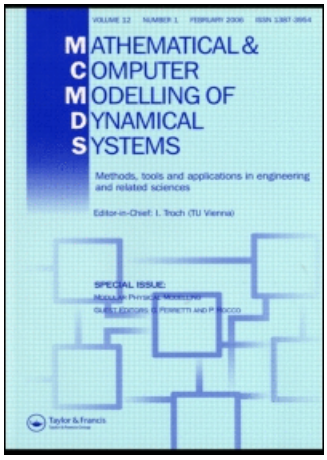


Mathematical and Computer Modelling of Dynamical Systems
Methods, Tools and Applications in Engineering and Related Sciences
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713682513

# Multivariate polynomial regression for identification of chaotic time series

Online Publication Date: 01 August 2007
To cite this Article: Vaccari, D. A. and Wang, H. -K. (2007) 'Multivariate polynomial regression for identification of chaotic time series', Mathematical and Computer Modelling of Dynamical Systems, 13:4, 395 - 412
To link to this article: DOI: 10.1080/13873950600883691
URL: http://dx.doi.org/10.1080/13873950600883691

*Mathematical and Computer Modelling of Dynamical Systems*
Vol. 13, No. 4, August 2007, 395–412





# Multivariate polynomial regression for identification of chaotic time series

D. A. VACCARI*† and H.-K. WANG‡

†Stevens Institute of Technology, Hoboken, NJ, USA
‡Cisco Systems, Inc., San Jose, CA, USA

Multivariate polynomial regression was used to generate polynomial iterators for time series exhibiting autocorrelations. A stepwise technique was used to add and remove polynomial terms to ensure the model contained only those terms that produce a statistically significant contribution to the fit. An approach is described in which datasets are divided into three subsets for identification, estimation, and validation. This produces a parsimonious global model that is can greatly reduce the tendency towards undesirable behaviours such as overfitting or instability. The technique was found to be able to identify the nonlinear dynamic behaviour of simulated time series, as reflected in the geometry of the attractor and calculation of multiple Lyapunov exponents, even in noisy systems.

The technique was applied to times series data obtained from simulations of the Lorenz and Mackey–Glass equations with and without measurement noise. The model was also used to determine the embedding dimension of the Mackey–Glass equation.

*Keywords*: Nonlinear modelling; Lyapunov coefficients; Multivariate polynomial Regression; Chaos; Identification

## Selected nomenclature

| | |
|---|---|
| $a_i$ | Model coefficient. |
| $b_{p,m}$ | Exponent of variable $m$ in term $p$. |
| $df$ | Degrees of freedom. |
| $D_i^2$ | Normalized error squared. |
| $e[i, j]$ | Error of variable $i$ taken at time step $j$. |
| $l_m$ | Maximum lag in the model. |
| $l_0$ | Index indicating if lag 0 is included in set of candidate lags. |
| $n_d$ | Number of data points. |
| $n_e$ | Number of candidate exponents. |
| $n_l$ | Number of candidate lags. |

*Corresponding author. Email: dvaccari@stevens.edu

*Mathematical and Computer Modelling of Dynamical Systems*
ISSN 1387-3954 print/ISSN 1744-5051 online © 2007 Taylor & Francis
http://www.tandf.co.uk/journals
DOI: 10.1080/13873950600883691

$n_m$ Maximum number of variables (multiplicands) in a term.
$n_p$ Number of terms in the model.
$n_t$ Number of candidate terms.
$n_v$ Number possible independent variables in a model, including lagged variables.
$n_x$ Number of independent variables, ignoring lags.
$MSE_i$ Mean square error for variable $i$.
$SSE_i$ Sum of squares of the error for variable $i$.
$TSS_i$ Total sum of squares of variable $i$.
$t_p$ $t$-statistic for parameter $a_p$.
$x[i, j]$ Measurement of variable $i$ taken at time step $j$.
$z_p[i]$ Polynomial term $p$ evaluated at point $i$.
$\lambda$ Lyapunov exponent.
$\tau$ Sampling interval.
$\sigma_p$ Standard error for $a_p$.



## 1. Introduction

In the absence of a fundamental understanding of a process one may turn to empirical system identification. A function is selected *a priori* to relate the outcome of the process to input and state variables, and the parameters of the function found by fitting to experimental data. Such a relationship may be used for forecasting, control, optimization, sensitivity analysis or correlation analysis. Although robust techniques have been developed for linear processes, there is much interest on extending the capabilities of these techniques to the modeling of nonlinear processes. A particularly challenging class of nonlinear processes are chaotic time series, in which correlations decay rapidly with the time interval between observations. A technique that proves capable of identifying chaotic processes may find wide applicability to other nonlinear systems. Analysis of nonlinear time series may also be used in the determination of whether or not a series represented by a set of data is chaotic.

This paper describes the use of multivariate polynomial regression to identify low-dimensional chaotic time series with a single, global model. A new model identification/estimation procedure is described in which the data are divided and model terms incorporated according to the statistical significance of their estimated coefficients in one dataset, coupled with their capacity to improve global predictions in the second. The method accurately forecasts differential equation-based chaotic time series and captures their dynamic characteristics as expressed by multiple Lyapunov exponents and the attractor geometry, even in the presence of measurement noise. The Lyapunov exponents are used to determine if the system is chaotic.

### 1.1 *Existing modeling approaches*

A great variety of approaches exist for modeling time series. The method used in this work can be compared to two which are most widely known and applied: the autoregressive moving average (ARMA) models [1] and artificial neural networks (ANN) [2]. The polynomial models described below can be considered to be a generalization of the former, and to be motivated by the computational intensity of the latter.

We will consider applications to data formed from $n_d$ measurements sampled at equal time intervals, $\tau$, of $n_x$ input, state and output variables. Here we will not distinguish among these types of variables. The data point $x[i, j]$ is the measurement of variable $i$ taken at time step $j$. In general, each variable may depend upon previous measurements of itself and the other variables, except for input variables which do not depend upon state and output variables.

A type of ARMA model applied to such systems is the vector autoregressive (VAR) model, in which a prediction is found by a linear combination of previous (lagged) measurements, $x[i, j - l]$:

$$x[i, j] = a_0[i] + \sum_{p=1}^{n_p} a_p[i] \cdot x[k_p, j - l_p] + e[i, j] \tag{1}$$

where $1 \le i \le n_x$, $1 \le k_p \le n_x$; $k_p \ne i$ if $l_p = 0$; $l_m + 1 \le j \le n_d$; $l_m \ge 1$ is the maximum lag; and $e[i, j]$ is the error in the model prediction. The parameters $a$ are determined by fitting the model to a set of data. An identification process is used to select which of the possible terms in equation (1) contribute significantly to the model, and only those terms are retained.

This model is adequate for many systems, at least over restricted ranges. However, it is incapable of reproducing certain kinds of process behaviours such as changing sensitivity to an individual variable (curvilinear response), conditions in which the sensitivity of one variable to another depends upon a third variable (interactions), and extreme sensitivity to initial conditions (chaotic behaviour). Furthermore, in the presence of such nonlinear behaviours a linear model may produce wrong and misleading results.

The ANN [2] is one of the few widely available methods that are capable of capturing these nonlinear effects. However, they have the following disadvantages: They are often overspecified, possessing many more weights than necessary. This can result in overfitting, or 'memorization', of the data, in which the fitting approaches interpolation. This, in turn, may produce degraded performance if the model is applied to independent data. The values obtained for the weights do not directly yield information on the behaviour of the model, particularly on its sensitivity or causal relationships. ANN models do not lend themselves to statistical analysis, and confidence intervals on forecasts or on individual weights are not easily computed. Finally, ANNs often require very large amounts of computational time to fit a single set of data; it is not unusual to allow fitting procedures to run for many hours or days on workstation computers. However, the robust pattern detection capability gained with ANNs leads to a recognition that such computational intensity may be less of a disadvantage in light of the low cost of computing when compared with the cost of the data and the value of having a good model. A further disadvantage of ANNs is that they are difficult to communicate to others; they cannot be easily written down in mathematical notation or incorporated into computer codes by those not posessing the same software. Thus investigators usually fit the models themselves rather than use ones identified by others.

Another nonlinear model that can capture chaotic dynamics is that of Farmer and Sidorowich [3] and of Sugihara and May [4]. This is a locally linear model with weights found by fitting to nearby points in the phase space of a dataset. This model is not parsimonious, since it requires that the entire learning dataset be available during model evaluation. It is complicated to use because it does not reduce to a single set of weights. Instead, the weights must be found repeatedly for each prediction by

successively fitting the model to nearby points from the dataset. It is desirable to have a global model that can be concisely stated. With the ARMA models and the polynomial models described in this paper, once the model form and its parameters are found using the dataset, the dataset is no longer needed for model evaluation.

### 1.2 *Nonlinear systems; chaotic systems, and Lyapunov exponents*

Some nonlinear systems may exhibit chaotic behaviour. Chaos is characterized by aperiodic motion confined to a region of phase space. The locus of trajectories in phase space describe a figure called a strange attractor. The chaotic dynamics of an $n$-dimensional system can also be characterized by a spectrum of $n$ Lyapunov exponents. The Lyapunov exponent is the coefficient describing the exponential rate at which the distance between two points in phase space, initially arbitrarily close together, changes with time. One of the Lyapunov exponents of a three-dimensional strange attractor will be positive, one will be zero, and the other will be negative. This is designated spectral type $(+, 0, -)$. This positive exponent results in long-term unpredictability, as small measurement errors grow exponentially. Nevertheless, prediction of chaotic motion is possible in the short term, and this capability can be exploited for controlling the process [5]. Other possible exponent spectra are $(0, -, -)$, a toroidal (perodic) attractor, and $(-, -, -)$ a fixed point.

Wolf *et al.* [6] developed a procedure for computing the Lyapunov exponents of systems described by differential equations. A trajectory is computed numerically, and simultaneously a set of orthogonal axes centered on the trajectory is evolved via a linearization of the differential equations. After each time step the exponential rates of growth of the axes are computed and Gram–Schmidt reorthonormalization applied to the axis to maintain their orthogonal orientation. The exponential rate of change in the length of the orthogonal axes are summed and averaged as the attractor is traversed, yielding the Lyapunov exponents directly.

A more difficult problem is computing Lyapunov exponents from experimental time series where the underlying model is unknown. Wolf *et al.* [6] describe a modification of the above procedure for this problem which involves the use of almost orientation-preserving replacement elements along the trajectory. This method introduces errors into the measurement and 'appears to be prohibitively expensive for attractors of dimension greater than 3 or 4'. Sugihara and May [4] exploit their modeling technique to detect chaos by a method which involves observing a 'signature' in a plot of the modeling correlation coefficient versus the embedding dimension. However, they indicated that this method might not clearly discriminate chaos from coloured noise with short-term correlations. Furthermore, this method does not directly yield an estimate of the Lyapunov exponents.

Wolf *et al.*'s method for time-series data amounts to forming a linear local map, in which nearby points are used to determine a model of the process that is only valid within the vicinity. Lyapunov coefficients are determined by successive use of the models as the trajectory is traversed. Brown *et al.* [7] used local polynomial mapping, up to fifth order. Briggs [8] used local multivariate polynomials up to third order for computing Lyapunov coefficients, but not for forecasting. Although these methods are capable of determining the Lyapunov coefficients, it would be desirable to have an approach that also produces a model that is useful for forecasting and other uses. Local mapping models require that all the original data be retained for use in forecasting. Casdagli [9] used global polynomial models for forecasting, but did not compute

Lyapunov coefficients with them. However, he formulated his polynomial models *a priori*, keeping terms in the model that may have been nonsignificant. This can have effects on the stability of the solutions, as described below. The method described here produces a global model better suited for forecasting because it is more parsimonious and consists only of terms that contribute significantly to the fit. Billings and Aguirre [10] used parsimonious polynomial models to identify chaotic systems, but they used a qualitative measure of fit, and did not measure the Lyapunov exponent. Guo and Billings [11] did confirm prediction of the Lyapunov exponent for a chaotic coupled map lattice. This paper extends these results by fitting global polynomial models to differential equation chaotic systems developed as time-series, by computing multiple Lyapunov exponents, including the more challenging task of computing even the negative Lyapunov coefficient, and also by demonstrating recovery of this dynamic information from chaotic systems with introduced noise.

## 2. Multivariate polynomial regression

More complex behaviours, such as coupled sensitivities between variables or curvature in the responses, could be included in an VAR model by adding polynomial terms to equation 1:

$$x[i, j] = \sum_{p=1}^{n_p} a_p[i] \cdot z_p[i] + e[i, j] \tag{2a}$$

$$z_p[i] = \prod_{m=1}^{n_m} x^{b_{p,m}[i]}[k_{p,m}, j - l_{p,m}] \tag{2b}$$

The additional parameters of this model compared to equation (1) are $b_{p,m}$, the (usually positive integer) exponents for each multiplicand in each term, and $n_m \leq n_v$, the maximum number of multiplicands in each term of the model ($n_v$ is the number of possible independent variables in the model, including lagged variables). The indices $i$, $j$, $k$, and $l$ are defined as for equation (1).

For the case of $b = \{0, 1, 2, \ldots, \infty\}$ and $n_m = n_v$ the model may approximate a Taylor polynomial. A Taylor polynomial, of course, can exactly represent any functional relationship among the independent variables (which may include any selected lags). In principle, one could achieve an arbitrary degree of accuracy with a finite set of terms.

Equations (2a) and (2b) are a polynomial form of the nonlinear autoregressive model with exogenous variables (NARX model) [12–14]. This, in turn, may be thought of as a generalization of Volterra models. Volterra models, however, do not include lagged values of the dependent variable. We also use the term multivariate polynomial regression (MPR) model as a more general term to reflect that this kind of model can also be used in nonlinear correlation analysis of non-time-series data.

Casdagli [9] used global polynomial models for prediction of chaotic time series but did not remove nonsignificant terms. Casdagli did not use the resulting models to compute Lyapunov coefficients. Including nonsignificant terms can result in parasitic solutions that grow rapidly, causing the model to 'blow up' upon iteration. Restricting the model to terms that are statistically significant helps ensure that behaviours in the model correspond to behaviours in the data. As a consequence of this restriction, if the data describes a stable process, the model should tend not to be unstable. However, this is not guaranteed (which is also true with ARMA or ANN models).

The model is made tractable by restricting the values that can be taken on by the exponents, $b$, the lags, and the value of $n_m$, and by including in the model only those terms that contribute significantly to the fit. The fitting procedure involves a stepwise selection process, described below, in which a set of candidate terms are tested for inclusion in the model.

The restricted set of candidate terms are formed as follows: First, a set of $n_e$ candidate exponents is selected, not including zero, which is always assumed. Then, a set of lags to be considered is formed. Determination of the sets of exponents and of lags may be an iterative process involving sequentially adding to each set until the model cannot be improved. In some cases discontinuous lags may be included to represent expected seasonal effects. If lag 0 is included in the list, then variable $i$ is being correlated to 'current' values of the other variables, and variable $i$, lag 0 (the dependent variable) must, of course, be excluded from the candidate terms. However, for forecasting purposes only prior values are used as independent variables, and lag zero is excluded. The total number of lags in the set, which may include lag zero, is $n_l$.

The stepwise procedure then selects a set of polynomial terms from the candidates which optimizes the fitting criteria. The resulting MPR model can be completely specified by a table containing the following information for each term:

$$k_{p,1}, l_{p,1}, b_1; k_{p,2}, l_{p,2}, b_2; \ldots k_{p,n_m}, l_{p,n_m}, b_{n_m}; a_p$$

### 2.1 *The number of candidate terms*

Adding lagged values increases the effective number of independent variables. The total number of independent variables, $n_v$, will be:

$$n_v = (n_x - 1) \cdot n_l + v_0 \cdot (n_l - l_0) \tag{3}$$

where $l_0$ equals one if the list of lags includes zero, and equals zero otherwise, and $v_0$ equals one if lagged dependent variables are included as independent variables, and zero otherwise.

The maximum value that $n_m$ can take is $n_v$. If $n_m = n_v$, then the total number of candidate terms is:

$$n_t = (n_e + 1)^{n_v} \tag{4}$$

This may result in a large number of terms to be tested for selection into the model. For example, if there are three independent variables ($n_x = 4$) and no lags ($n_l = 1$), and 10 exponents ($n_e = 10$), then there are $n_t = 1331$ possible terms. Experience with a wide variety of datasets has shown that $n_m$ can often be restricted to two or three. For $n_m \leq n_v$:

$$n_t = \sum_{m=0}^{n_m} C(n_v, m) \cdot {n_e}^m \tag{5}$$

where $C(n_v, m)$ is the number of combinations of $n_v$ objects taken $m$ at a time. For the example above, if $n_m = 2$, the number of candidate terms drops to 331. As $n_v$ and $n_e$ increase, this restriction becomes more important. For example, with $n_x = 10$, $n_l = 2$, $n_e = 2$, $l_0 = 1$, and $v_0 = 0$, $n_v = 18$. If $n_m = n_v$, then there are $n_t = 387\,420\,489$ possible terms. Restricting to $n_m = 2$ gives $n_t = 649$.

### 2.2 *Identification, estimation and validation*

The terms to be included in the model, together with estimates of the coefficients, can be obtained using stepwise multiple regression [15–17]. Stepwise multiple regression procedure selects from the many candidate terms only those that contribute significantly to the predictive capability of the model. Elimination of insignificant terms preserves parsimony and reduces the tendency towards overfitting and parasitic solutions that contribute to instability and inaccuracy.

In the stepwise procedure terms are added or removed one-at-a-time in a series of successive cycles. One cycle consists of a term-adding step followed by one or more term-removal steps. In the term-adding step the current model is augmented by each candidate term not currently in the model, one at a time. For each such candidate model a test statistic is computed to measure how strongly the candidate term improves the model. If the model can be significantly improved, then the best candidate model is made the current model. This is followed by a removal step in which new candidate models are formed by taking the current model and deleting each term one-at-a-time. If removal of any single term can improve the model, then the model which results in the greatest improvement is made the current model. The removal step is repeated until the model cannot be improved by removal of any single term, and then another addition step is begun. The procedure is started with no terms in the model ($x[i, j] = 0$), and continued until no further improvement can be obtained either by adding or removing any single term.

In order to compute test statistics the parameters of each candidate model must be estimated. This was done by a least squares regression using singular value decomposition (SVD) [18]. SVD was used instead of the more conventional approach involving matrix solution of the normal equations because SVD can detect and compensate for problems involving ill-conditioned matrices. Polynomial regression is notorious for such problems. It should be noted, however, that in the course of this work and numerous other MPR calculations, ill-conditioned matrices were rarely encountered. Apparently this is more likely to be a problem in situations involving low degrees of freedom, which approaches polynomial interpolation. Billings *et al*. [19] used orthogonal estimation to counter ill-conditioning. However, Piroddi and Spinelli [17] point out that this can result in models containing additional terms that may not contribute significantly to the model, and requiring a 'pruning' method to finish the model that does not use orthogonal polynomials. Here, in order to further reduce roundoff error problems the data were scaled by the mean of the absolute value of the data before fitting.

The statistic used in this work to select terms for adding or removing was the $t$-statistic for the hypothesis that the candidate term coefficient was different from zero:

$$t_p = a_p/\sigma_p \tag{6}$$

where $\sigma_p$ is the standard error of $a_p$, as computed using the SVD routine. Candidate terms were sorted according to their $t$-statistic, and if any exceeded the critical $t$ at a 95% confidence level, then the term with the largest absolute value of $t$ was added to the model. At each removal step, if the $t$-statistics of any terms were less than that corresponding to the 90% confidence level, then the term with the smallest absolute value of $t$ was removed from the model. All coefficients were then recomputed before beginning another step. A lower confidence level was used for removal than for addition to prevent an infinite loop of term addition and removal from occurring. In

some cases an additional constraint was placed on all changes to the current model in which the mean square error (*MSE*), defined below, of the new current model was required to be less than that of the previous one.

Various measures of the goodness-of-fit of the model may be computed. All are based on the sum of squares of the error for variable $i$ ($SSE_i$):

$$SSE_i = \sum_{j=l_m}^{n_d} e^2[i, j]. \tag{7}$$

The criterion that will be used here is the proportion of the variance in the data not explained by the model; it is the same measure of predictor error used by Casdagli [9]:

$$D_i^2 = SSE_i / TSS_i \tag{8}$$

where $TSS_i$ is the total sum of squares for $x[i, \cdot]$:

$$TSS_i = \sum_{j=l_m}^{n_d} (x[i, j] - \bar{x}[i])^2 \tag{9}$$

and where $\bar{x}[i]$ is the mean of $x[i, j]$. Note that Farmer and Sidorowich [3] call $D^2$ 'normalized error, $E$', Casdagli denotes it $\sigma$. It might be best to preserve the squared notation; it is the ratio of two squares and it is non-negative. In the case of the regression with the original fitting data, $D^2$ is related to the multiple-$R^2$ used in multiple regression by $D^2 = 1 - R^2$. However, when a model is applied to independent data that were not used in the fitting, it is possible for SSE to exceed the TSS if the prediction is poor enough. In such cases $D^2$ will be greater than 1.0.

A measure that penalizes models for having too many terms is the mean square error (the subscript $i$ indicating the variable being modelled is dropped for simplicity):

$$MSE = \frac{SSE}{df} \tag{10}$$

where $df = n_\mathrm{d} - l_m - n_p$ is the number of degrees of freedom in the model. The *MSE* is an unbiased estimate of the variance of the error.

An $F$-test can be used for testing disparate models; these may be compared with each other by forming a ratio between their MSE values:

$$F = \frac{MSE_1}{MSE_2}. \tag{11}$$

The probability of the MSE from model 1 ($MSE_1$) being greater than that of model 2 ($MSE_2$) can be determined by comparing the $F$-statistic from equation 11 to the $F$-distribution with degrees of freedom corresponding to the two *MSE* values. Thus, the performance of various models may be compared by using an independent dataset to compute the global model statistics $D^2$, *MSE*, or $F$ ratio, in increasing order of rigour.

We applied a unique cross-validation procedure for structure determination. The data were divided into three subsets, called the 'fit', 'test', and 'validation' datasets. A term is added to the MPR model only if it produced a $t$-statistic at the 5% significance level computed with the fit dataset, and also produced a decrease in a global goodness-of-fit criterion in the test dataset. In this work we used $MSE$ as the global criterion, but others such as the Akaike information criterion or Bayesian information criterion [20] could be used. Thus the information associated with the terms of the model, i.e. the coefficients and the associated $t$-statistics, are computed from the fit dataset during the identification procedure. When model identification is completed, the fit and test datasets may be reunited and used together for more accurate estimation of the model coefficients. This procedure avoids the overfitting that Lillekjendlie *et al.* [21] found polynomial models to be susceptible to.

The third, independent dataset is used for final validation. The validation dataset is not used for identifying or estimating the model, but may be used for comparing different modeling approaches or for reporting model performance. Note that in the results reported below, except where noted, all global model statistics are based on a validation dataset.

Billings and Voon [16] use a stepwise regression based on statistical tests of significance, but they do not use cross-correlation. Billings *et al.* [19] describe a stepwise method in which terms are selected based on their contribution to error reduction, with an *a priori* threshold for convergence instead of a computation of significance level (although they did use tests of significance in final validation). Piroddi and Spinelli [17] proposed replacing regression error with prediction error, but recognized that their approach would not work with chaotic systems due to the extreme sensitivity to initial conditions. The use of the $t$-test is similar to the analysis of variance method described by Lind and Ljung [22], except that they used all possible regressor combinations, instead of a stepwise algorithm. They also require that all factor level combinations be balanced, i.e. represented by an approximately equal amount of measurement data.

Even with the number of candidate terms restricted in the ways described above, a great many multilinear regressions (at least $n_p$ times $n_t$) often must be performed before the procedure converges to a final model. Nevertheless, as with ANNs, computational intensity is often relatively unimportant due to the currently low cost of computing, especially when compared to the cost of acquiring data or to the potential benefits of improved model performance.

Inclusion of the nonlinear terms makes the model capable of describing relationships between the dependent and independent variables that exhibit not only curvature, but also interactions in which the shape of the curvature depends upon other independent variables. This gives the MPR model a very general pattern-fitting capability similar to that often ascribed to ANNs. In one case the MPR model was found to produce a significantly better fit to experimental data than ANNs having more adjustable coefficients [23]. Because there are a finite number of possible models (once the lists of exponents and lags has been specified), the model converges absolutely. These procedures were implemented using Simetrica$^{TM}$ TaylorFit$^{TM}$ software [24].

### 2.3 *Using MPR models to compute Lyapunov coefficients*

The identification process described above converts any autocorrelated time series to a set of iteration polynomials, similar to the one-dimensional quadratic iterator or the

two-dimensional Henon iteration. This can be viewed as a filtering step to remove noise while retaining the underlying nonlinear dynamics. Wolf *et al.*'s [6] method for differential equations could then be used to compute the Lyapunov exponents for the resulting set of polynomials instead of using the replacement element technique with the dataset. The programme developed by Wolf [25] for differential equations was used in this work. Instead of incrementing time by numerical solution of the differential equations, the polynomials and their linearizations are used to step through time, and the exponential rate of divergence computed at each time step as above. This method is much simpler than the replacement element method described by Wolf *et al.* [6], and would be easily applied to attractors with higher dimension attractors.

Wolf *et al.*'s [6] method for differential equations was used with MPR models to compute Lyapunov exponents for time-series generated from the Lorenz system of equations and the Mackey–Glass equation.

## 3. Lorenz system

The Lorenz equations [26] were used to generate test data:

$$\begin{aligned} \dot{x} &= -a(x-y) \\ \dot{y} &= bx - y - xz \\ \dot{z} &= -cz + xy \end{aligned} \tag{12}$$

with $a = 10$, $b = 28$, and $c = 8/3$.

The differential equations were solved using the fourth-order Runge–Kutta method with a time increment of 1/64 time units. A dataset consisting of 2000 sets of points was generated from every fourth set of values produced by the simulation, producing a sampling interval of 1/16. The data were stored as six decimal digits. The resulting discretization error places a lower bound on $\log_{10}D^2$ on the order of $-6.0$.

Billings and Aguirre [10] showed that if the time increment is too short, then model identification is adversely affected. Conversely, if it is too long, then accuracy of parameter identification is degraded. They gave a rule of thumb for determining a range of values for the sampling interval. For the $x$ and $y$ variables, the appropriate range was to sample every 3.5 to 6.9 values from the simulated dataset described above. For $z$, the range was 1.9 to 3.8, although the latter limit could be relaxed to 7.6. This validates the choice mentioned above of selecting every fourth value from the dataset as a good compromise between identification and estimation.

A dataset for model fitting was selected from points 501 through 1000, and another for independent validation from points 1001 to 2000. Cross-validation was not used; thus the fit and test datasets are the same. Figure 1(a) displays the data from the fitting set in the time domain. Only $x$ and $z$ are shown in figure 1 (lower and upper curves, respectively); the trace of $y$ is similar to that of $x$ and is omitted for clarity. Figure 2(a) shows the familiar view of the strange attractor for this system viewed in the $xy$ phase plane.

A second pair of datasets was produced from the first by adding measurement noise to the values of $x$, $y$ and $z$. Lillekjendlie *et al.* [21] have indicated that 'too little work has been done on noisy chaotic time series'. The noise was drawn from a uniform distribution ranging from $-5.0$ to 5.0 (variance = 8.33). The ratios of the variance of the 'clean' (no noise) data to the noisy data were 0.885, 0.933 and 0.925 for $x$, $y$ and $z$,

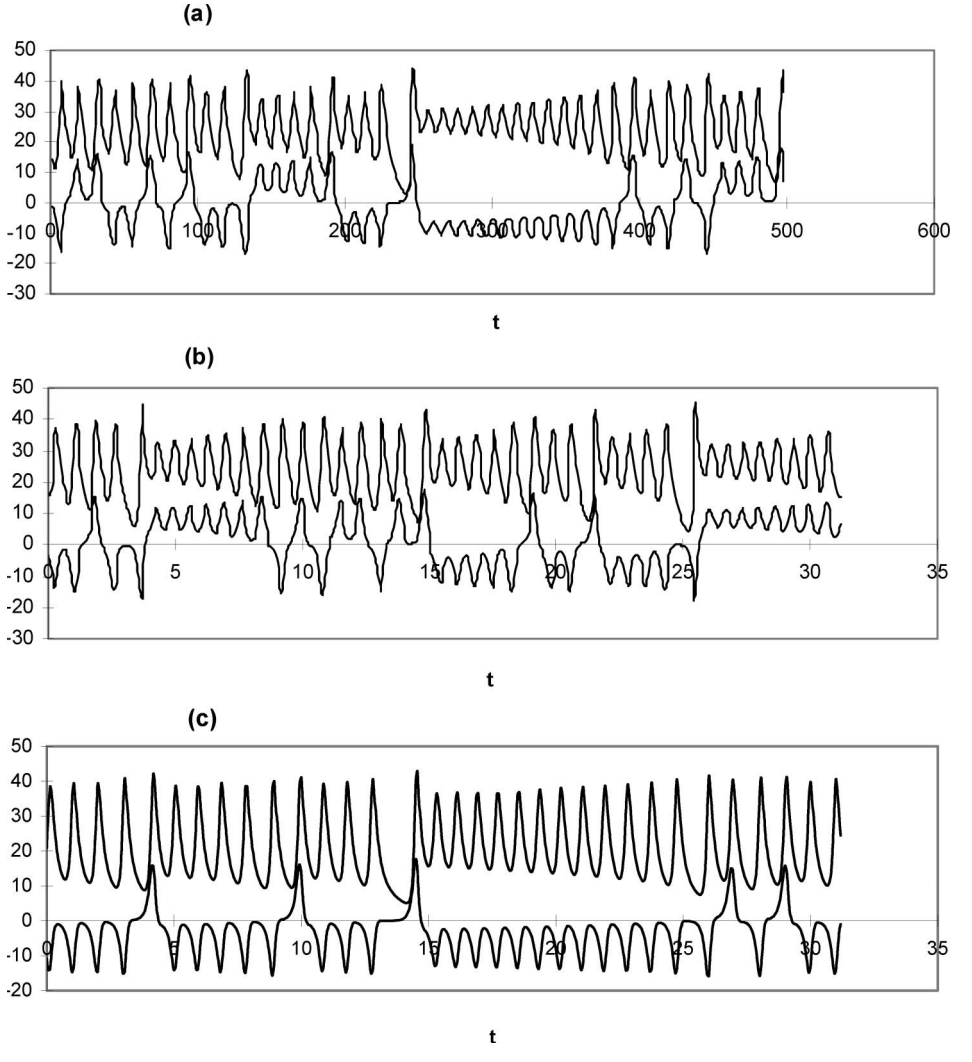


Figure 1. Time domain plot of Lorenz system; upper curve in each graph is $z$, lower curve is $x$; $y$ omitted for clarity; (a) results from differential equations; (b) results from MPR model of zero noise data; (c) results from MPR model of noise data with two lags.

respectively. This places a lower limit of 0.115 on $D^2$ for the prediction of $x$. Although the statistical tests of significance used in model structure selection are based on an assumption of normal error distribution, they are robust against violations of this assumption [22].

The fitting programme was used to generate multivariate polynomial models for the $x$, $y$ and $z$ values for both the clean and the noisy datasets. The values of $x$, $y$ and $z$ were associated with $x[i, j]$ with $i = 1$, 2 and 3, respectively. First order differences were used as dependent variables. For $x[i, j]$ the first order difference is defined as

$$\Delta x[i, j] = x[i, j] - x[i, j-1]. \tag{13}$$

In the case of the noisy datasets, the differencing was performed after the noise was added to the original data. A MPR model was then developed to predict the

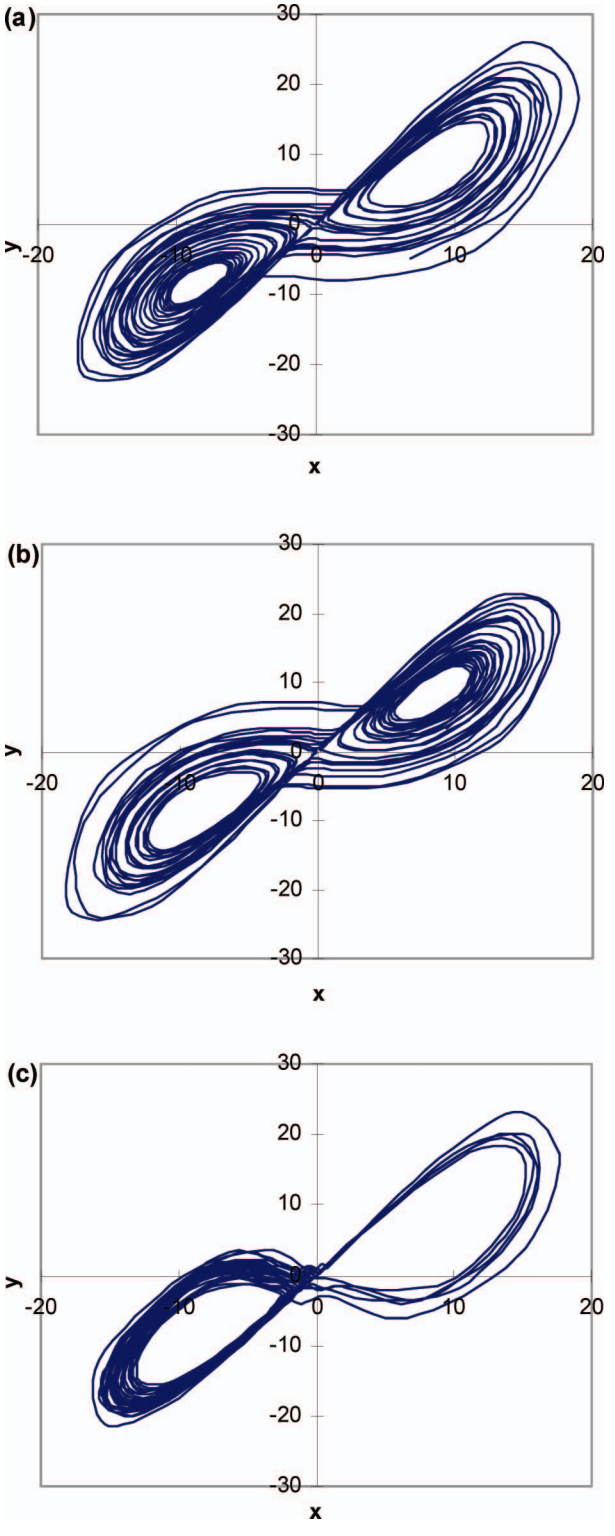


Figure 2. X – Y phase plot of Lorenz attractor; (a) results from differential equations; (b) results from MPR model of zero noise data; (c) results from MPR model of noise data with two lags.

differenced dependent variable based on the undifferenced independent variables. The fitting parameters for all fits were: candidate multiplicands were {1, 2} ($n_m = 2$); candidate exponents were {1, 2, ..., 9}; and the set of lags varied as described below. The range of exponents was selected to cover the order of accuracy in the calculations used to generate the time series. The resulting models were then converted to undifferenced form before being used for predictions or Lyapunov exponent calculation.

In the case of the noise-free data, the candidate lag set was {1}. That is, current values of $x$, $y$, and $z$ were used to predict the difference to the next value. Tests with additional lags were found not to improve the fit for this data set. Table 1 shows the differenced form of the model structure for predicting $x$, together with the $t$-statistic for the coefficient in the last column. The Table shows that the resulting model has 14 of the 271 possible terms. Written in the familiar form, and showing only the first several terms and omitting indices for clarity, this model is:

$$\Delta x = 3.4070 \cdot 10^{-7} \cdot xz^5 + 4.7659 \cdot 10^{-1} \cdot y - 3.4949 \cdot 10^{-3} \cdot xz^2 - 1.5647 \cdot 10^{-4} \cdot y^3 + \cdots$$

The values of the $t$-statistics indicate that all of these terms contribute to the model with a high degree of confidence. Table 1, together with similar models for $y$ and $z$, constitute a global model for generating the time series. Note that seven of these terms form the series $xz^i$, $i = 1, 2, \ldots, 7$.

The model in table 1 produced a very accurate fit to the independent test data. The value of $D^2$ for one-step-ahead prediction approaches the limit of the discretization error, as indicated in table 2 for the no-noise MPR model. The model showed no tendency to explode when iterated. Figures 1b and 2b show the time domain and phase-plane plots of the iterated MPR model, respectively. The similarity in appearance to the original data can be seen. The Lyapunov exponents ($\lambda$) found using the differential equations and the MPR models for the Lorenz system with no noise are also shown in table 2. The result shows that the MPR model gives an accurate estimate of the Lyapunov exponent using only the time series. It is notable that the method was even able to calculate the negative Lyapunov exponent accurately, which is considered a difficult feat [27].

Table 1. MPR model to predict $\Delta x$ from current values of $x$, $y$, and $z$.

| $P$ | $k_{p,1}$ | $l_{p,1}$ | $b_{p,1}$ | $k_{p,2}$ | $l_{p,2}$ | $b_{p,2}$ | $a_p$ | $t_p$ |
|---|---|---|---|---|---|---|---|---|
| 1 | 1 | 1 | 1.0 | 3 | 1 | 5.0 | 3.4070e−07 | 22.42 |
| 2 | 2 | 1 | 1.0 | 0 | 1 | 0.0 | 4.7659e−01 | 959.78 |
| 3 | 1 | 1 | 1.0 | 3 | 1 | 2.0 | −3.4949e−03 | −73.10 |
| 4 | 2 | 1 | 3.0 | 0 | 1 | 0.0 | −1.5647e−04 | −138.17 |
| 5 | 1 | 1 | 1.0 | 3 | 1 | 3.0 | 2.9533e−04 | 39.88 |
| 6 | 1 | 1 | 3.0 | 3 | 1 | 2.0 | −2.9164e−07 | −25.36 |
| 7 | 1 | 1 | 1.0 | 3 | 1 | 4.0 | −1.3313e−05 | −28.23 |
| 8 | 1 | 1 | 4.0 | 2 | 1 | 1.0 | 7.1386e−07 | 62.29 |
| 9 | 1 | 1 | 1.0 | 3 | 1 | 6.0 | −4.5779e−09 | −18.69 |
| 10 | 2 | 1 | 1.0 | 3 | 1 | 2.0 | −1.9111e−05 | −28.98 |
| 11 | 1 | 1 | 1.0 | 3 | 1 | 7.0 | 2.5209e−11 | 16.11 |
| 12 | 1 | 1 | 3.0 | 3 | 1 | 1.0 | −4.0800e−06 | −11.27 |
| 13 | 3 | 1 | 8.0 | 0 | 1 | 0.0 | −1.7893e−15 | −5.75 |
| 14 | 3 | 1 | 3.0 | 0 | 1 | 0.0 | 1.1256e−07 | 4.64 |

Table 2. Accuracy and Lyapunov exponents for fitted models.

| | $D^2$ | $\lambda_1$ | $\lambda_2$ | $\lambda_3$ |
|---|---|---|---|---|
| Differential equations | | 1.30 | −0.002 | −20.7 |
| MPR, no noise, lag = {1} | 8.7e−6 | 1.33 | −0.065 | −21.3 |
| MPR, with noise, lag = {1} | 0.74 | −0.46 | −0.460 | −88.7 |
| MPR, with noise, lag = {1, 2}* | 0.46 | 0.97 | −0.300 | −17.7 |
| MPR, with noise, lag = {1, 2, 3}* | 0.55 | 1.09 | −0.460 | −18.0 |

*The three largest Lyapunov exponents are shown.

When the MPR model with a single lag was fit to the noisy data the $D^2$ for predicting $x$ was a poor 0.74. More importantly, the behaviour of the iterated model differed significantly from the original system. All starting points converged to the lower fixed point of the attractor:

$$x = -7.684;\ y = -8.227;\ z = 25.872.$$

The Lyapunov exponents for this model shown in table 2, therefore, represent a local value at the stable fixed point, and differ greatly from the expected result. The original system has an unstable fixed point at $x = -8.485$; $y = -8.485$; $z = 27.00$. All points in the 'upper loop', where $x$ and $y$ are positive, escaped to the lower loop without making a complete cycle around the upper fixed point. There seems to be an unstable fixed point in the polynomial attractor at about $x = 17.6$, $y = 4.8$, $z = 42.5$.

A significant improvement was obtained by using lag set {1, 2}. In this case the next values of $x$, $y$, and $z$ were computed using the two previous sets of values. The prediction improved to $D^2 = 0.46$. The positive Lyapunov exponent reappears (table 2), and the attractor geometry of the iterated model appears similar to that of the original system (figure 2(c)).

Although these measures indicate reasonable accuracy for the model, graphical examination of the model behaviour as shown in figures 1(c) and 2(c) shows significant differences from the data. Whenever $x$ and $y$ crossed to the upper loop of the attractor it typically did not execute more than a single orbit before returning to the lower loop. The orbits stay away from the fixed points at the centres of the loops. Multiple orbits of the upper loop occur when the trajectory enters that loop sufficiently close to the fixed point at the centre. If the orbit enters near the outside of the loop, it will soon return to the lower loop. Examination of the original data used in the fitting shows that of the eight crossings from the lower loop to the upper, only three cases result in more than one orbit there, and one of these has only two orbits. It may be that the behaviour of the model of the noisy data is a result of the limited size of the data set which does not present enough examples of the complete Lorenz system behaviour.

The first and third Lyapunov exponents are reasonable estimates of those of the underlying model. However, the second appears to be significantly less than zero. This would change the type of attractor from a strange (chaotic) one to that of a two-torus [6]. However, if these data were being examined without prior knowledge of the underlying behaviour, the presence of the high level of noise should admit an increased uncertainty in the values of the Lyapunov exponents, and therefore the possibility that the second exponent is not different from zero. It should be noted that in the cases of lag sets {1} and {1, 2} $\lambda_2$ was estimated more accurately than $\lambda_1$ and $\lambda_3$. That is, looking at each model in turn, the absolute difference between $\lambda_2$ estimated using the noisy data

and the value for the ODEs is less than the same differences for $\lambda_1$ and $\lambda_3$. However, the amount of error would not be known *a priori* for a real time series. This suggests a need to understand the effect of the level of noise in a system upon the errors for estimated Lyapunov exponents.

The accuracy for the model using lags {1, 2, 3} was reduced somewhat: $D^2 = 0.55$. The fact that accuracy was reduced by the incorporation of another lag into the model suggests overfitting. Instead of contributing more information, the additional lag adds spurious components to the solution which only increase error and increase the potential for undesirable predicted behaviours.

## 4. Mackey – Glass equation

The MPR identification procedure was also tested using the Mackey – Glass delay-differential equation:

$$\frac{dx}{dt} = -0.1x + \frac{0.2x(t-\Delta)}{1 + x(t-\Delta)^{10}}. \tag{14}$$

This equation, which is known to produce chaotic behaviour for certain parameter values, is another challenge for identification methods.

When simulated numerically, equation (14) produces a univariate time-series. Two sets of data were created by solving the equation numerically to form clean datasets using fourth order Runge – Kutta integration with $\Delta t = 0.125$, and sampled every six time units. One set of data was created with $\Delta = 30$ to compare with the performance of predictions made by Casdagli [9]. Casdagli used 500 datapoints in the fitting procedure. In this work, cross-validation was used with 500 datapoints for the fit dataset and and additional 500 points for the test dataset. The global model statistics reported herein are those computed using the test dataset.

Another clean dataset was created with $\Delta = 31.8$, $N = 983$, to compare Lyapunov coefficients with those computed by Wolf *et al.* [6]. Noisy datasets were created by adding random numbers drawn from a uniform distribution with range $\pm$ 0.158. This corresponds to a noise level of 10%, based on the variance.

In order to model the univariate time-series, it is necessary to determine the number of lags required. This is equivalent to the embedding dimension, $n_M$. The procedure used for the Lorenz equations will serve here: Incrementally increase the number of lags until no improvement in prediction accuracy can be obtained. That is, using a lag set $= \{1, \ldots, n_l\}$. The MSE is computed for each model, and the embedding dimension is estimated to be the highest lag that produces a significant improvement in MSE over the model having one less lag. This approach was developed by Casdagli [9]. Here, Casdagli's procedure is modified by using the *F*-ratio (equation (11)) to test for significance of improvement as the embedding dimension is incremented.

The embedding dimension for the Mackey – Glass equation is known to be 6 [9]. As shown in table 3, the performance of the MPR models with the $\Delta = 30$ data improves for each increment of $n_l$ up to 6. However, the probability that the *MSE* has not improved with increment of embedding dimension, as computed using the *F*-statistic, increases abruptly from negligible values at $n_l = 6$ or less to 0.318 at $n_l = 7$. This confirms an embedding dimension of 6 for this Mackey – Glass equation, validating the use of MPR modelling for determining $n_M$.

Thus, the $n_l = 6$ model should be used for prediction and analysis. The prediction performance of this model is compared with the models developed by Casdagli [9] in table 4. By comparison with the results in table 3 the MPR model can be seen to be a highly significant improvement over any of the models tested by Casdagli, including artificial neural networks and polynomial models.

A similar model was developed using the second dataset with model parameters $\Delta = 31.8$ and sampling interval = 6.0 seconds, both with and without noise at the 10% of variance level. The Lyapunov coefficients were then computed as described above. Table 5 compares the results. The largest value for the clean data is about 50% high, the second is acceptably close. The errors in all three coefficients are much smaller than was obtained with the Lorenz equations, but the small values for the Mackey–Glass Lyapunov coefficients amplifies the relative error. However, the method is successful at finding the two positive values.

The limited accuracy may also be due to the quantity of data used for the fitting. Wolf *et al.* [6] estimated the number of datapoints necessary for accurately computing the Lyapunov exponents. In the case of the Mackey–Glass equation, a range of 4000 to 200 000 is given. In the work described here, about 500 points were used. Thus the

Table 3. Performance of MPR models using various embedding dimensions for Mackey–Glass equation with no noise.

| $n_l$ | $\log_{10}D^2$ | $F$ ratio | $p(F)$ |
|---|---|---|---|
| 1 | −0.489 | | |
| 2 | −0.732 | 1.744 | 3.8E−10 |
| 3 | −0.811 | 1.195 | 2.4E−02 |
| 4 | −1.046 | 1.706 | 2.3E−09 |
| 5 | −1.689 | 4.376 | 7.0E−55 |
| 6 | −2.187 | 3.129 | 2.7E−34 |
| 7 | −2.207 | 1.044 | 0.318 |

Table 4. Performance of models developed by Casdagli [9] for predicting the Mackey–Glass equation ($\Delta = 30$).

| Model | $\log_{10}D^2$ |
|---|---|
| MPR ($n_l = 6$) | −2.187 |
| Polynomial | −1.400 |
| Rational | −1.330 |
| Local ($d = 1$) | −1.240 |
| Local ($d = 2$) | −1.420 |
| Radial | −1.600 |
| Artificial neural networks | −1.500 |

Table 5. Lyapunov coefficients for Mackey–Glass equation ($\Delta = 31.8$).

| | $\lambda_1$ | $\lambda_2$ | $\lambda_3$ |
|---|---|---|---|
| Wolf *et al.*'s results | 0.00630 | 0.00262 | $< 8.0 \cdot 10^{-6}$ |
| MPR, no noise, $n_l = 6$ | 0.00999 | 0.00253 | −0.00195 |
| MPR, with noise, $n_l = 12$ | 0.00555 | −0.00153 | −0.01143 |

results obtained might be considered fairly good in light of small amount of data used in computing them.

In the case of the noisy data, it was necessary to include 12 lags to obtain sufficient accuracy. But even so, the method incorrectly computed the sign of the second Lyapunov coefficient. Nevertheless, in all cases the error is within the limit of the error observed in the case of the Lorenz equations. The presence of noise may further increase the amount of data needed to make an accurate computation. The data needed to compute the embedding dimension were not collected for the case of the Mackey–Glass equation with noise added.

## 5. Conclusion

Multivariate polynomial regression facilitates identification of nonlinear processes from time series data. The method was able to detect behaviour underlying noisy data, and thus acts as a filter. It generates global models that are simple and parsimonious making the resulting models easy to analyse. For example, the models were used for computation of Lyapunov exponents and embedding dimension. The method simplifies these calculations, which can be used to detect chaos in time series. By restricting the models to include only terms that are statistically significant, instability upon iteration is avoided and accuracy is improved. The method was also found to produce significantly greater accuracy in producing predictive models of the Mackey–Glass equation than a variety of other methods, including artificial neural networks, radial basis function models, and local models. The method was shown to be able to convert a flow to a map, as was done here for the Lorenz and Mackey–Glass system. The positive results of these investigations may encourage use of the method beyond the process control applications that they have been limited to, but also for nonlinear correlation and other prediction applications.

## Acknowledgements

Laura D'Ambrosa performed initial calculations that led to this work. The comments of the anonymous reviewers made this a greatly improved paper.

## References

[1] Box, G.E.P. and Jenkins, G.M., 1976, *Time Series Analysis, Forecasting and Control* (San Francisco, CA: Holden-Day).
[2] Tang, Z., de Almeida, C. and Fishwick, P.A., 1991, Time series forecasting using neural networks vs. Box-Jenkins methodology. *Simulation*, **57**(5), 303–310.
[3] Farmer, J.D. and Sidorowich, J.J., 1987, Predicting chaotic time series. *Phys. Rev. Lett.*, **62**, 845–848.
[4] Sugihara, G. and May, R.M., 1990, Nonlinear forecasting as a way of distinguishing chaos from measurement error in time series. *Nature*, **344**, 19 April, 734–741.
[5] Garfinkel, A., Spano, M.L., Ditto, W.L. and Weiss, J.N., 1992, Controlling cardiac chaos. *Science*, **257**, 1230–1235.
[6] Wolf, A., Swift, J.B., Swinney, H.L. and Vastano, J.A., 1985, Determining Lyapunov exponents from a time series. *Physica 16-D*, 235–317.
[7] Brown, R., Bryant, P. and Abarbanel, H.D.I., 1991, Computing the Lyapunov spectrum of a dynamical system from an observed time series. *Phys. Rev. A*, **43**(6), 2787–2806.
[8] Briggs, K., 1990, An improved method for estimating Liapunov exponents of chaotic time series. *Physics Letters A*, **151**(1), 27–32.

[9] Casdagli, M., 1989, Nonlinear prediction of chaotic time series. *Physica D*, **35**, 335–356.
[10] Billings, S.A. and Aguirre, L.A., 1995, Effects of the sampling time on the dynamics and identification of nonlinear model. *Int. J. Bifurc. & Chaos*, **5**(6), 1541–1556.
[11] Guo, L.Z. and Billings, S.A., 2005, A comparison of polynomial and wavelet expansions for the identification of chaotic coupled map lattices. *Int. J. Bifurc. & Chaos*, **15**(9), 2927–2938.
[12] Chen, S. and Billings, S.A., 1989, Representations of nonlinear systems: The NARMAX model. *Int. J. Control*, **49**(3), 1013–1032.
[13] Bard, Y. and Lapidus, L., 1970, Nonlinear system identification. *Ind. Eng. Chem. Fundam.*, **9**(4), 628–633.
[14] Wu, X. and Çinar, A., 1993, An automated knowledge-based system for nonlinear system identification. *Gensym User Society Meeting*, Cambridge, MA, 26–28 May.
[15] Draper, N.R. and Smith, H., 1966, *Applied Regression Analysis* (New York: John Wiley).
[16] Billings, S.A. and Voon, W.S.F., 1986, A prediction-error and stepwise-regression estimation algorithm for nonlinear-systems. *Int. J. Cont.*, **44**, 803–822.
[17] Piroddi, L, and Spinelli, W., 2003, Structure selection for polynomial NARX models based on simulation error minimization. In: P. van der Hof, B. Wahlberg and S. Weiland (Eds), *Proceedings of the 13th IFAC symposium on system identification*, Rotterdam, The Netherlands, pp. 371–376.
[18] Press, W.H., Flannery, B.P., Teukolsky, S.A. and Vetterling, W.T., 1989, *Numerical Recipes* (Cambridge, UK: Cambridge University Press).
[19] Billings, S.A., Chen, S. and Korenberg, M.J., 1989, Identification of MIMO nonlinear systems using a forward-regression orthogonal estimator. *Int. J. Cont.*, **49**, 2157–2189.
[20] Granger, C.W.J. and Terasvirta, T., 1993, *Modelling Nonlinear Economic Relationships* (New York: Oxford University Press).
[21] Lillekjendlie, B., Kugiumtzis, D. and Christophersen, N., 1994, Chaotic time series. 2. System identification and prediction. *Model. Ident. & Cont.*, **15**(4), 225–243.
[22] Lind, I. and Ljung, L., 2005, Regressor selection with the analysis of variance method. *Automatica*, **41**(4), 693–700.
[23] Vaccari, D.A. and Christodoulatos, C., 1992, Generalized multiple regression with interaction and nonlinearity for system identification in biological treatment processes. *Trans. Instr. Soc. Am.*, **31**(1), 97–102.
[24] TaylorFit Software, Simetrica, LLC, 2004. Available online at: http://simetrica-LLC.com (accessed day month year).
[25] Wolf, A., 1986, Quantifying chaos with Lyapunov exponents. In: A.V. Holden (Ed.), *Chaos* (Princeton, NJ: Princeton University Press).
[26] Lorenz, E.N., 1963, Deterministic nonperiodic flow. *J. Atmos. Sci.*, **20**, 130–141.
[27] Wolf, A., 1992, Personal communication.