International Journal of Control
Publication details, including instructions for authors and subscription information:
http://www.informaworld.com/smpp/title~content=t713393989

Orthogonal least squares methods and their application to non-linear system identification

S. Chen [a]; S. A. Billings [a]; W. Luo [a]
[a] Department of Control Engineering, University of Sheffield, Sheffield, England, U.K

Online Publication Date: 01 November 1989
To cite this Article: Chen, S., Billings, S. A. and Luo, W. (1989) 'Orthogonal least squares methods and their application to non-linear system identification', International Journal of Control, 50:5, 1873 - 1896
To link to this article: DOI: 10.1080/00207178908953472
URL: http://dx.doi.org/10.1080/00207178908953472

INT. J. CONTROL, 1989, VOL. 50, NO. 5, 1873–1896

Orthogonal least squares methods and their application to non-linear system identification

S. CHEN†, S. A. BILLINGS† and W. LUO†

Identification algorithms based on the well-known linear least squares methods of gaussian elimination, Cholesky decomposition, classical Gram–Schmidt, modified Gram–Schmidt, Householder transformation, Givens method, and singular value decomposition are reviewed. The classical Gram–Schmidt, modified Gram–Schmidt, and Householder transformation algorithms are then extended to combine structure determination, or which terms to include in the model, and parameter estimation in a very simple and efficient manner for a class of multivariable discrete-time non-linear stochastic systems which are linear in the parameters.

1. Introduction

Most systems encountered in the real world are non-linear and since linear models cannot capture the rich dynamic behaviour of limit cycles, bifurcations etc. associated with non-linear systems, it is important to investigate the development of identification procedures for non-linear models. The NARMAX (Non-linear AutoRegressive Moving Average with eXogenous inputs) model which was introduced by Leontaritis and Billings (1985) provides a basis for such a development. When a polynomial expansion of the NARMAX model is selected the model becomes linear-in-the-parameters. Providing the model structure, or which terms to include in the model, has been determined, only the values of the parameters are unknown and the identification can thus be formulated as a standard least squares problem which can be solved using various well-developed numerical techniques. Unfortunately the model structure of real systems is rarely known *a priori* and methods of model structure determination must therefore be developed and included as a vital part of the identification procedure. Because the number of all possible candidate terms can easily run into several thousands even for 'moderately' non-linear systems optimal multiple selection methods (Leontaritis and Billings 1987) are difficult to use and suboptimal methods of structure determination such as stepwise regression (Draper and Smith 1981) become very complicated and time consuming.

An orthogonal algorithm which efficiently combines structure selection and parameter estimation has been derived (Korenberg 1985) and extended to the identification of single-input single-output non-linear stochastic systems (Korenberg *et al.* 1988). A more reliable version of the algorithm has been developed by Billings *et al.* (1988 b) and applied to multi-input.multi-output non-linear stochastic systems (Billings *et al.* 1989 b). Various simulation studies and practical applications have shown that this algorithm provides a simple and powerful means of fitting parsimonious models to real systems. A similar structure selection algorithm incorporating some statistical tests for non-linear models without noise modelling has been reported

Received 16 December 1988.

† Department of Control Engineering, University of Sheffield, Mappin Street, Sheffield, S1 3JD, England, U.K.

by Kortmann and Unbehauen (1988). A slightly different structure determination algorithm using projection matrices (symmetric and idempotent matrices) has been given by Desrochers and Mohseni (1984).

Starting with a review of methods for solving least squares problems, the present study develops structure selection algorithms for the polynomial NARMAX model by modifying and augmenting some well-known techniques of orthogonal decomposition of the regression matrix. It is shown that the orthogonal algorithms developed here (Desrochers and Mohseni 1984, Korenberg *et al.* 1988, Billings *et al.* 1989 b) belong to this type. Advantages and disadvantages of using the different orthogonal decomposition techniques are discussed and a comparison of the resulting structure selection algorithms is given.

2. Non-linear system identification and linear least squares problems

Under some mild assumptions a discrete-time multivariable non-linear stochastic control system with m outputs and r inputs can be described by the NARMAX model (Leontaritis and Billings 1985)

$$y(t) = f(y(t-1), \dots, y(t-n_y), u(t-1), \dots, u(t-n_u), e(t-1), \dots, e(t-n_e)) + e(t) \qquad (1)$$

where

$$y(t) = \begin{bmatrix} y_1(t) \\ \vdots \\ y_m(t) \end{bmatrix}, \quad u(t) = \begin{bmatrix} u_1(t) \\ \vdots \\ u_r(t) \end{bmatrix}, \quad e(t) = \begin{bmatrix} e_1(t) \\ \vdots \\ e_m(t) \end{bmatrix} \qquad (2)$$

are the system output, input, and noise, respectively; n_y, n_u, and n_e are the maximum lags in the output, input, and noise; $\{e(t)\}$ is a zero mean independent sequence; and $f(\cdot)$ is some vector-valued non-linear function. Equation (1) can be decomposed into m scalar equations as follows:

$$\begin{aligned} y_i(t) = f_i(&y_1(t-1), \dots, y_1(t-n_y), \dots, y_m(t-1), \dots, y_m(t-n_y), \\ &u_1(t-1), \dots, u_1(t-n_u), \dots, u_r(t-1), \dots, u_r(t-n_u), \\ &e_1(t-1), \dots, e_1(t-n_e), \dots, e_m(t-1), \dots, e_m(t-n_e)) + e_i(t), \quad i = 1, \dots, m \qquad (3) \end{aligned}$$

A special case of the general NARMAX model (1) is the NARX (Non-linear AutoRegressive with eXogenous inputs) model

$$y(t) = f(y(t-1), \dots, y(t-n_y), u(t-1), \dots, u(t-n_u)) + e(t) \qquad (4)$$

or

$$\begin{aligned} y_i(t) = f_i(y_1(t-1), \dots, y_1(t-n_y), \dots, y_m(t-1), \dots, y_m(t-n_y), u_1(t-1), \dots, \\ u_1(t-n_u), \dots, u_r(t-1), \dots, u_r(t-n_u)) + e_i(t), \quad i = 1, \dots, m \qquad (5) \end{aligned}$$

In reality the non-linear form of $f_i(\cdot)$ in (5) is generally unknown. Any continuous $f_i(\cdot)$, however, can be aribtrarily well approximated by polynomial models (Chen and Billings 1989). Expanding $f_i(\cdot)$ as a polynomial of degree l gives the representation

$$\begin{aligned} y_i(t) = \theta_0^{(i)} + \sum_{i_1=1}^{n} \theta_{i_1}^{(i)} x_{i_1}(t) + \sum_{i_1=1}^{n} \sum_{i_2=i_1}^{n} \theta_{i_1 i_2}^{(i)} x_{i_1}(t) x_{i_2}(t) + \dots \\ + \sum_{i_1=1}^{n} \dots \sum_{i_l=i_{l-1}}^{n} \theta_{i_1 \dots i_l}^{(i)} x_{i_1}(t) \dots x_{i_l}(t) + e_i(t), \quad i = 1, \dots, m \qquad (6) \end{aligned}$$

where

$$n = m \times n_y + r \times n_u \tag{7}$$

and

$$\left.\begin{aligned} x_1(t) = y_1(t-1), \quad x_2(t) = y_1(t-2), \quad \ldots, \quad x_{m \times n_y}(t) = y_m(t-n_y) \\ x_{m \times n_y + 1}(t) = u_1(t-1), \quad \ldots, \quad x_n(t) = u_r(t-n_u) \end{aligned}\right\} \tag{8}$$

It is clear that each subsystem model in (6) belongs to the linear regression model

$$z(t) = \sum_{i=1}^{M} p_i(t)\theta_i + \xi(t), \quad t = 1, \ldots, N \tag{9}$$

where N is the data length, the $p_i(t)$ are monomials of $x_1(t)$ to $x_n(t)$ up to degree l—$p_1(t) = 1$ corresponding to a constant term—$\xi(t)$ is some modelling error, and the θ_i are unknown parameters to be estimated. In linear regression analysis, $z(t)$ is known as the dependent variable and the $p_i(t)$ are often referred to as regressors or predictors. Equation (9) can be written in the matrix form

$$\mathbf{z} = \mathbf{P}\mathbf{\Theta} + \mathbf{\Xi} \tag{10}$$

with

$$\mathbf{z} = \begin{bmatrix} z(1) \\ \vdots \\ z(N) \end{bmatrix}, \quad \mathbf{P} = [\mathbf{p}_1 \quad \cdots \quad \mathbf{p}_M], \quad \mathbf{\Theta} = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_M \end{bmatrix}, \quad \mathbf{\Xi} = \begin{bmatrix} \xi(1) \\ \vdots \\ \xi(N) \end{bmatrix} \tag{11}$$

and

$$\mathbf{p}_i = \begin{bmatrix} p_i(1) \\ \vdots \\ p_i(N) \end{bmatrix}, \quad i = 1, \ldots, M \tag{12}$$

In reality each subsystem in (6) may involve only a few significant terms which adequately characterize the system dynamics. If the significant terms are known *a priori* and only *they* are used to form the regression matrix $\mathbf{P}$, a linear least squares (LS) problem can be defined as follows:

$$\text{find parameter estimate } \hat{\mathbf{\Theta}} \text{ which minimizes } \|\mathbf{z} - \mathbf{P}\mathbf{\Theta}\| \quad \text{(LS)}$$

where $\|\cdot\|$ is the euclidean norm. It is well-known that the solution to this problem satisfies the normal equation

$$\mathbf{P}^{\mathrm{T}}\mathbf{P}\mathbf{\Theta} = \mathbf{P}^{\mathrm{T}}\mathbf{z} \tag{13}$$

where $\mathbf{P}^{\mathrm{T}}\mathbf{P}$ is called the information matrix. Several numerical methods of solving the least squares problem (LS) are summarized in § 3.

Because the system structure is generally unknown at the beginning of the identification, the experimenter is often forced to consider the full model set, that is all possible terms in (6). The determination of the structure or which terms to include in the final model is essential since a full model set can easily involve an excessive number of terms, most of which may be redundant and should be removed. A parsimonious model is highly desirable if the model is to be employed in controller design, prediction, and other applications. Let $\mathbf{P}$ represent the full model set. The combined problem of structure selection and parameter estimation (SSPE) can then be stated as follows:

$$\left.\begin{array}{l}\text{Select a subset } \mathbf{P}_s \text{ of } \mathbf{P} \text{ and find the corresponding parameter}\\ \text{estimate } \hat{\mathbf{\Theta}}_s \text{ which adequately fits the data}\end{array}\right\} \quad \text{(SSPE)}$$

One possible approach to the above problem is to use some optimal multiple selection methods based on the theory of hypothesis testing (Leontaritis and Billings 1987). Because the number of all the possible terms can easily become excessively large it is very difficult to attain the optimal solution since this would involve examining all the possible subset models. Some suboptimal methods have to be employed and § 4 considers a class of suboptimal algorithms based on the orthogonal decomposition of **P**.

So far only a polynomial expansion of the NARX model (4) has been discussed. If the same expansion is applied to the NARMAX model (1) a similar linear-in-the-parameters expression for (10) is obtained. Unlike the polynomial NARX case, however, now not all the columns of **P** can be measured or formed from the measurements directly, and (10) becomes a pseudo-linear regression model. In § 5, the results of § 4 are extended to the polynomial NARMAX model.

3. Review of methods for solving least squares problems

This section reviews numerical methods of solving the least squares problem (LS) defined in § 2. There are three approaches which may be considered competitive for computing $\hat{\mathbf{\Theta}}$ as follows:

(*a*) solve the normal equation by gaussian elimination or by forming the Cholesky decomposition of $\mathbf{P}^{\mathrm{T}}\mathbf{P}$;

(*b*) form an orthogonal decomposition of **P**;

(*c*) form a singular value decomposition of **P**.

Each of these approaches has advantages. If $\mathbf{P}^{\mathrm{T}}\mathbf{P}$ can be formed accurately, (*a*) offers the most economical way of computing $\hat{\mathbf{\Theta}}$ at about half the cost of the second approach (*b*), and one-quarter to one-eighth of the cost of the third approach (*c*). The second approach is generally the most accurate. It avoids possible ill-conditioning from the formation of $\mathbf{P}^{\mathrm{T}}\mathbf{P}$. The orthogonal decomposition may be carried out via (modified) Gram–Schmidt orthogonalization, a Householder transformation, or Givens method. Method (*c*) is particularly useful when the rank of **P** is unknown or when **P** is of full rank but is ill-conditioned in an unpredictable way. This method is computationally more expensive. Throughout the discussion in this section it is assumed that **P** has the dimension $N \times M$ with $M \leqslant N$.

3.1. *Methods based on the normal equation*

Assume that **P** is of full rank, then

$$\mathbf{B} = \mathbf{P}^{\mathrm{T}}\mathbf{P} \tag{14}$$

is positive definite. Gaussian elimination reduces **B** to an upper triangular matrix with positive diagonal elements. The reduction is achieved by a series of non-singular elementary row transformations in which multiples of each row of **B** are successively subtracted from the rows below to give zeros below the diagonal. Performing these transformations on the augmented matrix $[\mathbf{B}:\mathbf{P}^{\mathrm{T}}\mathbf{z}]$ gives rise to $[\bar{\mathbf{V}}:\mathbf{d}]$ where

$$\bar{\mathbf{V}} = \begin{bmatrix} \bar{v}_{11} & \bar{v}_{12} & \bar{v}_{13} & \dots & \bar{v}_{1M} \\ 0 & \bar{v}_{22} & \bar{v}_{23} & \dots & \bar{v}_{2M} \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \dots & \bar{v}_{MM} \end{bmatrix}, \quad \mathbf{d} = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_M \end{bmatrix} \tag{15}$$

The elements of $\hat{\mathbf{\Theta}}$ can readily be found by backward substitution

$$\left.\begin{aligned} \hat{\theta}_M &= \frac{d_M}{\bar{v}_{MM}} \\ \hat{\theta}_i &= \frac{d_i - \sum\limits_{k=i+1}^{M} \bar{v}_{ik}\hat{\theta}_k}{\bar{v}_{ii}}, \quad i = M-1, \dots, 1 \end{aligned}\right\} \tag{16}$$

If elementary row transformations are processed further, $\mathbf{B}$ can be reduced to the identity matrix $\mathbf{I}$ and the same transformations performed on $[\mathbf{I}:\mathbf{P}^{\mathrm{T}}\mathbf{z}]$ will lead to $[\mathbf{B}^{-1}:\mathbf{\Theta}]$. This is known as Jordan elimination.

The Cholesky decomposition method expresses $\mathbf{B}$ uniquely in the form

$$\mathbf{B} = \mathbf{U}^{\mathrm{T}}\mathbf{U} \tag{17}$$

where $\mathbf{U}$ is an upper triangular matrix with positive diagonal elements; $\hat{\mathbf{\Theta}}$ can then be found by solving two triangular systems

$$\left.\begin{aligned} \mathbf{U}^{\mathrm{T}}\mathbf{q} &= \mathbf{P}^{\mathrm{T}}\mathbf{z} \\ \mathbf{U}\mathbf{\Theta} &= \mathbf{q} \end{aligned}\right\} \tag{18}$$

using the forward and backward substitution algorithms. To avoid square root calculations in forming $\mathbf{U}$, the information matrix $\mathbf{B}$ can alternatively be decomposed into

$$\mathbf{B} = \mathbf{A}^{\mathrm{T}}\mathbf{D}\mathbf{A} \tag{19}$$

where $\mathbf{A}$ is an upper triangular matrix with unity diagonal elements and $\mathbf{D}$ is a positive diagonal matrix (Seber 1977). Denote $\mathbf{L} = \mathbf{A}^{\mathrm{T}}$ and $\mathbf{V} = \mathbf{D}\mathbf{A}$ then

$$\mathbf{B} = \mathbf{L}\mathbf{V} \tag{20}$$

The elements of $\mathbf{L}$ and $\mathbf{V}$ may be determined in M successive steps, and in each step a row of $\mathbf{V}$ and a column of $\mathbf{L}$ are computed

$$\left.\begin{aligned} &\left.\begin{aligned} v_{1j} &= b_{1j}, \quad j = 1, \dots, M \\ l_{j1} &= \frac{b_{j1}}{v_{11}}, \quad j = 2, \dots, M \end{aligned}\right\} \\ &\left.\begin{aligned} v_{ij} &= b_{ij} - \sum_{k=1}^{i-1} l_{ik}v_{kj}, \quad j = i, i+1, \dots, M \\ l_{ji} &= \frac{b_{ji} - \sum\limits_{k=1}^{i-1} l_{jk}v_{ki}}{v_{ii}}, \quad j = i+1, \dots, M \end{aligned}\right\} i = 2, \dots, M \end{aligned}\right\} \tag{21}$$

If $\mathbf{P}^T\mathbf{P}$ can be formed accurately, the methods based on $\mathbf{P}^T\mathbf{P}$ are computationally the cheapest to implement. Forming $\mathbf{P}^T\mathbf{P}$ however introduces round-off errors; and if $\mathbf{P}$ is ill-conditioned, that is a small change in the elements of $\mathbf{P}$ can cause large changes in $(\mathbf{P}^T\mathbf{P})^{-1}$ and hence $\hat{\mathbf{\Theta}} = (\mathbf{P}^T\mathbf{P})^{-1}\mathbf{P}^T\mathbf{z}$, any errors in the formation of $\mathbf{P}^T\mathbf{P}$ may have a serious effect on the stability of the least squares solution. Furthermore, round-off errors accumulate in the process of solving $\hat{\mathbf{\Theta}}$ and this makes the situation even worse. The problem of ill-conditioning frequently occurs in polynomial non-linear models where the columns of $\mathbf{P}$ can often be highly correlated. As an illustration, consider the example of a polynomial model with a single variable up to kth-degree (Seber 1977, p. 214)

$$z(t) = \sum_{i=0}^{k} \theta_i x^i(t) + e(t), \quad t = 1, \dots, N \tag{22}$$

Assume that $x(t)$ is distributed approximately uniformly on [0, 1]; then for large N it can be shown that $\mathbf{P}^T\mathbf{P}$ is approximately equal to the $(k+1) \times (k+1)$ principal minor of the so-called Hilbert matrix

$$\tilde{\mathbf{H}} = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{3} & \cdots \\ \frac{1}{2} & \frac{1}{3} & \frac{1}{4} & \cdots \\ \frac{1}{3} & \frac{1}{4} & \frac{1}{5} & \cdots \\ \vdots & \vdots & \vdots & \end{bmatrix} \tag{23}$$

multiplied by N. It is well-known that $\tilde{\mathbf{H}}$ is very ill-conditioned. For example, let $k = 9$ and $\tilde{\mathbf{H}}_{10}$ be the 10×10 principal minor of $\tilde{\mathbf{H}}$, then the inverse of $\tilde{\mathbf{H}}_{10}$ has elements of magnitude 3×10^{10}. Thus a small error of 10^{-10} in one element of $\mathbf{P}^T\mathbf{z}$ will lead to an error of about 3 in an element of $\hat{\mathbf{\Theta}}$.

A measure of the ill-conditioning of $\mathbf{P}$ is its condition number $\kappa[\mathbf{P}]$ which is defined as

$$\kappa[\mathbf{P}] = \text{ratio of the largest to smallest non-zero singular value of } \mathbf{P} \tag{24}$$

The singular values of $\mathbf{P}$ are the non-negative square roots of the eigenvalues of $\mathbf{P}^T\mathbf{P}$. Other definitions can also be used. Because

$$\kappa[\mathbf{P}^T\mathbf{P}] = \kappa^2[\mathbf{P}] \tag{25}$$

and $\kappa > 1$, $\mathbf{P}^T\mathbf{P}$ is more ill-conditioned than $\mathbf{P}$. Equation (25) indicates that, using k-digit binary arithmetic, we will not be able to obtain even an approximate solution to the normal equation (13) unless $\kappa[\mathbf{P}] \leqslant 2^{k/2}$ (Björck 1967). This is clearly an unsatisfactory feature of the methods based on the normal equation. Therefore unless $\kappa[\mathbf{P}]$ is of moderate magnitude and $\mathbf{P}^T\mathbf{P}$ can be formed accurately it is better not to form $\mathbf{P}^T\mathbf{P}$ at all, and methods based on the orthogonal decomposition of $\mathbf{P}$ are better alternatives. Although some authors (Golub 1969, Wilkinson 1974) have pointed out that the effect of $\kappa^2[\mathbf{P}]$ cannot be avoided entirely, solving least squares problems by forming an orthogonal decomposition of $\mathbf{P}$ is generally safer and more accurate than utilizing $\mathbf{P}^T\mathbf{P}$.

3.2. *Methods based on orthogonal decomposition of the regression matrix*

As shown in § 3.1 actually forming and solving the normal equation (13) numerically has serious drawbacks because $\mathbf{P}^T\mathbf{P}$ is often so ill-conditioned that the answer obtained is quite inaccurate. Now let $\mathbf{F} = \mathbf{PS}$ where $\mathbf{S}$ is an $M \times M$ non-singular matrix. From (13) it follows that

$$\mathbf{F}^{\mathrm{T}}\mathbf{P}\boldsymbol{\Theta} = \mathbf{F}^{\mathrm{T}}\mathbf{z} \tag{26}$$

This equation can be used instead of the normal equation for solving for $\hat{\boldsymbol{\Theta}}$ and $\mathbf{F}$ can be chosen in such a way that

$$\kappa[\mathbf{F}^{\mathrm{T}}\mathbf{P}] = \kappa[\mathbf{P}] \tag{27}$$

Indeed, since $\mathbf{P}$ is of full rank, it can be factorized as

$$\mathbf{P} = \mathbf{QR} \tag{28}$$

whered $\mathbf{Q}$ is an $N \times M$ orthogonal matrix ($\mathbf{Q}^{\mathrm{T}}\mathbf{Q} = \mathbf{I}$), that is the columns of $\mathbf{Q}$ are orthonormal, and $\mathbf{R}$ is an $M \times M$ upper triangular matrix. Choosing $\mathbf{F} = \mathbf{PR}^{-1} = \mathbf{Q}$, the matrix $\mathbf{F}^{\mathrm{T}}\mathbf{P} = \mathbf{Q}^{\mathrm{T}}\mathbf{QR} = \mathbf{R}$ is triangular and (26) can easily be solved by backward substitution. Moreover, the condition (27) is satisfied since

$$\kappa[\mathbf{F}^{\mathrm{T}}\mathbf{P}] = \kappa[\mathbf{R}] = \kappa[\mathbf{QR}] = \kappa[\mathbf{P}] \tag{29}$$

and $\mathbf{R}^{\mathrm{T}}\mathbf{R}$ is in fact the Cholesky decomposition of $\mathbf{P}^{\mathrm{T}}\mathbf{P}$ (17). The factorization of (28) can be obtained in several ways and these are summarized in the following.

Classical Gram–Schmidt

In the actual computation it is preferable to use a different factorization of $\mathbf{P}$ rather than (28) in order to avoid computing square roots. The factorization that corresponds to the Cholesky decomposition of (19) is

$$\mathbf{P} = \mathbf{WA} \tag{30}$$

where

$$\mathbf{A} = \begin{bmatrix} 1 & \alpha_{12} & \alpha_{13} & \cdots & \alpha_{1M} \\ & 1 & \alpha_{23} & \cdots & \alpha_{2M} \\ & & \ddots & \ddots & \vdots \\ & & & 1 & \alpha_{M-1M} \\ & & & & 1 \end{bmatrix} \tag{31}$$

is an $M \times M$ unit upper triangular matrix and

$$\mathbf{W} = [\mathbf{w}_1 \quad \ldots \quad \mathbf{w}_M] \tag{32}$$

is an $N \times M$ matrix with orthogonal columns that satisfy

$$\mathbf{W}^{\mathrm{T}}\mathbf{W} = \mathbf{D} \tag{33}$$

and $\mathbf{D}$ is the positive diagonal matrix in (19).

The classical Gram–Schmidt (CGS) procedure computes $\mathbf{A}$ one column at a time and orthogonalizes $\mathbf{P}$ as follows: at the kth stage make the kth column orthogonal to each of the $k-1$ previously orthogonaliazed columns and repeat the operations for

$k = 2, \dots, M$. The computational procedure is represented as

$$\left.\begin{aligned} &\mathbf{w}_1 = \mathbf{p}_1 \\ &\left.\begin{aligned} \alpha_{ik} &= \frac{\langle \mathbf{w}_i, \mathbf{p}_k \rangle}{\langle \mathbf{w}_i, \mathbf{w}_i \rangle}, \quad 1 \leqslant i < k \\ \mathbf{w}_k &= \mathbf{p}_k - \sum_{i=1}^{k-1} \alpha_{ik} \mathbf{w}_i \end{aligned}\right\} k = 2, \dots, M \end{aligned}\right\} \tag{34}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product, that is,

$$\langle \mathbf{w}_i, \mathbf{p}_k \rangle = \mathbf{w}_i^{\mathrm{T}} \mathbf{p}_k = \sum_{i=1}^{N} w_i(t) p_k(t) \tag{35}$$

Define

$$\mathbf{g} = \mathbf{D}^{-1} \mathbf{W}^{\mathrm{T}} \mathbf{z} \tag{36}$$

or

$$g_i = \frac{\langle \mathbf{w}_i, \mathbf{z} \rangle}{\langle \mathbf{w}_i, \mathbf{w}_i \rangle}, \quad i = 1, \dots, M \tag{37}$$

Then the parameter estimate $\hat{\boldsymbol{\Theta}}$ is readily computed from

$$\mathbf{A} \boldsymbol{\Theta} = \mathbf{g} \tag{38}$$

using backward substitution.

Modified Gram–Schmidt

If is well-known that the CGS procedure is very sensitive to round-off errors. The experimental results (Rice, 1966) and the theoretical analysis (Björck 1967) indicate that if $\mathbf{P}$ is ill-conditioned, using the CGS procedure, the computed columns of $\mathbf{W}$ will soon lose their orthogonality and reorthogonalization will be needed. On the other hand, the modified Gram–Schmidt (MGS) procedure is numerically superior.

The MGS procedure calculates $\mathbf{A}$ one row at a time and orthogonalizes $\mathbf{P}$ as follows: at the kth stage make the columns subscripted $k+1, \dots, M$ orthogonal to the kth column and repeat the operations for $k = 1, \dots, M-1$ (e.g., Björck 1967). Specifically, denoting $\mathbf{p}_i^{(0)} = \mathbf{p}_i$, $i = 1, \dots, M$, then

$$\left.\begin{aligned} &\left.\begin{aligned} &\mathbf{w}_k = \mathbf{p}_k^{(k-1)} \\ &\alpha_{ki} = \frac{\langle \mathbf{w}_k, \mathbf{p}_i^{(k-1)} \rangle}{\langle \mathbf{w}_k, \mathbf{w}_k \rangle}, \quad i = k+1, \dots, M \\ &\mathbf{p}_i^{(k)} = \mathbf{p}_i^{(k-1)} - \alpha_{ki} \mathbf{w}_k, \quad i = k+1, \dots, M \end{aligned}\right\} k = 1, 2, \dots, M-1 \\ &\mathbf{w}_M = \mathbf{p}_M^{(M-1)} \end{aligned}\right\} \tag{39}$$

The elements of $\mathbf{g}$ are computed by transforming $\mathbf{z}^{(0)} = \mathbf{z}$ in a similar way

$$\left.\begin{aligned} g_k &= \frac{\langle \mathbf{w}_k, \mathbf{z}^{(k-1)} \rangle}{\langle \mathbf{w}_k, \mathbf{w}_k \rangle} \\ \mathbf{z}^{(k)} &= \mathbf{z}^{(k-1)} - g_k \mathbf{w}_k \end{aligned}\right\} k = 1, 2, \dots, M \tag{40}$$

The CGS and MGS algorithms have distinct differences in computational behaviour.

The MGS procedure is more accurate and more stable than the CGS procedure. This is particularly remarkable since both methods perform basically the same operations, only in a different sequence. Indeed, if there were not computer round-off errors they would produce the same set of $\mathbf{w}_i$ with the same number of operations.

Householder transformation

An equivalent decomposition to (28) can be obtained by augmenting $\mathbf{Q}$ with $N-M$ further orthonormal columns to make up a full set of N orthonormal vectors for an N-dimensional euclidean space thus

$$\tilde{\mathbf{Q}} = [\mathbf{Q} : \tilde{\mathbf{q}}_{M+1} \dots \tilde{\mathbf{q}}_N] = [\mathbf{Q}_M : \tilde{\mathbf{Q}}_{N-M}] \tag{41}$$

Then

$$\mathbf{P} = \tilde{\mathbf{Q}}\tilde{\mathbf{R}} = \tilde{\mathbf{Q}}\begin{bmatrix}\mathbf{R}\\ 0\end{bmatrix} \tag{42}$$

where $\mathbf{R}$ is the $M \times M$ upper triangular matrix in (28), and $\tilde{\mathbf{Q}}^{\mathrm{T}}$ can be used to triangularize $\mathbf{P}$. If $\tilde{\mathbf{Q}}^{\mathrm{T}}\mathbf{z}$ is partitioned into

$$\tilde{\mathbf{Q}}^{\mathrm{T}}\mathbf{z} = \begin{bmatrix}\mathbf{z}_1\\ \mathbf{z}_2\end{bmatrix}\begin{matrix}\}M\\ \}N-M\end{matrix} \tag{43}$$

we have

$$\|\mathbf{z} - \mathbf{P}\Theta\| = \|\tilde{\mathbf{Q}}^{\mathrm{T}}(\mathbf{z} - \mathbf{P}\Theta)\| = \|\mathbf{z}_1 - \mathbf{R}\Theta\| + \|\mathbf{z}_2\| \tag{44}$$

The least squares estimates can therefore be obtained by solving the triangular system

$$\mathbf{R}\Theta = \mathbf{z}_1 \tag{45}$$

and the sum of the squares of the residuals is given by $\|\mathbf{z}_2\|^2$.

The Householder method triangularizes the $N \times (M+1)$ augmented matrix

$$\tilde{\mathbf{P}} = [\mathbf{P} : \mathbf{z}] = \begin{bmatrix}\tilde{p}_{11} & \cdots & \tilde{p}_{1M} & \tilde{p}_{1(M+1)}\\ \tilde{p}_{21} & \cdots & \tilde{p}_{2M} & \tilde{p}_{2(M+1)}\\ \vdots & & \vdots & \vdots\\ \tilde{p}_{N1} & \cdots & \tilde{p}_{NM} & \tilde{p}_{N(M+1)}\end{bmatrix} \tag{46}$$

to give

$$\tilde{\mathbf{P}}^{(M)} = \begin{bmatrix}\mathbf{R} & \mathbf{z}_1\\ 0 & \mathbf{z}_2\end{bmatrix} \tag{47}$$

using a series of Householder transformations

$$\mathbf{H}^{(k)} = \mathbf{I} - \beta^{(k)}\mathbf{v}^{(k)}(\mathbf{v}^{(k)})^{\mathrm{T}}, \quad k = 1, \dots, M \tag{48}$$

Here the $\mathbf{v}^{(k)}$ are N-vectors with certain properties, the $\mathbf{H}^{(k)}$ are $N \times N$ symmetric and orthogonal matrices, that is, $(\mathbf{H}^{(k)})^{\mathrm{T}} = \mathbf{H}^{(k)}$ and $(\mathbf{H}^{(k)})^{\mathrm{T}}\mathbf{H}^{(k)} = \mathbf{I}$. Furthermore,

$$\tilde{\mathbf{Q}}^{\mathrm{T}} = \mathbf{H}^{(M)}\mathbf{H}^{(M-1)} \dots \mathbf{H}^{(1)} \tag{49}$$

Denoting

$$\tilde{\mathbf{P}}^{(k)} = \mathbf{H}^{(k)}\tilde{\mathbf{P}}^{(k-1)}, \quad k = 1, \ldots, M \quad \text{and} \quad \tilde{\mathbf{P}}^{(0)} = \tilde{\mathbf{P}} \tag{50}$$

the kth Householder transformation is explcitly defined as (e.g., Golub 1965, Bierman 1977, Chap. IV)

$$\left.\begin{aligned} \sigma^{(k)} &= \left(\sum_{i=k}^{N} (\tilde{p}_{ik}^{k-1})^2\right)^{1/2} \\ \beta^{(k)} &= \frac{1}{\sigma^{(k)}(\sigma^{(k)} + |\tilde{p}_{kk}^{(k-1)}|)} \\ v_i^{(k)} &= \begin{cases} 0, & \text{for } i < k \\ \tilde{p}_{kk}^{(k-1)} + \operatorname{sgn}[\tilde{p}_{kk}^{(k-1)}]\sigma^{(k)}, & \text{for } i = k \\ \tilde{p}_{ik}^{(k-1)}, & \text{for } i > k \end{cases} \end{aligned}\right\} \tag{51}$$

and

$$\tilde{\mathbf{P}}^{(k)} = \tilde{\mathbf{P}}^{(k-1)} - \mathbf{v}^{(k)}(\beta^{(k)}(\mathbf{v}^{(k)})^{\mathrm{T}}\tilde{\mathbf{P}}^{(k-1)}) \tag{52}$$

Givens method

A Givens transformation rotates two row vectors

$$\left.\begin{matrix} 0, \ldots, 0, r_i, r_{i+1}, \ldots, r_k, \ldots \\ 0, \ldots, 0, \beta_i, \beta_{i+1}, \ldots, \beta_k, \ldots \end{matrix}\right\} \tag{53}$$

resulting in two new row vectors

$$\left.\begin{matrix} 0, \ldots, 0, \tilde{r}_i, \tilde{r}_{i+1}, \ldots, \tilde{r}_k, \ldots \\ 0, \ldots, 0, 0, \tilde{\beta}_{i+1}, \ldots, \tilde{\beta}_k, \ldots \end{matrix}\right\} \tag{54}$$

where

$$\left.\begin{aligned} \tilde{r}_k &= c r_k + s\beta_k \\ \tilde{\beta}_k &= -s r_k + c\beta_k \\ \tilde{r}_i &= (r_i^2 + \beta_i^2)^{1/2} \\ c &= \frac{r_i}{\tilde{r}_i} \\ s &= \frac{\beta_i}{\tilde{r}_i} \end{aligned}\right\} \tag{55}$$

There are two ways of applying a sequence of Givens transformations to triangularize $\tilde{\mathbf{P}}$ of (46).

For $j = 1, \ldots, M$, the first method rotates the jth row successively with the $(j+1)$th, ..., Nth row so that the last $N-j$ elements in the jth column are reduced to zeros. This leads to the triangular system, of (47). Here $\mathbf{z}_2$ can be transformed further to $(\delta, 0, \ldots, 0)^{\mathrm{T}}$, then the sum of the squares of the residuals is simply δ^2.

The second method processes one row of $\tilde{\mathbf{P}}$ at a time as follows:

$$\begin{bmatrix} \mathbf{R}^{(i-1)} & \mathbf{z}^{(t-1)} \\ & \delta^{(t-1)} \\ \tilde{p}_{t1}, \ldots, \tilde{p}_{tM} & \tilde{p}_{t(M+1)} \end{bmatrix} \rightarrow \begin{bmatrix} \mathbf{R}^{(t)} & \mathbf{z}_1^{(t)} \\ & \delta^{(t)} \\ 0, \ldots, 0 & 0 \end{bmatrix} \tag{56}$$

for $t = 1, \ldots, N$. The initial $\mathbf{R}^{(0)}$, $\mathbf{z}_1^{(0)}$, and $\delta^{(0)}$ can be taken as zeros. It is readily seen that this way of implementing the Givens method leads to on-line identification algorithms (e.g., Goodwin and Payne 1977, Result 7.2.2). Similar to the Gram–Schmidt procedure, the computation of square roots in (55) can be avoided (Gentleman 1973).

Comparisons

MGS and Householder algorithms are highly competitive. The MGS algorithm is easier to program but requires slightly more computation. It seems that the MGS algorithm is slightly more accurate than the Householder algorithm (Jordan 1968, Wampler 1970).

The reduction of $\mathbf{P}$ to the upper triangular form requires approximately $NM^2 - M^3/2$ multiplications and M square roots for the Householder method; and about $2NM^2$ multiplications and NM square roots for the Givens method if transformations are carried out in the form of (55)—see the work by Seber (1977). However, the modified Givens method (Gentleman 1973) requires no square roots and reduces multiplications to only three quarters of $2NM^2$. Furthermore, the Givens method processes one row at a time and has important on-line implementations.

3.3. *Singular value decomposition of the regression matrix*

If the rank of $\mathbf{P}$ is less than M the least squares problem no longer has a unique solution. Many of the methods discussed in §§ 3.1 and 3.2 can be adapted to handle this. The singular value decomposition, however, offers a general solution to the least squares problem.

Assume that the rank of $\mathbf{P}$ is η ($\leqslant M$). According to the singular value decomposition theorem (e.g., Golub and Reinsch 1970) $\mathbf{P}$ can be factorized as

$$\mathbf{P} = \tilde{\mathbf{U}}\mathbf{S}\tilde{\mathbf{V}}^{\mathrm{T}} \tag{57}$$

where $\tilde{\mathbf{U}}$ is an $N \times M$ orthogonal matrix consisting of M orthonormalized eigenvectors associated with the M largest eigenvalues of $\mathbf{P}\mathbf{P}^{\mathrm{T}}$, $\tilde{\mathbf{V}}$ is an $M \times M$ orthogonal matrix consisting of the orthonormalized eigenvectors of $\mathbf{P}^{\mathrm{T}}\mathbf{P}$, and

$$\mathbf{S} = \mathrm{diag}\,[s_1, \ldots, s_M] \tag{58}$$

has the singular values of $\mathbf{P}$ as its diagonal elements. The diagonal elements of $\mathbf{S}$ are in the order

$$s_1 \geqslant s_2 \geqslant \ldots \geqslant s_M \geqslant 0 \tag{59}$$

If $\eta < M$, $s_{\eta+1} = \ldots = s_M = 0$. There are alternative representations to (57)—e.g., as given by Hanson and Lawson (1969)—however, (57) is convenient for computational purposes.

The pseudo-inverse of $\mathbf{P}$ is defined as

$$\mathbf{P}^{+} = \tilde{\mathbf{V}}\mathbf{S}^{+}\tilde{\mathbf{U}}^{\mathrm{T}} \tag{60}$$

where

$$\mathbf{S}^+ = \operatorname{diag}\,[s_1^+, \dots, s_M^+] \tag{61}$$

and

$$s_i^+ = \begin{cases} \dfrac{1}{s_i}, & \text{for } s_i > 0 \\ 0, & \text{for } s_i = 0 \end{cases} \tag{62}$$

Then

$$\hat{\mathbf{\Theta}} = \mathbf{P}^+\mathbf{z} = \tilde{\mathbf{V}}\mathbf{S}^+(\tilde{\mathbf{U}}^{\mathrm{T}}\mathbf{z}) \tag{63}$$

is a solution of the least squares problem. Details of an algorithm for computing $\hat{\mathbf{\Theta}}$ using singular value decomposition can be found in work by Golub (1969), and Golub and Reinsch (1970).

4. Selection of subset methods using orthogonal algorithms

When identifying non-linear systems with an unknown structure, it is important to avoid losing significant terms which must be included in the final model, and consequently the experimenter is forced to start with large values of n_y and n_u, and a high polynomial degree i in (4). The number of columns in $\mathbf{P}$ is therefore often very large, even thousands, perhaps. For example, even in the single-input single-output (SISO) case, if $n_y = n_u = 10$ and $l = 3$, $\mathbf{P}$ has 1771 columns. Previous experience has shown that provided the significant terms in the model can be detected models with about 10 terms are usually sufficient to capture the dynamics of highly non-linear SISO processes (Billings 1986, Billings and Fadzil 1985, Billings *et al.* 1988 a, Billings *et al.* 1989 a). Throughout the rest of the discussion, $\mathbf{P}$ will be used to represent the full model set of a (sub)system so that $M \leqslant N$ becomes unnecessary. This section considers the combined problem of structure selection and parameter estimation (SSPE) defined in § 2. It is shown that by augmenting the orthogonal decomposition techniques of § 3.2, simple and efficient algorithms can be derived that determine $\mathbf{P}_s$, a subset of $\mathbf{P}$, in a forward-regression manner by choosing one column of $\mathbf{P}$ for which the sum of squares of residuals is maximally reduced at a time.

4.1. *Using the CGS procedure*

Assume that $\mathbf{P}_s$ has M_s ($M_s < M$ and $M_s \leqslant N$) columns. Factorize $\mathbf{P}_s$ into $\mathbf{W}_s\mathbf{A}_s$ as in (30) where $\mathbf{W}_s$ is an $N \times M_s$ matrix consisting of M_s orthogonal columns and $\mathbf{A}_s$ is an $M_s \times M_s$ unit upper triangular matrix. The residuals are defined by

$$\hat{\mathbf{\Xi}} = \begin{bmatrix} \xi(1) \\ \vdots \\ \xi(N) \end{bmatrix} = \mathbf{z} - \mathbf{P}_s\hat{\mathbf{\Theta}}_s = \mathbf{z} - (\mathbf{P}_s\mathbf{A}_s^{-1})(\mathbf{A}_s\hat{\mathbf{\Theta}}_s) = \mathbf{z} - \mathbf{W}_s\mathbf{g}_s \tag{64}$$

Equation (64) can be rewritten as

$$\mathbf{z} = \mathbf{W}_s\mathbf{g}_s + \hat{\mathbf{\Xi}} \tag{65}$$

The sum of squares of the dependent variable $\mathbf{z}$ is therefore

$$\langle \mathbf{z}, \mathbf{z} \rangle = \sum_{i=1}^{M_s} g_i^2 \langle \mathbf{w}_i, \mathbf{w}_i \rangle + \langle \hat{\mathbf{\Xi}}, \hat{\mathbf{\Xi}} \rangle \tag{66}$$

The error reduction ratio due to $\mathbf{w}_i$ is thus defined as the proportion of the dependent variable variance explained by $\mathbf{w}_i$

$$[\mathrm{err}]_i = \frac{g_i^2 \langle \mathbf{w}_i, \mathbf{w}_i \rangle}{\langle \mathbf{z}, \mathbf{z} \rangle} \tag{67}$$

Equation (67) suggests a way of computing $\mathbf{W}_s$ (and hence $\mathbf{P}_s$) from $\mathbf{P}$ by the CGS procedure. At the ith stage, by interchanging the i to M columns of $\mathbf{P}$ we can select a $\mathbf{p}_i$ which gives the largest $[\mathrm{err}]_i$ when orthogonalized into $\mathbf{w}_i$. The detailed procedure is as follows.

At this stage, for $i = 1, \ldots, M$, denote $\mathbf{w}_1^{(i} = \mathbf{p}_i$ and compute

$$g_i^{(i} = \frac{\langle \mathbf{w}_1^{(i}, \mathbf{z} \rangle}{\langle \mathbf{w}_1^{(i}, \mathbf{w}_1^{(i} \rangle}, \quad [\mathrm{err}]_1^{(i} = \frac{(g_1^{(i)})^2 \langle \mathbf{w}_1^{(i}, \mathbf{w}_1^{(i} \rangle}{\langle \mathbf{z}, \mathbf{z} \rangle} \tag{68}$$

Assume that $[\mathrm{err}]_1^{(i} = \max \{[\mathrm{err}]_1^{(i}, 1 \leqslant j \leqslant M\}$. Then $\mathbf{w}_1 = \mathbf{w}_1^{(j}$ $(= \mathbf{p}_j)$ is selected as the first column of $\mathbf{W}_s$ together with the first element of $\mathbf{g}_s$, $g_1 = g_1^{(j}$, and $[\mathrm{err}]_1 = [\mathrm{err}]_1^{(j}$.

At the second stage, for $i = 1, \ldots, M$ and $i \neq j$, compute

$$\left.\begin{aligned} \alpha_{12}^{(i} &= \frac{\langle \mathbf{w}_1, \mathbf{p}_i \rangle}{\langle \mathbf{w}_1, \mathbf{w}_1 \rangle}, \quad \mathbf{w}_2^{(i} = \mathbf{p}_i - \alpha_{12}^{(i} \mathbf{w}_1 \\ g_2^{(i} &= \frac{\langle \mathbf{w}_2^{(i}, \mathbf{z} \rangle}{\langle \mathbf{w}_2^{(i}, \mathbf{w}_2^{(i} \rangle}, \quad [\mathrm{err}]_2^{(i} = \frac{(g_2^{(i)})^2 \langle \mathbf{w}_2^{(i}, \mathbf{w}_2^{(i} \rangle}{\langle \mathbf{z}, \mathbf{z} \rangle} \end{aligned}\right\} \tag{69}$$

Assume that $[\mathrm{err}]_2^{(k} = \max \{[\mathrm{err}]_2^{(i}, 1 \leqslant i \leqslant M$ and $i \neq j\}$. Then $\mathbf{w}_2 = \mathbf{w}_2^{(k}$ $(= \mathbf{p}_k - \alpha_{12} \mathbf{w}_1)$ is selected as the second column of $\mathbf{W}_s$ together with the second column of $\mathbf{A}_s$, $\alpha_{12} = \alpha_{12}^{(k}$, the second element of $\mathbf{g}_s$, $g_2 = g_2^{(k}$, and $[\mathrm{err}]_2 = [\mathrm{err}]_2^{(k}$.

The selection procedure is continued until the M_sth stage when

$$1 - \sum_{i=1}^{M_s} [\mathrm{err}]_i < \rho \tag{70}$$

where ρ $(0 < \rho \leqslant 1)$ is a desired tolerance. Other criteria can also be used to stop the selection procedure, for example,

$$[\mathrm{err}]_{M_s+1} \times 100 < \text{a tolerance} \quad \text{or} \quad \frac{g_{M_s+1}^2 \langle \mathbf{w}_{M_s+1}, \mathbf{w}_{M_s+1} \rangle}{\langle \mathbf{z}, \mathbf{z} \rangle - \sum_{i=1}^{M_s} g_i^2 \langle \mathbf{w}_i, \mathbf{w}_i \rangle} \times 100 < \text{a tolerance}$$

The subset model parameter estimate $\hat{\mathbf{\Theta}}_s$ can easily be computed from $\mathbf{A}_s \mathbf{\Theta}_s = \mathbf{g}_s$ by backward substitution.

The geometrical interpretation of the above procedure is obvious. At the first stage, the vector $\mathbf{z}$ is projected onto the basis vectors $\{\mathbf{p}_i, i = 1, \ldots, M\}$ (implicitly). Then the scalar measures $[\mathrm{err}]_1^{(i}$ are calculated, and the maximum scalar measure $[\mathrm{err}]_1^{(i} = \max \{[\mathrm{err}]_1^{(i}, 1 \leqslant i \leqslant M\}$ is determined. This leads to the selection of $\mathbf{w}_1$ $(= \mathbf{p}_j)$ as the basis vector of the one-dimensional euclidean space $\mathbf{E}^1 = \mathbf{E}_{(1)}$. At the second stage, all the remaining basis vectors $\{\mathbf{p}_i, i = 1, \ldots, M$ and $i \neq j\}$ are transferred into an $(M - 1)$-dimensional euclidean space, which is orthogonal to $\mathbf{E}_{(1)}$, and the basis vector $\mathbf{w}_2$ for the one-dimensional euclidean space $\mathbf{E}_{(2)}$ is selected. The two-dimensional euclidean space $\mathbf{E}^2$ is then the union of $\mathbf{E}_{(1)}$ and $\mathbf{E}_{(2)}$, having the orthogonal basis vectors $\{\mathbf{w}_1, \mathbf{w}_2\}$. Finally after M_s stages, an M_s-dimensional euclidean space $\mathbf{E}^{M_s}$ is established which has the orthogonal basis vectors $\{\mathbf{w}_1, \mathbf{w}_2, \ldots, \mathbf{w}_{M_s}\}$. It is worth pointing out that the algorithm proposed by Desrochers and Mohseni (1984) is theoretically equivalent to the above procedure. In their algorithm, projection matrices are computed, and this makes the projection of $\mathbf{z}$ onto the basis vectors explicit and the geometrical insight more apparent. By using projection matrices, Desrochers and Mohseni (1984) have also been able to establish some properties of the algorithm. Forming projection matrices explicitly, however, is time consuming and computationally unnecessary.

It can easily be seen that the orthogonalization procedure used by Korenberg (1985), Korenberg *et al.* (1988), Billings *et al.* (1988 b, 1989 b), is the CGS procedure by simply comparing it with (34), and the algorithm of this subsection is in fact identical to the forward-regression orthogonal algorithm given by Billings *et al.* (1988 b, 1989 b). Application of this orthogonal algorithm to the identification of the polynomial NARX model is straightforward since the identification of any subsystem is decoupled from the others. From (66) and (67) it is seen that $1 - \sum_{i=1}^{M_s} [\mathrm{err}]_i$ is the proportion of the unexplained dependent variable variance. The value of ρ determines how many terms will be included in the final (sub)model and hence the complexity of the model. Let ρ_i be the desired tolerance for the ith subsystem. Ideal ρ_i should be closely related to $E[e_i^2(t)]/E[y_i^2(t)]$. Since the latter is not known *a priori*, the appropriate ρ_i may have to be found by trial and error.

The criterion of (70) concerns only the performance of the model (variance of residuals) and does not take into account the model complexity. A criterion that compromises between the performance and complexity of the model is Akaike's information criterion AIC(ϕ)

$$\mathrm{AIC}(\phi) = N \log C(\hat{\Theta}_s) + M_s \phi \tag{71}$$

where

$$C(\hat{\Theta}_s) = \frac{1}{N} \langle \hat{\Xi}, \hat{\Xi} \rangle \tag{72}$$

is the variance of the residuals, and ϕ is the critical value of the chi-square distribution with one degree of freedom for a given significance level. To use this criterion, the user is first required to specify a significance level. Leontaritis and Billings (1987) have pointed out that $\phi = 4$ is a convenient choice and it corresponds to the significance level of 0·0456. AIC(4) provides an alternative criterion to stop the above forward-regression model selection procedure. When the minimum of AIC(4) is reached the selection procedure is terminated. Other statistical criteria can also be employed to stop the selection, and the relationship between these criteria and AIC(ϕ) has been investigated by Söderström (1977), and Leontaritis and Billings (1987).

As mentioned before, the CGS orthogonalization is sensitive to computer round-off errors. Notice, however, that the model selection algorithm in this subsection is not designed to orthogonalize the whole $\mathbf{P}$ which is often ill-conditioned. It selects $\mathbf{P}_s$, usually small subset of $\mathbf{P}$ and typically about 10 columns for SISO systems. Since $\mathbf{P}_s$ contains only significant terms of the system it is usually well-conditioned and the problem of columns of $\mathbf{W}_s$ losing their orthogonality rarely occurs. Indeed this model selection algorithm performed well in many previous applications. Nevertheless, it may be desirable to employ the MGS orthogonalization procedure in a similar model selection algorithm because of its numerical superiority.

Using the CGS procedure to select terms *does* have two important advantages which are worth emphasizing. Storing the matrix $\mathbf{P}$ in the memory of a microcomputer could be a problem because its size is often huge. Notice that each column of $\mathbf{P}$ is a monomial function of the input–output data and the CGS algorithm computes one column of $\mathbf{A}$ and orthogonalizes a column of $\mathbf{P}$ at a time. Every time a column is to be orthogonalized it can be generated quickly from the input–output data. In this way, storing $\mathbf{P}$ can be avoided and only $\mathbf{W}_s$, which is often of modest dimensions, needs to be kept. Implementing the CGS algorithm in a microcomputer should therefore be straightforward. This will not be the case for MGS because storage for the whole $\mathbf{P}$ is required. The way that the CGS orthogonalization operates also makes the algorithm easier to extend to the polynomial NARMAX model where noise modelling is a part of the identification procedure. This is further discussed in § 5.

4.2. *Using the MGS procedure*

The development of a forward-regression orthogonal algorithm using MGS orthogonalization is straightforward. Because the MGS procedure makes the $(k+1)$th, ..., Mth columns orthogonal to the kth column at the kth stage, the matrix $\mathbf{P}$ must be kept in computer memory. The memory space for $\mathbf{A}$, up to M_s rows, must also be provided and this is of course much larger than the space for $\mathbf{A}_s$ required in the algorithm using CGS orthogonalization. Employing the same notation as (39) and (40) leads to the definition of $\mathbf{P}^{(k-1)}$ as

$$\mathbf{P}^{(k-1)} = [\mathbf{w}_1 \dots \mathbf{w}_{k-1} \mathbf{p}_k^{(k-1)} \dots \mathbf{p}_M^{(k-1)}] \tag{73}$$

If some of the columns $\mathbf{p}_k^{(k-1)}, \dots, \mathbf{p}_M^{(k-1)}$ in $\mathbf{P}^{(k-1)}$ haved been interchanged this will still be referred to as $\mathbf{P}^{(k-1)}$ for notational convenience. The forward-regression orthogonal algorithm using the MGS orthogonalization can now be summarized as follows.

At the kth stage, for $i = k, \dots, M$, compute

$$g_k^{(i} = \frac{\langle \mathbf{p}_i^{(k-1)}, \mathbf{z}^{(k-1)} \rangle}{\langle \mathbf{p}_i^{(k-1)}, \mathbf{p}_i^{(k-1)} \rangle}, \quad [\mathrm{err}]_k^{(i} = \frac{(g_k^{(i)})^2 \langle \mathbf{p}_i^{(k-1)}, \mathbf{p}_i^{(k-1)} \rangle}{\langle \mathbf{z}, \mathbf{z} \rangle}$$

Assume that $[\mathrm{err}]_k^{(i} = \max \{[\mathrm{err}]_k^{(i}, k \leqslant i \leqslant M\}$. Then the jth column of $\mathbf{P}^{(k-1)}$ is interchanged with the kth column; the jth column of $\mathbf{A}$ is interchanged up to the $(k-1)$th row with the kth column. The rest of the operations are as indicated in (39) and (40). The selection procedure can be terminated in the same ways as discussed in § 4.1. Notice that $\mathbf{z}^{(M_s)}$ is simply the residual vector and

$$\frac{\langle \mathbf{z}^{(M_s)}, \mathbf{z}^{(M_s)} \rangle}{\langle \mathbf{z}, \mathbf{z} \rangle} = 1 - \sum_{i=1}^{M_s} [\mathrm{err}]_i \tag{74}$$

Here $\mathbf{A}_s$ is the $M_s \times M_s$ principal minor of $\mathbf{A}$.

4.3. *Using the Household transformation*

The Householder transformation method can be employed to derive a forward-regression algorithm. Unfortunately as in the case of the MGS method, the whole of $\mathbf{P}$ must be stored in computer memory. Denote

$$\tilde{\mathbf{P}}^{(0)} = [\mathbf{P} : \mathbf{z}] = [\tilde{\mathbf{p}}_1^{(0)} \dots \tilde{\mathbf{p}}_M^{(0)} : \mathbf{z}^{(0)}] \tag{75}$$

and $\mathbf{R}_k$ the $k \times k$ principal minor of $\mathbf{R}$ where $\mathbf{R}$ is defined in (42).

After $\mathbf{H}^{(i)}, i = 1, \ldots, k-1$ have been successively applied to $\tilde{\mathbf{P}}^{(0)}$, it is transformed to

$$\tilde{\mathbf{P}}^{(k-1)} = \begin{bmatrix} \mathbf{R}_{k-1} & \tilde{\mathbf{p}}_k^{(k-1)} \ldots \tilde{\mathbf{p}}_M^{(k-1)} : \mathbf{z}^{(k-1)} \\ 0 & \end{bmatrix} \tag{76}$$

Two important properties of $\mathbf{H}^{(k)}$ should be noted as follows:

(a) it leaves the first $k-1$ rows of $\tilde{\mathbf{P}}^{(k-1)}$ unchanged;

(b) it leaves the column lengths invariant.

If the process were stopped at the $(k-1)$th stage and a subset model of $k-1$ parameters were chosen, the sum of the squares of residuals would be

$$\sum_{i=k}^{N} [z_i^{(k-1)}]^2 \tag{77}$$

and this is reduced to

$$\sum_{i=k+1}^{N} (z_i^{(k)})^2 = \sum_{i=k}^{N} (z_i^{(k-1)})^2 - (z_k^{(k)})^2 \tag{78}$$

after $\mathbf{H}^{(k)}$ has been applied to $\tilde{\mathbf{P}}^{(k-1)}$. The task is then to choose a column from $\tilde{\mathbf{p}}_k^{(k-1)}, \ldots, \tilde{\mathbf{p}}_M^{(k-1)}$ for which $(z_k^{(k)})^2$ is maximized, and this can be achieved as follows.

Denote $\tilde{\mathbf{p}}_j^{(k-1)} = (\tilde{p}_j^{(k-1)}, \ldots, \tilde{p}_{N_j}^{(k-1)})^{\mathrm{T}}, j = k, \ldots, M$. Compute

$$a_j^{(k)} = \left(\sum_{i=k}^{N} (\tilde{p}_{ij}^{(k-1)})^2 \right)^{1/2}, \quad b_j^{(k)} = \sum_{i=k}^{N} \tilde{p}_{ij}^{(k-1)} z_i^{(k-1)}, \quad \text{for } j = k, \ldots, M$$

Assume that the maximum of

$$\left(z_k^{(k-1)} - (\tilde{p}_{kj}^{(k-1)} + \operatorname{sgn}[\tilde{p}_{kj}^{(k-1)}] a_j^{(k)}) \times \frac{b_j^{(k)} + \operatorname{sgn}[\tilde{p}_{kj}^{(k-1)}] a_j^{(k)} z_k^{(k-1)}}{a_j^{(k)}(a_j^{(k)} + |\tilde{p}_{kj}^{(k-1)}|)} \right)^2 = \left(\frac{b_j^{(k)}}{a_j^{(k)}} \right)^2 \quad \text{for } k = k, \ldots, M$$

is achieved at $j = j_m$. Then interchange the j_mth column of $\tilde{\mathbf{P}}^{(k-1)}$ with the kth column. The rest of the operations are as indicated in (51) and (52). The procedure is terminated at the M_sth stage when

$$1 - \sum_{i=1}^{M_s} \frac{(z_j^{(M_s)})^2}{\langle \mathbf{z}, \mathbf{z} \rangle} = 1 - \sum_{i=1}^{M_s} \frac{(z_j^{(i)})^2}{\langle \mathbf{z}, \mathbf{z} \rangle} < \rho \tag{79}$$

or when AIC(4) is minimized. It can be seen that $(z_i^{(i)})^2/\langle \mathbf{z}, \mathbf{z} \rangle$ corrdsponds to $[\mathrm{err}]_i$ in the CGS and MGS algorithms and if the notation $[\mathrm{err}]_i = (z_i^{(i)})^2/\langle \mathbf{z}, \mathbf{z} \rangle$ is used (79) becomes identical to (70). The subset model parameter estimate $\hat{\mathbf{\Theta}}_s$ is computed from

$$\mathbf{R}_{M_s} \mathbf{\Theta}_s = \begin{bmatrix} z_1^{(M_s)} \\ \vdots \\ z_s^{(M_s)} \end{bmatrix}, \quad \text{that is } \mathbf{R}_{M_s} \mathbf{\Theta}_s = \begin{bmatrix} z_1^{(1)} \\ \vdots \\ z_{M_s}^{(M_s)} \end{bmatrix} \tag{80}$$

using backward substitution.

This algorithm seems to require less computer memory space than the algorithm based on the MGS method because $\mathbf{R}$ is stored by overwriting part of $\mathbf{P}$. Using the

Householder transformation method to select predictors in such a forward-regression manner has been mentioned by Golub (1965); $a_j^{(k)}$ and $b_j^{(k)}$ for $j = k, \ldots, M$ can be calculated quickly in the following way. Given

$$(a_j^{(1)})^2 = \sum_{i=1}^{N} (\tilde{p}_{ij}^{(0)})^2, \quad b_j^{(1)} = \sum_{i=1}^{N} \tilde{p}_{ij}^{(0)} z_i^{(0)}, \quad j = 1, \ldots, M \tag{81}$$

After $\tilde{\mathbf{P}}^{(k)}$ has been computed, $(a_j^{(k+1)})^2$ and $b_j^{(k+1)}$ for $j = k+1, \ldots, M$ are updated according to

$$(a_j^{(k+1)})^2 = (a_j^{(k)})^2 - (\tilde{p}_{kj}^{(k)})^2, \quad b_j^{(k+1)} = b_j^{(k)} - \tilde{p}_{kj}^{(k)} z_k^{(k)}, \quad j = k+1, \ldots, M \tag{82}$$

Naturally, if the columns of $\tilde{\mathbf{P}}^{(k-1)}$ are interchanged, the $a_j^{(k)}$ must be interchanged accordingly and so must the $b_j^{(k)}$.

5. Iterative schemes for polynomial NARMAX models

For the polynomial NARMAX model, delayed noise terms are included in each subsystem model and these are generally unmeasured. The solution to this problem is to replace $e(t)$ by the prediction errors or residuals $\varepsilon(t)$ in the identification process. Let $\mathbf{P}_i$ represent the full submodel set of subsystem i which is partitioned into

$$\mathbf{P}_i = [\mathbf{P}_{p_i} : \mathbf{P}_{n_i}] \tag{83}$$

where each column in $\mathbf{P}_{p_i}$ is a monomial function of the input–output data only and each column in $\mathbf{P}_{n_i}$ is a monomial function of the prediction errors and the input–output data. Here $\mathbf{P}_{p_i}$ may therefore be referred to as the full ith process sub-model and $\mathbf{P}_{n_i}$ the full ith noise sub-model. A subset $\mathbf{P}_{si}$ of $\mathbf{P}_i$ is similarly represented as

$$\mathbf{P}_{si} = [\mathbf{P}_{p_{si}} : \mathbf{P}_{n_{si}}] \tag{84}$$

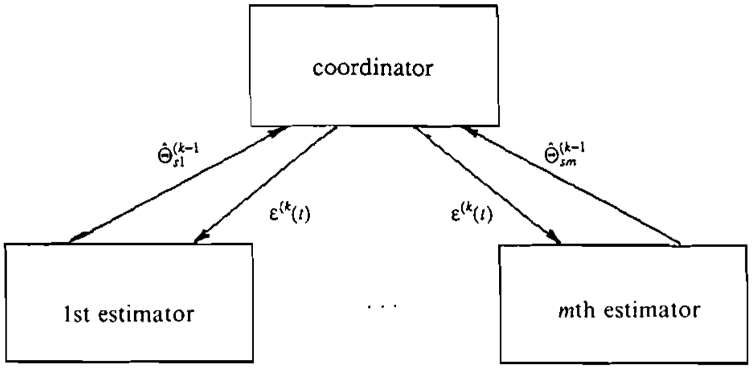

Figure 1. Iterative scheme for the polynomial NARMAX model.

Referring to Fig. 1, a general iterative scheme using the orthogonal algorithms of § 4 can be described as follows.

Initially the ith estimator selects columns of $\mathbf{P}_{p_{si}}$ from $\mathbf{P}_{p_i}$. The selection is terminated after $M_{p_{si}}$ columns have been selected and the condition

$$1 - \sum_{j=1}^{M_{p_{si}}} [\text{err}]_j < \rho_{p_i} \tag{85}$$

is satisfied where ρ_{p_i} is the tolerance for the ith process sub-model. The initial submodel parameter estimate $\hat{\Theta}_{si}^{(0}$ (containing only $M_{p_{si}}$ elements) can be computed and then sent to the coordinator. Based on $\hat{\Theta}_{si}^{(0}$, $i = 1, \ldots, m$, the coordinator can generate the initial prediction error sequence $\{\varepsilon^{(1}(t)\}$.

At the kth iteration ($k \geqslant 1$) each estimator receives $\{\varepsilon^{(k}(t)\}$ from the coordinator. This allows the ith estimator to form $\mathbf{P}_{n_i}^{(k}$ and to select columns from $\mathbf{P}_{n_i}^{(k}$. Assume that after $M_{n_{si}}^{(k}$ more columns have been added to $\mathbf{P}_{p_{si}}$ the condition

$$1 - \sum_{j=1}^{M^{p_{si}} + M^{k}_{n_{si}}} [\mathrm{err}]_j < \rho_{n_i} \tag{86}$$

is satisfied, where ρ_{n_i} ($< \rho_{p_i}$) is the tolerance for the ith noise sub-model, the selection procedure is stopped, $\hat{\Theta}_{si}^{(k}$ (containing $M_{p_{si}} + M_{n_{si}}^{(k}$ elements) is computed and sent to the coordinator.

Previous experience involving the CGS orthogonal algorithm has shown that typically four to six iterations are usually adequate. Since the MGS and Householder transformation algorithms are numerically more accurate than the CGS algorithm, four to six iterations should also be adequate for the iterative schemes using these two methods.

Notice that the selection of the process and noise model parameters is decoupled. However, there is no particular reason why the process model terms should be selected first and the noise model terms selected later other than that this way is convenient for updating $\{\varepsilon(t)\}$. For SISO systems, an additional stage may be added to the above scheme: after a reasonable $\{\varepsilon(t)\}$ has been obtained, we may reselect process and noise model terms together from $\mathbf{P}$ and stop the procedure when AIC(4) is minimized. It is also possible to terminate the iterative scheme using AIC(ϕ) instead of the error-reduction-ratio criterion for SISO systems. For example, the process model regression is stopped when AIC(4) is minimized and, at each iteration, the noise model regression is stopped when AIC(2) is minimized. Theoretical analysis (Leontaritis and Billings 1987) and simulation studies (Leontaritis and Billings 1987; Chen and Billings 1988) indicate that AIC(2) often overestimates the number of necessary parameters. Ideally we should treat the process and noise terms equally. When AIC(4) is used to terminate the process model regression, however, we are in fact trying to fit a NARX model to the system which may be better modelled by a NARMAX model and some unnecessary process terms may be included in the initial stage. AIC(2) is therefore deliberately used for the noise model regression at each iteration in order to avoid the possibility of losing significant noise model terms. For multivariable systems, AIC(ϕ) becomes

$$\mathrm{AIC}(\phi) = N \log \det C(\hat{\Theta}_s) + M_s \phi \tag{87}$$

where

$$C(\hat{\Theta}_s) = \frac{1}{N} \sum_{t=1}^{N} \varepsilon(t) \varepsilon^{\mathrm{T}}(t) \tag{88}$$

and M_s is the number of all the subsystem parameters. It is difficult to apply this criterion to terminate the above iterative scheme for multivariable systems because the model structure determination is done in a decentralized way.

Computational aspects of different schemes obtained using different orthogonal methods applied to the NARMAX model are now discussed.

CGS scheme

The CGS algorithm orthogonalizes one column at a time. This makes the iterative CGS scheme a simple and natural extension of the forward-regression orthogonal procedure of § 4.1. After the initial stage, $\mathbf{W}_{si}$ contains $M_{p_{si}}$ orthogonal columns and the first $M_{p_{si}}$ columns of $\mathbf{A}_{si}$ have been computed. At the kth iteration, the ith estimator simply computes $M_{p_{si}}+1, \ldots, M_{p_{si}}+M_{n_{si}}^{(k}$ columns of $\mathbf{A}_{si}$ and selects corresponding columns of $\mathbf{W}_{si}$ from $\mathbf{P}_{n_i}^{(k}$ successively just as the $M_{p_{si}}+1, \ldots, M_{p_{si}}+M_{n_{si}}^{(k}$ stages of the forward-regression orthogonal procedure of § 4.1.

MGS scheme

For the ith estimator, when $\mathbf{P}_{n_i}^{(k}$ has been formed at the kth iteration, each column of $\mathbf{P}_{n_i}^{(k}$ must be made orthogonal to the $\bar{\imath}$th, $\bar{\imath}=1, \ldots, M_{p_{si}}$, column of $\mathbf{W}_{si}$ and the corresponding $\alpha_{\bar{\imath}j}$ must be computed first; $\mathbf{z}^{(M_{p_{si}})}$ must also be restored at the beginning of each iteration. After these operations, the rest of the kth iteration is as the $M_{p_{si}}+1, \ldots, M_{p_{si}}+M_{n_{si}}^{(k}$ stages of the forward-regression orthogonal procedure of § 4.2.

Householder transformation scheme

For the ith estimator, the initial stage consists of applying the forward-regression orthogonal procedure of § 4.3 to

$$[\mathbf{P}_{p_i} \vdots \mathbf{y}_i] \tag{89}$$

where $\mathbf{y}_i=(y_i(1), \ldots, y_i(N))^{\mathrm{T}}$. After $M_{p_{si}}$ Householder orthogonal matrices have been applied to the matrix of (89) it is transformed to

$$\begin{bmatrix} \mathbf{R}_{M_{p_{si}}} & & \\ & \cdots & \mathbf{y}_i^{(M_{p_{si}})} \\ 0 & & \end{bmatrix} \tag{90}$$

These $M_{p_{si}}$ orthogonal transformations must be preserved (e.g., stored in the space below the diagonal of $\mathbf{R}_{M_{p_{si}}}$).

At the kth iteration, when $\mathbf{P}_{n_i}^{(k}$ has been formed, the $\bar{\imath}$th, $\bar{\imath}=1, \ldots, M_{p_{si}}$, Householder transformations must be applied to $\mathbf{P}_{n_i}^{(k}$ successively. Denoting the resulting matrix as $\bar{\mathbf{P}}_{n_i}^{(k}$, the rest of the kth iteration is as the $M_{p_{si}}+1, \ldots, M_{p_{si}}+M_{n_{si}}^{(k}$ stages of the forward-regression orthogonal procedure of § 4.3 applied to the matrix

$$\begin{bmatrix} \mathbf{R}_{M_{p_{si}}} & \bar{\mathbf{P}}_{ni}^{(k} \vdots \mathbf{y}_i^{(M_{p_{si}})} \\ 0 & \end{bmatrix} \tag{91}$$

6. Simulation study

The main purpose of this simulation study was to compare the performance of the three algorithms and only SISO examples will be used. Some multivariable examples using the CGS scheme can be found in work by Billings *et al.* (1989 b). The program is written on a Sun 3/50 workstation and all calculations ard carried out in single precision.

Example 1

The data was collected from a large pilot scale liquid level system where the input was a zero mean gaussian signal. A description of this process is given by Billings and Voon (1986). The inputs and outputs of the system are illustrated in Fig. 2.

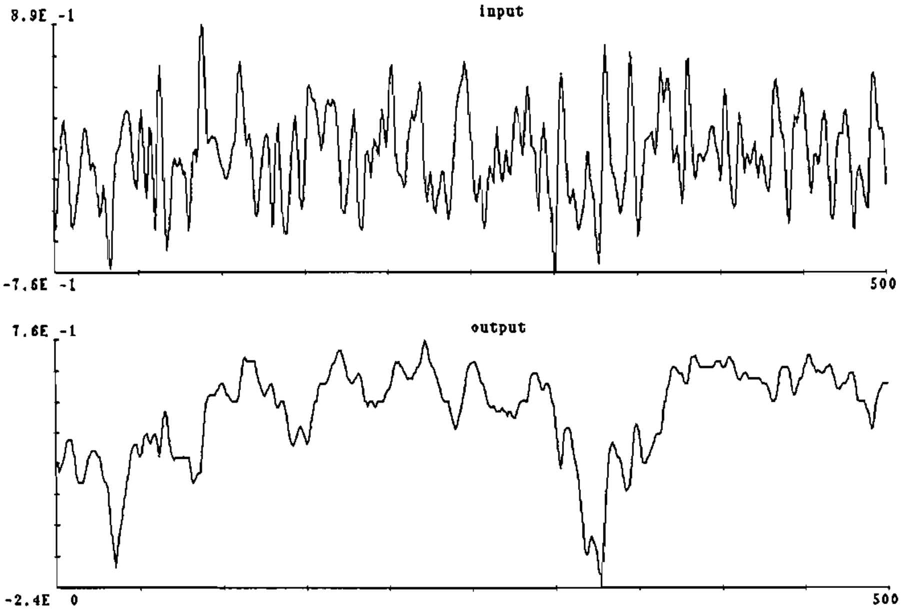

Figure 2. Inputs and outputs of Example 1.

A polynomial NARX model ($l = n_y = n_u = 3$ and $n_e = 0$) was used to fit the data. The full model set consisted of 84 terms. The results obtained by applying the forward-regression variant of CGS algorithm are given in Table 1. Using the same $\rho = 0{\cdot}0048$ the MGS and Householder transformation (HT) algorithms produced the same model as in Table 1.

Terms	Estimates	$[\mathrm{err}]_i$	Standard deviations
$y(t-1)$	0·80916E + 0	0·97444E + 0	0·47816E − 1
$u(t-1)$	0·41813E + 0	0·14158E − 1	0·15593E − 1
$u(t-2)$	−0·10762E + 0	0·26158E − 2	0·16974E − 1
$y(t-2)y(t-3)u(t-2)$	−0·36722E + 0	0·12047E − 2	0·53508E − 1
$y(t-1)u(t-1)$	−0·33901E + 0	0·18040E − 2	0·27629E − 1
$y(t-2)y(t-3)u(t-3)$	0·14530E + 0	0·31312E − 3	0·20979E − 1
$u(t-1)u^2(t-2)$	−0·16590E + 0	0·17932E − 3	0·43352E − 1
$y(t-1)u(t-2)$	0·16459E + 0	0·73725E − 4	0·31417E − 1
$y^2(t-2)y(t-3)$	−0·39164E − 1	0·90227E − 4	0·60504E − 2
$y(t-1)y(t-2)$	−0·58358E − 1	0·18795E − 3	0·10442E − 1
$y(t-2)$	0·16186E + 0	0·12685E − 3	0·46291E − 1
$y^2(t-3)u(t-2)$	0·13916E + 0	0·64359E − 4	0·54245E − 1

Tolerance $\rho = 0{\cdot}0048$, variance of residuals $\sigma_\varepsilon^2 = 0{\cdot}18585\mathrm{E} - 2$, residual variance and output variance ratio $= 0{\cdot}47429\mathrm{E} - 2$.

Table 1. Selected model of Example 1 (using CGS and ERR criterion).

Example 2

This is a simulated system. The data was generated by

$$y(t) = 0{\cdot}5y(t-1) + u(t-2) + 0{\cdot}1u^2(t-1) + 0{\cdot}5e(t-1)$$
$$+ 0{\cdot}2u(t-1)e(t-2) + e(t)$$

where the system noise $e(t)$ was a gaussian white sequence with mean zero and variance 0·04 and the system input $u(t)$ was an independent sequence of uniform distribution with mean zero and variance 1·0. The inputs and outputs of the system are shown in Fig. 3.

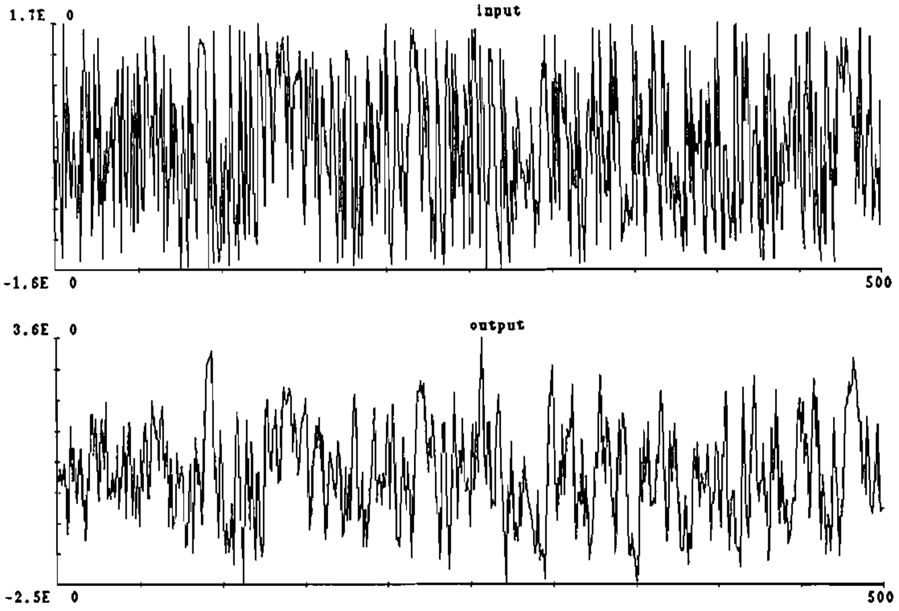

Figure 3. Inputs and outputs of Example 2.

A polynomial NARMAX model with $l = n_y = n_u = n_e = 2$ was used to fit the data. The full model set contained 28 terms. Using the iterative scheme discussed in § 5, that is, selecting process model terms first and then entering an iterative loop to select the noise model terms, the MGS algorithm gave the correct final model, and this can be seen from Table 2 where $\rho_p = 0{\cdot}034$, $\rho_n = 0{\cdot}026$ and five iterations (computing the

Terms	Estimates	$[\text{err}]_i$	Standard deviations
$u(t-2)$	0·10032E + 1	0·67047E + 0	0·89932E − 2
$y(t-1)$	0·50276E + 0	0·28735E + 0	0·73754E − 2
$u^2(t-1)$	0·91139E − 1	0·85640E − 2	0·68862E − 2
$e(t-1)$	0·54833E + 0	0·69294E − 2	0·45770E − 1
$u(t-1)e(t-2)$	0·23770E + 0	0·16084E − 2	0·42268E − 1

Variance of residuals $\sigma_\varepsilon^2 = 0{\cdot}41436\text{E} - 1$, residual variance and output variance ratio = 0·25072E − 1.

Table 2. Selected model of Example 2 (using MGS and ERR criterion).

residual sequence 5 times) were involved. Under the same conditions the CGS and HT algorithms produced identical results.

As mentioned in § 5, other iterative strategies can also be employed. The following procedure was also tested on Example 2. First use AIC(4) to terminate the process model regression and use this to produce the initial residual sequence $\{\varepsilon^{(1}(t)\}$ and noise model set $\mathbf{P}_n^{(1}$. Next use AIC(2) to terminate the noise model regression. Having obtained $\{\varepsilon^{(k}(t)\}$ $k \geqslant 2$ then select the process and noise model terms together from the full model set or the model set obtained from the previous iteration. Each of such regressions is terminated when AIC(4) is minimized. When the model set contains the noise terms, to compute the exact AIC(ϕ) value each time a term is selected would require the computation of all the selected parameter values $\hat{\theta}_i$ and the recalculation of the residual sequence. This can be avoided by computing the approximate AIC(ϕ) value using the approximate variance of the residuals. For a model of M_s terms, if the residual sequence is approximated by

$$\hat{\boldsymbol{\varepsilon}} = \mathbf{z} - \sum_{i=1}^{M_s} g_i \mathbf{w}_i \tag{92}$$

then the approximate variance of the residuals is readily given by

$$\sigma_z^2 \left(1 - \sum_{i=1}^{M_s} [\text{err}]_i \right) \tag{93}$$

where

$$\sigma_z^2 = \frac{1}{N} \langle \mathbf{z}, \mathbf{z} \rangle \tag{94}$$

Notice that for polynomial NARX models (92) gives the exact residual sequence and (93) the exact variance of the residuals. Using this procedure and involving only two iterations ($k = 2$), the HT algorithm produced the model shown in Table 3. The results obtained by the CGS and MGS algorithms were identical to those of Table 3.

Terms	Estimates	$[\text{err}]_i$	Standard deviations
$u(t-2)$	0·10034E + 1	0·67047E + 0	0·90317E − 2
$y(t-1)$	0·50355E + 0	0·28735E + 0	0·74037E − 2
$u^2(t-1)$	0·89880E − 1	0·85640E − 2	0·69221E − 2
$e(t-1)$	0·51440E + 0	0·65584E − 2	0·44959E − 1
$u(t-1)e(t-2)$	0·20321E + 0	0·12277E − 2	0·41518E − 1

Variance of residuals $\sigma_\varepsilon^2 = 0{\cdot}41754\text{E} - 1$, residual variance and output variance ratio = 0·25265E − 1.

Table 3. Selected model of Example 2 (using HT and AIC criterion).

7. Conclusions

Several orthogonal forward-regression estimators have been derived for the identification of polynomial NARMAX systems by modifying and augmenting some well-known orthogonal least squares methods. It has been shown that these estimators efficiently combine structure determination with parameter estimation to provide very powerful procedures for identifying parsimonious models of structure-unknown systems. The application to both simulated and real data has been demonstrated.

Whilst the iterative CGS scheme is easier to implement on a microcomputer and its coding is simpler, experience has shown that the iterative MGS and HT schemes work faster. The first (off-line) version of the Givens method discussed in § 3.2 can also be used to develop a similar model structure selection routine but it will require more computations compared with the three routines discussed in this work.

Acknowledgments

Two of the authors, S. Chen and S. A. Billings, gratefully acknowledge financial support for the work presented above from the Science and Engineering Research Council (Grant Ref. GR/D/30587).

References

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

Billings, S. A., 1986, Introduction to nonlinear ststems analysis and identification. *Signals Processing for Control*, edited by K. Godfrey and P. Jones (Berlin: Springer-Verlag), pp. 261–294.

Billings, S. A., Chen, S., and Backhouse, R. J., 1989 a, The identification of linear and nonlinear models of a turbocharged automotive diesel engine. *Mechanical Systems and Signal Processing*, **3**, 123–142.

Billings, S. A., Chen, S., and Korenberg, M. J., 1989 b, Identification of MIMO nonlinear systems using a forward-regression orthogonal estimator. *International Journal of Control*, **49**, 2157–2189.

Billings, S. A., and Fadzil, M. B., 1985, The practical identification of systems with nonlinearities. *Proceedings of the 7th IFAC Symposium on Identification and System Parameter Estimation*, York, U.K., pp. 155–160.

Billings, S. A., Fadzil, M. B., Sulley, J., and Johnson, P. M., 1988 a, Identification of a nonlinear difference equation model of an industrial diesel generator. *Mechanical Systems and Signal Processing*, **2**, 59–76.

Billings, S. A., Korenberg, M. J., and Chen, S., 1988 b, Identification of non-linear output-affine systems using an orthogonal least-squares algorithm. *International Journal of Systems Science*, **19**, 1559–1568.

Billings, S. A., and Voon, W. S. F., 1986, A prediction-error and stepwise-regression estimation algorithm for non-linear systems. *International Journal of Control*, **44**, 803–822.

Björck, A., 1967, Solving linear least squares problems by Gram–Schmidt orthogonalization. *Nordisk Tidshrift for Informationsbehadlung*, **7**, 1–21.

Chen, S., and Billings, S. A., 1988, Prediction-error estimation algorithm for non-linear output-affine systems. *International Journal of Control*, **47**, 309–332; 1989, Representations of nonlinear systems: the NARMAX model. *Ibid.*, **49**, 1013–1032.

Desrochers, A., and Mohseni, S., 1984, On determining the structure of a non-linear system. *International Journal of Control*, **40**, 923–938.

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

Gentleman, W. M., 1973, Least squares computations of Givens transformations without square roots. *Journal of Institute of Mathematics and its Applications*, **12**, 329–336.

Golub, G. H., 1965, Numerical methods for solving linear least squares problems. *Numerische Mathematik*, **7**, 206–216; 1969, Matrix decompositions and statistical calculations. *Statistical Computation*, edited by R. C. Milton and J. A. Nelder (New York: Academic Press).

Golub, G. H., and Reinsch, C., 1970, Singular value decomposition and least squares solutions. *Numerische Mathematik*, **14**, 403–420.

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

Hanson, R. J., and Lawson, C. L., 1969, Extensions and applications of the Householder algorithm for solving linear least squares problems. *Mathematics of Computation*, **23**, 787–812.

Jordan, T. L., 1968, Experiments on error growth associated with some linear least-squares procedures. *Mathematics of Computation*, **22**, 579–588.

KORENBERG, M. J., 1985, Orthogonal identification of nonlinear difference equation models. *Mid. West Symposium on Circuits and Systems*, Louisville.

KORENBERG, M. J., BILLINGS, S. A., LIU, Y. P., and MCILOY, P. J., 1988, Orthogonal parameter estimation algorithm for non-linear stochastic systems. *International Journal of Control*, **48,** 193–210.

KORTMANN, M., and UNBEHAUEN, H., 1988, A model structure selection algorithm in the identification of multivariable nonlinear systems with application to a turbogenerator set. *12th IMACS World Congress on Scientific Computation*, Paris, 18–22 July, 1988.

LEONTARITIS, I. J., and BILLINGS, S. A., 1985, Input–output parametric models for non-linear systems. Part I: deterministic non-linear systems; Part II: stochastic non-linear systems. *International Journal of Control*, **41,** 303–344; 1987, Model selection and validation methods for non-linear systems. *Ibid.*, **45,** 311–341.

RICE, J. R., 1966, Experiments on Gram–Schmidt orthogonalization. *Mathematics of Computation*, **20,** 325–328.

SEBER, G. A. F., 1977, *Linear Regression Analysis* (New York: Wiley).

SÖDERSTRÖM, T., 1977, On model structure testing in system identification. *International Journal of Control*, **26,** 1–18.

WAMPLER, R. H., 1970, A report on the accuracy of some widely used least squares computer programs. *Journal of American Statistical Association*, **65,** 549–565.

WILKINSON, J. H., 1974, The classical error analysis for the solution of linear systems. *Institute of Mathematics and its Applications Bulletin*, **10,** 175–180.