Identification of Dynamic Process Systems with Surrogate Data Methods

Jakobus P. Barnard, Chris Aldrich, and Marius Gerber
Dept. of Chemical Engineering, University of Stellenbosch, Stellenbosch 7602, South Africa

Identifying the underlying dynamics of chemical process systems from experimental data is complicated, owing to a mixture of influences that cause erratic fluctuations in the time series. These influences can be notoriously difficult to disentangle. The development of process models is usually subject to considerable human judgment and can therefore be very unreliable. This is especially the case when the model priors are unknown and the model is validated empirically, such as with cross-validation or holdout methods. A case study shows that more reliable identification of systems is possible by using surrogate methods to classify the data, as well as to validate models derived from these data.

Introduction

Many real-world systems exhibit changing behavior. Chemical and metallurgical systems in particular can be high-dimensional, nonlinear, and ill-defined in that little knowledge of plant dynamics and chemical reaction mechanisms is available. As a consequence, engineers usually try to develop dynamic process models for these systems direct from input-output data, rather than attempting to develop time-consuming, expensive analytical models.

A promising method for dealing with complex nonlinear systems is the identification of global models based on the reconstruction of the dynamic attractor(s) of these systems. This strategy enables successful identification of state-space model structures and can be applied to all deterministic systems. In theory, a properly identified state-space model ensures a complete description of the system dynamics and allows prediction of the system output from any initial condition within the basin of attraction or validity region of the model. Most current state-space models are linear and depend upon the availability of differential equations of the system that allow only well-defined, periodic, or quasiperiodic dynamic systems to be identified successfully. With these methods, the system dynamics either have to be linearized in toto, such as with the standard Kalman filter or locally, such as with the extended Kalman filter in the case of nonlinear systems. Unfortunately, these approximations cannot always capture the nonlinearities frequently found in chemical process systems and can, therefore, result in grossly inadequate process models in terms of prediction horizon and generalization.

Despite the many successful models reported in literature (MacMurray and Himmelblau, 1995; Pham et al., 1995; Su et al., 1992), some chemical process systems remain difficult to model, especially those nonlinear deterministic systems that manifest apparently unstable aperiodic behavior. Although these and other intractable systems can sometimes be accommodated by adaptive local modeling techniques, this approach is not always satisfactory. The problem can be attributed in part to the fact that highly nonlinear systems with chaotic behavior appear random under linear analysis, even though they are actually deterministic (Farmer and Sidorowich, 1987). Since highly nonlinear systems are not periodic or even quasiperiodic, linear analysis, such as Fourier transforms, yields results similar to broadband noise. Also, linear filters cannot remove noise from such nonlinear data without distorting the underlying dynamics.

Moreover, a major problem with empirical systems is to determine *a priori* whether deterministic dynamics underlie the data in the first place, that is, whether dynamic attractors are present at all. To make matters worse, nonlinear identification algorithms that calculate the system dimension from a time series do not always return an infinite value for stochastic processes (that have infinite dimension) as would be expected. Osborne and Provenzale (1989) have shown that

Correspondence concerning this article should be addressed to C. Aldrich.

stochastic data with power law spectra also yield correlation dimensions with finite values, so that statistics characterizing the dimensionality of a system cannot be reliably used for the identification of determinism. Some stochastic processes generate so-called colored noise, which have fractal curves in phase space, but no dynamic attractors. Nonlinear identification algorithms cannot distinguish between fractal curves and fractal attractors. This can be problematic, since reliable classification of the data as a first step in system identification is important; otherwise, the resulting model will not generalize beyond the training data set.

A second, similar problem arises especially with nonlinear systems, namely, validation of a model fitted to observed data. Proper validation ensures the reliable application of the model on new observations from the same process. Model validation is usually based on statistical tests, such as the significance of R^2 or RMS criteria derived from actual and predicted values, or empirical methods such as cross-validation or holdout. Although statistical tests of model validity are the preferred approach (Rivals and Personnaz, 1999), they can unfortunately only be applied when the statistical properties of the system are known. This may be the exception, rather than the rule. Likewise, empirical methods such as cross-validation can perform poorly when used in the selection of linear models (Zhu and Rohwer, 1996; Goutte, 1997) and is highly unlikely to perform any better with nonlinear models.

Also, model validation is often based on the one-step-ahead prediction, which is not necessarily a good indicator of the ability of the model to generalize the underlying process represented by the data. A free-run prediction, in which the model has to predict the long-term future behavior of the system without being updated with succeeding observed states, is a considerably more rigorous test of the validity of the model. To achieve this, one has to generate a free-run time series with the model, reconstruct the dynamic attractor from these predicted values and characterize the attractor with some discriminating statistic. The dynamic attractor for the actual system is likewise reconstructed from the observed time series, and characterized. The reliability of the model can therefore be assessed systematically by comparing the discriminating statistic of the model with that of the experimental data.

As shown in this article, state-space techniques and surrogate methods (Takens, 1993) provide a more powerful approach to the evaluation of dynamic models than conventional statistical and empirical validation methods such as the R^2 test and cross-validation. Although Theiler et al. (1992) and Takens (1993) have used surrogate data methods to classify dynamic systems, these methods can be incorporated in a general framework for the identification and visualization of process systems. The concepts of surrogate data methods are first described, followed by the proposed framework for the identification of process systems. Finally, the methodology is illustrated by way of two case studies.

Surrogate Data

The method of surrogate data (Takens, 1993; Theiler and Pritchard, 1996; Theiler and Rapp, 1996) involves a null hypothesis against which the data are tested, as well as a discriminating statistic. The data are first assumed to belong to a specific class of dynamic processes. Surrogate data are subsequently generated, based upon the given data set, by using the assumed process. An appropriate discriminating statistic is calculated for both the surrogate and the original data (Theiler et al., 1992). If the calculated statistics of the surrogate and the original data are significantly different, then the null hypothesis is rejected, that the process that has generated the original data is of the same class as the system that has generated the surrogate data. By means of a trial-and-error eliminating procedure, it is then possible to get a good idea of the characteristics of the original process.

More specifically, let $\boldsymbol{x} \in \Re^N$ be a time series consisting of N observations, let ψ be a specific hypothesis, let $\Im_\psi$ be the set of process systems consistent with the hypothesis, and let $T: \Re^N \rightarrow U$ be a statistic that will be used to evaluate the hypothesis ψ that x was generated by some process $\Im \in \Im_\psi$. Generally, the statistic $U \in \Re$, and it will be possible to discriminate between the original data $\boldsymbol{x}$ and the surrogate data $\boldsymbol{x}$ consistent with the hypothesis given by the probability density of T, given $\Im$, that is, $p_{T,\Im}(t)$.

Classes of hypotheses

Three classes of hypotheses are widely used; those equivalent to the assumption that the data are identically, independently distributed noise (Type 0), linearly filtered noise (Type 1), and a monotonic nonlinear transformation of linearly filtered noise (Type 2).

Type 2 surrogates are also known as amplitude adjusted Fourier transform (AAFT) surrogates (Small and Judd, 1998a). Combining procedures for Type 0 and 1 surrogate data, the procedure for generating Type 2 surrogate data consists of the following steps:

(i) Generation of a normally distributed data set $\boldsymbol{y}$, reordered to have the same rank distribution as $\boldsymbol{x}$, the observed (original) data set, producing type 0 surrogate data.

(ii) Generation of a Type 1 surrogate data set $\boldsymbol{y}_s$ from $\boldsymbol{y}$ (by phase-shuffling the Fourier transform of $\boldsymbol{y}$, as explained in the Appendix).

(iii) Finally, rank ordering of $\boldsymbol{y}_s$ and replacing the amplitudes y_{sj} with that of x_i of corresponding rank.

Pivotal test statistics

Theiler (1995) has suggested that a distinction can be made between so-called pivotal and nonpivotal statistics. A test statistic T is considered to be pivotal, if the probability distribution $p_{T,\Im}$, is the same for all processes $\Im$ consistent with the hypothesis ψ; thus, $p_{T,\Im}$ is invariant for all $\Im \in \Im_\psi$. Moreover, a distinction can be made between simple and composite hypotheses. If the set of all processes consistent with the hypothesis $(\Im_\psi)$ is a singleton, then the hypothesis is simple. Otherwise, the hypothesis is composite and can be used not only to generate surrogate data consistent with a particular process $\Im$, but also to estimate $\Im \in \Im_\psi$. In fact, $\Im$ has to be specified when the hypothesis is composite, unless T is a pivotal statistic (Theiler, 1995).

Constrained realization (Schreiber and Schmitz, 1996) schemes can be employed when nonpivotal statistics are applied to composite hypotheses. That is, apart from generating surrogate data that represent typical realizations of a model of the system, the surrogate data should also be representative of a process yielding identical estimates of the parameters of the process when compared to the estimates of the process parameters obtained from the original data. Put in a different way, if $\mathfrak{I}_{\text{est}} \in \mathfrak{I}_{\psi}$ is the process estimated from the original data $\boldsymbol{x}$, and $\boldsymbol{x}_s$ is a surrogate data set generated by $\mathfrak{I}' \in \mathfrak{I}_{\psi}$. Then, $\boldsymbol{x}$ is a constrained realization of $\mathfrak{I}_{\text{est}} \in \mathfrak{I}'$.

As an example, if ψ is the hypothesis that $\boldsymbol{x}$ is generated by linearly filtered independent identically distributed noise, then *nonconstrained* surrogate data $\boldsymbol{x}_s'$ can be generated from a Monte Carlo simulation based on the best linear model estimated from $\boldsymbol{x}$. The data $\boldsymbol{x}_s'$ can be constrained by shuffling the phases of the Fourier transform of the data, producing a set of random data $\boldsymbol{x}_s''$ with the same power spectra (and autocorrelation) as the original data $\boldsymbol{x}$. The autocorrelation, rank order statistics, nonlinear prediction error, and so on, would all be nonpivotal test statistics characterizing dynamic manifold structures, since the distributions of these statistics would all depend on the form of the noise source and the type of linear filter. In contrast, the Lyapunov exponents and the correlation dimension (fractal dimension) would be pivotal test statistics, since the probability distribution of these quantities would be the same for all processes, regardless of the source of the noise of the estimated model. Since recent investigations have shown that Lyapunov exponents can be misleading in the presence of noise, the correlation dimension has gained favor as the pivotal statistic of choice.

Correlation dimension

The correlation dimension d_c is defined as follows

$$d_c = \lim_{\epsilon \to 0} \lim_{N \to \infty} \frac{\log C_N}{\log \epsilon} \tag{1}$$

C_N is the correlation function and is defined by

$$C_N(\epsilon) = \binom{N}{2}^{-1} \sum_{0 \le i < j \le N} I(\| v_i - v_j \| < \epsilon) \tag{2}$$

$I(\| \cdot \|)$ is a Heavyside function that returns one if the distance between point i and j is within ϵ, and zero otherwise, while N is the number of observations in the data set.

The idea is illustrated in Figure 1a, which shows a hypersphere of a radius ϵ centered on one of the points (v_i) defining the trajectory of the attractor. With the Heavyside function, the points enclosed in a given hypersphere are counted, while the hypersphere itself is moved from point-to-point along the trajectory. The cumulative sum of the points is subsequently divided by $N(N-1)$ to give the correlation sum for a hypersphere of a particular size. The maximum value that the averaged correlation sum defined by Eq. 2 can attain is unity, while the minimum value is $2/[N(N-1)]$, when only the closest two points on the attractor are counted.

The correlation sum scales with the hypersphere radius according to a power law of the form $C_N(\epsilon) \approx \epsilon^{d_c}$, where d_c is the correlation dimension of the attractor. As indicated in Figure 1b, d_c is obtained from the slope of the curve generated by plotting $\log(\epsilon)$ vs. $\log[C_N(\epsilon)]$. The slope is based on the central part of the curve, which is approximately linear and not on the extremes of the curve coinciding with very large and very small scales, which tend to be nonlinear.

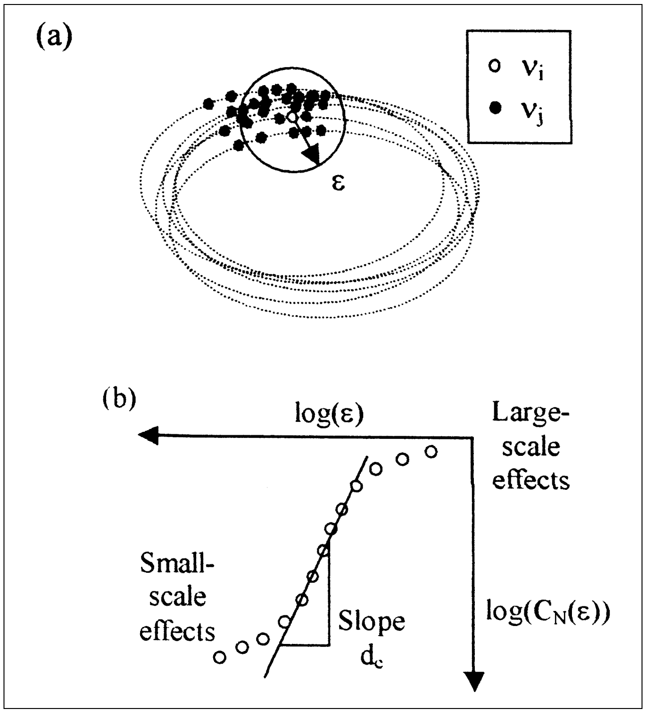


Figure 1. Calculation of the correlation dimension for an attractor.

(a) Probing hypersphere on an attractor, and (b) calculation of the correlation dimension d_c as the slope of the curve defined by plotting the logarithms of the correlation sum C_N and the size of the hypersphere (ϵ).

Reliable calculation of the correlation dimension is not as straightforward as first thought when the Grassberger-Procaccia (1983) algorithm appeared. Using this algorithm requires a linear scaling region to reliably calculate the correlation dimensions. Noise strongly influences the approximation of the correlation dimension, according to Stefanovska et al. (1997). When working with measured (empirical) data, they stressed in particular the problem of using the Grassberger-Procaccia algorithm to obtain an adequate scaling region for a valid approximation of the correlation dimension. Lai and Lerner (1998) showed that this region is sensitive to the choice of the embedding lag. Linear correlation in the data set misleads the algorithm to falsely show convergence to some low dimension, which could then be misinterpreted for inherent low-dimensional dynamics (Judd, 1994).

Earlier, Judd (1992) pointed out the deficiencies of the Grassberger-Procaccia algorithm and proposed a different algorithm for calculation of the correlation dimension. This algorithm replaces the requirement for a linear scaling region on the graph $C_N(\epsilon)$ by fitting a polynomial of the order of the topological dimension in that region. It expresses the correlation dimension for interpoint distances below a specific scale ϵ_0. Instead of comparing estimates of the scalar correlation dimension (d_c), as indicated in Figure 1b, one rather compares curves of correlation dimension vs. scale $[d_c(\epsilon)]$, as calculated by the Judd algorithm. This allows the correlation dimension to be used for examining the macro- and microscale of the reconstructed dynamic attractor. For large data sets, it asymptotically approaches the value of the true correlation dimension as ϵ_0 goes to zero. Also, the algorithm is not easily confused by linear correlation in the data (Judd, 1994).

Judd proposed that the correlation dimension be estimated as a function of scale ϵ_0 using the following equation, valid for $\epsilon < \epsilon_0$

$$C_N(\epsilon) \approx \epsilon^{d_c} q(\epsilon) \tag{3}$$

where $q(\cdot)$ is a polynomial of order of the topological dimension.

The topological dimension is the smallest dimension of embedding space in which the attractor fully unfolds. The function that is fitted is $\epsilon^{d_c} q(\epsilon) \approx C_N(\epsilon)$, instead of the conventional $\epsilon^{d_c} \approx C_N(\epsilon)$, which assumes a linear scaling region.

The Judd-algorithm estimates d_c, as well as the coefficients of the polynomial $q(\cdot)$. The estimated d_c value is retained for classification purposes. The polynomial coefficients are discarded for the polynomial's only purpose which is to improve the fit through $C_N(\epsilon)$ and to avoid the conventional, suboptimal assumption of a linear scaling region. Kevin Judd provided a C-implementation of the Judd-algorithm to us, which we could port for application in the MS NT environment.

Finally, accurate calculation of the correlation dimension depends on the minimum length of a time series. Stefanovska et al. (1997) have shown that too few points in a neighborhood leads to overestimation of the correlation dimension when using the Grassberger-Procaccia algorithm. The Judd algorithm is less sensitive to the number of observations by an order of magnitude, compared to the Grassberger-Procaccia algorithm. In practical terms, a data set of approximately 1,000 observations is usually sufficient for the Judd algorithm.

General Framework for Identifying Dynamic Process Systems

A dynamic system can be represented mathematically by a state equation in a number of state variables, that is, a state vector. Starting from some initial conditions, the state vector follows over time a trajectory that is confined to some closed subspace of the total available state space. The dynamic attractor, to which the trajectory thus converges, is a smooth, nonlinear manifold of this state space and defines the true dynamics of the system. The output of such a system can be observed through some measurement function, which may be nonlinear. According to Takens (1981), in the absence of noise and under certain conditions, one can reconstruct an equivalent representation of the system state space from a time series observation of a single observed output. Such a reconstruction is called an embedding of the observed time series by way of delay coordinates (equivalent state variables). The number of these coordinates is the embedding dimension m, and the time delay k (in multiples of sample period) is the lag between each coordinate. A brief discussion of the theoretical background to the embedding of time series can be found in an article by Osborne and Provenzale (1989), among others.

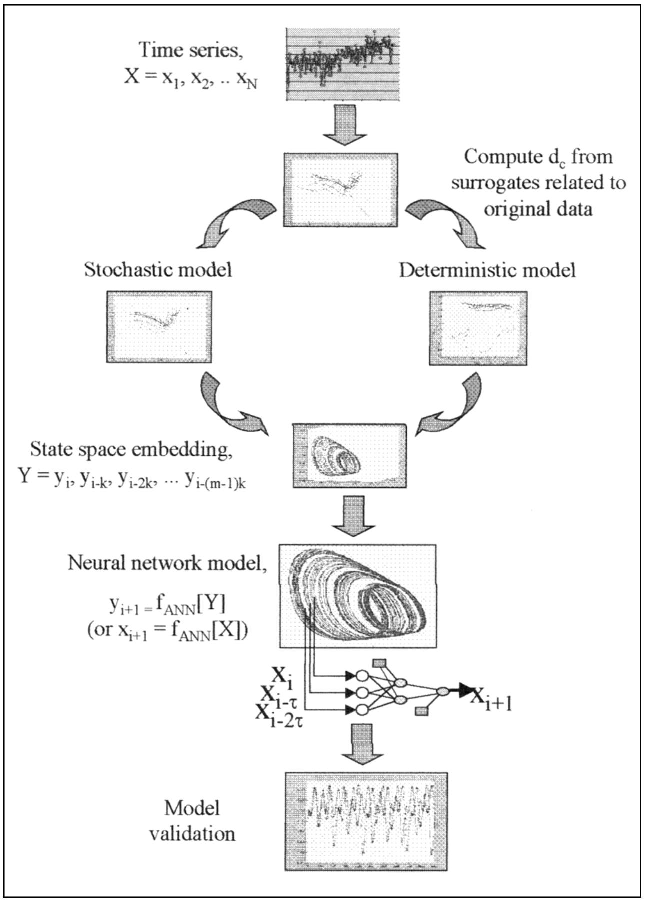


Figure 2. Methodology of nonlinear process identification with surrogate data.

The time lag between the state variable is usually determined by the method of average mutual information (Frazer and Swinney, 1986), while the number of state variables is typically calculated using the method of false nearest neighbors (Kennel et al., 1992). In an alternative approach both these properties can be calculated simultaneously by use of the method of false strands (Kennel et al., 1992).

After reconstructing the dynamic attractor of the system, a one-step prediction model is built to map the attractor onto the observed time series, such as by using a neural network or radial basis functions. Finally, the model is validated before it is applied to the system. The methodology is summarized in Figure 2.

The presence of determinism in the observed (measured) time series is assessed via comparison with suitable surrogate data sets. If the time series proves to be stochastic, it is modeled stochastically. If not, a deterministic model can be constructed via embedding of the time series, and fitting of the proposed model to the attractor. The model can also be fitted direct to the time series, as is done conventionally. In the final stage, the model is validated by testing one-step and free-run prediction against a section of observed data not used in the construction of the model.

Classification of observed time series

The observed time series, which is returned by some measurement function operating on the system, is first classified as either stochastic or deterministic by use of surrogate data. The hypothesis that the time series was generated by a nonlinearly transformed stochastic process is then tested. A set of type 2 (AAFT) surrogate data is generated based on the observed time series. In practice, a set consisting of approximately 15 to 30 surrogate data series is required for statistical significance. The data sets are different random realizations of these surrogate data series. Correlation dimension curves are now calculated for the observed time series, as well as for the set of surrogate data. A Cartesian plot of the various correlation dimension curves provides a qualitative test of the hypothesis. Deterministic data will have a correlation dimension curve that falls significantly below and outside the cluster of curves representing the surrogate data set, depending on the degree of determinism. In constrast, stochastic data will yield correlation dimension curves coinciding with the cluster of curves representing the surrogate data. If the data are deterministic, the dynamic attractor can be reconstructed through an optimal embedding of the time series.

Calculation of lag

In order to embed the data, the lag between embedded variables is typically calculated from the average mutual information algorithm proposed by Frazer and Swinney (1986). That is, the average mutual information (AMI) between the observation $x(t)$, at time t, and the observations $x(t-k)$, at time $(t-k)$, is

$$I(k) = \Sigma_t P(x_t,x_{t-k}) \log_2 \{ P(x_t,x_{t-k}) / [P(x_t) P(x_{t-k})] \} \tag{4}$$

where $P[\cdot]$ is the probability function and t is the time. In practice, $P(x_t)$ is estimated as the histogram of x_t and $P(x_t,x_{t-k})$, the joint histogram of x_t and x_{t-k}. The function $I(k)$ can be seen as a nonlinear autocorrelation function that is used to determine when the values of x_t and x_{t-k} are sufficiently uncorrelated to be useful as coordinates in a time delay vector, but not so uncorrelated as to have no connection with each other. The optimal embedding lag is taken as the value of k where the first minimum in $I(k)$ occurs.

Embedding of time series

Once the optimum embedding time lag has been established, the embedding time series can be expressed as follows

$$\boldsymbol{x}_t = [y_t, y_{t-k}, y_{t-2k}, \dots y_{t-(m-1)k}]^T \tag{5}$$

where $\boldsymbol{x}_i$ is the embedded state vector, while y_i is the observed output. In the case studies considered below, the number of coordinates (m), in which the time series should be embedded, was calculated by the method of false nearest neighbors. A point $\boldsymbol{x}_j$ is a false neighbor of $\boldsymbol{x}_i$ in embedding space when $\boldsymbol{x}_j$ moves away from $\boldsymbol{x}_i$ at more than some arbitrary ϵ at higher embedding dimension, based on Euclidian distance. For a given lag, this algorithm finds the optimum embedding dimension by progressively embedding the time series in state space with increasing dimension. The presence of false nearest neighbors around individual embedded vectors is usually indicated by a change in the number of nearest neighbors over two consecutive embeddings. The optimal embedding dimension is the minimum dimension at which there is no more decrease in the number of false nearest neighbors.

Fitting the model

Having calculated both embedding lag and dimension, the dynamic attractor is reconstructed by embedding the observed time series, after which a model structure is selected and fitted to a section from the first part of the observed time series. This training set is embedded to form the input space to be used during the construction of the model (such as for a neural network, the input-output training data are sampled from this space).

The model is of the form $\mathfrak{M}\colon \boldsymbol{x}_t \rightarrow Y_{t+1}$, where $\boldsymbol{x}$ is the embedding vector at time t. A state space parameterization of $\boldsymbol{Y}$ is formed by using time series embedding, as follows

$$\begin{aligned} \boldsymbol{x}_{t+1} &= f(\boldsymbol{x}_t) \\ y_t &= g(\boldsymbol{x}_t) \end{aligned} \tag{6}$$

In the above nonlinear state space formulation, $\boldsymbol{x} = \Lambda(\boldsymbol{y}, m, k)$, where Λ is the embedding operator. The model is an approximation of $Y_{t+1} = \text{gof}(\boldsymbol{x}_t)$ as $\hat{Y}_{t+1} = \hat{g}(\boldsymbol{x}_t)$. The model structure that is estimated can be any appropriate nonlinear regressor, such as a multilayer perceptron network or a radial-basis function network. The phase variables that constitute the embedding vector are not directly related to the original states, but are an equivalent parameterization of the system. The equivalence is tested by correlation dimension or fitting a model and qualifying this model, using correlation dimension, as was done in this article.

Validation

Model validation is done first by predicting the validation data one step ahead and comparing the R^2 statistic for the predicted and observed data. Secondly, a free-run prediction is done on the validation data set and inspected. If the model output resembles the observed data, nonlinear surrogate data sets are generated using the model in free-run mode. Correlation dimension curves are calculated for these data sets, as well as for the observed validation data. Inspection of a Cartesian plot of these indicates the validity of the model in dynamic terms. If the correlation dimension curve for the observed data lies significantly outside the cluster of these curves for the nonlinear surrogates, the model is rejected.

Case Study: Modeling of an Autocatalytic Process

The case study concerns an autocatalytic process in a continuous stirred-tank reactor originally considered by Gray and Scott (1983, 1984) and subsequently considered by Lynch

(1992). The system is capable of producing self-sustained oscillations based on cubic autocatalysis with catalyst decay and proceeds mechanically as follows

$$A+2B \rightarrow 3B\left(-r_A = k_1 c_A c_B^2\right)$$

$$B \rightarrow C\left(r_C = k_2 c_B\right)$$

$$D+2B \rightarrow 3B\left(-r_D = k_3 c_D c_B^2\right)$$

This process is represented by the following set of ordinary differential equations

$$\frac{dX}{dt} = 1 - X - aXZ^2$$

$$\frac{dY}{dt} = 1 - Y - bYZ^2$$

$$\frac{dZ}{dt} = 1 - (1+c)Z + daXZ^2 + ebYZ^2 \qquad (7)$$

where X, Y, and Z denote the dimensionless concentrations of the products, while a, b, c, d, and e denote the dimensionless concentrations of the reactants. The process is chaotic, with a well-defined attractor for specific ranges of the two parameters d and e. For the settings: $a = 18{,}000$; $b = 400$; $c = 80$; $d = 1.5$; $e = 4.2$, and for initial conditions $[0,0,0]^T$, the set of equations was solved by using a 5th-order Runge Kutta numerical method over 100 simulated seconds. This gave approximately 10,000 observations, which were resampled with a constant sampling period of 0.01 s. Figure 3 shows the attractor of the data reconstructed from the observed states X, Y, and Z.

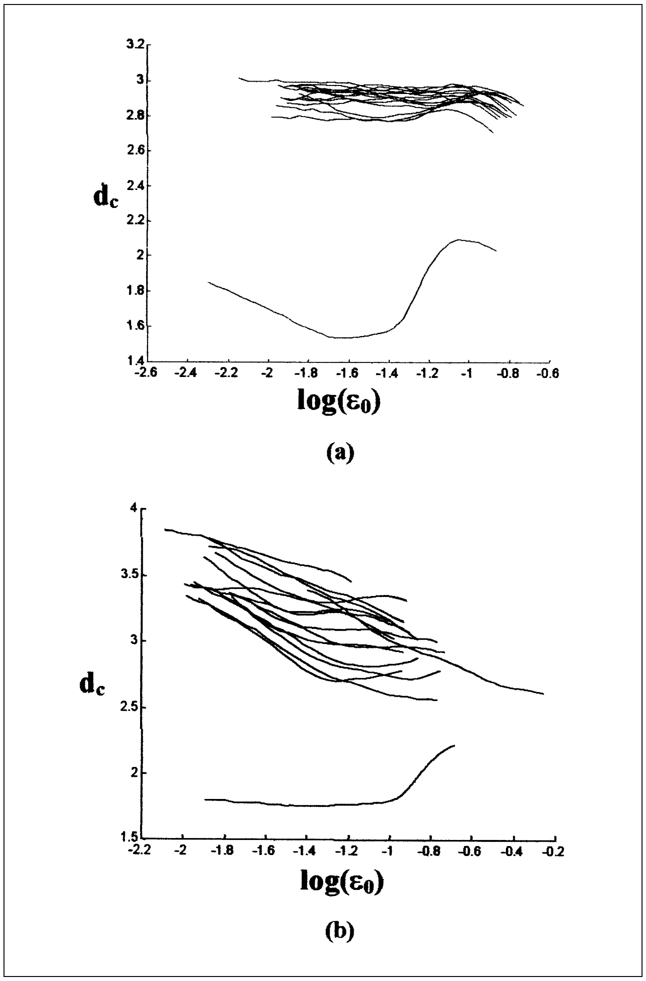


Figure 4. Correlation dimension curves for autocatalytic process Y-state (bottom curve) and its AAFT surrogates for the smaller data set (a) and larger data set (b).

For the case study, the Y state was taken as the observed state variable on which a model was based. Two different data sets were considered in order to assess the effect of the size of the data set on the identification method. The smaller of the two sets consisted of the first 2,000 observations of the original data set of 10,000 observations, while the larger of the two sets consisted of the first 8,000 observations. In each case the remainder of the data was used to validate subsequent models, that is, 8,000 observations were used to validate models based on the smaller set, while 2,000 observations were available for validation in the case of the larger set.

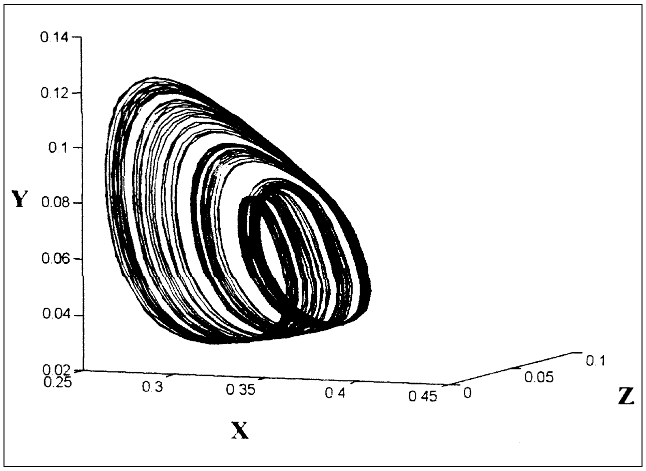


Figure 3. Attractor of autocatalytic process constructed from observed states X, Y, Z.

Classification of the data with Type 2 (AAFT) surrogates was performed first. This entailed calculation of the correlation dimension for each of the two data sets, as well as for 15 surrogate data sets generated from each data set. These results are shown in Figures 4a and 4b, for the small and the large data set, respectively. The deterministic character of the data is evident from these figures. In both cases the curve representing the data set is well separated from the cluster of curves representing the 15 different, random realizations of the surrogate data series, generated from the particular data set.

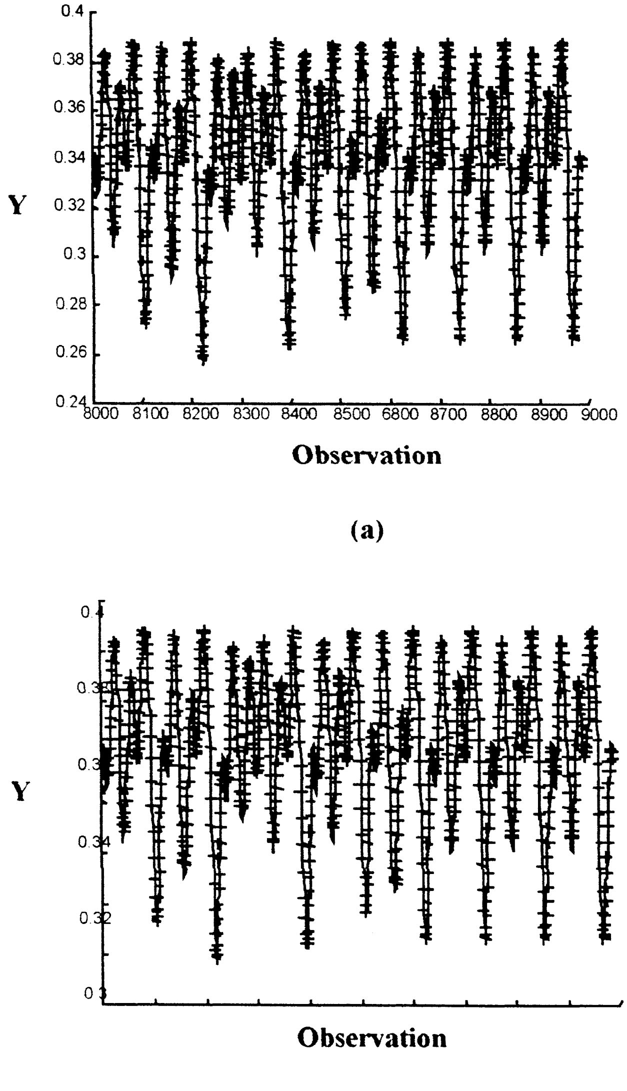
The next step involved the embedding of each of the training and validation data sets in an appropriate state space. By making use of the method of false nearest neighbors discussed earlier, both the smaller and the larger data set could be optimally embedded in a three-dimensional space ($m = 3$) with a time lag of $k = 7$. Two nonlinear models were subsequently fitted to the data.

Neural network model

The first model consisted of a multilayer perceptron neural network with an input layer of three nodes, a hidden layer with six bipolar sigmoidal nodes (activation functions of the form $g(\cdot) = (1 - \exp(\cdot)/[1 + \exp(\cdot)])$, and a single linear output node. The neural network was trained with the Levenberg-Marquardt algorithm and the optimal model structure was determined via cross-validation on the test data, for both the smaller and the larger data set.

The model based on the smaller data set was able to predict the data one-step-ahead in the associated validation data set very accurately ($R^2 = 0.999$), as indicated in Figure 5a. It could predict the data one-step-ahead in the validation data set associated with the larger data set with a similar degree of accuracy, as shown in Figure 5b.

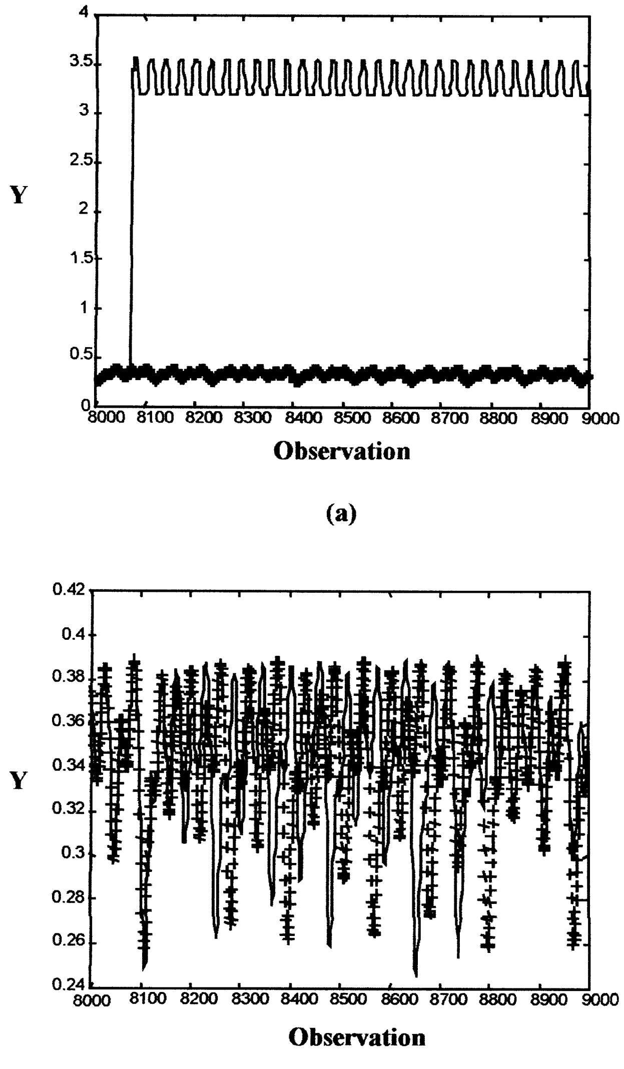
The free-run predictions for the two data sets are shown in Figure 6a for the smaller training data set and Figure 5b for the larger training data set. As can be from Figure 6a, the neural network could predict the data accurately in a free-run mode, up to about the 60th observation, after which it systematically and grossly overestimated the actual values of the observations. It performed significantly better with the larger training data set, but after approximately 180 observations the model predictions also started to break down, as indicated in Figure 6b.



(a)

(b)

Figure 5. One-step prediction of observed autocatalytic Y-state (+marker) vs. Y-state using feed-forward neural network trained on (a) the smaller data set and (b) the larger data set.



(a)

(b)

Figure 6. Free-run prediction of autocatalytic Y-state with feed-forward neural network model (x marker), trained on (a) the smaller data set and (b) the larger data set.

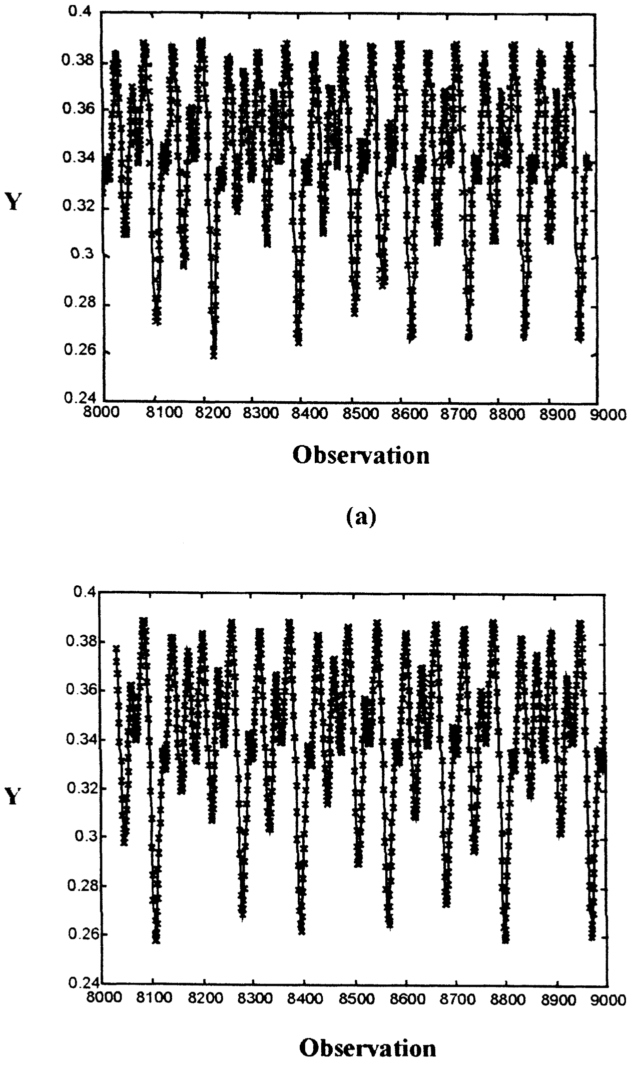
Pseudolinear radial basis function model

The pseudolinear radial basis function (PL-RBF) model previously proposed by Small and Judd (1998a) was also fitted to the data set by using an algorithm that optimizes model size via Rissasen's minimum description length (MDL) of the model during each iteration (Judd and Mees, 1995; Small and Judd, 1998a).

The PL-RBF model was comprised of a combination of linear terms and a number of Gaussian radial basis function terms. The algorithm of Small and Judd (1998a) determines the combination and number of these terms by using minimum description length as a criterion. Like the neural network model, the PL-RBF model was able to predict the data one-step-ahead in the validation data associated with both the smaller and the larger data sets very accurately ($R^2 = 0.999$), as indicated in Figures 7a and 7b respectively.

The free-run predictions for the two data sets are shown in Figure 8a for the smaller training data set and Figure 8b for the larger training data set. As can be seen from these figures, the PL-RBF model could predict the data more accurately in free-run mode than the neural network model.

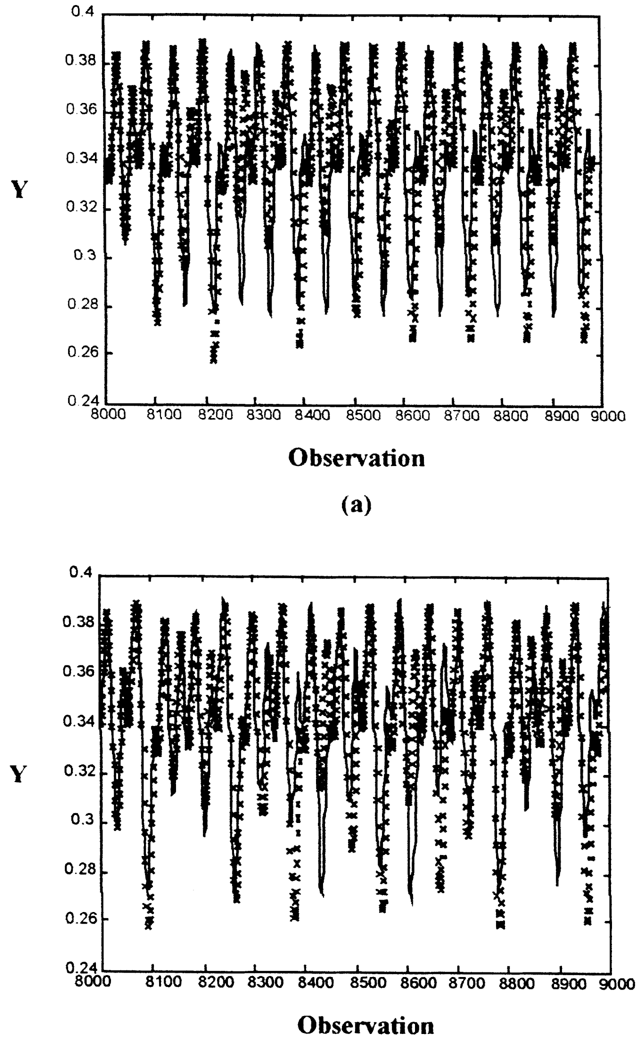
Based on these results, it is clear that analyses of one-step-ahead predictions are comparatively poor indicators of the quality of the models and that the free-run predictions provide a better idea of the aqequacy of the models representing the dynamics of the system. These analyses can also be conducted by comparing the surrogate data derived from the models with the actual data. The results for the feedforward neural network model based on the larger training data set are shown in Figure 9. The results pertaining to the smaller data set are not shown, owing to the obvious poor quality of the model shown in Figure 6a. From Figure 9, it can be seen that the broken curve at the bottom that represents the neu-



(a)

(b)

Figure 7. One-step prediction of observed autocatalytic Y-state with PL-RBF model vs. state (x marker) based on (a) the smaller data set and (b) the larger data set.



(a)

(b)

Figure 8. Free-run prediction of observed autocatalytic Y-state with PL-RBF model vs. Y-state (x marker) based on (a) the smaller data set and (b) the larger data set.

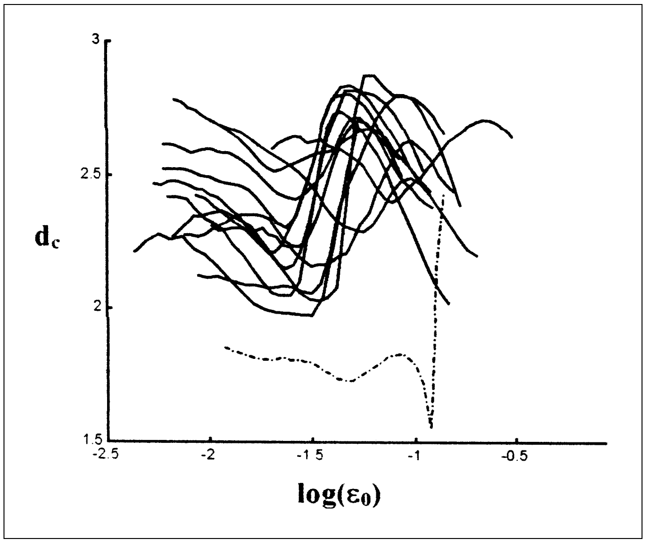


Figure 9. Correlation dimension curves of nonlinear surrogates of the feed-forward neural network and that of the observed data (broken line, bottom) from the larger data set (curves for the smaller data set are not shown).

ral network model has not captured the structure of the data completely, except in the large-scale region ($\epsilon_0 > -0.9$). Similar analyses show the better quality of the PL-RBF model. In Figure 10a, it can be seen that the PL-RBF model based on the smaller data set has captured most of the large-scale structure of the data ($\epsilon_0 > -1.3$). Likewise, the PL-RBF model based on the larger data set has evidently captured most of the structure of the data, as indicated by Figure 10b.

Reconstructions of the dynamic attractor of the data, based on the actual data and the free-run predicted data (PL-RBF model on larger data set) are shown in Figures 11a and 11b. As can be seen from these figures, the two attractors are remarkably similar in appearance, and to the attractor reconstructed from the X, Y and Z variables in Figure 3. This is confirmed by the position of the correlation dimension curve for observed data among the cluster of nonlinear surrogates in Figure 10b.

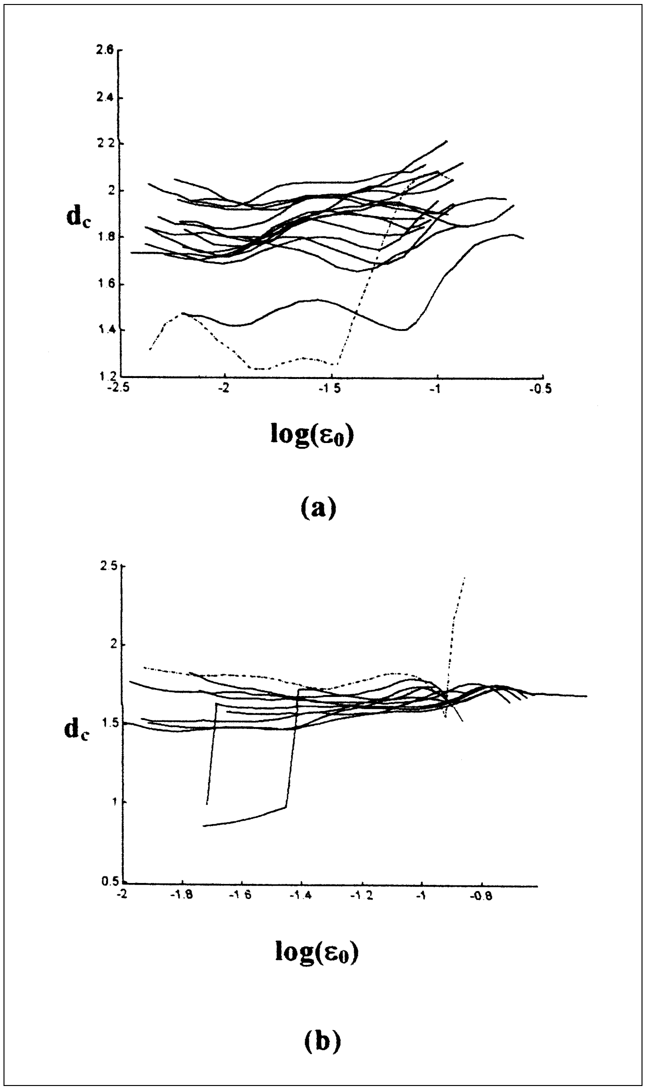


Figure 10. Correlation dimension curves of nonlinear surrogates of PL-RBF model and that of the observed data (broken line, bottom) based on (a) the smaller data set and (b) the larger data set.

Effect of measurement and dynamic noise

To test the effectiveness of the identification method on noisy data, Gaussian measurement noise, as well as dynamic noise were added to the autocatalytic process. The noise level was set at 0.1σ (10% of the sample standard deviation of the training data set) for measurement noise. Dynamic noise was added by a modified one-step prediction of the training data with the PL-RBF model based on the larger data set. Noise of 0.1σ (10% of the sample standard deviation of the training data set) was added to the ith point, which was then included in the embedding for prediction of the $(i+1)$ th point.

In Figure 12a, the correlation dimension curves of the noisy data and their associated surrogates are shown, while the correlation dimension curves of the data with, and without, noise are shown in Figure 12b. The curve at the bottom of Figure 12a is the same as the curve at the top of Figure 12b. It is interesting to note the relative positions of the small and large-scale sections on the correlation dimension curves in Figure 12b. For the data with noise, the whole correlation dimension curve has higher values than for the data without noise. Moreover, the correlation dimension curve for the noisy data converges on that for the noiseless data at larger scales. This trend is typical of data containing measurement and dynamic noise, as opposed to noiseless data with the same dynamics. A higher correlation dimension at small scales indicates more intricate microstructure in the attractor, which is typical for data containing measurement noise. A higher correlation dimension at large scales indicates more complex macrostructure in the attractor or higher dimensionality, which is typical of data containing dynamic noise (or, alternatively, more complex dynamics).

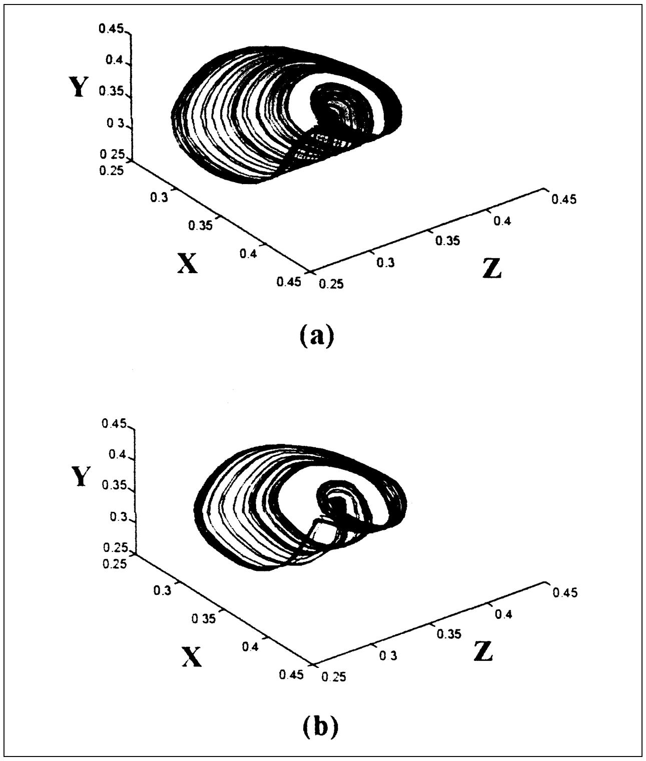


Figure 11. Dynamic attractor of autocatalytic process reconstructed from (a) the Y-state, and (b) the PL-RBF free-run model of the Y-state.

Figure 13a shows the ability of the model to predict the data with measurement and dynamic noise one-step-ahead, while Figure 13b shows the free-run predictions of the model based on the same data. These figures indicate that though the model was able to make acceptable one-step predictions, it failed to capture the detail dynamics by not following the peaks and troughs in the observed data during free-run predictions.

Discussion and Conclusions

With surrogate data methods, model validation is based on criteria related to the topology of the trajectory of the system in state space. Instead of comparing models based on single-valued statistical criteria, they can be compared on multiple scales of attractor topology by means of surrogate data methods. This allows better discrimination between models, and can in principle also aid in the development of better models where one model is not consistently better than the other over the entire range of scales.

From this investigation, it is evident that different nonlinear models may produce excellent one-step predictions, from which it may be very difficult to assess or compare the general validity of the models. For example, from the free-run predictions of the neural net and PL-RBF models, it is quite clear that the PL-RBF model was better able to capture the process dynamics from the smaller data set, than from the neural network.

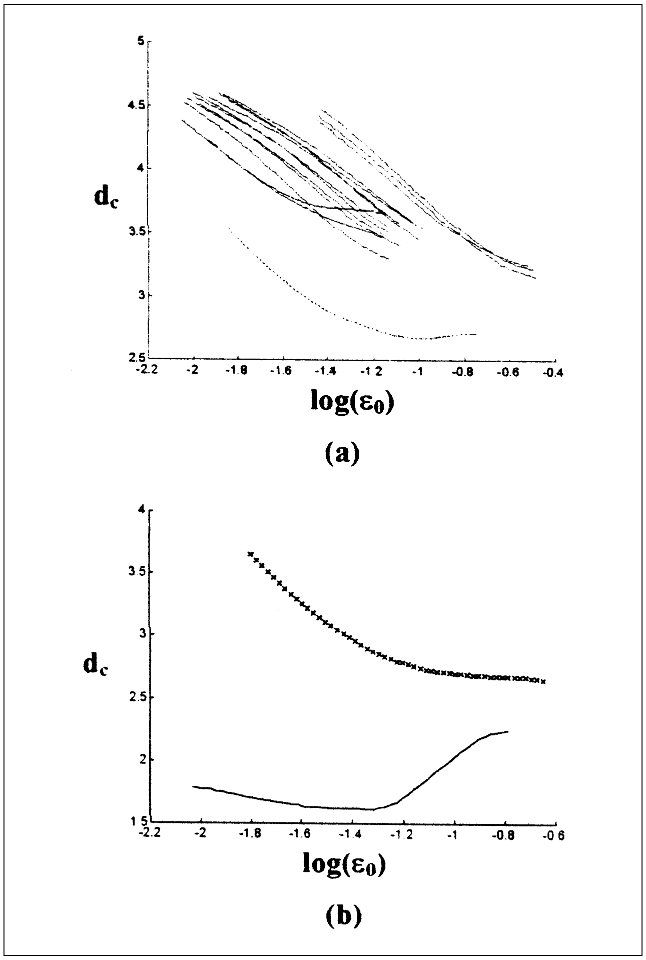


Figure 12. Correlation dimension curves.

(a) For autocatalytic Y-state with noise (broken line, bottom) and its AAFT surrogates, and (b) for Y-state with noise (top) and Y-state without noise (bottom). The bottom curve in (a) and the top curve in (b) are the same.

As far the application of surrogate data methods on the autocatalytic process in this investigation is concerned, the following can be concluded:

- Surrogate data are particularly valuable for the screening of data prior to model building. It is not always easy to determine the degree of determinism or stochasticity of real-world data, and the technique allows the modeler to inspect the data prior to building a model.
- Since the correlation dimension characterizes the topology of the attractor of the system in state space, it is a more rigorous criterion for the validation of dynamic process models than statistical or empirical criteria often used in practice.
- Smaller data sets are less likely to represent the full-range of the dynamic behavior of a system and can therefore lead to the construction of less accurate global models. This can be readily assessed by the use of surrogate data methods to visualize the performance of the system.
- Although a multilayer perceptron, as well as a pseudolinear radial basis function model, was capable of similar, accurate one-step-ahead prediction of a chaotic autocatalytic pro-

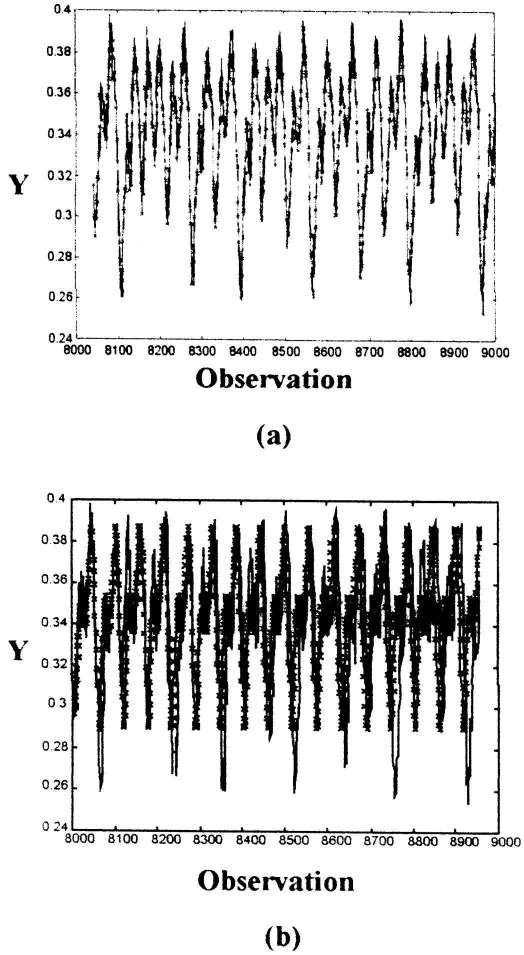


Figure 13. One-step predictions of autocatalytic Y-state (+ marker) with the PL-RBF model vs. the Y-state with measurement and dynamic noise (a), and the free-run prediction of the same data with the PL-RBF model (b).

cess, the pseudolinear radial basis function model was better able to capture the underlying dynamics of the system.

Literature Cited

Farmer, J. D., and J. J. Sidorowich, "Predicting Chaotic Time Series," *Phys. Rev. Lett.*, **34**, 845 (1987).

Frazer, A. M., and H. L. Swinney, "Independent Coordinates for Strange Attractors," *Phys. Rev. Lett.*, **33**, 1134 (1986).

Goutte, C., "Note on Free Lunches and Cross-Validation," *Neural Comput.*, **9**, 1245 (1997).

Grassberger, P., and I. Procaccia, "Characterization of Strange Attractors," *Phys. Rev. Lett.*, **50**, 346 (1983).

Gray, P., and S. K. Scott, "Autocatalytic Reactions in the Isothermal, Continuous Stirred Tank Reactor: Isolas and Other Forms of Multistability," *Chem. Eng. Sci.*, **38**, 29 (1983).

Gray, P., and S. K. Scott, "Autocatalytic Reactions in the Isothermal, Continuous Stirred Tank Reactor. Oscillations and Instabilities in the System A + 2B → 3B, B → C," *Chem. Eng. Sci.*, **39**, 1087 (1984).

Judd, K., "An Improved Estimator of Dimension and Some Comments on Providing Confidence Intervals," *Physica D*, **56**, 216 (1992).

Judd, K., "Estimating Dimension from Small Samples," *Physica D*, **71**, 421 (1994).

Judd, K., and A. Mees, "On Selecting Models for Non-Linear Time Series," *Physica D*, **82**, 426 (1995).

Kennel, M. B., R. Brown, and H. D. I. Abarbanel, "Determining Minimum Embedding Dimension Using a Geometrical Construction," *Physica Review A*, **45**, 3403 (1992).

Lai, Y.-C., and D. Lerner, "Effective Scaling Region for Computing the Correlation Dimension from Chaotic Time Series," *Physica D*, **115**, 1 (1998).

Lynch, D. T., "Chaotic Behavior of Reaction Systems: Parallel Cubic Autocatalators," *Chem. Eng. Sci.*, **47**, 347 (1992).

MacMurray, J. C., and D. M. Himmelblau, "Modeling and Control of a Packed Distillation Column Using Artificial Neural Networks," *Comp. Chem. Eng.*, **19**, 1077 (1995).

Osborne, A. R., and A. Provenzale, "Finite Correlation Dimension for Stochastic Systems with Power-Law Spectra," *Physica D*, **35**, 357 (1989).

Pham, D. T., X. Liu, and S. J. Oh, "Dynamic System Identification Using Elman and Jordan Neural Networks," *Neural Networks for Chemical Engineers*, A. B. Bulsari, Elsevier, ed., Chap. 23, The Netherlands, p. 573 (1995).

Rivals, I., and L. Personnaz, "On Cross Validation for Model Selection," *Neural Comput.*, **11**, 863 (1999).

Schreiber, T., and A. Schmitz, "Improved Surrogate Data for Nonlinearity Tests," *Phys. Rev. Lett.*, **77**, 635 (1996).

Small, M., and K. Judd, "Comparison of New Non-Linear Modelling Techniques With Applications to Infant Respiration," *Physica D*, **117**, 283 (1998a).

Small, M., and K. Judd, "Correlation Dimension: A Pivotal Statistic for Non-Constrained Realizations of Composite Hypotheses in Surrogate Data Analysis," *Physica D*, **129**, 386 (1998b).

Stefanovska, A., S. Strle, and P. Kroselj, "On the Overestimation of the Correlation Dimension," *Phys. Lett. A*, **235**, 24 (1997).

Su, H-T., T. J. McAvoy, and P. Werbos, "Long Term Predictions of Chemical Processes Using Recurrent Neural Networks: A Parallel Training Approach," *Ind. Eng. Chem. Res.*, **31**, 1338 (1992).

Takens, F., "Detecting Non-Linearities in Stationary Time Series," *Int. J. of Bifurcations and Chaos.*, **3**, 241 (1993).

Takens, F., "Detecting Strange Attractors in Turbulence," *Lecture Notes in Mathematics*, **898**, Springer, Berlin, 366 (1981).

Theiler, J., "On the Evidence for Low-Dimensional Chaos in an Epileptic Encephalogram," *Physics Letters A*, **196**, 335 (1995).

Theiler, J., E. Eubank, A. Longtin, and B. Galdrikian, "Testing for Non-Linearity in Time Series: The Method of Surrogate Data," *Physica D*, **58**, 77 (1992).

Theiler, J., and D. Pritchard, "Constrained Realization Monte Carlo Method for Hypothesis Testing," *Physica D*, **94**, 221 (1996).

Theiler, J., and P. E. Rapp, "Reexamination of the Evidence for Low-Dimensional Non-Linear Structure in the Human Electroencephalogram," *Encephalography and Clinical Neurophysiology*, **98**, 213 (1996).

Zhu, H., and R. Rohwer, "No Free Lunch for Cross-Validation," *Neural Computation*, **8**, 1421 (1996).

Appendix: Example of Generation of Surrogate Data

In this appendix the generation of surrogate data of different types for time series is illustrated. In all cases, the time series $y = 1.9, 3.2, 1.2, 2.2, 0.9, 2.1, 2.9, 2.3$ is considered.

Generating index-shuffled surrogates (Type 0)

(1) Generate normally distributed data

$$y_r = \text{randn}(\text{size}(y))$$

as before $y_r = -0.30, -1.28, 0.24, 1.28, 1.20, 1.73, -2.18, -0.23$

(2) Sort random data in ascending order and store sequence of indices

$\boldsymbol{y}_{r2} = \mathrm{sort}(\boldsymbol{y}_r)$, giving $\boldsymbol{y}_{r2} = -2.18,\ -1.28,\ -0.30,\ -0.23,\ 0.24, 1.20, 1.28, 1.73$

$\mathrm{indices}(\boldsymbol{y}_{\mathrm{r}2}) = 7, 2, 1, 8, 3, 5, 4, 6$

(3) Sort observed data to rank order of random data

$\boldsymbol{y}_{s0} = y(7), y(2), y(1), \ldots y(6) = 2.9, 3.2, 1.9, 2.3, 1.2, 0.9, 2.2, 2.1$

where $\boldsymbol{y}_{s0}$ is the surrogate data of type 0. Different surrogate data sets will be generated by different random samplings ($\boldsymbol{y}_r$).

Generating phase-shuffled surrogates (Type 1)

To generate phase-shuffled surrogate data, the following procedure can be used:

(1) Calculate the Fourier transform of the time series

$$\boldsymbol{y}_{\mathrm{fft}} = FFT(\boldsymbol{y})$$

$\boldsymbol{y}_{\mathrm{fft}} = 16.70, 1.8485 + 0.9929i,\ -1.3000 - 0.8000i,\ 0.1515 - 2.4071i,\ -2.9000, 1.8485 - 0.9929i\ -1.3000 + 0.8000i,\ 0.1515 + 2.4071i$

(2) Generate uniformly distributed random complex numbers with modulus 1.

$$\boldsymbol{b} = 2\pi\ \mathrm{rand}(\mathrm{size}(\boldsymbol{y}))$$

such as $\boldsymbol{b} = 5.9698, 1.4523, 3.8129, 3.0535, 5.6002, 4.7884, 2.8681, 0.1163$

$$\boldsymbol{z} = \exp(i\boldsymbol{b})$$

$\boldsymbol{z} = 0.9513 - 0.3082i, 0.1182 + 0.9930i,\ -0.7830 - 0.6220i,\ -0.9961 + 0.0880i, 0.7757 - 0.6311i,\ 0.0759 - 0.9971i,\ -0.9628 + 0.2701i, 0.9932 + 0.1160i$

(3) Determine the indices for the first half of the FFT conjugate pairs, that is, not counting first and symmetry point, indices = 2, 3, 4 and indices (conjugates) = 6, 7, 8.

(4) Multiply original Fourier Transform with random complex number vector, as follows:

$\boldsymbol{ys}^{+}_{\mathrm{fft}} = \boldsymbol{y}_{\mathrm{fft}}(\mathrm{indices})\ \boldsymbol{z}(\mathrm{indices})$

$\boldsymbol{ys}^{-}_{\mathrm{fft}} = \mathrm{conj}[\boldsymbol{y}_{\mathrm{fft}}(\mathrm{indices})][\mathrm{conj}(\boldsymbol{z}(\mathrm{indices})],$ where indices are ordered in reversed ranking

$$ys_{\mathrm{fft}}(1) = y_{\mathrm{fft}}(1) = 16.70$$
$$ys_{\mathrm{fft}}(2) = y_{\mathrm{fft}}(2)^{*} z(2) = -0.7674 - 1.9529i$$
$$ys_{\mathrm{fft}}(3) = y_{\mathrm{fft}}(3)^{*} z(3) = 0.5203 - 1.4350i$$
$$ys_{\mathrm{fft}}(4) = y_{\mathrm{fft}}(4)^{*} z(4) = 0.0609 - 2.4111i$$
$$ys_{\mathrm{fft}}(5) = y_{\mathrm{fft}}(5) = -2.9000$$
$$ys_{\mathrm{fft}}(6) = \mathrm{conj}[y_{\mathrm{fft}}(4)]^{*}\mathrm{conj}[z(4)] = 0.0609 + 2.4111i$$
$$ys_{\mathrm{fft}}(7) = \mathrm{conj}[y_{\mathrm{fft}}(3)]^{*}\mathrm{conj}[z(3)] = 0.5203 + 1.4350i$$
$$ys_{\mathrm{fft}}(8) = \mathrm{conj}[y_{\mathrm{fft}}(1)]^{*}\mathrm{conj}[z(1)] = -0.7674 + 1.9529i$$

(5) Finally, calculate the type 1 surrogate data set as the real part of the inverse Fourier transform of $\boldsymbol{ys}_{\mathrm{fft}}$.

$\boldsymbol{y}_{s1} = \mathrm{real}\left[FFT^{-1}(\boldsymbol{ys}_{\mathrm{fft}})\right]$

$\boldsymbol{y}_{s1} = 1.6784, 1.1734, 1.7095, 2.1837, 2.0317, 3.0091, 1.4804, 3.4338$

where $\boldsymbol{y}_{s1}$ is a surrogate data set of Type 1.

Generating Type 2 surrogate data

(1) Generate random data (as with Type 0 surrogates)

$$\boldsymbol{y}_r = \mathrm{randn}[\mathrm{size}(\boldsymbol{y})]$$

$\boldsymbol{y}_r = -0.30,\ -1.28, 0.24, 1.28, 1.20, 1.73,\ -2.18,\ -0.23$

(same set used as above)

(2) Randomly shuffle the original data and store the new sequence of indices

$$\boldsymbol{y}_2 = \mathrm{sort}(\boldsymbol{y})$$

$\boldsymbol{y}_2 = 0.9, 1.2, 1.9, 2.1, 2.2, 2.3, 2.9, 3.2$

$\boldsymbol{i}_2 = \mathrm{Indices}(\boldsymbol{y}_2) = 5, 3, 1, 6, 4, 8, 7, 2$

$\boldsymbol{y}_{r2} = \mathrm{sort}(\boldsymbol{y}_r)$

$\boldsymbol{y}_{r2} = -2.18,\ -1.28,\ -0.30,\ -0.23, 0.24, 1.20, 1.28, 1.73$

$\boldsymbol{i}_{r2} = \mathrm{Indices}(\boldsymbol{y}_{r2}) = 7, 2, 1, 8, 3, 5, 4, 6$

Let $\boldsymbol{y}_{t1}(\boldsymbol{i}_{r2}) = \boldsymbol{y}_{r2}$

that is, $y_{t1} = y_{r2}(7), y_{r2}(2), \ldots, y_{r2}(6) = 2.9, 3.2, 1.9, 2.3, 1.2, 0.9, 2.2, 2.1$

(3) Shuffle phase of random vector y_{s1} as explained above for generation of Type 1 surrogate data.

$$\boldsymbol{y}_{t2} = \mathrm{shuffle\_phase}(\boldsymbol{y}_{t1}),$$

which gives $\boldsymbol{y}_{t2} = -0.0671, 1.1552,\ -0.9412, 2.2162,\ -0.5305,\ -0.7462,\ -0.9412, 0.3148$

(4) Sort phase-shuffled random data in ascending order

$$\boldsymbol{y}_{t3} = \mathrm{sort}(\boldsymbol{y}_{t2});$$

$\boldsymbol{y}_{t3} = -0.9412,\ -0.9412,\ -0.7462,\ -0.5305,\ -0.0671,\ 0.3148, 1.1552, 2.2162$

$\boldsymbol{i}_{yt3} = \mathrm{Indices}(\boldsymbol{y}_{t3}) = 3, 7, 6, 5, 1, 8, 2, 4$

(5) Sort data to rank order of random data

$$\boldsymbol{y}_{s2}(\boldsymbol{i}_{yt3}) = \boldsymbol{y}_2 = y_2(3), y_2(7), \ldots, y_2(4)$$
$$\boldsymbol{y}_{s2} = 2.2, 2.9, 0.9, 3.2, 2.1, 1.9, 1.2, 2.3$$

where $\boldsymbol{y}_{s2}$ is a surrogate data set of Type 2.

Manuscript received June 23, 1999, and revision received Dec. 6, 2000.