boldsymbol{x}_{k(i)}\|), \tag{8}$$

where $\boldsymbol{x}_{k(i)}$ denotes the i-th closest vector to $\boldsymbol{x}_k$ in $\mathscr{R}^m$, s is the number of neighbours, and $\|\cdot\|$ denotes a norm. Usually ϕ is a weight function increasing from zero to one when $\boldsymbol{x}_k$ approaches $\boldsymbol{x}_{k(i)}$. Note that there is no parameter estimation involved here, ϕ is a fixed function. Thus, this method is efficient in terms of computation time. However, the approximating maps are generally not continuously differentiable, and the search for neighbours become more time consuming as more vectors are stored.

The other class of methods fits a surface in graph space $\mathscr{R}^{m+1}$, as described in section 2, to the measurement points $(x_{k(i)+1}, \boldsymbol{x}_{k(i)})$, $i = 1, 2, \ldots, s$. This may be a plane, but polynomials of higher degrees may also be used to interpolate between neighbours. Taking $s > m+1$ and fitting a plane, one obtains a local AR model, also called a local linear model. For chaotic time series, this was, as far as we know, first done in Farmer and Sidorowich (1987). They experimented with such local AR models, as well as with higher order polynomials, but did not observe significant improvements moving to higher order. For comparison, they also applied a global AR model as a 'standard forecasting technique'. Casdagli (1991) continued to explore the relation between global and local AR models. Other work applying local methods is Casdagli (1989), Tenorio and Lee (1989), Pawelzik and Schuster (1991), Townshend (1992), and Hunter (1992). Various versions of these techniques are well known in system identification, see for example Tong (1990).

5. Semi-local methods

Semi-local methods may combine the best of two worlds, the smoothness of global predictors and the localized dependence on new information of local predictors. Well-known classes of semi-local approximants are splines and radial basis functions (RBF). For radial basis functions, three research communities exist. Approximation theorists are concerned with topics like convergence properties, for example Powell (1987), people in the neural network community approach the problem from a more algorithmic point of view, see Lee and Kil (1991), and statisticians have their well developed field of kernel estimators, as described in Scott (1992). To our knowledge, prediction of chaotic time series has only been considered from the standpoint of neural networks.

RBF-approximation can be thought of as a combination of the fitting and weighting approaches described in the previous section on local methods—weights are assigned according to the distance from the basis function centres, but these weights adjust parameters, not the next sample value as in (8). Applying s basis functions, such approximants take the form

$$\hat{f}(\boldsymbol{x}_k) = \sum_{i=1}^{s} w_i \phi_i(\|\boldsymbol{x}_k - \xi^i\|), \tag{9}$$

where the function $\phi_i(r)$ is radially symmetric around a centre value ξ^i in $\mathscr{R}^m$, and w_i are weights to be chosen. If w_i are the only parameters to estimate, the normal equations can be utilized. If, however, there are parameters inside ϕ_i which enter the problem nonlinearly, time consuming iterative optimization methods must be applied. As an intuitive visualization of how radial basis functions work, consider Fig. 3 and imagine the smooth step functions replaced by, for example, Gaussian hats, (and a linear combination at the last level).

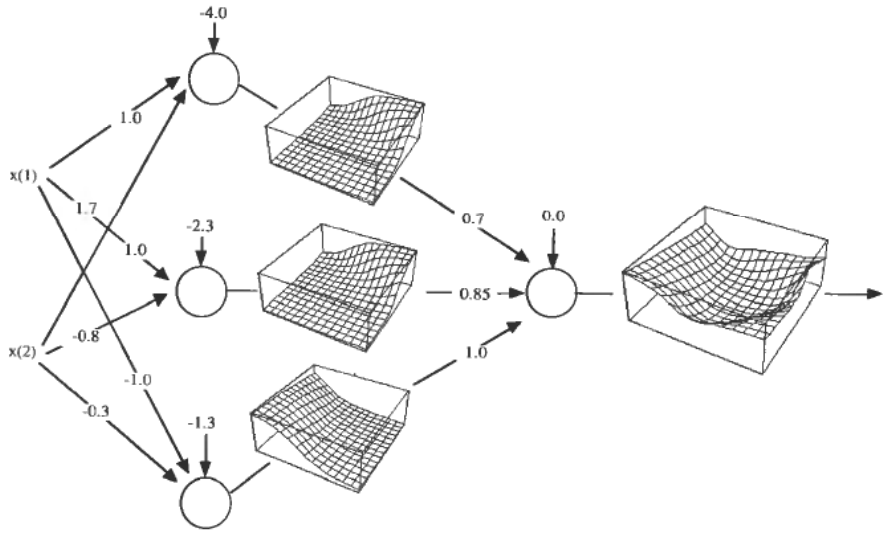


Figure 3. A multi-layer perceptron with two input variables, three hidden nodes and one output variable implements a function from $\mathscr{R}^2$ to $\mathscr{R}$.

There are two main areas for modifying the basic scheme: experiments with various types of basis functions ϕ_i (section 5.1), and experiments with various algorithms for determining the parameters, especially the centres ξ and the number s of basis functions (section 5.2).

5.1. *Choosing the basis function type*

Various results prove that different types of RBF functions ϕ_i are universal approximants, that is, any smooth function can be approximated to any degree of accuracy given enough basis functions. This, in turns, requires an infinite amount of noise free measurement data. We refer to Park and Sandberg (1993) for a neural network approach, Scott (1992) for a statistical approach and Michelli (1986) for a result from approximation theory.

The most popular type is rotation symmetric Gaussian hats of the form $\phi_i(r)=\exp(-r^2/2\sigma_i^2)$, where the hat widths σ_i are fixed constants. Even though Gaussian hats are global in theory, they decrease fast enough to have finite support for all practical purposes. The basis function can also be non-local like the multi-quadratic $\phi_i(r)=(r^2+\alpha^2)^{-1/2}$, α and r real. For a more extensive list of basis functions, see Carlin (1991), and compare Franke (1982) for a number of related methods viewed from numerical mathematics.

In higher dimensions, the locality advantage of Gaussian hats turns into a disadvantage. The basis functions are local, and since data is almost always scarce in higher dimension, most points in state space have no basis function cover. Another source of difficulty is that some input variables x_{k-j} may be almost uncorrelated with the output variable x_k, especially if the embedding dimension m is estimated too large. There are two typical ways to improve this situation, either by letting the hats smear out in some directions and become Gaussian 'ridges', or by normalizing the basis functions. The normalized Gaussian hats going into (9) are written

$$\phi_i(\|\boldsymbol{x}_k-\xi^i\|)=\frac{\exp(-\|\boldsymbol{x}_k-\xi^i\|^2/2\sigma_i^2)}{\Sigma_{j=1}^{s}\exp(-\|\boldsymbol{x}_k-\xi^j\|^2/2\sigma_j^2)}, \tag{10}$$

and they are named weight constant predictors (WCP) in Stokbro, Umberger and Hertz (1990) and Stokbro and Umberger (1992). They also suggested a method called weighted linear predictors (WLP) where the simple weights w_i in (9) are replaced by the linear term $v_i + (\boldsymbol{x}_k - \xi^i) \cdot \boldsymbol{d}_i$ where v_i is a scalar parameter and $\boldsymbol{d}_i$ is a parameter vector. In Mead *et al.* (1991, 1992), the same method was tested on the Mackey–Glass equation, and in Abarbanel *et al.* (1989, 1990), a slightly modified weighting function was used. The WLP method was applied in Pawelzik and Schuster (1991) with e^{-K_i} as the weight function, and neighbour samples taken from identified unstable periodic orbits (UPO) close to $\boldsymbol{x}_k$. Generic strange attractors can be approximated by unstable periodic orbits of a given length, and methods to identify such orbits are given in the same article. K_i was the sum of the positive Lyapunov exponents of each neighbour UPO.

In Hartman and Keeler (1991), ridge functions were applied to the Mackey–Glass equation. An alternative view was proposed by Deco and Ebmeyer (1993), observing that a multi-dimensional Gaussian is equal to a product of scalar Gaussian with adjustable centres and widths. During learning, hat parameters are adjusted when a suitable Gaussian covers the input, otherwise a new hat is generated. In the product they use as few terms as possible, and new variables are introduced only when required, thus converting ridges into hats one variable at the time. In this way, the system identification method itself decides which previous variables that are sufficiently correlated to warrant inclusion into the model, estimation of the best embedding dimension m becomes an integral part of the model building algorithm. This algorithm is named 'CC-RAN'. Another related tree-building algorithm was devised in Sanger (1990, 1991).

5.2. *The number of basis functions and their centres*

In the most basic RBF method, there is one basis function at each sample so that $\xi^i = \boldsymbol{x}_i$. To reduce the computational burden, only the s nearest samples are often taken into account. Now, $\xi^i = \boldsymbol{x}_{k(i)}$ and the number of terms in (9) is reduced. This approach was taken in Casdagli (1989), using non-local basis functions r^3 and the $s = 50$ nearest samples. The standard RBF method interpolates the data, and is thus sensitive to noise. To reduce this problem, Broomhead and Lowe (1988) put a fixed number of basis functions on a regular grid. By letting the number of basis functions be less than the number of samples, the data was smoothed. They applied this method to predict the logistic map.

In higher dimensions, regular grids become infeasible because the number of basis functions grows exponentially with the dimension of the grid. In addition a chaotic attractor occupies only a small subset of the entire space, so most of the basis functions are superfluous. The solution is to represent only those parts of the embedding space $\mathscr{R}^m$ where data exists, that is the manifold occupied by the attractor of the system. Thus, the memory requirements can be made proportional to the attractor's size, which also improves model estimation time and noise robustness.

One such algorithm, here denoted 'hashing RBF', is described in Moody (1989a). Basis functions are maintained on a regular grid with spacing λ, but are only constructed at those grid points where data exists. A grid coordinate is related to the corresponding function parameters through a hash table, and once the neighbours are found, we are back to a summation model $\hat{f}_\lambda(\boldsymbol{x}_k)$ on the form (9). This hash table scheme was originally invented by Albus (1975a, b) for real time motion control in robotics, which is possible since a hash table makes the look-up extremely fast. Without detailed *a priori* information of the distribution of data in the input space, it is difficult to choose

an optimal lattice spacing—a coarse grid will smooth data well, whereas a fine grid will capture details. A hierarchy of hash models spaced on grids with increasing resolution λ, written $\hat{f}(\boldsymbol{x}_k)=\Sigma_\lambda \hat{f}_\lambda(\boldsymbol{x}_k)$, will thus represent the major function structure on the coarsest scale grids, and add model details at finer grid resolutions.

Another idea for reducing the number of basis functions, is to cluster neighbouring sample vectors and represent them all with one basis function at the cluster centre ξ^i. When all training samples are collected, the clusters may be formed with for example the k-means clustering algorithm (MacQueen 1967). In Moody and Darken (1988, 1989b), they applied a variation of this k-means clustering algorithm to centre the clusters, used the average distance of the neighbouring hats as hat width, and estimated the weights w_i.

The cluster centres and widths may also be built as an integral part of the parameter estimation, and one of the first descriptions of such algorithms was in Lee (1988). These algorithms add a new basis function only when no existing function covers the new sample, or if the existing functions cannot easily be changed to approximate the new sample. As more samples arrive, the hat widths are decreased gradually, leading to increased estimation accuracy. Usually such methods call for iterative estimation algorithms. In Platt (1991) a version of this algorithm was implemented, named 'RAN' as an acronym for resource allocating nets.

The automatic addition of basis functions during model identification represents one solution to the so-called problem of model structure selection or model realization. In statistics, there is an emerging theory of how to select the number of basis function, with cross validation and bootstrap methods as some of the themes (Efron 1982). Another challenge lies in selecting the form of the basis functions themselves, as well as identifying which variables should go into the model. This is different from estimating the parameters in a fixed structure, which is usually what identification amounts to. Realization theory as described in Casti (1977) is well developed for deterministic linear systems, but a fundamental theory lacks for non-linear systems (compare Deller 1989).

6. Discussions and comparisons

At this stage, a set of rules recommending certain methods for certain classes of problems, would be desirable. Unfortunately, such general advice is yet unavailable—there is too little experience gained from actual use. Instead, we have settled for the more modest goal of comparing experimental results given in the literature by compiling tables and figures. The figures and tables will hopefully give a feeling of the performance one can expect from the various methods.

The quality of the constructed approximants depends on many factors including: the underlying system, the number of samples available, the embedding dimension, the noise in process and measurements, the kind of approximator, the number of parameters in the approximator, the amount of computer and human resources invested in constructing the approximator, and the prediction interval. Based on results reported in the literature, we have chosen interesting experimental results where only one, or just a few, of these factors vary, the remaining factors remaining 'relatively' fixed. The following tables and figures have been collated:

- Table (1) hints at how well different methods approximate the Mackey–Glass delay differential equation.
- Table (2) compares the approximation error for local AR models (local linear polynomials) with global AR models for different systems.

- Figure (4) shows how fast the prediction error increases with time for different methods and different test sets.
- Figure (5) shows how prediction accuracy is connected with the number of parameters and the number of samples for some methods applied to the Mackey–Glass equation.

Even though we have attempted to extract experimental results which are as comparable as possible, comparisons like these can never be completely fair, and only show the applicability of each method to the type of data chosen.

In all tables and figures, prediction error means normalized root mean square prediction errors as given in (4).

To give an impression of how well the various methods approximate a fixed system, we have in Table 1 collected prediction errors from experiments on the standard noise free Mackey–Glass equation with delay parameter $\Delta = 17$, with 500 training sets, embedding dimension $m = 4$, and sampling interval $\tau = 6$ time units. The difference Mackey–Glass equation is written

$$x_{k+1} = \frac{0{\cdot}2x_{k-\Delta}}{1+[x_{k-\Delta}]^{10}} - 0{\cdot}1x_k.$$

Most of the results are reported by the originators of the methods, presumably assuring maximum performance. Different number of samples were used in the test sets, typically either 500 or 1000. The number of model parameters differed between the tests. This is reasonable since each method should be allowed to apply an optimal number of parameters. Unfortunately, some reported prediction error six time units ahead, others eighty-four, so the table reports results for both prediction intervals. In spite of this, the table should give an indication of how well the different methods approximate this Mackey–Glass system.

As can be seen from Table 1, no single method excels for one-step prediction six time units ahead, but global rationals and the weighted constant map (WCP) give the worst

	Prediction time = 6		Prediction time = 84	
Method	e (%)	Reference	e (%)	Reference
Global polynomials	1·1	Casdagli (1989)		
Global rationals	7·2	Casdagli (1989)		
Multi-layer perceptron	1·0	Lapedes (1987)	5·0	Platt (1991)
Method of analogy			25·1	Moody (1988)
Local linear polynomials	3·3	Casdagli (1989)	4·5	Stokbro (1992)
Local quadratic polynomials	1·3	Casdagli (1989)		
Standard RBF	1·1	Casdagli (1989)	15·8	Moody (1988)
K-means RBF			9·3	Moody (1988)
Adaptive clustering RBF			7·0	Platt (1991)
Hashing RBF			5·0	Moody (1989)
Weighted constant map	6·0	Stokbro (1992)		
Weighted linear map	1·3	Stokbro (1992)	3·0	Stokbro (1992)
Ridge functions			8·0	Hartman (1991)
Coarse coding RBF			5·5	Deco (1993)

Table 1. Normalized root mean square prediction error for a number of different approximation schemes with two different lengths of the prediction interval. Data is from the Mackey–Glass delay difference equation.

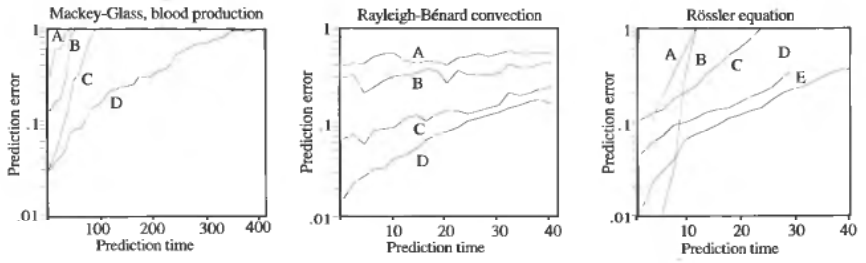


Figure 4. The prediction error (ordinate) as a function of prediction time (abscissa). Mackey–Glass: Curves A, B, C, and D are iterated and non-iterated global polynomials, and non-iterated and iterated MLP respectively. Rayleigh–Bénard: Curve A is a global linear model, B, C, and D are local linear methods with embedding dimensions 4, 6 and 15. Rössler: Curve A is a global linear model, B a local linear model, C global universal periodic orbits (UPO), D local UPO, and E a global MLP model.

predictions. For fourteen iterated one-step predictions (eighty-four time units), multilayer perceptrons, local AR models, hashing RBF, weighted linear predictors (WLP), and CC-RAN all give prediction errors of approximately the same size. The only 'bad' approximants here were the method of analogy and standard RBF.

Note that this table indicates that the prediction error increases with increasing prediction interval, which is quite natural. To illustrate this point further, we have in Fig. 4 collected prediction error as a function of prediction interval for three different systems and different system identification methods. The three systems are the Mackey–Glass equation, the Rayleigh–Bénard convection, and the Rössler equation. Again, the experimental conditions are similar enough to compare the results.

In Fig. 4, curves for the Mackey–Glass equation with delay parameter $\Delta = 30$ are reproduced from Lapedes (1987). The curves A, B, C and D are iterated and non-iterated global polynomials, and non-iterated and iterated MLP, respectively. These curves show how the prediction error increases with prediction time, and how non-linear methods may increase the prediction interval. Further, it can be seen that the iterated MLP predictor outperforms the direct long-time MLP predictor.

The Rayleigh–Bénard part of the same figure is based on Farmer and Sidorowich (1987), which used convection in an ^{3}He–^{4}He mixture with Rayleigh number $R/R_c = 12{\cdot}24$ and fractal dimension $\approx 3{\cdot}1$. Curve A is global linear model with embedding dimension $m = 15$ (i.e. $15+1$ parameters). Curves B, C, and D are local linear methods with embedding dimension 4, 6 and 15 respectively; all predictors iterated. The improvement of local AR models over the global one should be obvious from this figure.

In the third part, we reproduce results for predicting the Rössler equation. The embedding dimension was 4, the sampling interval 0·87 time units, and 100 samples were used to train the hierarchical MLP net. The curve A is from a global linear model, B a local linear model, C from global universal periodic orbits (UPO), D from local UPO, and E a global MLP model. Curves A to D are taken from Pawelzik and Schuster (1991), and E from Deppish, Bauser and Geisel (1991). For this system, both global AR, but also local AR models give predictions which soon become totally unreliable, the universal periodic orbits give better predictions, but the MLP net is the most accurate approximator in this case.

Instead of varying the approximation method keeping most of the other factors constant, it is interesting to apply a small number of methods to a variety of processes,

Data set description	Data type	m	e_{NL}	e_{AR}
Mackey–Glass	Simulated	4	0·2	0·4
Mackey–Glass	Simulated	6	0·03	0·4
Ikeda map 0% noise	Simulated	5	<0·02	0·9
Ikeda map, 2% noise	Simulated	5	0·06	0·9
Ikeda map, 20% noise	Simulated	5	0·5	0·9
Two coupled diodes	Lab. data	0·3	0·9	
Four coupled diodes	Lab. data	7	0·5	0·9
Weak fluid turbulence	Lab. data	20	0·01	0·4
Strong fluid turbulence	Lab. data	20	≈0·22	0·16
Flames, non-chaotic	Lab data	20	0·05	0·1
Flames, weak-chaotic	Lab. data	20	0·12	0·25
Flames, strong-chaotic	Lab. data	20	≈0·7	0·56
Speech	Natural data	20	0·2	0·3
EEG, resting patient,	Natural data	20	≈0·7	0·54
EEG, with anaesthesia,	Natural data	20	≈1·2	0·9
Measles	Natural data	2	0·23	0·27
Sunspots	Natural data	6	0·36	0·44

Table 2. Normalized root mean square prediction errors for a global AR model (e_{AR}) and the best non-linear model found with local linear AR models (e_{NL}) are compared. m is the dimension of the coordinate delay vector. From Casdagli (1991).

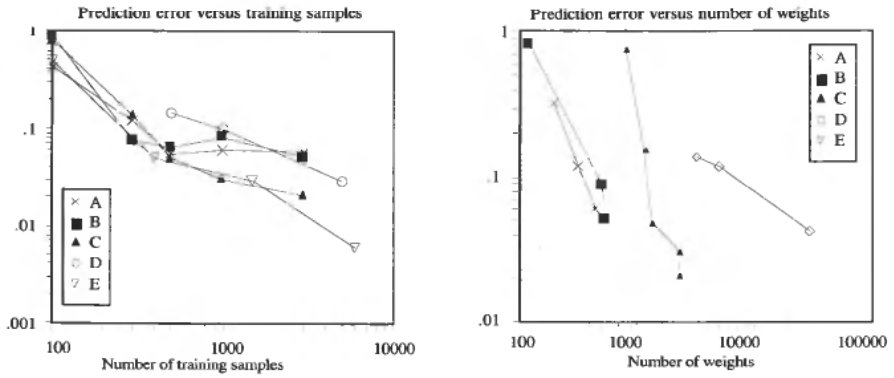


Figure 5. The prediction error as a function of the number of training data and the number of parameters. Curves A to E are CC-RAN method, adaptive clustering RBF, hashing RBF, K-means RBF, and Standard RBF respectively.

both simulated laboratory experiments and real life data sets. Such a table can be compiled from Casdagli (1991). In his article, local AR models are computed for all degrees of 'locality' varying from the method of analogy to a global AR model. We have in Table 2 chosen the optimal local AR model and compared it with the estimated global AR model. The results can briefly be summarized as follows. With much noise and/or high dimensional systems, global AR models seem to outperform the local AR models; otherwise the local linear models are more accurate.

On noise free data, most of the approximation methods are, at least in principle, able to approximate any reasonably well behaved system to any degree of accuracy, given enough data. The approximation error will therefore depend on the number of

Data set	Authors
Taylor–Couchette flow	Farmer 1987
Rayleigh–Bénard convection	Farmer 1987
Driven semiconductors	Linsay 1991
Measles	Sugihara 1990, Casdagli 1991
Chickenpox	Sugihara 1990
Marine plankton	Sugihara 1990
Sea surface temperature	Elsner 1992, Huber 1992
Coupled diodes	Casdagli 1991
Fluid turbulence	Casdagli 1991, Elsner 1992
Flame dynamics	Casdagli 1991
Electroencephallograms	Casdagli 1991
Sunspots	Casdagli 1991, Weigend 1990
Geology, ground elevation	Cortini 1993
Computational ecosystem	Weigend 1990
Double potential well	Hunter 1992

Table 3. Some data sets from real processes used in experiments with forecast of chaotic time series.

samples used in the approximation process. For most methods, there is a strong relation between the number of samples used in training, and the number of parameters/weights in the model. We have therefore collated two graphs showing these relations in Fig. 5. All these curves are based on experiments with the standard Mackey–Glass equation as described above, with delay parameter $\Delta = 17$. The first graph shows prediction error as a function of the number of training samples, the second the prediction error as a function of the number of parameters. In both graphs, A is the CC-RAN method (Deco and Ebmeyer 1993), B is adaptive clustering RBF (Platt 1991), C is hashing RBF (Moody and Darken 1989b), D is K-means RBF (Moody and Darken 1989b) and E is standard RBF (Moody and Darken 1988). As can be seen, there are no major differences between most of the methods, except K-means RBF which requires far more training sets. Concerning the number of parameters against prediction error, it is clear that standard and K-means RBF requires many parameters compared to the other methods.

Since much of the discussions in this section has circled around deterministic, simulated systems, we conclude this section with Table 3 which summarizes some of the real data sets analysed. We have not been able to find a reasonable way of comparing these data sets, so the reader is referred to the original work.

7. Conclusion

Prediction of chaotic time series is a fairly new research topic, dating back to 1987. The underlying philosophy is geometrical, fitting non-linear functions to samples in the embedding space $\mathscr{R}^{m+1}$. The area of chaotic prediction is still relatively small within the larger domain of non-linear system identification and prediction, which offers a multitude of approaches still not tested on chaotic systems. Within non-linear studies at large, some knowledge of chaos is desirable since such behaviour is a pervasive non-linear phenomenon that may be manifest in various models in certain regions of the parameter space.

The fact that chaotic systems are 'persistently excited' in the sense that accumulating points become dense on the attractor, is an advantage when modeling chaotic

series. In particular, this could hold promise both for the local methods and the adaptive semi-local methods where the data determine the location and shape of the basis functions.

The paper has focused on methods that up to now have been applied to chaotic time series, including global polynomials, local polynomials, multi-layer perceptrons and semi-local methods. Chaotic time series frequently resemble white noise having broadband Fourier spectra, and the non-linear predictors clearly outperform the standard linear methods like global AR models if the noise level is limited and the dimension of the attractor is low. The local and semi-local methods generally seem to be the best, but no non-linear method ranks top in all situations.

Most of the schemes can approximate any well behaved function to any desired accuracy level, provided enough samples and basis/parameters are available. The crucial question is therefore the number of samples and parameters that are required to achieve a certain accuracy. From this point of view, the adaptive semi-local methods generally seem to be the best.

A crucial aspect is the robustness of the approximation schemes to noise. With noisy data we have, in the language of numerical mathematicians, a fitting problem and not an approximation problem. If the model is too 'small', it will underfit the data and ignore important characteristics. If, on the other hand, the model is too 'large', it will overfit the data, reproducing the noise as well as the underlying behaviour. It is impossible to use the number of parameters to measure the risk of overfitting, since in most methods the parameters are internally correlated. Currently, cross validation is the standard tool for selection of an appropriate model avoiding overfitting. However, the recent appearance of non-linear filtering schemes for time series allow for pretreatment of the data before constructing the model. This is certainly an interesting future line of development, since up to now, too little work has been done on noisy chaotic time series.

Acknowledgments

Support for this research was provided by the Norwegian Research Council (NFR) for D. Kugiumtzis and B. Lillekjendlie. B. Lillekjendlie was also supported by SINTEF-SI. The authors would also like to thank Tom Kavli and Erik Weyer, both at SINTEF-SI, for valuable discussions and careful reading of the manuscript.

References

Abarbanel, H. D. I., Brown, R., and Kadtke, J. B. (1989). Prediction and system identification-in chaotic nonlinear systems: Time series with broadband spectra. *Phys. Lett.* A, **138,** 401–408.

Abarbanel, H. D. I., Brown, R. and Kadtke, J. B. (1990). Prediction in chaotic nonlinear systems: Methods for time series with broadband Fourier spectra. *Phys. Rev.* A, **41,** 1782–1807.

Albus, J. (1975a). A new approach to manipulator control: The cerebellar model articulation controller (CMAC). *J. Dyn. Syst. Meas. Control, Trans. ASME*, 220–227.

Albus, J. (1975b). Data storage in the cerebellar model articulation controller (CMAC). *J. Dyn. Sys. Meas. Control, Trans. ASME*, 228–233.

Barnsley, M. (1988). *Fractals everywhere.* (Academic Press, Inc., New York).

Broomhead, D. S. and Lowe, D. (1988). Multivariable functional interpolation and adaptive networks. *Complex Systems,* **2,** 321–355.

Cadden, D. T. (1991). Neural networks and the mathematics of chaos—an investigation of these methodologies as accurate predictors of corporate bankruptcy. In *Proc. of The First International Conference on Artificial Intelligence Applications on Wall Street*, New York, Oct. 1991.

CARLIN, M. (1991). Neural nets for empirical modelling. Master's thesis, Norwegian Institute of Technology (NTH).

CASDAGLI, M. (1989). Nonlinear prediction of chaotic time series. *Physica* D, **35,** 335–356.

CASDAGLI, M. (1992). Chaos and deterministic versus stochastic and non-linear modelling. *J. R. Statist. Soc.* B, **54,** 303–328.

CASTI, J. L. (1977). *Dynamical systems and their applications—Linear theory* (Academic Press, New York).

CORTINI, M., CILENTO. L. and RULLO, A. (1991). Vertical ground movements in the campi flegrei caldera as a chaotic dynamic phenomenon. *J. Vulcan. Geotherm. Res.*, **48,** 199–222.

CORTINI, M. and BARTON, C. C. (1993). Nonlinear forecasting analyses of inflation–deflation patterns of an active caldera (campi flegrei, Italy). *Geology*, **21,** 239–242.

CYBENKO, G. (1989). Approximation of superpositions of a sigmoidal function. *Math. Control, Signals Sys.*, **2,** 303–314.

DECO, G. and EBMEYER, J. (1993). Coarse coding resource-allocating networks. *Neur. Comp.* **5,** 105–114.

DELLER, J. R., (1989). Set membership identification in digital signal processing. *IEEE ASAP Magazine*, **4,** 4–20.

DEPPISH, J., BAUSER, H. U. and GEISEL, T. (1991). Hierarchical training of neural networks and prediction of chaotic time series. *Phys. Lett.* A, **158,** 57–63.

EFRON, B. (1982). *The Jackknife, the Bootstrap and Other Resampling Plans.* CBMS-NSF Conference Series in Applied Mathematics. Society of Industrial and Applied Mathematics, Philadelphia, 1982.

ELSNER, J. B. (1992). Predicting time series using a neural network as a method of distinguishing chaos from noise. *J. Phys. A: Math. Gen.* A, **25,** 843–850.

ELSNER, J. B., and TSONIS, A. A. (1992). Nonlinear predicting, chaos and noise. *Bull. Am. Meteor. Soc.*, **73,** 49–60.

FARMER, J. D. and SIDOROWICH, J. J. (1987). Predicting chaotic time series. *Phys. Rev. Lett.*, **59,** 845–848.

FRANKE, R. (1982). Scattered data interpolation: Tests of some methods. *Math. Comput.*, **38,** 181–200.

GIONA, M., LENTINI, F. and CIMAGALLI, V. (1991). Functional reconstruction and local prediction of chaotic time series. *Phys. Rev.* A, **44,** 3496–3502.

GRASSBERGER, P., SCHREIBER, T. and SCHAFFRATH, C. (1991). Non-linear time sequence analysis. *Int. J. Bifurc. Chaos*, **1,** 521.

HARTMAN, E. and KEELER, D. (1991). Predicting the future: Advantages of semilocal units. *Neur. Comput.*, **3,** 566–577.

HÉNON, M. (1976). A two dimensional mapping with a strange attractor. *Comm. Math. Phys.*, **50,** 69–77.

HUNTER, N. F. JR. (1992). Nonlinear prediction of speech signals. In Martin Casdagli and Stephen Eubanks, editors, *Nonlinear Modelling and Forecasting*, pages 467–492. (Addison Wesley), pp. 467–492.

KENNEL, M. B. and ISABELLE, S. (1992). Method to distinguish possible chaos from colored noise and to determine embedding parameters. *Phys. Rev.* A, **46,** 3111–3118.

KIRKPATRICK, S., GELATT, C. D. JR, and VECHII, M. P. (1983). Optimization by simulated annealing. *Science*, **220,** 671–680.

KUGIUMTZIS, D., LILLEKJENDLIE, B. and CHRISTOPHERSEN, N. (1994). Chaotic time series, Part I, Estimation of some invariant properties in state space. *Modeling, Identification and Control*, **15,** 205–224.

LAPEDES, A. and FARBER, R. (1987). How neural nets work. In D. Z. Anderson, editor, *Neural Information Processing Systems* (American Institute of Physics, New York), pp. 442–456.

LEE, S. and KIL, R. M. (1988). Multilayer feedforward potential function network. In *IEEE International Conference on Neural Networks*, pages I-161–I-171, San Diego, 1988.

LEE, S. and KIL, R. M. (1991). A gaussian potential function network with hierarchically self-organizing learning. *Neural Networks*, **4,** 207–224.

LICHTENBERG, A. J. and LIEBERMAN, M. A. (1992). *Regular and chaotic dynamics.* 2nd edition, (Springer-Verlag, New York).

LINSAY, P. S. (1991). An efficient method of forecasting chaotic time series using linear interpolation. *Phys. Lett.* A, **153,** 353–356.

LJUNG, L. (1991). Issues in system identification. *IEEE Control Syst. Mag.*, **11,** 25–29.

LORENZ, E. N. (1963). Deterministic non-periodic flows. *J. Atmos. Sci.*, **20,** 130–141.
LORENZ, E. N. (1969). Atmospheric predictability as revealed by naturally occurring analogies. *J. Atmos. Sci.*, **26,** 636.
LOWE, D. and WEBB, A. R. (1991). Time series prediction by adaptive networks: a dynamical system perspective. *IEEE Proc.* F, **138,** 17–24.
MACQUEEN, J. (1967). Some methods for classification and analyses of multivariate observations. In L. M. LeCam and J. Neyman, editors, *Proc. of the 5th Berkeley Symp. on Mathematics, Statistics and Probability*, 1967.
MACKEY, M. and GLASS, L. (1977). Oscillation and chaos in physiological control systems. *Science*, **197,** 287.
MEAD, W. C., JONES, R. D., LEE, Y. C., BARNES, C. W., FLAKE, G. W., LEE, L. A. and O'ROURKE, M. K. (1991). Using cnls-net to predict the Mackey–Glass chaotic time series. In *Proc. IEEE Int. Joint Conf. on Neural Networks (IJCNN)* (New York, IEEE), pp. II485–490.
MEAD, W. C., JONES, R. D., LEE, Y. C., BARNES, C. W., FLAKE, G. W., LEE, L. A. and O'ROURKE, M. K. (1992). Prediction of chaotic time series using cnls-net, example: The Mackey–Glass equation. In M. Casdagli and S. Eubank, editors, *Nonlinear Modelling and Forecasting* (Addison-Wesley), pp. 39–71.
MEES, A. I. (1992). Tesselation and dynamical systems. In M. Casdagli and S. Eubank, editors, *Nonlinear Modelling and Forecasting* (Addison-Wesley), pp. 3–24.
MICHELLI, C. A. (1986). Interpolation of scattered data: distance matrixes and conditionally positive definite functions. *Construct. Approx.*, **2,** 11.
MOODY, J. and DARKEN, C. J. (1988). Learning with localized receptive fields. In D. Touretzky, G. Hinton, and T. Sejnowski, editors, *Proceedings of the 1988 Connectionist Models Summer School* (Morgan-Kaufmann), pp. 133–143.
MOODY, J. (1989a). Fast learning in multi-resolution hierarchies. In D. S. Touretzky, editors, *Advances in Neural Information Processing Systems 1* (Morgan Kaufmann).
MOODY, J. and DARKEN, C. J. (1989b). Fast learning in networks of locally-tuned processing units. *Neur. Comput.*, **1,** 281–294.
PARK, J. and SANDBERG, I. W. (1993). Approximation and radial-basis-function networks. *Neur. Comput.*, **5,** 305–316.
PAWELZIK, K. and SCHUSTER, H. G. (1991). Unstable periodic orbits and prediction. *Phys. Rev.* A, **43,** 1808–1812.
PLATT, J. (1991). A resource-allocating network for function interpolation. *Neur. Comput.*, **3,** 213–225.
POWELL, M. J. D. (1987). Radial basis functions for multivariable interpolation: A review. In J. C. Mason and M. G. Cox, editors, *Algorithms for Approximation* (Clarendon Press, London).
PREPARATA, F. R. and SHAMOS, M. I. (1985). *Computational Geometry: An Introduction* (New York: Springer-Verlag).
PRESS, W. H., FLANNERY, B. P., TEUKOLSKY, S. A. and VETTERLING, W. T. (1988). *Numerical Recipes in C* (Cambridge University Press, Cambridge).
PRIESTLEY, M. B. (1981). *Spectral Analysis and Time Series* (Academic Press, London).
RÖGVALDSSON, T. S. (1993). Brownian motion updating of multi-layered perceptrons. In Stan Gielen and Bert Kappen, editors, *Proc. Int. Conf. on Artificial Neural Networks* (Springer-Verlag, London), pp. 527–532.
RÖSSLER, O. E. (1976). An equation for continuous chaos. *Phys. Lett.* A, **57,** 397.
RUMELHART, D. E., MCLELLAND, J. L. and PDP RESEARCH GROUP. (1986). *Parallel Distributed Processing* Vol. 1–2 (The MIT Press).
SANGER, T. D. (1990). Basic-function trees for approximation in high-dimensional spaces. In *Proceedings of the 1990 Connectionist Models Summer School* (Morgan Kaufmann).
SANGER, T. D. (1991). A tree-structured algorithm for reducing computation in networks with separable basis functions. *Neur. Comput.*, **3,** 67–78.
SAUER, T., YORKE, J. A. and CASDAGLI, M. (1991). Embedology. *J. Statist. Phys.*, **65,** 579–615.
SCOTT, D. W. (1992). *Multivariate Density Estimation* (Wiley, New York).
SÖDERSTRÖM, T. and STOICA, P. (1989). *System Identification* (Prentice-Hall, New York).
STOKBRO, K., UMBERGER, D. K. and HERTZ, J. A. (1990). Exploiting neurons with localized receptive fields to learn chaos. *Complex Syst.*, **4,** 603.
STOKBRO, K. and UMBERGER, D. K. (1992). Forecasting with weighted maps. In M. Casdagli and S. Eubank, editors, *Nonlinear Modelling and Forecasting*, (Addison-Wesley, Readwood City), pp. 73–94.

STONE, M. (1977). Cross-validation, a review. *Math. Oper. Stat. Ser. Stat.*, **9,** 127–139.
SUGIHARA, G. and MAY, R. M. (1990). Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series. *Nature*, **344,** 734–741.
TAKENS, F. (1981). Detecting strange attractors in turbulence. In D. A. Rand and L. S. Yound, editors, *Dynamical Systems and Turbulence* (Springer Verlag, Berlin), pp. 366–381.
TENORIO, M. F. and LEE, WEI-TSIH (1989). Self organizing neural networks for the identification problem. In D. S. Touretzky, editor, *Advances in Neural Information Processing Systems 1* (Morgan Kaufmann).
TONG, H. (1990). *Nonlinear Time Series: A Dynamical System Approach* (Oxford University Press).
TOWNSHEND, B. (1992). Nonlinear prediction of speech signals. In Martin Casdagli and Stephen Eubanks, editors, *Nonlinear Modelling and Forecasting* (Addison-Wesley), pp. 433–453.
WEIGEND, A., HUBERMAN, B. and RUMELHART, D. E. (1990). Predicting the future: a connectionist approach. *Int. J. Neur. Syst.*, **1,** 193–209.
WEIGEND, A., HUBERMAN, B. and RUMELHART, D. E. (1992). Predicting sunspots and exchange rates with connectionist networks. In M. Casdagli and S. Eubank, editors, *Nonlinear Modelling and Forecasting* (Addison-Wesley), pp. 395–432.
WELSTEAD, S. T., (1991). Multilayer feedforward networks can learn strange attractors. In *IEEE Proc. Int. Joint Conference of Neural Networks* (New York, IEEE), pp. II 139–144.
WEST, B. J. and MACKEY, H. J. (1992). Forecasting chaos: A review. *J. Sci. Ind. Res.*, **51,** 634–643.
WOLPERT, D. M. and MIALL, R. C. (1990). Detecting chaos with neural network. *Proc. R. Soc. London* B, **242,** 82–86.