Monograph:
Chen, S. and Billings, S.A. (1987) A Prediction-Error Estimation Algorithm for Nonlinear Output-Affine Systems. Research Report. Acse Report 314 . Dept of Automatic Control and System Engineering. University of Sheffield

A PREDICTION-ERROR ESTIMATION ALGORITHM FOR NONLINEAR OUTPUT-AFFINE SYSTEMS

S. Chen and S.A. Billings

Department of Control Engineering

University of Sheffield

Mappin Street

Sheffield S1 3JD

Research Report No.314

April 1987

A PREDICTION-ERROR ESTIMATION ALGORITHM FOR NONLINEAR OUTPUT-AFFINE SYSTEMS

Abstract

A prediction-error estimation algorithm is developed for nonlinear discrete-time systems which can be represented by the output-affine difference equation model. The theory of hypothesis testing is employed to select a model with the correct structure. Problems relating to the use of output-affine models in the identification of nonlinear systems are discussed, and a comparison with the NARMAX (Nonlinear ARMAX) model is given.

1. Introduction

The choice of model that is used to represent a nonlinear system is vitally important since this will influence its usefulness in prediction and control. The results of Sontag (1979) on polynomial response maps has produced the nonlinear output-affine difference equation model which is valid globally.

The internal behaviours of a wide range of nonlinear systems can be described by state-affine models (Sontag, 1979). A technique has been developed which involves patching a series of linear signal dependent state-space models together to yield an approximate nonlinear state-affine model (Cyrot-Normand and Dang Van Mien, 1980). The technique is efficient in computing time and can be easily implemented on a microprocessor. However, the final model obtained may be input sensitive and may only be valid for relatively slow moving inputs and levels of plant operation.

Previous research has revealed several difficulties in using the linear least squares method to construct an output-affine model from input-output data (Billings, Korenberg and Chen, 1987). In the present paper, a prediction-error estimation algorithm (Goodwin and Payne, 1977) is derived for the output-affine model based on input-output data measurements. The log determinant ratio test (Leontaritis and Billings, 1987) is employed to determine the correct model structure. Stepwise backward elimination of parameters (Draper and Smith, 1981) is integrated with the prediction-error algorithm to provide an efficient procedure for fitting parsimonious output-affine models to nonlinear systems. The present study is an extension of previous results (Leontaritis and Billings, 1986) to nonlinear output-affine systems. Model validation based on correlation tests (Billings and Voon, 1986a) is briefly described. A simulation study is included to illustrate the algorithm. Finally, it is shown that, although the output-affine model is valid globally, there are disadvantages in using it as a basis for the identification of nonlinear systems compared with using a NARMAX (Nonlinear ARMAX) model (Leontaritis and Billings, 1985).

For simplicity, only the single-input, single-output case is considered throughout the discussion. However, the results are readily valid for multi-input, multi-output systems.

2. Output-affine model

The realization of discrete-time polynomial input-output maps or response functions has been studied in great detail (Sontag, 1979). It was proved in Sontag's work (1979) that the response function of a deterministic nonlinear system is finitely realizable and bounded iff it satisfies the affine difference equation

$$\sum_{i=0}^{r} a_i(u(t-1),...,u(t-r))\hat{y}(t-i) = a_{r+1}(u(t-1),...,u(t-r)) \tag{1}$$

where u(t) and $\hat{y}(t)$ are the input and output of the deterministic system at time t respectively, a_i, i=0,1,...,r+1 are polynomials and r is the order of the system. The output-affine difference equation (1) is an input-output model valid globally.

It is more appropriate to define a model in the stochastic environment if the model is to be used as a basis for the development of identification and digital controller design techniques. Let the input and output of a general stochastic discrete-time system at time t be u(t) and y(t) respectively. The observation of the system is assumed to start from time 1. Denote

$$\begin{aligned} y^t &= (y(1),...,y(t))^\tau \\ u^t &= (u(1),...,u(t))^\tau \end{aligned} \tag{2}$$

The system can then be described by the conditional probability density function of y(t) given all the past inputs and outputs u^t and y^{t-1}

$$p(y(t)|y^{t-1},u^t) \tag{3}$$

The prediction of the output at time t is given by

$$\hat{y}(t) = E[y(t)|y^{t-1},u^t] = f(y^{t-1},u^t) \tag{4}$$

The prediction error or innovation e(t) is defined as

$$e(t) = y(t)-\hat{y}(t) = y(t)-f(y^{t-1},u^t) \tag{5}$$

That is, the output is separated as two parts. The part of the output that can be predicted from the past is given by a deterministic function $f(y^{t-1},u^t)$ and the unpredictable part is defined as the innovation e(t). Let

$$e^t = (e(1),...,e(t))^\tau \tag{6}$$

The elements of the vector e^{t-1} can be calculated from the vectors y^{t-1} and u^{t-1} using eqn. (5) recursively. Similarly the vector y^{t-1} can be evaluated from e^{t-1} and u^{t-1}. Therefore, knowing (e^{t-1},u^t) is equivalent to knowing (y^{t-1},u^t) and

$$p(y(t)|y^{t-1},u^t) = p(y(t)|e^{t-1},u^t) \tag{7}$$

The prediction $\hat{y}(t)$ can thus alternatively be given by

$$\hat{y}(t) = E[y(t)|e^{t-1},u^t] = f^*(e^{t-1},u^t) \tag{8}$$

f^* can be considered as the response function of a deterministic system where the input is $(u(t),e(t))^\tau$ and the output is $\hat{y}(t)$. As the consequence of Sontag's results, the following input-output model describes the system

$$\sum_{i=0}^{r} a_i(u(t-1),...,u(t-r),e(t-1),...,e(t-r))\hat{y}(t-i) = a_{r+1}(u(t-1),...,u(t-r),e(t-1),...,e(t-r)) \tag{9}$$

or

$$\sum_{i=0}^{r} a_i(u(t-1),...,u(t-r),e(t-1),...,e(t-r))y(t-i) = a_{r+1}(u(t-1),...,u(t-r),e(t-1),...,e(t-r)) + \sum_{i=0}^{r} a_i(u(t-1),...,u(t-r),e(t-1),...,e(t-r))e(t-i) \tag{10}$$

A special case of eqn. (10) occurs when the noise enters the system additively at the output

$$y(t) = \hat{y}(t) + e(t) \tag{11}$$

where $\hat{y}(t)$ satisfies eqn. (1). In this case, the input-output description of the system can be simplified to

$$\sum_{i=0}^{r} a_i(u(t-1),...,u(t-r))y(t-i) = a_{r+1}(u(t-1),...,u(t-r)) + \sum_{i=0}^{r} a_i(u(t-1),...,u(t-r))e(t-i) \tag{12}$$

The model that is actually used for the development of the prediction-error estimation algorithm in the following sections is a slight generalization of eqn. (12)

$$y(t) = \frac{1}{a_0(u(t-1),...,u(t-r))}\{\sum_{i=1}^{r} a_i(u(t-1),...,u(t-r))y(t-i) + a_{r+1}(u(t-1),...,u(t-r)) + \sum_{i=1}^{r} a_{r+1+i}(u(t-1),...,u(t-r))e(t-i)\} + e(t) \tag{13}$$

However, the results can easily be extended to the more general case of eqn. (10).

Assume that each $a_i[.]$, i=0,1,...,2r+1 is expanded as an L degree polynomial and is parametrized accordingly. Then for a given value of the parameter vector θ, the residual is given as

$$\varepsilon(t,\theta) = y(t) - \frac{1}{a_0(u(t-1),...,u(t-r))}\{\sum_{i=1}^{r} a_i(u(t-1),...,u(t-r))y(t-i) + a_{r+1}(u(t-1),...,u(t-r)) + \sum_{i=1}^{r} a_{r+1+i}(u(t-1),...,u(t-r))\varepsilon(t-i,\theta)\} \tag{14}$$

For example, consider a first order system (r=1) with a linear expansion (L=1) of $a_i[.]$, i=0,1,2,3

$$\varepsilon(t,\theta) = y(t) - \frac{1}{\theta_1+\theta_2 u(t-1)}\{(\theta_3+\theta_4 u(t-1))y(t-1)+(\theta_5+\theta_6 u(t-1))+(\theta_7+\theta_8 u(t-1))\varepsilon(t-1,\theta)\}$$

The dimension of the parameter vector θ is

$$n_\theta = (2r+2)n \tag{15}$$

where

$$n = \sum_{i=0}^{L} n_i,\ n_0=1,\ n_i = n_{i-1}(r+i-1)/i,\ i=1,...,L \tag{16}$$

3. Prediction-error estimation

The prediction-error estimation method produces an estimate of the parameter vector θ by minimizing a loss function (Goodwin and Payne, 1977). The asymptotic properties of the method are very similar to those of the maximum likelihood estimator which can be shown to produce consistent, asymptotically normally distributed and asymptotically efficient estimates. The covariance matrix of the maximum likelihood estimator reaches the Cramer-Rao bound asymptotically. In order to apply the maximum likelihood method, however, the probability density function of the innovations must be known. The prediction-error method, on the other hand, does not require any knowledge of the distribution of the innovations and is equivalent to the maximum likelihood method for Gaussian innovations. It can be shown that the performance of the prediction-error method is only slightly inferior to the maximum likelihood method for bell-shaped density functions of the innovations.

Consider the following loss function

$$J(\theta) = \frac{1}{2}\log \det Q(\theta) = \frac{1}{2}\log Q(\theta) \tag{17}$$

in the scalar case where

$$Q(\theta) = \frac{1}{N}\sum_{t=1}^{N}\varepsilon^2(t,\theta) \tag{18}$$

N is the data length and the residual $\varepsilon(t,\theta)$ is given in (14). The gradient and the approximate Hessian of $J(\theta)$ are given by (Goodwin and Payne, 1977)

$$\frac{\partial J}{\partial \theta_i} = \frac{1}{NQ(\theta)}\sum_{t=1}^{N}\varepsilon(t,\theta)\frac{\partial \varepsilon(t,\theta)}{\partial \theta_i} \quad i=1,...,n_\theta \tag{19}$$

$$\frac{\partial^2 J}{\partial \theta_i \partial \theta_j} = \frac{1}{NQ(\theta)}\sum_{t=1}^{N}\frac{\partial \varepsilon(t,\theta)}{\partial \theta_i}\frac{\partial \varepsilon(t,\theta)}{\partial \theta_j} \quad i,j=1,...n_\theta \tag{20}$$

The derivatives of the residuals can be obtained by differentiating (14). Define

$$\bar{i} = \underline{[}\frac{i-1}{n}\underline{]},\ i\geq n+1;\quad \tilde{i} = \underline{[}\frac{i-(r+1)n-1}{n}\underline{]},\ i\geq (r+2)n+1 \tag{21}$$

where $\underline{[}\ -\ \underline{]}$ denotes the integer part of the result of the division. Then for

$1 \le i \le n$:
$$\frac{\partial \varepsilon(t,\theta)}{\partial \theta_i} = \frac{1}{a_0^2[.]}\{\sum_{j=1}^{r} a_j[.]y(t-j)+a_{r+1}[.]+\sum_{j=1}^{r} a_{r+1+j}[.]\varepsilon(t-j,\theta)\}\frac{\partial a_0[.]}{\partial \theta_i} - \frac{1}{a_0[.]}\{\sum_{j=1}^{r} a_{r+1+j}[.]\frac{\partial \varepsilon(t-j,\theta)}{\partial \theta_i}\}$$

$n+1 \le i \le (r+1)n$:
$$\frac{\partial \varepsilon(t,\theta)}{\partial \theta_i} = -\frac{1}{a_0[.]}\{\frac{\partial a_{\bar{i}}[.]}{\partial \theta_i} y(t-\bar{i}) + \sum_{j=1}^{r} a_{r+1+j}[.]\frac{\partial \varepsilon(t-j,\theta)}{\partial \theta_i}\} \qquad (22)$$

$(r+1)n+1 \le i \le (r+2)n$:
$$\frac{\partial \varepsilon(t,\theta)}{\partial \theta_i} = -\frac{1}{a_0[.]}\{\frac{\partial a_{r+1}[.]}{\partial \theta_i} + \sum_{j=1}^{r} a_{r+1+j}[.]\frac{\partial \varepsilon(t-j,\theta)}{\partial \theta_i}\}$$

$(r+2)n+1 \le i \le (2r+2)n$:
$$\frac{\partial \varepsilon(t,\theta)}{\partial \theta_i} = -\frac{1}{a_0[.]}\{\frac{\partial a_{\bar{i}}[.]}{\partial \theta_i} \varepsilon(t-\bar{i},\theta) + \sum_{j=1}^{r} a_{r+1+j}[.]\frac{\partial \varepsilon(t-j,\theta)}{\partial \theta_i}\}$$

For the example given in Section 2

$$\frac{\partial \varepsilon(t,\theta)}{\partial \theta_1} = \frac{1}{(\theta_1+\theta_2 u(t-1))^2}\{(\theta_3+\theta_4 u(t-1))y(t-1)+(\theta_5+\theta_6 u(t-1))+(\theta_7+\theta_8 u(t-1))\varepsilon(t-1,\theta)\} - \frac{1}{\theta_1+\theta_2 u(t-1)}(\theta_7+\theta_8 u(t-1))\frac{\partial \varepsilon(t-1,\theta)}{\partial \theta_1}$$

$$\frac{\partial \varepsilon(t,\theta)}{\partial \theta_2} = \frac{1}{(\theta_1+\theta_2 u(t-1))^2}\{(\theta_3+\theta_4 u(t-1))y(t-1)+(\theta_5+\theta_6 u(t-1))+(\theta_7+\theta_8 u(t-1))\varepsilon(t-1,\theta)\}u(t-1) - \frac{1}{\theta_1+\theta_2 u(t-1)}(\theta_7+\theta_8 u(t-1))\frac{\partial \varepsilon(t-1,\theta)}{\partial \theta_2}$$

$$\frac{\partial \varepsilon(t,\theta)}{\partial \theta_3} = -\frac{1}{\theta_1+\theta_2 u(t-1)}\{y(t-1)+(\theta_7+\theta_8 u(t-1))\frac{\partial \varepsilon(t-1,\theta)}{\partial \theta_3}\}$$

$$\frac{\partial \varepsilon(t,\theta)}{\partial \theta_4} = -\frac{1}{\theta_1+\theta_2 u(t-1)}\{u(t-1)y(t-1)+(\theta_7+\theta_8 u(t-1))\frac{\partial \varepsilon(t-1,\theta)}{\partial \theta_4}\}$$

$$\frac{\partial \varepsilon(t,\theta)}{\partial \theta_5} = -\frac{1}{\theta_1+\theta_2 u(t-1)}\{1+(\theta_7+\theta_8 u(t-1))\frac{\partial \varepsilon(t-1,\theta)}{\partial \theta_5}\}$$

$$\frac{\partial \varepsilon(t,\theta)}{\partial \theta_6} = -\frac{1}{\theta_1+\theta_2 u(t-1)}\{u(t-1)+(\theta_7+\theta_8 u(t-1))\frac{\partial \varepsilon(t-1,\theta)}{\partial \theta_6}\}$$

$$\frac{\partial \varepsilon(t,\theta)}{\partial \theta_7} = -\frac{1}{\theta_1+\theta_2 u(t-1)}\{\varepsilon(t-1,\theta)+(\theta_7+\theta_8 u(t-1))\frac{\partial \varepsilon(t-1,\theta)}{\partial \theta_7}\}$$

$$\frac{\partial \varepsilon(t,\theta)}{\partial \theta_8} = -\frac{1}{\theta_1+\theta_2 u(t-1)}\{u(t-1)\varepsilon(t-1,\theta)+(\theta_7+\theta_8 u(t-1))\frac{\partial \varepsilon(t-1,\theta)}{\partial \theta_8}\}$$

The minimization of the loss function $J(\theta)$ can be performed very efficiently using Newton's method. The procedure consists of the following steps:

(a) Set k=0 and choose an initial value θ^0 of the parameter vector.

(b) Evaluate the gradient vector $\nabla_\theta J = \frac{\partial^\tau J}{\partial \theta}$ and the Hessian matrix $H = \frac{\partial^2 J}{\partial \theta^2}$ *at* θ^k.

(c) Calculate the direction vector $d^k = -H^{-1}\nabla_\theta J$.

(d) Perform a linear search to find the scalar μ^k such that

$$J(\theta^k+\mu^k d^k) = \min_{\mu} J(\theta^k+\mu d^k).$$

(e) Set $\theta^{k+1} = \theta^k + \mu^k d^k$.

(f) If $J(\theta^k)-J(\theta^{k+1})<$ a small tolerance, stop the algorithm. Otherwise, set k=k+1 and go to (b).

It is obvious from an inspection of eqn. (22) that the initial θ^0 cannot be chosen as the zero vector. An acceptable choice is to assume initially $a_0[.]=1$ and to use the least squares estimate as θ^0. The initial values for the derivatives of residuals are clearly zero since the residuals do not depend on the parameters for $t\leq 0$. The first r pairs of inputs and outputs can be used as the initial values of inputs and outputs. The initial values for residuals may be assumed to be zero. To avoid numerical difficulties, $a_0(u(t-1),...,u(t-r))$ should be replaced by a threshold value of the same sign if $|a_0[.]|$ is less than this threshold.

4. Model selection

Model selection methods for nonlinear systems based on the theory of hypothesis testing have been investigated by Leontaritis and Billings (1987). In this section, some of the results given in the above reference are employed to select models for output-affine systems.

There are two basic approaches to model selection. One is the model reduction method and the other is the model expansion method. In the model expansion approach, a basic model is initially chosen such that it contains parameters known to belong to the final model. The basic model is too simple to explain the data, and is improved by adding more parameters until the best model is found according to some statistical criterion. Since a knowledge of which terms exist in the data generating mechanism is not generally known a priori, a usual choice is to use the model with no parameters as the basic model. The application of the model expansion method to the output-affine model is involved. This is because the model eqn. (13) contains a denominator polynomial $a_0(u(t-1),...,u(t-r))$. A zero-parameter model cannot be used as a basic model. Therefore, only the model reduction method is discussed in the present study.

To initialise the model reduction procedure, the most complicated model must be selected. This model is referred to as the full model. All the other models to be considered are special cases of the full model with some of the parameters of the full model equal to zero or other constant values. For an output-affine system, the order r of the system and the degree L of polynomial expansion must be selected first in order to define the structure of the full model. Partition the parameter vector θ of a full model as

$$\theta = \begin{bmatrix} b \\ c \end{bmatrix} \tag{23}$$

where b is a vector of dimension $n_\theta - s$ and c is a vector of dimension s. The null hypothesis considered here is that the vector c is equal to a specific value c^*. The alternative hypothesis is that the vector c is unrestricted. The vector c^* is usually taken as a zero vector and the null hypothesis represents a reduced model with s parameters fewer than the full model. The statistic used to test the hypotheses is

$$d(y) = 2N\,J(\hat{\theta}_o) - 2N\,J(\hat{\theta}_1) = N \log Q(\hat{\theta}_o) - N \log Q(\hat{\theta}_1) = N \log \frac{Q(\hat{\theta}_o)}{Q(\hat{\theta}_1)} \tag{24}$$

where $\hat{\theta}_o$ and $\hat{\theta}_1$ are the restricted and unrestricted prediction-error estimates respectively. The test based on this statistic is referred to as the log determinant ratio test (Leontaritis and Billings, 1987) where the multi-input, multi-output case is studied, and (24) takes a general form

$$d(y) = N \log \frac{\det Q(\hat{\theta}_o)}{\det Q(\hat{\theta}_1)}$$

Under the null hypothesis, the statistic $d(y)$ is asymptotically distributed as a chi-squared distribution with s degrees of freedom.

$$d(y) \rightarrow \chi^2(s) \tag{25}$$

the values of $d(y)$ and of $\hat{\theta}_o$ can easily be computed (Leontaritis and Billings, 1987) using the formulae

$$d(y) = N(c^* - \hat{c}_1)^T [H_{cc} - H_{bc}^T H_{bb}^{-1} H_{bc}](c^* - \hat{c}_1) \tag{26}$$

$$\hat{b}_o = \hat{b}_1 - H_{bb}^{-1} H_{bc}(c^* - \hat{c}_1) \tag{27}$$

where H is the Hessian of $J(\theta)$ at the minimum $\hat{\theta}_1$, and is partitioned as

$$H = \begin{bmatrix} H_{bb} & H_{bc} \\ H_{bc}^T & H_{cc} \end{bmatrix} \tag{28}$$

$\hat{c}_1$ is the unrestricted prediction-error estimate of the assumed true value c^* and $\hat{b}_o$ is the restricted prediction-error estimate of b. The null hypothesis is accepted if

$$d(y) = N \log \frac{Q(\hat{\theta}_o)}{Q(\hat{\theta}_1)} < k_\alpha(s) \tag{29}$$

where $k_\alpha(s)$ is the critical value of the chi-squared distribution with s degrees of freedom and a given significant level α. If $d(y)$ is greater than $k_\alpha(s)$ there is strong evidence against the null hypothesis and it is thus rejected.

The model selection has to be carried out on the full model and all its reduced variants. This is a multiple selection problem. The requirement of non-conflicting pairwise comparisons leads to a C-criterion (Leontaritis and Billings, 1987) which the model selected from all the competing models must minimize

$$C = N \log Q(\bar{\theta}) + \bar{n}_\theta\, k_\alpha(1) \tag{30}$$

where $\bar{\theta}$ is the parameter vector of the particular model and $\bar{n}_\theta$ its dimension. The critical value $k_\alpha(1)$ must now be specified. The choice $k_\alpha(1) = 2$ is known as Akaike's information criterion (AIC)

$$AIC = N \log Q(\bar{\theta}) + 2\bar{n}_\theta \tag{31}$$

However, it is well-known that AIC may consistently overestimate the true parameter vector and a more reasonable choice of $k_\alpha(1)$ has been shown to be 4 (Leontaritis and Billings, 1987)

$$C = N \log Q(\bar{\theta}) + 4\bar{n}_\theta \tag{32}$$

Consider all the reduced models with s elements of the full parameter vector θ equal to zero. The number of all such models is

$$C(n_\theta,s) = \frac{n_\theta!}{s!(n_\theta-s)!}, \quad 0 \le s \le n_\theta \tag{33}$$

The maximun number of all the competing models therefore is

$$\sum_{s=0}^{n_\theta} C(n_\theta,s) = 2^{n_\theta} \tag{34}$$

The optimal model that minimizes (32) must be selected from all the 2^{n_θ} models. Such a method of choosing the best model is known as the combinatorial method. It is apparent that for a moderate n_θ the computation may already become excessive even though the C-criterion can be computed extremely efficiently using eqn. (26). In an attempt to reduce the computational burden, the stepwise backward elimination (SBE) method (Draper and Smith, 1981) is employed.

The SBE algorithm initially calculates the C-criterion for the full model (s=0). It then considers the class of all the models with one parameter fewer than the full model. If the minimum value of the criterion for this class is greater than that of the full model the procedure is terminated and the full model is accepted. Otherwise, the model that mimimizes the criterion is accepted and the parameter that is missing in this model is deleted and never considered again. This model is then treated as a new full model and the above step is repeated. The advantage of the SBE method is that the maximum number of times that the C-criterion needs to be evaluated is

$$1 + n_\theta(n_\theta + 1)/2 \tag{35}$$

This is considerably less than that of the combinatorial method (e.g. n_θ=30, $2^{n_\theta}=1.07\times10^9$, $1 + n_\theta(n_\theta + 1)/2 = 466$). Although the SBE method does not always select the optimal model with the minimum value of the C-criterion as given by the combinatorial method, it provides an efficient means to select models for output-affine systems.

For an output-affine model, the situation $s=n_\theta$ is impossible and the parameter vector must contain at least one coefficient of the polynomial $a_0(u(t-1),...,u(t-r))$. The actual maximum number of all the competing models will therefore be less than that given by eqn. (34) or (35) depending on whether the combinatorial method or the SBE method is used. In the derivation of eqn. (26), it was assumed that the parameters to be eliminated were the last elements in the full parameter vector. This is not necessary as it will be shown in the next section.

5. Numerical aspects

The calculation of the direction vector d^k, step (c) in the prediction-error algorithm, does not actually require the inversion of the Hessian. Since the approximate Hessian matrix H is always symmetric positive definite, the square root method can be utilized to factorize H into

$$H = W^\tau W \tag{36}$$

where W is an upper triangular square matrix. The problem of evaluating d^k is thus equivalent to that of solving the following linear equations

$$W^\tau(Wd^k) = -\nabla_\theta J \tag{37}$$

Let

$$H = \left[h_{ij}\right] \quad i,j=1,\ldots,n_\theta \tag{38}$$

and

$$W = \begin{bmatrix} w_{11} & w_{12} & \cdot & \cdot & \cdot & w_{1n_\theta} \\ & w_{22} & \cdot & \cdot & \cdot & w_{2n_\theta} \\ & & \cdot & & & \cdot \\ & 0 & & \cdot & & \cdot \\ & & & & \cdot & \cdot \\ & & & & & w_{n_\theta n_\theta} \end{bmatrix} \tag{39}$$

It is straightforward to verify

$$\begin{aligned} &w_{11} = \sqrt{h_{11}},\ w_{1j} = \frac{h_{1j}}{w_{11}},\ j = 2,\ldots,n_\theta \\ &\text{for } i=2,\ldots,n_\theta \quad w_{ii} = \sqrt{h_{ii} - \sum_{k=1}^{i-1} w_{ki}^2} \\ &\qquad\qquad w_{ij} = \frac{h_{ij} - \sum_{k=1}^{i-1} w_{ki}w_{kj}}{w_{ii}} \quad j = i+1,\ldots,n_\theta \end{aligned} \tag{40}$$

Define

$$-\nabla_\theta J = \begin{bmatrix} g_1 \\ \cdot \\ \cdot \\ \cdot \\ g_{n_\theta} \end{bmatrix}, \quad Wd^k = \begin{bmatrix} x_1 \\ \cdot \\ \cdot \\ \cdot \\ x_{n_\theta} \end{bmatrix}, \quad d^k = \begin{bmatrix} d_1 \\ \cdot \\ \cdot \\ \cdot \\ d_{n_\theta} \end{bmatrix} \tag{41}$$

Equation (37) can then be solved efficiently using the substitution algorithms

$$x_1 = \frac{g_1}{w_{11}}, \quad x_i = \frac{g_i - \sum_{k=1}^{i-1} w_{ki}x_k}{w_{ii}} \quad i=2,\ldots,n_\theta \tag{42}$$

$$d_{n_\theta} = \frac{x_{n_\theta}}{w_{n_\theta n_\theta}}, \quad d_i = \frac{x_i - \sum_{k=i+1}^{n_\theta} w_{ik}d_k}{w_{ii}} \quad i=n_{\theta-1},\ldots,1 \tag{43}$$

After the Hessian H at the unrestricted minimum $\hat{\theta}_t$ has been decomposed into $W^\tau W$, the evaluation of the statistic $d(y)$ for all the reduced models can be performed more efficiently

using the Householder orthogonal transformation (Bierman, 1977) than using equation (26) directly. It holds approximately for θ near $\hat{\theta}_t$

$$NJ(\theta) = NJ(\hat{\theta}_t) + \frac{N}{2}(\theta - \hat{\theta}_t)^T W^T W(\theta - \hat{\theta}_t) = NJ(\hat{\theta}_t) + \frac{N}{2}(W\theta - W\hat{\theta}_t)^T(W\theta - W\hat{\theta}_t) \tag{44}$$

Assume that a reduced model has s parameters fewer than the full model. Denote the parameter vector of this reduced model as b and those s parameters missing in the reduced model as c respectively. Rearrange $W\theta$ into

$$W\theta = \overline{W}_b b + \overline{W}_c c \tag{45}$$

where $\overline{W}_b$ is the $n_\theta \times (n_\theta - s)$ matrix which consists of the columns of W that correspond to the vector b and $\overline{W}_c$ is the $n_\theta \times s$ matrix which contains the columns of W that are associated with the vector c. Let

$$v = W\hat{\theta}_t \tag{46}$$

For the vector c restricted to zero, the loss function can be written as

$$NJ(\theta) = NJ(\hat{\theta}_t) + \frac{N}{2}(\overline{W}_b b - v)^T(\overline{W}_b b - v) \tag{47}$$

The orthogonal matrix T that triangulates $\overline{W}_b$ can be found using the Householder transformation. Since $T^T T = I$, eqn. (47) becomes

$$NJ(\theta) = NJ(\hat{\theta}_t) + \frac{N}{2}(\overline{W}_b b - v)^T T^T T(\overline{W}_b b - v) = NJ(\hat{\theta}_t) + \frac{N}{2}(T\overline{W}_b b - Tv)^T(T\overline{W}_b b - Tv) \tag{48}$$

$T\overline{W}_b$ is an $n_\theta \times (n_\theta - s)$ matrix having the form

$$T\overline{W}_b = \begin{bmatrix} W_b \\ 0 \end{bmatrix} \tag{49}$$

where W_b is a $(n_\theta - s) \times (n_\theta - s)$ upper triangular square matrix. Partition the vector Tv as

$$Tv = \begin{bmatrix} z \\ \eta \end{bmatrix} \tag{50}$$

where z and η have dimensions $n_\theta - s$ and s respectively. The loss function can then be written as

$$NJ(\theta) = NJ(\hat{\theta}_t) + \frac{N}{2}(W_b b - z)^T(W_b b - z) + \frac{N}{2}\eta^T\eta \tag{51}$$

The restricted minimum $\hat{\theta}_o$ minimizes the above $NJ(\theta)$, that is,

$$W_b \hat{b}_o = z \tag{52}$$

The value of $d(y)$ is thus given by

$$d(y) = 2NJ(\hat{\theta}_o) - 2NJ(\hat{\theta}_t) = N\eta^T\eta \tag{53}$$

The value of the C-criterion corresponding to this reduced model is

$$C = 2N\, J(\hat{\theta}_o) + 4(n_\theta - s) = N \log Q(\hat{\theta}_t)) + 4(n_\theta - s) + N\eta^T\eta \tag{54}$$

It can be seen that the prediction-error algorithm needs to be called only once for the full model. The model reduction routine then selects the optimal model from all the competing

models using the above scheme. The estimates of the reduced parameter vectors are provided by eqn. (52). If the SBE algorithm is employed, each time a parameter is eliminated, the corresponding column in W is removed into $\overline{W}_c$ permanently. The Householder transformation technique is well documented elsewhere and will not be detailed here.

For a complicated full model, the Hessian may become almost singular. To avoid numerical difficulties, a diagonal matrix ρI, where ρ is a small positive scalar, can be added to the Hessian before the factorization. Thus

$$H + \rho I = W^{\tau}W \tag{55}$$

This alters the direction d^k slightly. This is unimportant since d^k is only used as a direction along which a linear search for the minimum can be performed. However, it will affect the calculation of the statistic $d(y)$. A more accurate approximation of $d(y)$ in this case is given in (Leontaritis and Billings, 1986)

$$d(y) = N\eta^{\tau}\eta - N\rho(\hat{\theta}_o - \hat{\theta}_1)^{\tau}(\hat{\theta}_o - \hat{\theta}_1) \tag{56}$$

6. Model validation

Model validation forms the final stage of any identification procedure. For the model reduction approach, the full model should be complicated enough to include all the significant terms of the true system. If this is not the case, the final model selected will not be a good representation of the system. The objective of model validity tests is to reveal such a deficient model. Only three simple correlation-based tests (Billings and Voon, 1986a) will briefly be described. Other more sophisticated parametric and non-parametric model validity tests are also available (Leontaritis and Billings, 1987). They will not be discussed here.

If the model structure and parameter estimate $\hat{\theta}$ are correct then the prediction error sequence $\varepsilon(t,\hat{\theta})$ should be unpredictable from all linear and nonlinear combinations of past inputs and outputs. This condition will hold iff (Billings and Voon, 1986a)

$$\Phi_{\varepsilon\varepsilon}(k) = \frac{E[\varepsilon(t,\hat{\theta})\varepsilon(t-k,\hat{\theta})]}{E[\varepsilon^2(t,\hat{\theta})]} = \delta(k)$$

$$\Phi_{u\varepsilon}(k) = \frac{E[u(t)\varepsilon(t-k,\hat{\theta})]}{\sqrt{E[u^2(t)]E[\varepsilon^2(t,\hat{\theta})]}} = 0, \quad \text{for } all\ k \tag{57}$$

$$\Phi_{\varepsilon(\varepsilon u)}(k) = \frac{E[\varepsilon(t,\hat{\theta})\varepsilon(t-1-k,\hat{\theta})u(t-1-k)]}{\sqrt{E[\varepsilon^2(t,\hat{\theta})]E[\varepsilon^2(t,\hat{\theta})u^2(t)]}} = 0, \quad \text{for } k\geq 0$$

The tests in (57) can be applied provided that a noise model is fitted as part of the estimation procedure so that it is possible to reduce $\varepsilon(t,\hat{\theta})$ to an unpredictable sequence. This requirement is satisfied for the prediction-error estimation algorithm. Note that the traditional linear correlation tests $\Phi_{\varepsilon\varepsilon}(k)$ and $\Phi_{u\varepsilon}(k)$ are not sufficient.

In practice, the normalized correlation function between two sequences $\psi(t)$ and $\omega(t)$ is estimated using the sampled correlation function

$$\hat{\Phi}_{\psi\omega}(k) = \frac{\sum_{t=1}^{N-k} \psi(t)\omega(t+k)}{\sqrt{\sum_{t=1}^{N} \psi^2(t) \sum_{t=1}^{N} \omega^2(t)}} \tag{58}$$

The normalization ensures that $-1 \le \hat{\Phi}_{\psi\omega}(k) \le 1$. Confidence intervals are used to indicate if the correlation between variables is significant or not. For large N, the standard deviation of the correlation estimate is $\frac{1}{\sqrt{N}}$, the 95% confidence limits are therefore approximately $\pm\frac{1.96}{\sqrt{N}}$.

7. Simulation study

Two simulated examples are used to demonstrate the application of the prediction-error algorithm and the SBE routine. The linear search in the optimisation algorithm is carried out by a golden section search routine and $k_\alpha(1) = 4$ is used to calculate the C-criterion.

Example 1

$$y(t) = \frac{1}{u(t-1)+u(t-2)}\{0.2u(t-1)y(t-2)+0.6u(t-1)+0.7u(t-2)-0.2u(t-1)e(t-2)\}+e(t)$$

or

$$(u(t-1)+u(t-2))y(t) = 0.2u(t-1)y(t-2)+0.6u(t-1)+0.7u(t-2)-0.2u(t-1)e(t-2)+(u(t-1)+u(t-2))e(t)$$

where the noise $e(t)$ is a Gaussian white sequence with zero mean and variance 0.04. An input-output sequence of 500 points was generated. The input u(t) was an independent sequence of uniform distribution with zero mean and variance 1. The input-output data set is illustrated in Fig.1.

A full model with r=2 and L=1 was used to fit the data. The full model contains all the terms of the true system. The prediction-error estimates of the parameters and their standard deviations are given in TABLE 1. The estimates are unbiased since the true values of the parameters are all within two standard deviations of the estimated values. This was to be expected. The predicted outputs of the model and the residuals are plotted in Fig.2. The correlation functions $\Phi_{\varepsilon\varepsilon}(k)$, $\Phi_{u\varepsilon}(k)$ and $\Phi_{\varepsilon(u\varepsilon)}(k)$ shown in Fig. 3 also indicate that the model is adequate. However, this full model is heavily over-parametrized. The SBE routine is used to reduced the parameters of the model and the results obtained are shown in TABLE 2. It is seen from TABLE 2 that AIC over-estimates the number of required parameters and it retains the term $e(t-1)$ while the C-criterion finds the correct model. After the SBE procedure, the prediction-error algorithm was called a second time to improve the parameter estimates of the final model. The final estimates, the predicted outputs of the final model and the residuals are given in TABLE 3 and Fig.4 respectively. The correlation tests for the final model are shown

in Fig.5.

Example 2

$$y(t) = \frac{1}{1+u(t-1)+u^2(t-2)}\{0.4u^2(t-2)y(t-1)+0.2u(t-1)y(t-2)+0.6u(t-1)u(t-2) \\ -0.4u^2(t-2)e(t-1)-0.2u(t-1)e(t-2)\}+e(t)$$

or

$$(1+u(t-1)+u^2(t-2))y(t) = 0.4u^2(t-2)y(t-1)+0.2u(t-1)y(t-2)+0.6u(t-1)u(t-2) \\ -0.4u^2(t-2)e(t-1)-0.2u(t-1)e(t-2)+(1+u(t-1)+u^2(t-2))e(t)$$

The noise sequence $e(t)$ was the same as in Example 1. 500 input-output data points were generated using the input

$$u(t) = sin(\frac{\pi}{20}t) + \beta(t)$$

where $\beta(t)$ was an independent sequence of uniform distribution with zero mean and variance 0.04. The inputs and outputs of the system are illustrated in Fig.6.

Initially, an inadequate full model with r=1 and L=1 was used to identify the system. The estimates are given in TABLE 4 and they are biased as expected. The predicted outputs of the model and the residuals are shown in Fig.7. The SBE routine cannot reduce the model any further. The correlation tests plotted in Fig.8 indicate that the model is deficient.

An over-parametrized model with 22 parameters was also used to fit the data. This full model includes all the 8 terms of the system. The estimation results and the correlation tests are shown in TABLE 5, Fig.9 and Fig.10 respectively. The SBE routine was then used to simplify the full model. The results of the SBE procedure are given in TABLE 6. It is seen that the C-criterion deleted 12 redundant terms in the model while AIC removed only 9 redundant terms. The SBE routine was entered again after the prediction-error algorithm had been called to improve the accuracy of the estimates of the reduced model. The results given in TABLE 7 indicate that the two remaining redundant terms have been deleted. The estimates of the final model, the predicted outputs and the residuals are shown in TABLE 8 and Fig.11 respectively. The correlation tests for the final model are plotted in Fig.12.

8. Comparison with the NARMAX model

Leontaritis and Billings (1985) rigorously proved that a nonlinear system can be represented by the difference equation model

$$y(t) = F(y(t-1),...,y(t-r),u(t-1),...,u(t-r),e(t-1),...,e(t-r)) + e(t) \qquad (59)$$

in a region around an equilibrium point, where F is some nonlinear function, provided that the system is finitely realizable and a linearized model exists if the system is operated close to an

equilibrium point. The difference equation model (59) is known as the NARMAX model. Unlike the traditional representations of nonlinear systems, the Volterra and Wiener functional series, both the NARMAX model and the globally valid output-affine model provide very concise representations for nonlinear systems. This section presents a brief comparison between these two models. The emphasis is on the application of these two models to the identification problem.

The validity of the model (59) in a region around an equilibrium point may impose no serious restriction to its application since pratical systems are often regulated around some operating point. A nonlinear model is needed if the system is not operated closed enough to the desired operating point for a linear model to be useful. The nonlinear function F can be approximated to arbitrary accuracy using polynomial functions valid in some region. It is important to emphasize that such a region is generally much larger than the region where a linear model would be valid. Let a polynomial representation of (59) be

$$y(t) = F^L(y(t-1),...,y(t-r),u(t-1),...,u(t-r),\varepsilon(t-1,\theta),...,\varepsilon(t-r,\theta)) + \varepsilon(t,\theta) \tag{60}$$

where L is the degree of the polynomial and θ is the vector of polynomial coefficients required to be estimated. Equation (60) is a linear-in-the-parameters model. This is an obvious advantage since many linear regression methods are readily applicable and unbiased estimates of θ can be obtained using the extended least squares algorithm (Billings and Voon, 1984; Korenberg, Billings and Liu, 1986).

The output-affine model (13) is not linear-in-the-parameters, and therefore, the linear least squares principle cannot be applied directly. If a polynomial function is used to approximate an output-affine model then both its special structure of affine in outputs and the global validity are lost. To illustrate this consider the simple example of Section 2, and set $\theta_1 = \theta_2 = 1$. Using

$$\frac{1}{1+u(t)} = \sum_{i=0}^{\infty} (-1)^i u^i(t) \simeq 1 - u(t) + u^2(t), \quad |u(t)| < 1$$

a NARMAX model of the form of eqn. (60) is actually obtained, which is valid for $|u(t)| < 1$.

Multiplying both sides of equation (13) by $a_0(u(t-1),...,u(t-r))$, defining

$$\xi(t) = a_0(u(t-1),...,u(t-r))\varepsilon(t,\theta), \tag{61}$$

keeping only one term of $a_0(u(t-1),...,u(t-r))y(t)$ on the left hand side of the equation and moving the rest of terms to the right hand side results in the following linear-in-the-parameters expression (Billings, Korenberg and Chen, 1987)

$$Y(t) = \sum_{i=1}^{n_0-1} q_i(t)\bar{\theta}_i + \xi(t) \tag{62}$$

Two difficulties arise. Firstly, if a term is used as $Y(t)$ which does not exist in the data

generating mechanism, the final fitting will be useless. In order to find the correct choice of $Y(t)$, all the n possible expressions of eqn. (62) must be fitted and the results analyzed. This apparently complicates the identification task. Secondly, the estimates of $\bar{\theta}_i$ are biased because some of $q_i(t)s$ have $y(t)$ as their component and are therefore correlated with $\xi(t)$.

The prediction-error estimation method is a general estimation method and is applicable to both models. The linear-in-the-parameters nature of eqn. (60), however, makes the task of structure determination easier. Both the model expansion method and the model reduction method can be employed to determine the correct model structure (Leontaritis and Billings, 1987). Moreover, a combination of forward and backward regression techniques, the stepwise regression (Draper and Smith, 1981), can be used to detect significant terms in the model prior to final estimation. The combined algorithm of prediction-error estimation and stepwise regression provides a powerful procedure for fitting parsimonious NARMAX models to nonlinear systems (Billings and Voon, 1986b). For the output-affine model, it is difficult to apply structure determination technques based on linear regression such as the stepwise regression routine.

9. Conclusions

A prediction-error estimation algorithm has been developed for nonlinear stochastic output-affine systems. The task of structure determination was performed using the model reduction method. The numerical implementation of the prediction-error algorithm and the associated model selection technique has been discussed in detail. The combined routine consisting of a prediction-error estimator and a stepwise backward elimination algorithm provides a practical means of constructing output-affine models from input-output data measurements.

The full model should not be too complicated otherwise it may contain an excessive number of redundant parameters. However, if the full model does not include all the significant terms of the true system a deficient final model will be produced. Simple correlation-based tests can be employed to detect such a case.

A comparison between the output-affine model and the NARMAX model for the identification of nonlinear systems has been presented.

Acknowledgments

The authors gratefully acknowledge financial support for the work presented above from the Science and Engineering Research Council, Ref. GR/D/30587.

References

[1] Bierman, G.J. (1977). *Factorization Methods for Discrete Sequential Estimation.* Academic Press, New York.

[2] Billings, S.A., M.J. Korenberg and S. Chen (1987). Identification of nonlinear output-affine systems using orthogonal least squares algorithm. *Research Report No.313,* Department of Control Eng., University of Sheffield, Sheffield, U.K.

[3] Billings, S.A., and W.S.F. Voon (1984). Least squares parameter estimation algorithms for non-linear systems. *Int. J. Systems Sci.,* Vol.15, No.6, pp601-615.

[4] Billings, S.A., and W.S.F. Voon (1986a). Correlation based model validity tests for non-linear models. *Int. J. Control,* Vol.44, No.1, pp235-244.

[5] Billings, S.A., and W.S.F. Voon (1986b). A prediction-error and stepwise-regression estimation algorithm for non-linear systems. *Int. J. Control,* Vol.44, No.3, pp803-822.

[6] Cyrot-Normand, D., and H. Dang Van Mien (1980). Nonlinear state-affine identification methods: applications to electrical power plants. *Preprints of IFAC Symp. on Auto. Control in Power Generation, Distribution and Protection.* Pretoria, South Africa, 1980, pp449-462.

[7] Draper, N.R., and H. Smith (1981). *Applied Regression Analysis.* Wiley, New York.

[8] Goodwin, G.C., and R.L. Payne (1977). *Dynamic System Identification: Experiment Design and Data Analysis.* Academic Press, New York.

[9] Korenberg, M.J., S.A. Billings, and Y.P. Liu (1986). An orthogonal parameter estimation algorithm for nonlinear stochastic systems. *Research Report No.307,* Department of Control Eng., University of Sheffield, Sheffield, U.K. (Submitted for publication).

[10] Leontaritis, I.J., and S.A Billings (1985). Input-output parametric models for non-linear systems, Part I: deterministic non-linear systems; Part II: stochastic non-linear systems. *Int. J. Control,* Vol.41, No.2, pp303-344.

[11] Leontaritis, I.J., and S.A Billings (1986). A prediction error estimator for nonlinear stochastic systems. *Int. J. Systems Sci.,* (to appear).

[12] Leontaritis, I.J., and S.A Billings (1987). Model selection and validation methods for non-linear systems. *Int. J. Control,* Vol.45, No.1, pp311-341.

[13] Sontag, E.D. (1979). *Polynomial Response Maps,* Lecture Notes in Control and Information Sciences 13. Springer Verlag, Berlin.

TABLE 1. Estimates and standard deviations (full model for Example 1)

terms	estimates	standard deviations	terms	estimates	standard deviations
$y(t)$	-0.22003E-2	0.15728E-2	*constant*	0.71647E-2	0.12947E-1
$u(t-1)y(t)$	0.10237E+1	0.28513E+0	$u(t-1)$	0.61298E+0	0.17217E+0
$u(t-2)y(t)$	0.10259E+1	0.28573E+0	$u(t-2)$	0.69903E+0	0.19574E+0
$y(t-1)$	-0.15351E-2	0.12883E-1	$e(t-1)$	0.20291E-1	0.16920E-1
$u(t-1)y(t-1)$	-0.59215E-3	0.28970E-1	$u(t-1)e(t-1)$	0.35238E-1	0.49928E-1
$u(t-2)y(t-1)$	0.60053E-2	0.24355E-1	$u(t-2)e(t-1)$	0.30087E-1	0.48559E-1
$y(t-2)$	-0.93303E-2	0.12453E-1	$e(t-2)$	0.19852E-1	0.16562E-1
$u(t-1)y(t-2)$	0.19928E+0	0.62117E-1	$u(t-1)e(t-2)$	-0.20110E+0	0.72372E-1
$u(t-2)y(t-2)$	0.64485E-2	0.30157E-1	$u(t-2)e(t-2)$	0.29343E-2	0.47129E-1

TABLE 2. Model reduction using SBE (full model for example 1)

elimination step	eliminated parameter	AIC value	C-criterion value	standard deviation of residuals
		-0.15613E+4	-0.15253E+4	0.20114E+0
1	$u(t-1)y(t-1)$	-0.15633E+4	-0.15293E+4	0.20114E+0
2	$u(t-2)e(t-2)$	-0.15653E+4	-0.15333E+4	0.20114E+0
3	$y(t-1)$	-0.15673E+4	-0.15373E+4	0.20115E+0
4	$u(t-2)y(t-2)$	-0.15692E+4	-0.15412E+4	0.20117E+0
5	$u(t-2)y(t-1)$	-0.15709E+4	-0.15449E+4	0.20122E+0
6	*constant*	-0.15724E+4	-0.15484E+4	0.20134E+0
7	$y(t-2)$	-0.15741E+4	-0.15521E+4	0.20138E+0
8	$u(t-1)e(t-1)$	-0.15756E+4	-0.15556E+4	0.20149E+0
9	$u(t-2)e(t-1)$	-0.15774E+4	-0.15594E+4	0.20154E+0
10	$e(t-2)$	-0.15789E+4	-0.15629E+4	0.20164E+0
11	$y(t)$	-0.15799E+4	-0.15659E+4	0.20183E+0
12	$e(t-1)$	-0.15791E+4	-0.15671E+4	0.20241E+0
13	$u(t-1)e(t-2)$	-0.15750E+4	-0.15650E+4	0.20364E+0

TABLE 3. Estimates and standard deviations (final model for Example 1)

terms	estimates	standard deviations	terms	estimates	standard deviations
$u(t-1)y(t)$	0.10102E+1	0.26547E+0	$u(t-1)$	0.59580E+0	0.15679E+0
$u(t-2)y(t)$	0.10105E+1	0.26555E+0	$u(t-2)$	0.69993E+0	0.18403E+0
$u(t-1)y(t-2)$	0.20762E+0	0.55308E-1	$u(t-1)e(t-2)$	-0.21480E+0	0.58266E-1

TABLE 4. Estimates and standard deviations (inadequate full model for Example 2)

terms	estimates	standard deviations	terms	estimates	standard deviations
$y(t)$	0.11839E+1	0.28105E+0	*constant*	0.98532E-1	0.33423E-1
$u(t-1)y(t)$	0.10610E+1	0.25186E+0	$u(t-1)$	0.80105E-1	0.28501E-1
$y(t-1)$	0.82058E+0	0.19755E+0	$e(t-1)$	-0.27334E+0	0.10201E+0
$u(t-1)y(t-1)$	0.74906E+0	0.17994E+0	$u(t-1)e(t-1)$	-0.25397E+0	0.92647E-1

TABLE 5. Estimates and standard deviations (adequate full model for Example 2)

terms	estimates	standard deviations	terms	estimates	standard deviations
$y(t)$	0.85876E+0	0.11830E+0	*constant*	-0.93936E-2	0.13869E-1
$u(t-1)y(t)$	0.72095E+0	0.12299E+0	$u(t-1)$	0.14123E-1	0.30210E-1
$u(t-2)y(t)$	0.52964E+0	0.15185E+0	$u(t-2)$	0.16364E-1	0.41674E-1
$u^2(t-2)y(t)$	0.11689E+1	0.14642E+0	$u(t-1)u(t-2)$	0.49657E+0	0.76176E-1
$y(t-1)$	-0.79704E-2	0.99154E-1	$e(t-1)$	0.90533E-2	0.10657E+0
$u(t-1)y(t-1)$	-0.36507E-2	0.10838E+0	$u(t-1)e(t-1)$	-0.40174E-1	0.12154E+0
$u(t-2)y(t-1)$	0.35440E+0	0.13668E+0	$u(t-2)e(t-1)$	-0.13886E+0	0.14730E+0
$u^2(t-2)y(t-1)$	0.60393E+0	0.98296E-1	$u^2(t-2)e(t-1)$	-0.40556E+0	0.12348E+0
$y(t-2)$	0.68006E-1	0.68830E-1	$e(t-2)$	-0.14503E+0	0.82840E-1
$u(t-1)y(t-2)$	0.23962E+0	0.70256E-1	$u(t-1)e(t-2)$	-0.25209E+0	0.99739E-1
$u(t-2)y(t-2)$	-0.78181E-1	0.72690E-1	$u(t-2)e(t-2)$	0.20734E-1	0.11931E+0

TABLE 6. Model reduction using SBE

(adequate full model for example 2)

elimination step	eliminated parameter	AIC value	C-criterion value	standard deviation of residuals
		-0.15637E+4	-0.15197E+4	0.19906E+0
1	$u(t-1)y(t-1)$	-0.15657E+4	-0.15237E+4	0.19906E+0
2	$y(t-1)$	-0.15677E+4	-0.15277E+4	0.19906E+0
3	$e(t-1)$	-0.15697E+4	-0.15317E+4	0.19907E+0
4	$u(t-2)e(t-2)$	-0.15716E+4	-0.15356E+4	0.19907E+0
5	$u(t-2)$	-0.15735E+4	-0.15395E+4	0.19910E+0
6	$u(t-1)e(t-1)$	-0.15750E+4	-0.15430E+4	0.19920E+0
7	*constant*	-0.15764E+4	-0.15464E+4	0.19932E+0
8	$y(t-2)$	-0.15776E+4	-0.15496E+4	0.19948E+0
9	$u(t-1)$	-0.15779E+4	-0.15519E+4	0.19982E+0
10	$e(t-2)$	-0.15773E+4	-0.15533E+4	0.20035E+0
11	$u(t-2)y(t-2)$	-0.15758E+4	-0.15538E+4	0.20105E+0
12	$u(t-2)e(t-1)$	-0.15740E+4	-0.15540E+4	0.20181E+0
13	$u(t-2)y(t)$	-0.15676E+4	-0.15496E+4	0.20353E+0

TABLE 7. Model reduction using SBE (reduced model for example 2)

elimination step	eliminated parameter	AIC value	C-criterion value	standard deviation of residuals
		-0.15780E+4	-0.15580E+4	0.20100E+0
1	$u(t-2)y(t)$	-0.15777E+4	-0.15597E+4	0.20146E+0
2	$u(t-2)y(t-1)$	-0.15785E+4	-0.15625E+4	0.20172E+0
3	$u(t-1)e(t-2)$	-0.15750E+4	-0.15610E+4	0.20284E+0

TABLE 8. Estimates and standard deviations (final model for Example 2)

terms	estimates	standard deviations	terms	estimates	standard deviations
$y(t)$	0.9950E+0	0.32723E+0	$u(t-1)y(t-2)$	0.20733E+0	0.69510E-1
$u(t-1)y(t)$	0.10099E+1	0.32999E+0	$u(t-1)u(t-2)$	0.58342E+0	0.19187E+0
$u^2(t-2)y(t)$	0.10271E+1	0.33850E+0	$u^2(t-2)e(t-1)$	-0.34646E+0	0.14054E+0
$u^2(t-2)y(t-1)$	0.41023E+0	0.14034E+0	$u(t-1)e(t-2)$	-0.16349E+0	0.66904E-1

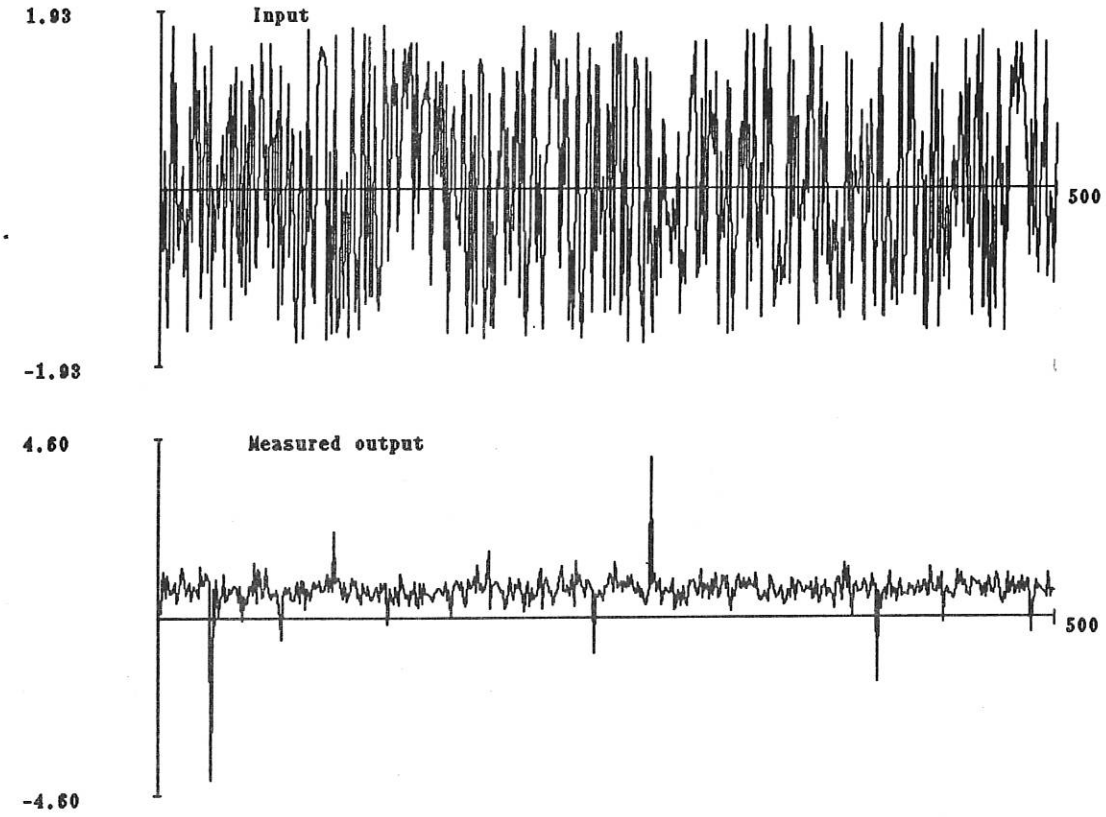

Fig.1. Inputs and outputs of Example 1

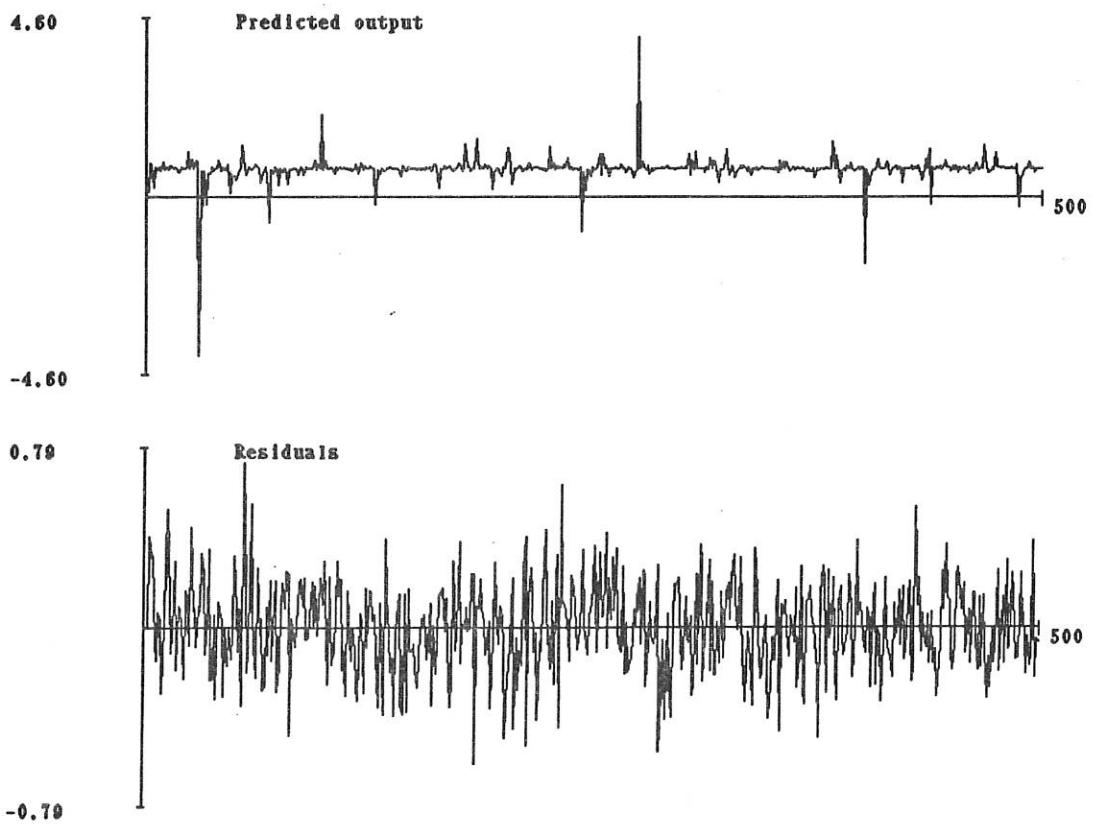

Fig.2. Predicted outputs and residuals (full model for Example 1)

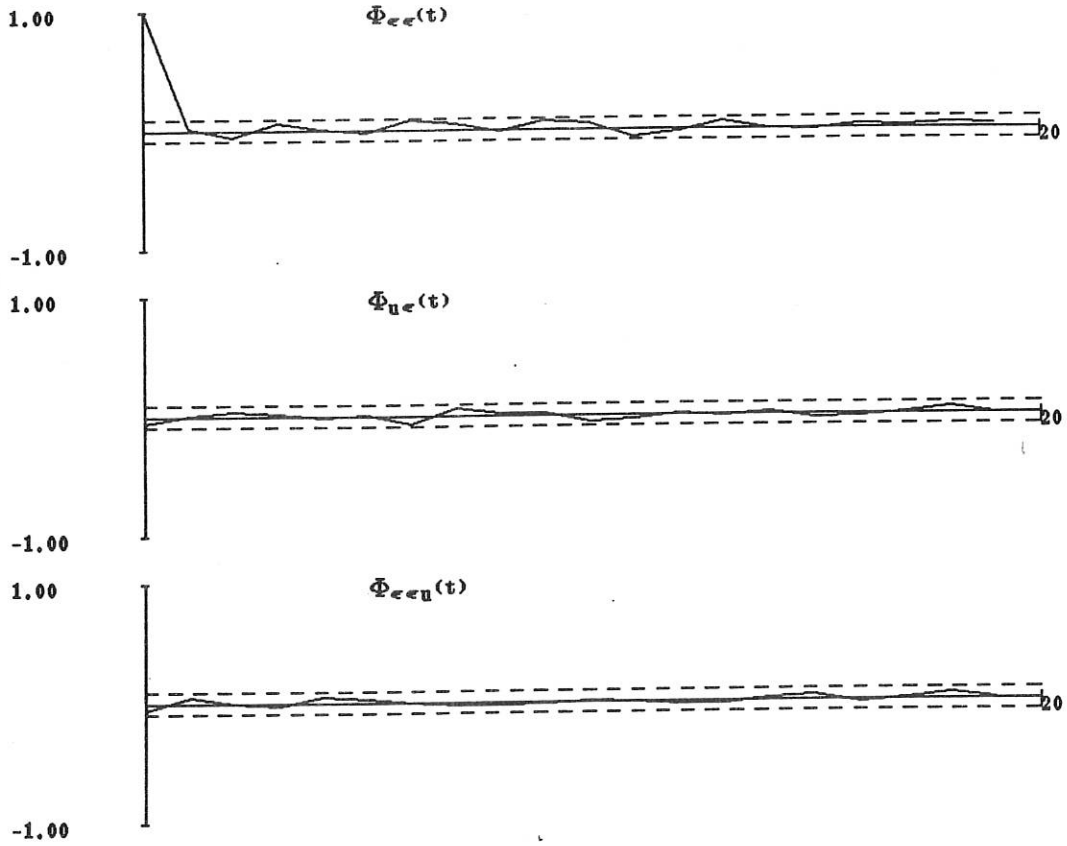

Fig.3. Correlation tests (full model for Example 1)

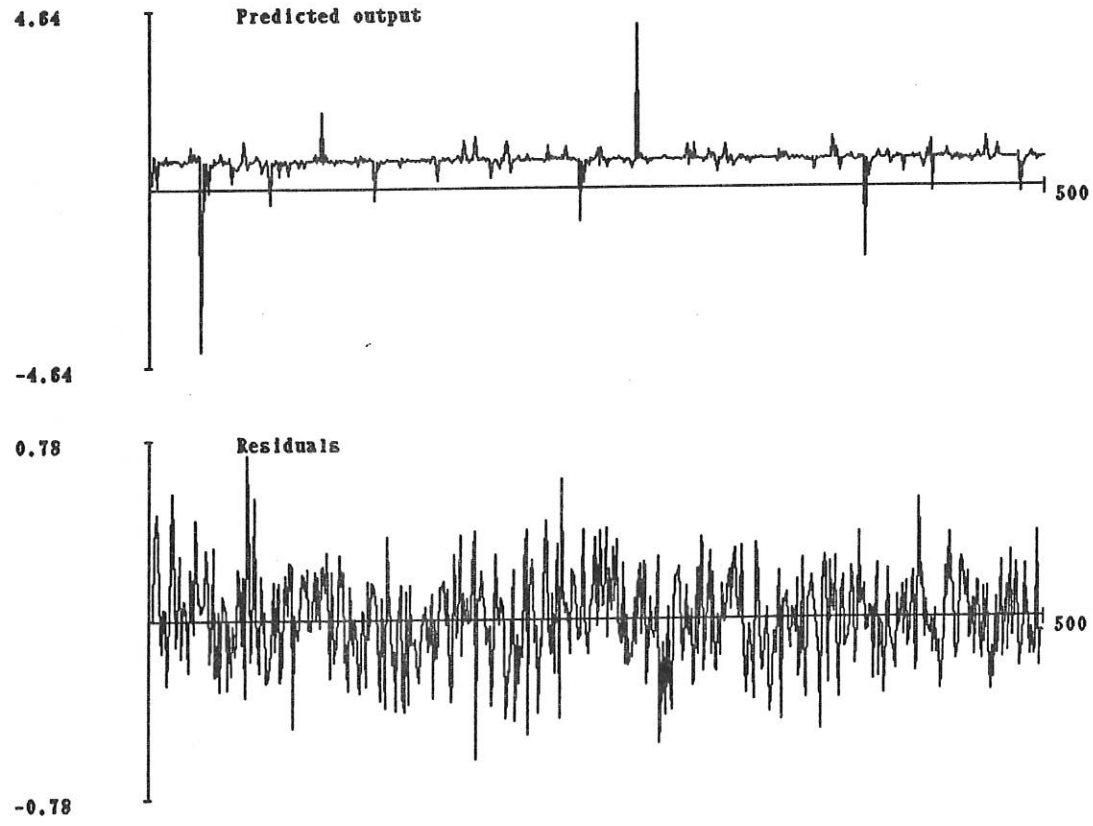

Fig.4. Predicted outputs and residuals (final model for Example 1)

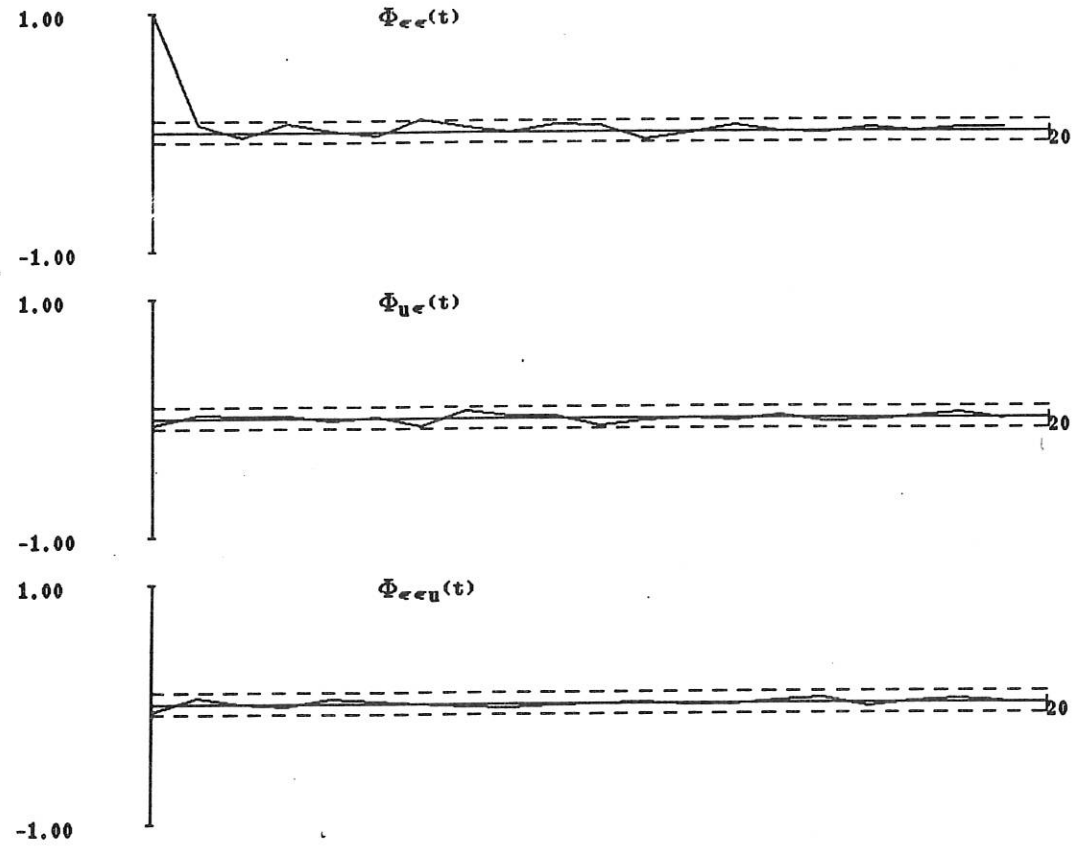

Fig.5. Correlation tests (final model for Example 1)

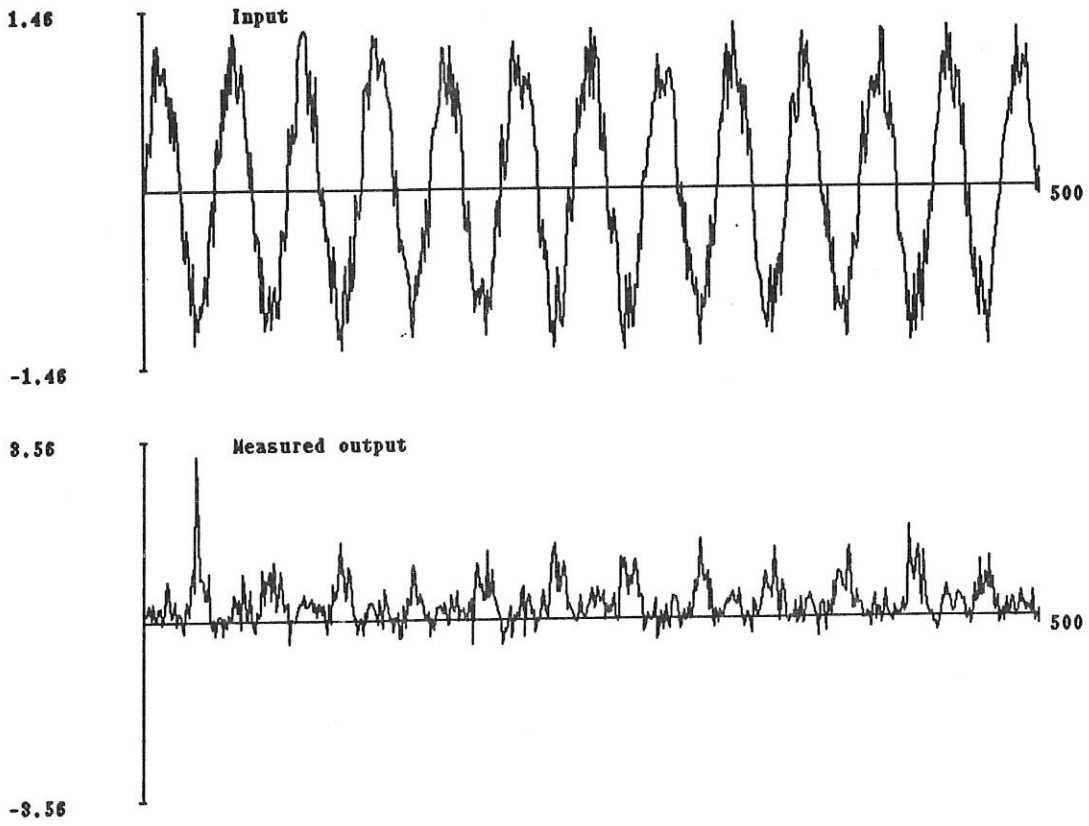

Fig.6. Inputs and outputs of Example 2

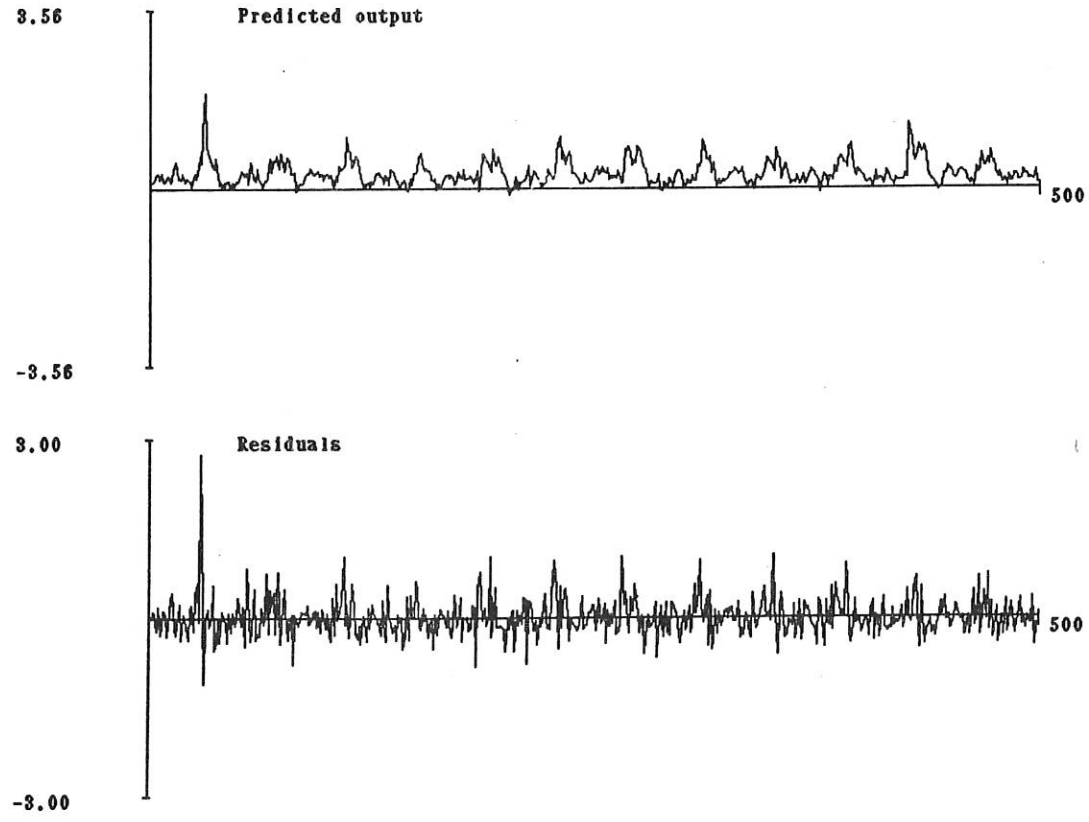

Fig.7. Predicted outputs and residuals (inadequate full model for Example 2)

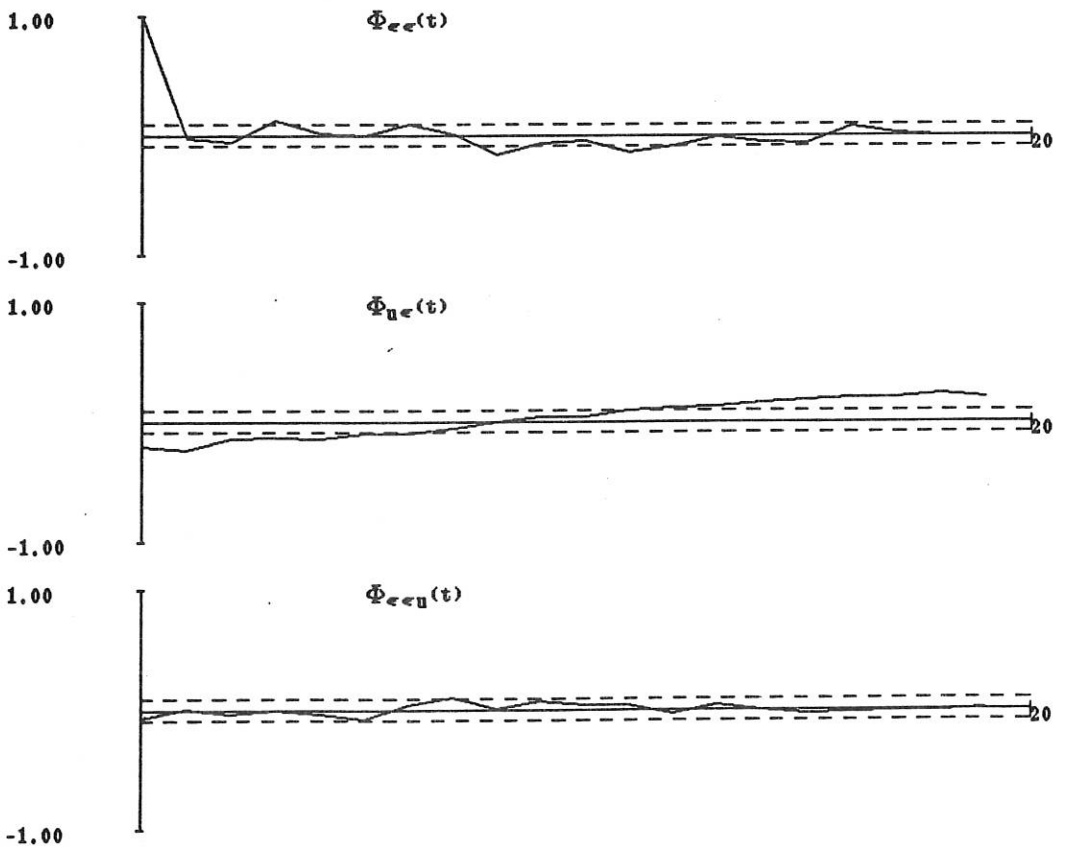

Fig.8. Correlation tests (inadequate full model for Example 2)

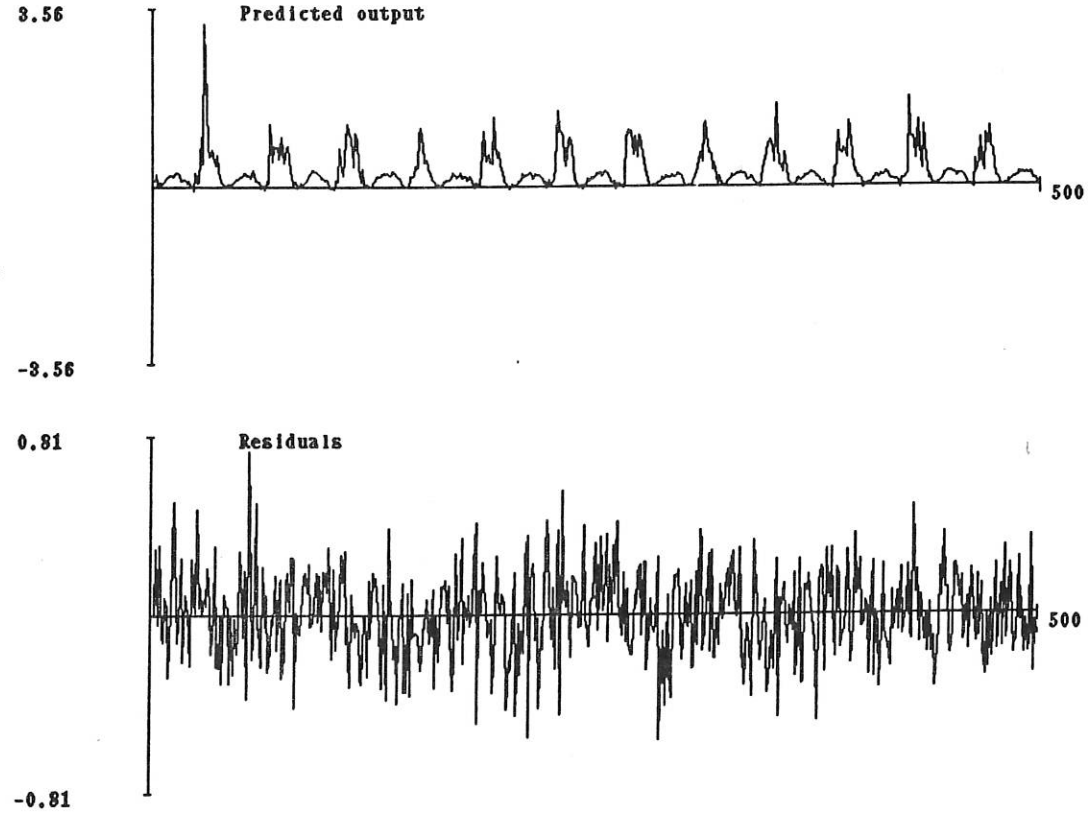

Fig.9. Predicted outputs and residuals (adequate full model for Example 2)

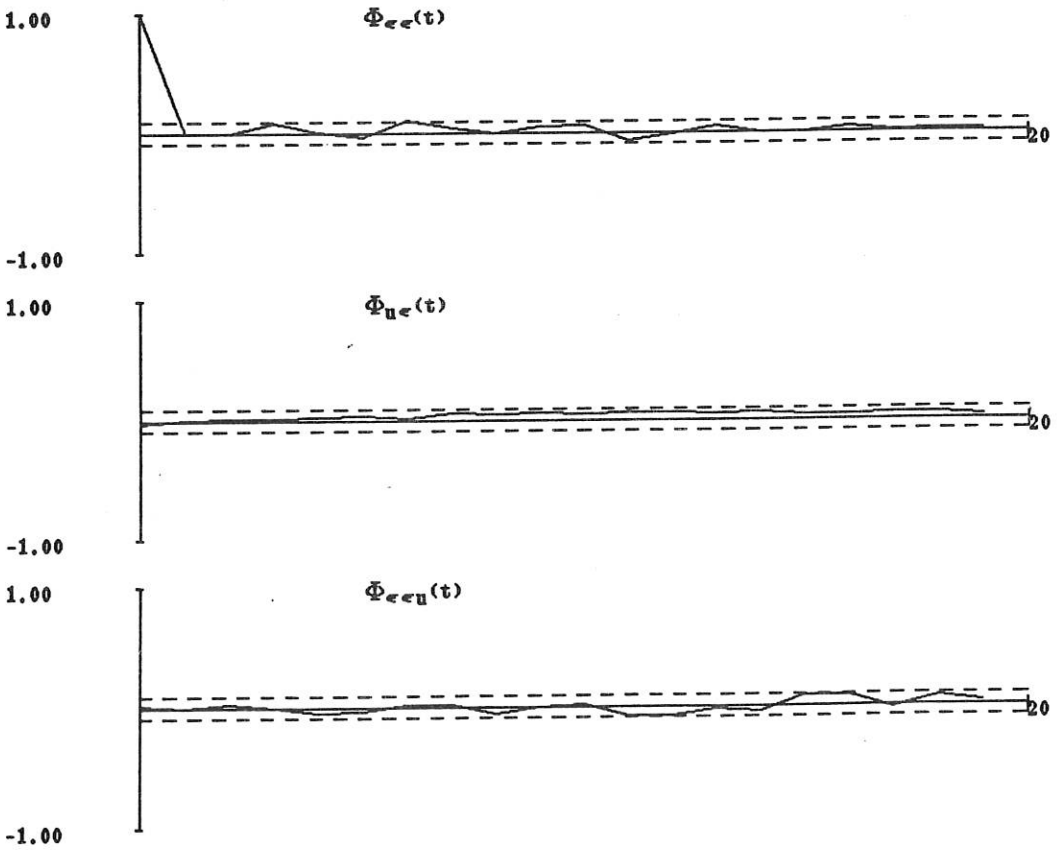

Fig.10. Correlation tests (adequate full model for Example 2)