This is a repository copy of *On-Line Structure Detection and Parameter Estimation with Exponential Windowing for Nonlinear Systems*.

White Rose Research Online URL for this paper:
http://eprints.whiterose.ac.uk/79631/

Monograph:
Luo, W., Billings, S.A. and Tsang, K.M. (1994) On-Line Structure Detection and Parameter Estimation with Exponential Windowing for Nonlinear Systems. Research Report. ACSE Research Report 503 . Department of Automatic Control and Systems Engineering

On-line Structure Detection and Parameter Estimation with Exponential Windowing for Nonlinear Systems

W. Luo [*]   *S. A. Billings* [*]   *and*   *K. M. Tsang* [+]

[*]Department of Automatic Control and Systems Engineering

University of Sheffield

Sheffield, UK

[+]Department of Electrical Engineering

Hong Kong Polytechnic

Hung Hom, Kowloon

Hong Kong

February 1994

On-line Structure Detection and Parameter Estimation with Exponential Windowing for Nonlinear Systems

W. Luo [*]   *S. A. Billings* [*]   *and   K. M. Tsang* [+]

Abstract

A new recursive orthogonal estimation algorithm is derived which updates both the model structure and the parameters of nonlinear models on-line. Techniques developed for linear systems are not generally suitable when nonlinear models are considered because the complexity of nonlinear systems and the demands of recursive processing make on-line structure detection much more difficult. In the present study a new on-line orthogonal estimation algorithm based on the polynomial NARMAX model is derived by extending the family of orthogonal QR decomposition algorithms to include on-line model structure selection. The new algorithm which includes exponential data windowing based on a stable Givens routine minimizes the loss function at every selection step by selecting significant regression variables, computes the parameter estimates and maintains the orthogonality of the vector space for continuous computation. Simulated examples are included to demonstrate the performance of the new algorithm.

1. Introduction

Most real systems are nonlinear. If the nonlinearities are relatively mild techniques of linearization can provide simple methods of modelling nonlinear systems. However such conditions are often not satisfactory and the obvious extension is to consider the application of nonlinear models. For nonlinear systems with time-varying structure nonlinear threshold models can provide an adequate representation but the analysis of the transient behaviour between operating regions can be difficult if the nonlinearities are severe or the input changes rapidly. A solution to these systems is to use time-varying models and on-line algorithms. The polynomial NARMAX (Nonlinear AutoRegressive Moving Average with Exogenous inputs) model (Leontaritis and Billings 1985) can be used to describe a large class of nonlinear systems (usually with less than 10 terms) and exhibits a low computational cost compared with other

[*] Department of Automatic Control and Systems Engineering. University of Sheffield, Sheffield, UK
[+] Department of Electrical Engineering. Hong Kong Polytechnic, Hung Hom, Kowloon, Hong Kong

nonlinear descriptions (Billings 1985). The extension of this model to include time-varying terms will be considered in the present study.

Almost all recursive parameter estimation algorithms operate under the assumption that the system structure or which terms should be in the model has been predetermined. But real systems can be time-variant and the model structure can change so it becomes necessary to detect such structure changes on-line to ensure accurate estimates are obtained. Lattice algorithms developed for linear systems can perform on-line adjustment for the order of linear AR or ARMA models using order-recursion methods based on the shifting property of the regressors used in these models. But such order-recursion techniques are not suitable for most nonlinear models because these properties no longer hold. The recursive modified Gram Schmidt algorithm (RMGS) (Ling et al 1986) does not require the shifting property as a prerequisite for performing order-recursion, but the expansion of candidate variables must be in a specified form for easy adjustment of the model order. Assume that m candidate variables can be arranged in the order of ϕ_1, ϕ_2, ϕ_3, ..., ϕ_m. Such a sequence ensures that if ϕ_1, ..., ϕ_{i-1} cannot properly describe the system dynamics a reasonable solution can be obtained by a successive expansion of the variables ϕ_i, ϕ_{i+1}, ... Since most nonlinear models do not have such a simple structure, the order-recursive procedure of RMGS cannot readily be applied to adjust the structure of these models. New recursive algorithms which do not depend on the shifting property or the successive expansion of candidate variables should therefore be developed to detect the system structure on-line.

The recursive parameter estimation algorithms based on orthogonal QR decomposition (Golub and Styan 1973, Bierman 1977) are numerically stable and accurate but the adjustment of the model structure is restricted. Although the off-line QR decomposition algorithms with forward selection derived in (Chen et al 1989) can provide a good starting point for developing on-line structure detection methods the selection procedures cannot be easily carried over to the on-line case. There are three reasons for this. First a finite storage space requires that the orthogonal transformation must be implemented sequentially. Second, when an orthogonal space has been formed, the change of the matrix position of any orthogonal vector, for example the i'th orthogonal vector, will destroy the orthogonality of the $i+1$, $i+2$, ..., m'th orthogonal vectors. To ensure orthogonality these vectors must be reorthogonalized but this becomes difficult if the off-line method is employed, because the finite memory space cannot store the complete data set. Thirdly, the system structure is unknown, therefore the computation has to start with all the candidate variables some of which may be

linearly dependent, so the regression matrix may not be of full rank. This can induce singularities and the least squares normal equations will no longer have a unique solution.

These problems are avoided in the present study by using the Givens rotation method to develop a new recursive QR decomposition algorithm for time-varying polynomial NARMAX models. This new algorithm which will be referred to as the GFSE algorithm (Givens rotation with Forward Selection and Exponential windowing algorithm) can detect the model structure on-line and estimate the unknown parameters. The GFSE algorithm like other recursive QR parameter estimation algorithms is numerically stable and accurate but provides more flexibility for on-line selection of regression variables. This algorithm can therefore be applied to systems with time-varying structure. Although only SISO systems will be considered in the present study, the new method can readily be extended to MIMO systems. The paper is organized as follows. First the time-varying polynomial NARMAX model is defined in Section 2. The recursive orthogonal transformation is described in Section 3 and the mechanics of the on-line structure detection are derived in Section 4. In Section 5 the initialization of the algorithm is discussed and Section 6 provides a summary of the procedures. The properties of the new GFSE algorithm are described in Section 7 and simulated results are included in Section 8.

2. System Representation

Subject to some mild assumptions, a discrete time SISO nonlinear system can be represented by the NARMAX model (Leontaritis and Billings 1985)

$$y(t){=}F\left[y(t-1),...,y(t-n_y),u(t-d),...,u(t-d-n_u+1),e(t-1),...,e(t-n_e)\right] + e(t). \qquad (2\text{-}1)$$

where t is the t'th time point, $y(t)$, $u(t)$ and $e(t)$ represent the output, input, and noise respectively, n_y, n_u and n_e are the corresponding orders, $F[.]$ is some nonlinear functions and d is the minimum time delay of the input. Since $F[.]$ may include cross product-terms and higher degree terms in $y(.)$, $u(.)$ and $e(.)$, the degree of the power terms in $y(.)$, $u(.)$ and $e(.)$ will be referred to as the degree of nonlinearity denoted as n_l. In practical computations the noise will not be measurable and is replaced by the residual or prediction error

$$\epsilon(t) = y(t) - \hat{y}(t) \qquad (2\text{-}2)$$

where ^ represents one step ahead predicted output. Eqn (2-1) can be rewritten as

$$y(t) = F[y(t-1),...,y(t-n_y),u(t-d),...,u(t-d-n_u+1),\varepsilon(t-1),...,\varepsilon(t-n_\varepsilon)] + \varepsilon(t). \qquad (2\text{-}3)$$

The model (2-3) can be used to specify a general finite-dimensional input-output nonlinear system (Chen and Billings 1989, Fnaiech and Ljung 1987). Various possibilities of parametrizing $F[.]$ exist (Chen and Billings 1989). If functions $\phi_i(.)$ are chosen as monomials of lagged $u(t)$, $y(t)$ and/or $\varepsilon(t)$, e.g. $u^2(t-1)y(t-2)$ etc, a polynomial model is obtained. Transforming the lagged input, output and residual in (2-3) into regressors yields the pseudo-linear regression model

$$y(t) = \sum_{i=1}^{m} \phi_i(y(t-1),...,y(t-n_y),u(t-d), ..., u(t-d-n_u+1),\varepsilon(t-1),...,\varepsilon(t-n_\varepsilon))\theta_i(t) + \varepsilon(t) \qquad (2\text{-}4)$$

or more concisely

$$y(t) = \sum_{i=1}^{m} \phi_i(t)\theta_i(t) + \varepsilon(t). \qquad (2\text{-}5)$$

In equation (2-5) $\phi_i(t)$ expresses the ith regression variable (regressor), m is the number of the regressors and $\theta_i(t)$ is the unknown parameter corresponding to $\phi_i(t)$. Combining all the data at time points 1, 2, ..., t produces the model

$$\mathbf{y}(t) = \mathbf{\Phi}(t)\mathbf{\theta}(t) + \mathbf{\varepsilon}(t) \qquad (2\text{-}6)$$

where $\mathbf{\Phi}(t)$ is the $t \times m$ regression matrix, $\mathbf{\theta}(t)$ is the $m \times 1$ parameter matrix, $\mathbf{y}(t)$ and $\mathbf{\varepsilon}(t)$ are $t \times 1$ matrices. Since the linear-in-the-parameter property is preserved, the models can be applied to severely nonlinear systems using efficient off-line estimation algorithms (Chen et al 1989).

Replacing m with a variant $m(t)$ in Eqn. (2-4), the variable structure NARMAX model is defined. This should provide an acceptable basis for predicting the output during time variation and produce a smooth transient between different operating regimes. Such a model is time-dependent not signal-dependent (Billings and Voon 1987). Like most nonlinear models, the structure of the NARMAX model does not satisfy the shift-invariant property or successive expansion of candidate variables. However the polynomial structure of the NARMAX model provides some significant properties which can be utilized to develop new algorithms for both on-line structure detection and parameter estimation of nonlinear systems.

3. A Recursive Orthogonal Transformation with Exponential Weighting

The Euclidean norm of a vector is unitarily invariant,

$$\| \mathbf{y}(t) - \mathbf{\Phi}(t)\mathbf{\theta}(t) \| = \| \mathbf{v}(t) - \mathbf{Q}^T(t)\mathbf{\Phi}(t)\mathbf{\theta}(t) \|$$

where $\mathbf{v}(t) = \mathbf{Q}^T(t)\mathbf{y}(t)$ and $\mathbf{Q}(t)$ is a $t \times t$ orthogonal matrix given by

$$\mathbf{Q}(t) = [\mathbf{q}_1(t) \quad \mathbf{q}_2(t) \quad \cdots \quad \mathbf{q}_m(t) \quad \mathbf{q}_{m+1}(t) \quad \cdots \quad \mathbf{q}_n(t)]$$

$\mathbf{Q}^T(t)\mathbf{Q}(t)=\mathbf{I}$ and hence $\mathbf{Q}(t)$ is a $t \times t$ orthonormal matrix. Choose $\mathbf{Q}(t)$ to be of the form

$$\mathbf{Q}^T(t)\mathbf{\Phi}(t) = \begin{bmatrix} \mathbf{R}(t) \\ \mathbf{O}(t) \end{bmatrix}$$

where $\mathbf{O}(t)$ is a $(t-m) \times m$ zero matrix and

$$\mathbf{R}(t) = \begin{bmatrix} r_{11}(t) & r_{12}(t) & . & . & . & r_{1m}(t) \\ 0 & r_{22}(t) & . & . & . & r_{2m}(t) \\ . & & 0 & & & . \\ . & & & . & . & . \\ . & & & & . & . \\ 0 & & . & . & 0 & r_{mm}(t) \end{bmatrix}$$

is an $m \times m$ upper triangular matrix. Premultiply $\mathbf{y}(t)=\mathbf{\Phi}(t)\mathbf{\theta}(t)$ by $\mathbf{Q}^T(t)$ to obtain

$$\mathbf{Q}^T(t)\mathbf{y}(t) = \mathbf{Q}^T(t)\mathbf{\Phi}(t)\mathbf{\theta}(t) = \begin{bmatrix} \mathbf{R}(t) \\ \mathbf{O}(t) \end{bmatrix}\hat{\mathbf{\theta}}(t) + \mathbf{Q}^T(t)\mathbf{\varepsilon}(t)$$

and hence

$$\mathbf{v}(t) = \begin{bmatrix} \mathbf{v}_m(t) \\ \mathbf{v}_{t-m}(t) \end{bmatrix} = \begin{bmatrix} \mathbf{R}(t) \\ \mathbf{O}(t) \end{bmatrix}\hat{\mathbf{\theta}}(t) + \mathbf{Q}^T(t)\mathbf{\varepsilon}(t)$$

where $\mathbf{v}_m(t)$ contains the first m components of $\mathbf{v}(t)$ and $\mathbf{v}_{t-m}(t)$ contains the remainder. The estimation of $\hat{\mathbf{\theta}}(t)$ is easily achieved by solving the triangular system $\mathbf{R}(t)\hat{\mathbf{\theta}}(t) = \mathbf{v}_m(t)$, and consequently the norm of the residual vector

$$\|\mathbf{\varepsilon}(t)\| = \| \mathbf{y}(t) - \mathbf{\Phi}(t)\hat{\mathbf{\theta}}(t) \| = \|\mathbf{v}_{t-m}(t)\| = (\sum_{i=m+1}^{t} v_i^2(t))^{1/2}$$

The whole factorization process involves a decomposition of $\mathbf{\Phi}(t)$ and it is hence called the orthogonal triangular decomposition or the **QR** algorithm.

Suppose that there are $t-1$ observations of the system $\mathbf{y}(t-1)=\mathbf{\Phi}(t-1)\mathbf{\theta}(t-1)$ which contains m regression variables, augmenting the $t-1$ normal equations with a zero element denoted $\mathbf{O}_2$ and a $1 \times m$ zero row vector $\mathbf{O}_1$ yields

$$\begin{bmatrix} \mathbf{y}(t-1) \\ \mathbf{O}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{\Phi}(t-1) \\ \mathbf{O}_1 \end{bmatrix}\mathbf{\theta}(t-1) \tag{3-1}$$

Premultiplying by a $t \times t$ orthogonal matrix $\mathbf{Q}^T(t-1)$ yields the QR orthogonal decomposition

$$\mathbf{v}(t-1) = \begin{bmatrix} \mathbf{v}_m(t-1) \\ \mathbf{v}_{t-1-m}(t-1) \\ \mathbf{O}_2 \end{bmatrix} = \begin{bmatrix} \mathbf{R}(t-1) \\ \mathbf{O}_{t-1-m} \\ \mathbf{O}_1 \end{bmatrix} \hat{\theta}(t-1) + \mathbf{Q}^T(t-1)\varepsilon(t-1) \tag{3-2}$$

where $\mathbf{v}_m(t-1)$ and $\mathbf{v}_{t-1-m}(t-1)$ have dimensions $m \times 1$ and $(t-1-m) \times 1$ respectively, $\mathbf{O}_{t-1-m}$ is a $(t-1-m) \times m$ zero matrix, the residual vector

$$\varepsilon(t-1) = \begin{bmatrix} \varepsilon_m(t-1) \\ \varepsilon_{t-1-m}(t-1) \\ 0 \end{bmatrix} , \tag{3-3}$$

and $\varepsilon_m(t-1)$, $\varepsilon_{t-1-m}(t-1)$ are $m \times 1$ and $(t-1-m) \times 1$ matrices respectively. Suppose that a new observation is added to improve the previous estimates. The regression equations can be written as

$$\begin{bmatrix} \mathbf{v}(t-1) \\ \mathbf{v}_{t-1-m}(t-1) \\ y(t) \end{bmatrix} = \begin{bmatrix} \mathbf{R}(t-1) \\ \mathbf{O}_{t-1-m} \\ \phi(t) \end{bmatrix} \hat{\theta}(t-1) + \mathbf{Q}^T(t-1)\varepsilon(t-1) + \begin{bmatrix} \mathbf{O}_{\varepsilon 1} \\ \mathbf{O}_{\varepsilon 2} \\ \underline{\varepsilon}(t) \end{bmatrix} . \tag{3-4}$$

where $\mathbf{O}_{\varepsilon 1}$ and $\mathbf{O}_{\varepsilon 2}$ are $m \times 1$ and $(t-1-m) \times 1$ zero matrices respectively, and $\underline{\varepsilon}(t)$ is the a priori prediction error defined by

$$\underline{\varepsilon}(t) = y(t) - \phi(t)\hat{\theta}(t-1)$$

Premultiplying (3-4) with a $t \times t$ orthogonal matrix $\mathbf{Q}_t^T(t)$ updates the QR decomposition so that

$$\mathbf{v}(t) = \begin{bmatrix} \mathbf{v}_m(t) \\ \mathbf{v}_{t-1-m}(t) \\ v_t(t) \end{bmatrix} = \begin{bmatrix} \mathbf{R}(t) \\ \mathbf{O}_{t-1-m} \\ \mathbf{O}_1 \end{bmatrix} \hat{\theta}(t) + \mathbf{Q}^T(t)\varepsilon(t) , \tag{3-5}$$

where $v_t(t)$ is the t' th element of $\mathbf{v}(t)$, the orthogonal matrix $\mathbf{Q}(t)=\mathbf{Q}(t-1)\mathbf{Q}_t(t)$, the residual vector is given by

$$\varepsilon(t) = \begin{bmatrix} \varepsilon_m(t) \\ \varepsilon_{t-1-m}(t) \\ \varepsilon(t) \end{bmatrix} .$$

where the t' th element of $\varepsilon(t)$, $\varepsilon(t)$, is the residual (the a posteriori prediction error) defined as

$$\varepsilon(t) = y(t) - \phi(t)\hat{\theta}(t) .$$

The parameter vector $\hat{\theta}(t)$ can then be computed using a back-substitution with the relationship $\mathbf{v}_m(t)=\mathbf{R}(t)\hat{\theta}(t)$. It is worth noting that (3-5) provides a framework which can easily be employed for on-line detection of system structure by means of measuring the residual sum of squares (RSS).

Inspection of (3-5) shows that removing the equations between the $(t-1-m)$' th and $(t-1)$' th row of (3-5) does not affect the solution of the estimate $\hat{\theta}(t)$ ($=\mathbf{R}^{-1}(t)\mathbf{v}_m(t)$) and the residual sum of squares (RSS) can still be obtained from

$$RSS(t) = \|\varepsilon(t)\|^2 = \|\mathbf{y}(t)\|^2 - \|\mathbf{v}_m(t)\|^2 = \mathbf{y}^T(t)\mathbf{y}(t) - \mathbf{v}_m^T(t)\mathbf{v}_m(t) \tag{3-6}$$

Inspection of the derivation of (3-5) also shows that the orthogonal transformation matrix $\mathbf{Q}^T(t)$ or $\mathbf{Q}(t)$ does not need to be expressed explicitly for computing $\mathbf{R}(t)$ and $\mathbf{v}_m(t)$. To provide a suitable framework for on-line computation using recursive orthogonal decomposition conveniently (e.g. the Givens rotation described later), Eqn (3-5) can be represented as an augmented matrix

$$\begin{bmatrix} \mathbf{R}(t) & \mathbf{v}_m(t) \\ \mathbf{O}_1 & \mathbf{O}_2 \end{bmatrix} \tag{3-7}$$

To track variations in the parameters of the system under investigation, it is necessary to weight the most recent input-output and residuals data. Define the weighting matrix Λ as a diagonal matrix

$$\Lambda(t) = diag \ [\ \lambda^{t-1} \quad \lambda^{t-2} \quad \cdots \quad \lambda \quad \lambda^0 \ ] \tag{3-8}$$

where $0<\lambda<1$ is called the weighting factor or the forgetting factor. Therefore the least squares error function, namely, the residual sum of squares (RSS) in (3-6), is modified to

$$R\tilde{S}S(t) = \tilde{\varepsilon}^T(t)\tilde{\varepsilon}(t) = \varepsilon^T(t)\Lambda(t)\varepsilon(t) = \sum_{i=1}^{t}\lambda^{t-i} \ \varepsilon^2(i) \tag{3-9}$$

where ~ denotes the weighted variables, matrices and vectors which are multiplied by $\lambda^{1/2}$ or $\Lambda^{1/2}$, and the weighted error vector $\tilde{\varepsilon}(t) = \tilde{\mathbf{y}}(t) - \tilde{\Phi}(t)\hat{\theta}(t)$. When implemented with the forgetting factor $\lambda \neq 1$, $\mathbf{R}(t-1)$ and $\mathbf{v}_m(t-1)$ in Eqn (3-4) should be weighted and correspondingly the orthogonal transformation between (3-4) and (3-5) can be represented as

$$\begin{bmatrix} \lambda^{1/2}\tilde{\mathbf{R}}(t-1) & \lambda^{1/2}\tilde{\mathbf{v}}_m(t-1) \\ \phi(t) & y(t) \end{bmatrix} \quad => \quad \begin{bmatrix} \tilde{\mathbf{R}}(t) & \tilde{\mathbf{v}}_m(t) \\ \mathbf{O}_1 & \mathbf{O}_2 \end{bmatrix} \tag{3-10}$$

To ensure clarity, the symbol $\tilde{}$ will not be used in the remainder of this paper, because the following discussion only relates to the exponential windowing method.

The key problem is how to find an orthogonal transformation matrix $Q_i^T(t)$ which can implement the above transformation. Such a matrix, $Q_i^T(t)$, can be formed as the product of m Givens rotations (Gentleman 1973, Gentleman 1974) as

$$Q_i^T(t) = G_m(t)\, G_{m-1}(t) \,\cdots\, G_1(t)$$

where the Givens rotation factors are

$$G_i(t) = \begin{bmatrix} I_{i11} & O_{i12} & O_{i13} & O_{i14} \\ O_{i21} & \cos\beta_i(t) & O_{i23} & \sin\beta_i(t) \\ O_{i31} & O_{i32} & I_{i33} & O_{i34} \\ O_{i41} & -\sin\beta_i(t) & O_{i43} & \cos\beta_i(t) \end{bmatrix} \qquad (i=1,...,m) ,$$

where the zero matrices $O_{(.)}$ and unity matrices $I_{(.)}$ have corresponding dimensions (here the subscriptions are not dimensions). $G_i(t)$ is of dimension $t \times t$, β_i is the rotation angle and the subscript i denotes the values associated with the processing of the ith column.

As mentioned above the reorthogonalization can be executed using the augmented matrix in which orthogonal transformation matrices do not need to be expressed explicitly. At time t, define the $(m+1)\times(m+1)$ matrix

$$\underline{C} = \begin{bmatrix} \lambda^{1/2}\bar{R}(t-1) & \lambda^{1/2}\bar{v}_m(t-1) \\ \phi(t) & y(t) \end{bmatrix} \tag{3-11}$$

Then the orthogonal QR transformation can be applied to $\underline{C}$ row by row. For example the transformation of the i'th and j'th row is presented as follows

$$row\ i: 0, ..., 0, \underline{c}_{i,k}, \underline{c}_{i,k+1}, ..., \underline{c}_{i,m+1} \quad => \quad 0, ..., 0, \underline{c}_{i,k}^*, \underline{c}_{i,k+1}^*, ..., \underline{c}_{i,m+1}^*$$

$$row\ j: 0, ..., 0, \underline{c}_{j,k}, \underline{c}_{j,k+1}, ..., \underline{c}_{j,m+1} \quad => \quad 0, ..., 0, 0, \underline{c}_{j,k+1}^*, ..., \underline{c}_{j,m+1}^*$$

where $*$ denotes the new value after transformation. The elements can be computed using the following equations:

$$\underline{c}_{i,k}^* = \sqrt{\underline{c}_{i,k}^2 + \underline{c}_{j,k}^2} \tag{3-12a}$$

$$\cos\beta = \frac{\underline{c}_{i,k}}{\underline{c}_{i,k}^*} , \qquad \sin\beta = \frac{\underline{c}_{j,k}}{\underline{c}_{i,k}^*} \tag{3-12b,c}$$

$$\underline{c}_{i,p}^* = \underline{c}_{i,p} \cos\beta + \underline{c}_{j,p} \sin\beta , \qquad \underline{c}_{j,p}^* = -\underline{c}_{i,p} \sin\beta + \underline{c}_{j,p} \cos\beta \tag{3-12d,e}$$

where $p = k+1, k+2, ..., m+1$. The augmented matrix (3-7) is then obtained and $\hat{\theta}(t)$ is computed using back-substitution.

If the regression matrix $\Phi(t)$ is of full rank (rank $m_r = m$) the Givens procedure described above should be numerically stable and accurate just like other recursive QR algorithms. But if the system structure is unknown, the computation inevitably involves all candidate variables and some of these variables may be linearly dependent so that $\Phi(t)$ will be less than full rank ($m_r < m$). This means that $\Phi^T(t)\Phi(t)$ may be singular and there may no longer be a unique solution. Nevertheless, $\Phi(t)$ can still be decomposed as

$$Q^T(t)\Phi(t) = \begin{bmatrix} R(t) \\ O(t) \end{bmatrix}$$

by using the orthogonal transformations, where $R(t)$ is a real upper triangular matrix with $(m-m_r)$ zero diagonal elements and m_r positive diagonal elements and the row vectors of $R(t)$ corresponding to the zero diagonal elements become zero vectors. If the columns of $\Phi(t)$ can be permuted so that the selected column vector at each stage minimizes RSS, then $\Phi(t)$ can be decomposed as

$$Q^T\Phi\Pi = \begin{bmatrix} R_{11} & R_{12} \\ O_3 & O_4 \end{bmatrix} \tag{3-13}$$

where

$$R_{11} = \begin{bmatrix} r_{11} & r_{12} & \cdot & \cdot & \cdot & r_{1m_r} \\ 0 & r_{22} & \cdot & \cdot & \cdot & r_{2m_r} \\ \cdot & & 0 & & & \cdot \\ \cdot & & & \cdot & & \cdot \\ \cdot & & & & \cdot & \cdot \\ 0 & 0 & \cdot & \cdot & 0 & r_{m_r m_r} \\ & & O_5 & & & \end{bmatrix} = \begin{bmatrix} R_{m_r} \\ O_5 \end{bmatrix},$$

$$R_{12} = \begin{bmatrix} r_{1,m_r+1} & \cdot & \cdot & \cdot & r_{1m} \\ r_{2,m_r+1} & \cdot & \cdot & \cdot & r_{2m} \\ \cdot & & \cdot & & \cdot \\ \cdot & & & \cdot & \cdot \\ \cdot & & & & \cdot \\ r_{m_r,m_r+1} & \cdot & \cdot & \cdot & r_{mm} \\ & & O_6 & & \end{bmatrix}.$$

O_3, O_4, O_5, O_6 are zero matrices with dimensions $(t-m)\times m_r$, $(t-m)\times(m-m_r)$, $(m-m_r)\times m_r$, $(m-m_r)\times(m-m_r)$ respectively and Π is a $t \times t$ permutation matrix. To clearly express the operations of the argument "(t)" was ignored in (3-13) and such a simplification will be used throughout the remainder of this section.

A generalized inverse of Φ, e.g. Φ^*, can always be found and $\Phi^* y$ is a solution of the normal equations with all the candidate variables. But there are many such generalized inverses and corresponding solutions. So the significant regression variables must

be found from the linearly independent columns and the corresponding parameters determined.

The symbol $(\ )^{\Pi}$ will be used to represent a permuted matrix. Inspection of (3-13) shows that the first m_r column vectors of Φ^{Π} $(=\Phi\Pi)$ are linearly independent. Partitioning yields

$$Q = [\ Q_{m_r}\ \ Q_{m-m_r}\ \ Q_{l-m}\ ] \tag{3-14}$$

and

$$\Phi^{\Pi} = [\Phi_{m_r}^{\Pi}\ \ \Phi_{m-m_r}^{\Pi}\ ]$$

where Q_{m_r}, Q_{m-m_r} and Q_{l-m} are of dimension $t \times m_r$, $t \times (m-m_r)$ and $t \times (t-m)$ respectively, and $\Phi_{m_r}^{\Pi}$, $\Phi_{m-m_r}^{\Pi}$ are of dimension $t \times m_r$ and $t \times (m-m_r)$. Eqn. (3-13) can be written as

$$\begin{bmatrix} Q_{m_r}^T \\ Q_{m-m_r}^T \\ Q_{l-m}^T \end{bmatrix} [\Phi_{m_r}^{\Pi}\ \ \Phi_{m-m_r}^{\Pi}] = \begin{bmatrix} Q_{m_r}^T \Phi_{m_r}^{\Pi} & Q_{m_r}^T \Phi_{m-m_r}^{\Pi} \\ Q_{m-m_r}^T \Phi_{m_r}^{\Pi} & Q_{m-m_r}^T \Phi_{m-m_r}^{\Pi} \\ Q_{l-m}^T \Phi_{m_r}^{\Pi} & Q_{l-m}^T \Phi_{m-m_r}^{\Pi} \end{bmatrix} . \tag{3-15}$$

Inspection of (3-15) shows that

$$\mathbf{R}_{11} = \begin{bmatrix} Q_{m_r}^T \Phi_{m_r}^{\Pi} \\ Q_{m-m_r}^T \Phi_{m_r}^{\Pi} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{m_r} \\ \mathbf{O}_5 \end{bmatrix} , \quad \mathbf{R}_{12} = \begin{bmatrix} Q_{m_r}^T \Phi_{m-m_r}^{\Pi} \\ Q_{m-m_r}^T \Phi_{m-m_r}^{\Pi} \end{bmatrix} = \begin{bmatrix} Q_{m_r}^T \Phi_{m-m_r}^{\Pi} \\ \mathbf{O}_6 \end{bmatrix} .$$

Based on (3-14) the orthogonal output vector is given by

$$Q^T y = \begin{bmatrix} Q_{m_r}^T y \\ Q_{m-m_r}^T y \\ Q_{l-m}^T y \end{bmatrix} = \begin{bmatrix} v_{m_r} \\ v_{m-m_r} \\ v_{l-m} \end{bmatrix} .$$

Using the candidate variables corresponding to the first m_r column vectors, which are linearly independent, to represent the dynamics of the underlying system the equations

$$Q_{m_r}^T \Phi_{m_r}^{\Pi} \hat{\theta}_{m_r} = Q_{m_r}^T y$$

can have a unique solution and the estimate $\hat{\theta}_{m_r}$ can hence be obtained using the triangular system $\mathbf{R}_{m_r} \hat{\theta}_{m_r} = v_{m_r}$. In practice roundoff errors always exist in Φ and further roundoff errors are involved in forming the elements of $\mathbf{R}$. This means that some diagonal elements r_{ii} and the corresponding rows may be approximately but not exactly zero. It is necessary to identify which column vectors are linearly independent. Using specified criteria the significant regressors can be selected from all the candidate variables. Assuming that the first m_s column vectors are selected as the significant

regressors, the estimate $\hat{\theta}_{m_s}$ can be obtained from $R_{m_s} \hat{\theta}_{m_s} = v_{m_s}$, where R_{m_s} is the top-left portion of the upper triangular matrix R_{m_r}. The above operations can still be implemented conveniently in the augmented matrix

$$
\begin{bmatrix} R & v_m \\ O_1 & O_2 \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & v_m \\ O_1 & & O_2 \end{bmatrix} = \begin{bmatrix} R_{m_r} & R_{m-m_r} & v_{m_r} \\ O_5 & O_6 & v_{m-m_r} \\ O_1 & & O_2 \end{bmatrix} .
\tag{3-16}
$$

Therefore, it is important to ensure that all significant regressors are permuted to the left portion of Φ^Π in the recursive orthogonal process.

4. On-line Structure Detection

Since complete input-output data and orthogonal vector storage are not possible in on-line processing the extension of the off-line detection procedure (Chen et al 1989) to the on-line case must resolve two problems: i) which elements or functions of elements represent the contribution of a regression variable to the output; ii) how to reformulate the orthogonal vectors based on the selected optimal regressors at every selection step.

In the remainder of this section the notation "(t)" will be ignored when all quantities being discussed are available up to the time t. For example, $\Phi(t)$ will be written as Φ.

Dividing (3-6) by $y^T y$ gives the normalized RSS ($NRSS$), given by

$$
NRSS = 1 - \frac{v_{m_s}^T v_{m_s}}{y^T y} = 1 - \sum_{i=1}^{m_s} ERR_i .
\tag{4-1}
$$

where $ERR_i = v_i^2 / y^T y$. This is defined as the error reduction ratio (ERR) of the orthogonal vector q_i (Liu et al 1987, Billings and Chen 1989), it is obtained by transforming the first i columns of Φ. Geometrically, v_i represents the projection of the output in the direction of the ith orthogonal vector. This constituent has been normalized, because $\|q_i\|=1$. The value of v_i^2 can be conveniently utilized to select significant regressors from all the candidate regression variables by using a forward search procedure so that every selected variable minimizes $NRSS(t)$ in every selection step. The number of the selected regressors at time t, named m_s, will normally be less than the number of candidate variables, m. Therefore the selection will be continued m_s steps until

$$
NRSS_{m_s}(t) = 1 - \sum_{i=1}^{m_s} ERR_i \le \xi_s .
\tag{4-2}
$$

where ξ_s is a pre-set tolerance. Generally, the selected variables at every computational period are different. The number of these regressors is time-varying, named $m_s(t)$. If $m_s(t)$ regression variables (corresponding to the first $m_s(t)$ columns of $\Phi(t)$), are determined as significant regressors, the detailed form of (4-2) is

$$NRSS_{m_s(t)}(t) = 1 - \sum_{i=1}^{m_s(t)} ERR_i(t) \leq \xi_s \ , \tag{4-3}$$

It is important to select the optimal regressors and to compute $NRSS_{m_s(t)}(t)$ with $v_i(t)$ for limited memory space in on-line identification.

At the j'th selection step off-line structure detection initially considers all the Φ_p, $p=j,...,m$, as candidates one of which will be combined with the previously formed orthogonal vectors to form $\mathbf{q}_j$. Then for $p=j,...,m$, compute $v_{j(p)}^2=(\mathbf{q}_{j(p)}\mathbf{y})^2$ and select the j'th optimal regressor with the maximum $v_{j(p)}^2$ to minimize RSS. Since in on-line processing all the previous input-output data or the formed orthogonal vectors cannot be stored, the computation must be limited within the augmented matrix (3-16). The key procedure for on-line structure detection is to utilize $\mathbf{R}$ and $\mathbf{v}_m$ efficiently to obtain $v_{j(p)}$, $p=1,...,m$, $j=1,...,m_s(t)$, and to permute implicitly $m_s(t)$ optimal regressors in the first $m_s(t)$ columns of Φ^{fi}, where ^{fi} denotes the permuted matrix at the $m_s(t)$ selection step. Since $\mathbf{R}$ indicates the relationship between $\mathbf{Q}$ and Φ, the different triangular forms of $\mathbf{R}$ correspond to different constituents of the orthogonal space mapped from the columns of Φ. The corresponding rotated form of $\mathbf{v}_m$ can indicate the contribution of all the new orthogonal vectors to the output in the new orthogonal space. Consider the matrix

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} & \cdot & \cdot & \cdot & r_{1m} & v_1 \\ 0 & r_{22} & r_{23} & \cdot & \cdot & \cdot & r_{2m} & v_2 \\ \cdot & 0 & r_{33} & \cdot & \cdot & \cdot & r_{3m} & v_3 \\ \cdot & \cdot & 0 & \cdot & & & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & & & \cdot & \cdot \\ \cdot & \cdot & \cdot & & \cdot & & \cdot & \cdot \\ 0 & 0 & 0 & \cdot & \cdot & 0 & r_{mm} & v_m \end{bmatrix} \cdot \tag{4-4}$$

This shows that ϕ_1 is used to compose $\mathbf{q}_1$, both ϕ_1 and ϕ_2 are used to constitute $\mathbf{q}_2$, etc, where here v_1 is equivalent to $v_{1(1)}$. This is based on the expansion of $\mathbf{Q}=\Phi\mathbf{R}^{-1}$, i.e.

$$\mathbf{q}_1=\frac{1}{r_{11}}\phi_1 \ , \quad \mathbf{q}_2=\frac{1}{r_{22}}\phi_2-\frac{r_{12}}{r_{22}r_{11}}\phi_1 \ , \quad ......$$

and $v_1=\mathbf{q}_1\mathbf{y}$. But the following matrix

$$
\begin{bmatrix}
\underline{r}_{11} & \underline{r}_{12} & \underline{r}_{13} & \cdot & \cdot & \cdot & \underline{r}_{1m} & \underline{v}_1 \\
\underline{r}_{21} & 0 & \underline{r}_{23} & \cdot & \cdot & \cdot & \underline{r}_{2m} & \underline{v}_2 \\
0 & \cdot & \underline{r}_{33} & \cdot & \cdot & \cdot & \underline{r}_{3m} & \underline{v}_3 \\
\cdot & \cdot & & 0 & \cdot & & \cdot & \cdot \\
\cdot & \cdot & & \cdot & & & \cdot & \cdot \\
\cdot & \cdot & & \cdot & & & \cdot & \cdot \\
0 & 0 & 0 & \cdot & \cdot & 0 & \underline{r}_{mm} & \underline{v}_m
\end{bmatrix} ,
\tag{4-5}
$$

which is a variant of a triangular matrix shows that ϕ_2 is used to compose the first orthogonal vector $\mathbf{q}_1$ and ϕ_2 and ϕ_1 are used to constitute the second orthogonal vector $\mathbf{q}_2$, where $\underline{v}_1$ is taken as equivalent to $v_{1(2)}$. This produces the factor of $\mathbf{Q}=\Phi\underline{\mathbf{R}}^{-1}$ described as

$$
\mathbf{q}_1 = \frac{1}{\underline{r}_{12}}\phi_2 , \quad \mathbf{q}_2 = \frac{1}{\underline{r}_{21}}\phi_1 - \frac{\underline{r}_{11}}{\underline{r}_{21}\underline{r}_{12}}\phi_2 , \quad \dots\dots
$$

and $\underline{v}_1 = \mathbf{q}_1 \mathbf{y}$. The underlines in the above notation is only used to distinguish these quantities from those of (4-4). To easily execute the back-substitution the columns of $\mathbf{R}$, named $\mathbf{r}_i$, $i=1.....m$, will be permuted and retriangularized.

Expanding $\Phi=\mathbf{QR}$ gives

$$
\phi_1 = \mathbf{q}_1 r_{11}
$$

$$
\cdots
$$

$$
\phi_j = \sum_{i=1}^{j} \mathbf{q}_i r_{ij} = \sum_{i=1}^{j-1} \mathbf{q}_i r_{ij} + \sum_{i=j}^{j} \mathbf{q}_i r_{ij} \qquad (j = 2, \dots, m)
\tag{4-6}
$$

Inspection of the above shows that if a candidate vector, for example ϕ_k $(j\leq k)$, which is selected as the j'th optimal regressor is permuted in the j'th column of Φ, the operation of reorthogonalization only needs to be applied to r_{ip}, $i=j.....m$, $p=j.....m$ using the Givens rotation in (3-12) where the orthogonal formation matrix is not expressed explicitly. Such a computational process can be expressed concisely in the following equations:

$$
\mathbf{Q}^T \Phi = \begin{bmatrix} \mathbf{R} \\ \mathbf{O}_1 \end{bmatrix} = \begin{bmatrix} \mathbf{r}_1 \ \mathbf{r}_2 & \cdots & \mathbf{r}_j & \cdots & \mathbf{r}_k & \cdots & \mathbf{r}_m \\ & & & \mathbf{O}_1 & & & \end{bmatrix}
\tag{4-7a}
$$

$$
\mathbf{Q}^T \Phi \Pi_{jk} = \begin{bmatrix} \mathbf{R} \\ \mathbf{O}_1 \end{bmatrix} \Pi_{jk} = \begin{bmatrix} \mathbf{r}_1 \ \mathbf{r}_2 & \cdots & \mathbf{r}_k & \cdots & \mathbf{r}_j & \cdots & \mathbf{r}_m \\ & & & \mathbf{O}_1 & & & \end{bmatrix}
\tag{4-7b}
$$

$$
\mathbf{Q}_{re_j}^T \mathbf{Q}^T \Phi \Pi_{jk} = \mathbf{Q}_{re_j}^T \begin{bmatrix} \mathbf{R} \\ \mathbf{O}_1 \end{bmatrix} \Pi_{jk} = \begin{bmatrix} \mathbf{r}_1 \ \mathbf{r}_2 & \cdots & \mathbf{r}_j^* & \cdots & \mathbf{r}_k^* & \cdots & \mathbf{r}_m^* \\ & & & \mathbf{O}_1 & & & \end{bmatrix}
\tag{4-7c}
$$

where Π_{jk} is the permutation matrix for exchanging ϕ_j and ϕ_k (the equivalent effect of which is to exchange $\mathbf{r}_j$ and $\mathbf{r}_k$), $\mathbf{Q}^T_{re_j}$ denotes the orthogonal transformation matrix of the j'th selection step which retriangularizes $\begin{bmatrix} \mathbf{R} \\ \mathbf{O}_1 \end{bmatrix} \Pi_{jk}$ (namely reorthogonalization) and * denotes the new results after retriangularization. Since $\mathbf{R}$ contains the information from the previously formed orthogonal vectors and candidate vectors, the operations are only implemented in part of $\mathbf{R}$ (see the right-hand sides of (4-7a)-(4-7c)), so that the whole process avoids the storage of the orthogonal vectors and candidate vectors.

Note that

$$\mathbf{v}_m = [\ \mathbf{v}_1 \ \mathbf{v}_2 \ \cdots \ \mathbf{v}_j \ \cdots \ \mathbf{v}_k \ \cdots \ \mathbf{v}_m\ ]^T$$
$$= [\ \mathbf{q}_1^T\mathbf{y} \ \ \mathbf{q}_2^T\mathbf{y} \ \cdots \ \mathbf{q}_j^T\mathbf{y} \ \cdots \ \mathbf{q}_k^T\mathbf{y} \ \cdots \ \mathbf{q}_m^T\mathbf{y}\ ]^T \ . \tag{4-8}$$

This shows that when the j'th optimal regressor has been determined the reorthogonalizing computation of the output vector is only applied to some elements of $\mathbf{v}_m$, $v_i (=\mathbf{q}_i^T\mathbf{y})$, $i=j,...,m$ and this computation can be implemented at the same time as retriangularizing $\begin{bmatrix} \mathbf{R} \\ \mathbf{O}_1 \end{bmatrix} \Pi_{jk}$, say.

Based on (3-6) the j'th optimal regressor should be the candidate variable with the maximum of $v_{j(p)}^2$, $p=j,...,m$, so that the RSS can be minimized. Such a computation for $v_{j(p)}^2$ can only use $\mathbf{R}$. If a candidate variable, for example ϕ_k, is selected as the j'th optimal regressor, $v_{j(k)}^2$ should be the the maximum of all the $v_{j(p)}^2$ when the elements r_{ik}, $i=j+1,...,m$, are transformed to zero in the retriangularization. Since $\mathbf{R}$ is upper triangular, the columns of $\mathbf{R}$, $\mathbf{r}_p$, $p=j,...,m$, have the following form

$$\mathbf{r}_p = [r_{1p} \ r_{2p} \ \cdots \ r_{jp} \ \cdots \ r_{pp} \ 0 \ \cdots \ 0]^T \ . \tag{4-9}$$

This means that only the nonzero elements r_{ip}, $i=j,...,p$ are involved in the computation, namely, $v_{j(p)}$ can be obtained by applying $p-j$ Givens rotations to v_j using $\mathbf{r}_p$ and v_i, $i=j,...,p$. First set $v_{j(j)}=v_j$ and two auxiliary variables $r_{j(p)}^{(j)}=r_{jp}$ and $v_{j(p)}^{(j)}=v_j$. Using (3-12) $v_{j(p)}$ can be calculated using the following procedure:

For $i=j+1,...,p$,

$$r_{j(p)}^{(i)} = \sqrt{(r_{j(p)}^{(i-1)})^2 + r_{ip}^2} \tag{4-10a}$$

$$v_{j(p)}^{(i)} = v_{j(p)}^{(i-1)} \frac{r_{j(p)}^{(i-1)}}{r_{j(p)}^{(i)}} + v_i \frac{r_{ip}}{r_{j(p)}^{(i)}} \ . \tag{4-10b}$$

Finally $v_{j(p)}^2 = (v_{j(p)}^{(p)})^2$.

Once the optimal regressor is determined the new orthogonal space can be formed using the procedures of (4-7). The selection procedures are summarized in Section 6.

5. Initialization

To initialize $R(t)$ and $v_m(t)$, let all the elements in $R(0)$ and $v_m(0)$ equal a small positive number to prevent division by zero.

The candidate variables represent a library of regressors from which sub-models can be generated. These variables should be sufficient to describe the dynamics of the system under test in a wide range of operation. The GFSE algorithm also allows the addition of some "empty" variables at the beginning of the computation by assigning all the elements of the associated columns of $R(0)$ as very small numbers. Variables which have not been involved in the initial regression model can then replace these "empty" variables in the computational process. Removing some useless candidate variables can also be easily realized by substituting all the elements of the associated columns of $R(t)$ with very small numbers. An alternative method of adjusting candidate variables is to extend or contract on-line the dimension of the augmented matrix (3-7). All the new elements are initialized to very small numbers at time t and then the data associated with these new variables are added successively to the computation so that the data will have the effect on the estimates from time t.

A constant forgetting factor λ slightly less than 1 is usually chosen. Define

$$N_a = \frac{1}{1 - \lambda} \tag{5-1}$$

as the asymptotic memory length (Clarke and Gawthop 1975). This means that after N_a samples the square of the current error will have about 36% of the current contribution to least squares error criterion $RSS(t)$. If the system under test remains approximately time invariant over N_a samples, a suitable choice of λ can then be made from (5-1). The choice of λ affects both the tracking ability and noise-sensitivity of the estimators. Typical choices of λ are between 0.98 to 0.995. In some applications where the system dynamics change suddenly λ can be designed as a time-varying parameter to enhance the tracking performance.

The design of the tolerance ξ_s in (4-3) depends on a knowledge of the system under test, but it is often difficult to decide on an optimal value at the beginning of the identification. A very small ξ_s will lead to unnecessarily complex sub-models. To avoid such an inadequate design, some statistical tests can be used.

6. Summary of the GFSE's Procedure

First, initialize the estimator according to Section 5, including initial values of the elements of the augmented matrix, candidate variables, forgetting factor λ and tolerance ξ_s. Then perform the computation. At the beginning of the computation at time instant t, the augmented matrix is

$$\begin{bmatrix} \mathbf{R}(t-1) & \mathbf{v}_m(t-1) \\ \mathbf{O}_1 & \mathbf{O}_2 \end{bmatrix} \tag{6-1}$$

(i) Multiply $\mathbf{R}(t-1)$ and $\mathbf{v}_m(t-1)$ by $\lambda^{1/2}$, and put the data, $[\phi_1(t) \cdots \phi_m(t) \ y(t)]$, at the $(m+1)$'th row to obtain

$$\begin{bmatrix} \lambda^{1/2}\mathbf{R}(t-1) & \lambda^{1/2}\mathbf{v}_m(t-1) \\ \phi_1(t) \quad \cdots \quad \phi_m(t) & y(t) \end{bmatrix} = \begin{bmatrix} \lambda^{1/2}\mathbf{R}(t-1) & \lambda^{1/2}\mathbf{v}_m(t-1) \\ \phi(t) & y(t) \end{bmatrix} \tag{6-2}$$

(ii) Using Givens rotation, produce the new augmented matrix

$$\begin{bmatrix} \mathbf{R}(t) & \mathbf{v}_m(t) \\ \mathbf{O}_1 & \mathbf{O}_2 \end{bmatrix}. \tag{6-3}$$

(iii) With (4-10), compute $v_{j(p)}(t)$, $p=1,....m$, $j \geq 1$ and choose the j'th optimal regressor with the maximum $v_{j(p)}^2$ by selecting the columns of $\mathbf{R}(t)$.

(iv) According to the result from Step (iii) (e.g., the kth variable has been selected) exchange the positions of the current j'th and k'th columns of $\mathbf{R}(t)$, and then retriangularize $\mathbf{R}(t)$ and rotate $\mathbf{v}_m(t)$ using Givens rotation.

(v) Compute $NRSS(t)$ using (4-3). If the critical value,

$$NRSS_j(t) = 1 - \sum_{i=1}^{j} ERR_i(t) \leq \xi_s .$$

perform the next step; otherwise perform statistical tests to decide whether to return to step (iii) to select more regressors. Suppose that m_s regressors have been selected, then the computational matrix is

$$
\begin{bmatrix}
r_{11}(t) & . & . & . & r_{1,m_s}(t) & . & . & . & v_1(t) \\
0 & . & & & . & & & & . \\
. & . & . & & . & & & & . \\
. & & . & . & r_{m_s m_s}(t) & & & & v_{m_s}(t) \\
. & & & & 0 & . & & & . \\
. & & & & . & & . & & . \\
. & & & & . & & & . & . \\
0 & . & . & . & 0 & & & . & 0
\end{bmatrix} .
$$

(vi) Using back-substitution solve for the parameters $\hat{\theta}_i(t)$, $i=1,...,m_s$, from $\mathbf{R}_{m_s}(t)$, which is the top-left triangular portion of the final $\mathbf{R}(t)$, and $\mathbf{v}_{m_s}(t)$ which consists of the first m_s elements.

(vii) Compute the residual at the time instant t $\varepsilon(t) = y(t) - \sum_{i=1}^{m_s} \phi_i(t)\,\hat{\theta}_i(t)$ This result will become part of the next input signal if the initially designed model includes noise terms.

7. Properties of the GFSE Algorithm

Although the properties of QR decomposition depend on the condition number $\kappa[\Phi]$ which is defined as the ratio of the largest to the smallest nonzero singular value of Φ (Golub and Styan 1973, Chen et al 1989), the problems relevant to positive definitness of $(\Phi^T\Phi)^{-1}$ and the effects of accumulated roundoff errors due to updating the parameter vector (Ardalan 1986, Ardalan and Alexander 1987) are largely avoided. When the GFSE algorithm is applied in parameter estimation the numerical stability will be similar to other recursive QR algorithms. When the GFSE algorithm is applied in structure detection the procedure consists of permutation and reorthogonalization. It is obvious that the permutation does not introduce additional roundoff errors. The reorthogonalization utilizes the 2×2 Givens rotation factor on each of the two column vectors formed by the two corresponding rows. Such an orthogonal rotation does not increase the magnitude of roundoff errors.

For systems with time-varying structure, the GFSE algorithm always, based on the contribution of the candidate variables make to the output, determines which variables should be included in the final model. Only after the model structure have been determined can the corresponding parameter estimates be calculated. If the system structure does not change over time the parameter estimates will exhibit the same convergent properties of the underlying recursive parameter estimation algorithms.

The stability of a recursive estimator is related to the stability of the noise model. For the recursive prediction error method (RPEM) applied in on-line parameter identification for the NARMAX model Chen and Billings (1988) utilized the differential equation approach to investigate the convergence property of this algorithm. They were able to show that the stability of the estimator is associated with a stable noise model and power terms in $e(t)$ should be avoided because these can become explosive and can induce instability. This consideration is carried over to the GFSE algorithm. But it should be noted that the GFSE algorithm is distinguished from other RLS parameter estimation algorithms because of the on-line detection of the structure. During computation, both the model structure and parameters can be identified. Therefore the GFSE algorithm provides a possibility to track nonlinear noise models on-line. In practical simulation tests which will be described in Section 8, the power of $\varepsilon(k)$, $(k = t-1, ..., t-n_\varepsilon)$ has been set to a value larger than one (e.g. two) and the estimator still worked very well. However nonlinear systems are in general very complex and the convergence analysis for on-line structure and parameter identification becomes very complex and more work is required to investigate these issues.

8. Numerical Results

To illustrate the GFSE algorithm, a time-varying system will be simulated where both the structure and parameters change suddenly. The first 250 data points were generated using a linear model

$$z(t) = 0.5\, z(t-1) + u(t-1)$$

$$y(t) = z(t) + e(t)$$

and the second 250 data points by a NARMAX model

$$z(t) = 0.2\, z(t-1) + 0.8\, u(t-1) + 0.1\, u^2(t-1)$$

$$y(t) = z(t) + e(t)$$

Notice that the models relating $y(t)$ to $u(t)$ will involve coloured noise terms.

In the test, the input signal $u(t)$ was an independent sequence of uniform distribution with zero mean and the variance 1.03, and the noise signal $e(t)$ was a Gaussian white noise with zero mean and variance 0.005. The output and input signals are illustrated in Fig. 1. To demonstrate that it is not possible to fit a single global model to this system and that the GFSE algorithm can give a reasonable time-varying model, the noise-free case was considered initially. With the off-line method described in

(Chen et al 1989), two global NARMAX models were estimated based on the specifications 1)$n_y=n_u=n_\varepsilon=2$, $n_l=2$; 2)$n_y=n_u=n_\varepsilon=2$, $n_l=3$. The former model has 28 candidate variables and the latter has 84. These two models represent two possible expansions, one including only up to quadratic and the other only up to cubic terms, which were fitted to the total data set. To aid the estimation, Akaike's information criterion was applied to the computation with a cutoff of $AIC=4.0$. The two models gave large RSS $(=\sum_{t=1}^{500}\varepsilon^2(t))$ values, 9.926 and 7.602, respectively and both have a complex structure (with 8 and 13 terms, see Table 1), compared with the real structure. The predicted outputs and residuals produced from the models are plotted in Fig. 2 and 3. Although the data were noise-free the residuals are large. The off-line validity tests illustrated in Fig. 4 and 5 (Billings and Voon 1983, Billings and Voon 1986) indicate that the two fitted models are an inadequate representation for this system. The on-line validity tests (see Appendix) show that neither of these global models can match the dynamics at all time points, Fig. 6 and 7.

The new on-line structure detection and parameter estimation GFSE algorithm was then applied to the same data set. The initial design was $n_y=n_u=n_\varepsilon=n_l=2$, $\xi_s=0.01$, $\lambda=0.97$. The estimates after 250 and 500 iterations are given in Table 2 and the predicted output and residuals are shown in Fig. 8. The value of $\sum_{t=1}^{500}\varepsilon^2(t)$ becomes much smaller (0.374), compared with the values obtained in the previous global models. It is significant that the residuals become relatively smooth in two intervals where the system structure is stable. In the initial stage of processing, the estimates are still converging and the predicted output can not track the real output well and the residuals have a few large jumps. This phenomenon also occurs between the 251th and 325th point when the sudden model change occurs. Because the system structure at the 251th point suddenly changes the estimator requires time to respond and to modify the previous structure and parameters. The error suddenly increases in this short interval and then decreases exponentially subject to the pre-set forgetting factor λ. This variation in the structure can also be observed on the plots of the three parameters, Fig. 9. The on-line validity tests (Fig. 10) provide the same information and show that this time-varying model is acceptable.

Noise was then added to the above data set by operating on the noisy output $y(t)$ in the above models. The input and output are shown in Fig. 11. the initial design was given by $n_y=n_u=n_\varepsilon=2$, $n_l=2$, $\xi_s=0.01$, $\lambda=0.97$. The fitted sub-models at the 250 th and 500 th data points are listed in Table 3 and the predicted output and the residuals are plotted in Fig. 12. Due to the presence of noise, the value of $\sum_{t=1}^{500}\varepsilon^2(t)$ increases to 1.91 and the

curve of the residuals becomes ragged. Notice that even when the noise is added into the system the loss function is still lower than that obtained from the two global models in the noise-free case. The corresponding on-line validity tests (Fig. 14) indicate that the time-varying model is adequate. The analysis has shown that the three estimated parameters (Fig. 13) are tracked over the variation of the structure even when the data is disturbed by noise. The values of the parameters are still very close to the true values.

The effect of the forgetting factor λ has also been tested for this system. If λ has a small value, e.g. 0.90, the tracking ability is good, see Fig. 15 and Table 4. This can easily be observed from the parameter curves of $y(t-1)$ and $u^2(t-1)$, Fig. 16. But the predicted output quickly deteriorates due to the noise disturbances and this can also be clearly seen at 350'th to 400'th point on the curves of $y(t-1)$ and $u^2(t-1)$. If λ is set to a large value, e.g. 0.99, the response to a large variation in the system structure is slow so that the parameters are still converging even when the system structure has become stabilized. Therefore the error is large and the value of $\sum_{t=1}^{500} \varepsilon^2(t)$ rises to 6.899. This result is illustrated in Fig. 17, 18 and Table 5.

9. Conclusions

Most existing recursive parameter estimation algorithms work on the assumption that the system structure is time-invariant and has been determined. Algorithms based on these principles cannot produce an accurate solution for systems with time-varying structure. Lattice algorithms and the RMGS algorithm can provide the facility for the on-line adjustment of the structure of linear models using order-recursion techniques. However some key properties of the regressors, which are used in these algorithms, do not exist in most nonlinear models so it is difficult to extend these algorithms to general nonlinear systems. It is therefore necessary to develop new recursive algorithms for both detecting the system structure and estimating the system parameters on-line.

A recursive orthogonal QR decomposition algorithm, called the GFSE has been derived in this paper. This new algorithm overcomes the disadvantages of lattice algorithms, RMGS and other existing recursive parameter estimation algorithms and also preserves the numerical stability and accuracy of orthogonal decomposition methods. The algorithm does not require the storage of all the original data, nor the orthogonal vectors, and consequently memory space is saved. Both the structure and parameters can be updated on-line for a wide class of nonlinear systems. Overparameterisation is

avoided and flexibility in the selection of the regressors is maintained. Simulations using the GFSE algorithm illustrate the ability to track the variation of both system structure and parameters.

Since the estimation algorithm involves structure detection, the computational cost is higher than would be the case if just parameter estimation were performed and hence the use of parallel processing techniques may be appropriate in some applications which require fast computation.

The results in the present paper only expresses the generalized computational procedures of the new algorithm and do not refer to the special treatment required in some applications. For example, some control problems do not require variation of the structure at every time step and updating of the structure may therefore be reduced. When the system structure and parameters are stable, some computational procedures become redundant and this leads to the development of selective updating algorithms which only update the structure and parameters when a sufficient improvement can be achieved. These and other related problems will be addressed in forthcoming papers.

Acknowledgments

Table 1a: Global Model with 28 terms, Noise-free Case			
term	$\hat{\theta}$	ERR	St.Dev.
$u(t-1)$	$9.070386E-01$	$7.978188E-01$	$6.356812E-03$
$y(t-1)$	$4.243559E-01$	$1.678550E-01$	$1.590431E-02$
$e(t-1)$	$7.309806E-01$	$9.946108E-03$	$4.938950E-02$
$u^2(t-1)$	$4.340994E-02$	$3.868522E-03$	$5.056084E-03$
$e(t-2)$	$2.518882E-01$	$1.594725E-03$	$4.915041E-02$
$u(t-2)$	$-4.610006E-02$	$4.645953E-04$	$1.619390E-02$
$u(t-1)y(t-1)$	$1.431958E-02$	$2.055899E-04$	$6.000943E-03$
$e^2(t-1)$	$2.528252E-01$	$7.248319E-05$	$1.791946E-01$
$n \quad = 500$		$RSS \quad = 9.926377E+00$	
criterion $AIC=4.0$		$\varepsilon^T\varepsilon/y^Ty = 1.817415E-02$	

Table 1b: Global Model with 84 terms, Noise-free Case			
term	$\hat{\theta}$	ERR	St.Dev.
$u(t-1)$	$8.886380E-01$	$7.978188E-01$	$5.702131E-03$
$y(t-1)$	$3.880198E-01$	$1.678550E-01$	$1.221655E-02$
$e(t-1)$	$7.419301E-01$	$6.596605E-03$	$7.706479E-02$
$y^3(t-1)$	$2.296590E-01$	$4.004659E-03$	$2.565859E-02$
$u^2(t-1)$	$4.384641E-02$	$2.758204E-03$	$4.239363E-03$
$e(t-2)u(t-1)y(t-2)$	$2.946119E-01$	$1.082497E-03$	$3.612769E-02$
$e(t-2)y(t-2)y(t-1)$	$2.086115E-01$	$1.507739E-03$	$3.137829E-02$
$e(t-1)y^2(t-2)$	$-2.708210E-01$	$7.486877E-04$	$4.441615E-02$
$e(t-1)u(t-1)y(t-1)$	$2.493311E-01$	$7.152694E-04$	$3.620446E-02$
$u(t-2)y^2(t-1)$	$-2.328737E-01$	$7.695575E-04$	$2.609124E-02$
$y(t-2)y^2(t-1)$	$-1.001797E-01$	$1.504718E-03$	$1.338755E-02$
$e^3(t-1)$	$-1.579338E-00$	$3.922463E-04$	$4.301757E-01$
$u(t-1)y(t-1)$	$1.840856E-02$	$3.278574E-04$	$5.368260E-03$

$n \quad = 500$ $\qquad\qquad$ $RSS \quad = 7.601820E+00$

criterion $AIC=4.0$ $\qquad\qquad$ $\varepsilon^T\varepsilon/y^T y = 1.391813E-02$

Table 2: Time-Varying Model, Noise-free Case, $\lambda=0.97$				
point	terms	θ	$\hat{\theta}$	ERR
250 th	$u(t-1)$	$1.000000E+00$	$1.000000E+00$	$7.478421E-01$
	$y(t-1)$	$5.000000E-01$	$5.000000E-01$	$2.521579E-01$
	$u^2(t-1)$	none	none	*
500 th	$u(t-1)$	$8.000001E-01$	$8.000001E-01$	$9.273681E-01$
	$y(t-1)$	$2.000001E-01$	$2.000001E-01$	$5.111581E-02$
	$u^2(t-1)$	$1.000000E-01$	$9.999995E-02$	$2.151604E-02$

$t \quad = 500$ $\qquad$ $\lambda \quad = 0.97$ $\qquad$ $\displaystyle\sum_{t=1}^{500}\varepsilon^2(t) = 3.743278E-01$

Table 3: Time-Varying Model, Noise Case, λ=0.97				
point	terms	θ	$\hat{\theta}$	ERR
250 th	$u(t-1)$	$1.000000E+00$	$1.011648E+00$	$7.329127E-01$
	$y(t-1)$	$5.000000E-01$	$5.109544E-01$	$2.638053E-01$
	$u^2(t-1)$	none	none	*
500 th	$u(t-1)$	$8.000000E-01$	$7.927485E-01$	$9.236030E-01$
	$y(t-1)$	$2.00000E-01$	$2.015239E-01$	$5.194039E-02$
	$u^2(t-1)$	$1.000000E-01$	$9.814149E-02$	$2.101408E-02$
$t = 500$		$\lambda = 0.97$	$\sum_{t=1}^{500}\varepsilon^2(t) = 1.913597E+00$	

Table 4: Time-Varying Model, Noise Case, λ=0.90				
point	terms	θ	$\hat{\theta}$	ERR
250 th	$u(t-1)$	$1.000000E+00$	$1.014191E+00$	$7.318673E-01$
	$y(t-1)$	$5.000000E-01$	$5.204697E-01$	$2.648013E-01$
	$u^2(t-1)$	none	none	*
500 th	$u(t-1)$	$8.000000E-01$	$7.922933E-01$	$9.238081E-01$
	$y(t-1)$	$2.00000E-01$	$1.966080E-01$	$5.022280E-02$
	$u^2(t-1)$	$1.000000E-01$	$9.961569E-02$	$2.314501E-02$
$t = 500$		$\lambda = 0.90$	$\sum_{t=1}^{500}\varepsilon^2(t) = 2.010585E+00$	

point	terms	θ	$\hat{\theta}$	ERR
250 th	$u(t-1)$	$1.000000E+00$	$1.005268E+00$	$7.258623E-01$
	$y(t-1)$	$5.000000E-01$	$4.923638E-01$	$2.642085E-01$
	$u^2(t-1)$	none	none	*
500 th	$u(t-1)$	$8.000000E-01$	$8.172578E-01$	$8.905356E-01$
	$y(t-1)$	$2.00000E-01$	$7.319452E-01$	$7.505734E-02$
	$u^2(t-1)$	$1.000000E-01$	$8.752618E-02$	$1.600140E-02$
	$u(t-2)$	none	$-4.035374E-01$	$1.434520E-03$
	$y(t-2)$	none	$-1.159932E-01$	$1.240755E-03$
	$u^2(t-2)$	none	$-4.255118E-02$	$1.603582E-03$
	$e(t-1)$	none	$-1.107034E-01$	$1.379275E-04$
t = 500		λ = 0.99	$\sum_{t=1}^{500}\varepsilon^2(t)$ = 6.898567E+00	

Table 5: Time-Varying Model, Noise Case, λ=0.99

Appendix

On-line Model Validity Tests with Exponential Windowing

For off-line identification model validity tests based on correlation methods have been developed in (Billings and Voon 1983, Billings and Voon 1986). If the fitted model is correct and unbiased the residual sequence will be uncorrelated with all linear and nonlinear combinations of past inputs and outputs. That is,

$$\Psi_{\varepsilon'\varepsilon'}(\tau) = E\left[(\varepsilon(t)-\overline{\varepsilon})(\varepsilon(t+\tau)-\overline{\varepsilon})\right] = \delta(\tau)$$

$$\Psi_{u'\varepsilon'}(\tau) = E\left[(u(t)-\overline{u})(\varepsilon(t+\tau)-\overline{\varepsilon})\right] = 0 \quad \forall \quad \tau$$

$$\Psi_{\varepsilon'(\varepsilon'u)}(\tau) = E\left[(\varepsilon(t)-\overline{\varepsilon})(\varepsilon(t-1-\tau)-\overline{\varepsilon})(u(t-1-\tau)-\overline{u})\right] = 0 \quad \forall \quad \tau \geq 0$$

$$\Psi_{u^2{}'\varepsilon'}(\tau) = E\left[(u(t)^2-\overline{u^2})(\varepsilon(t+\tau)-\overline{\varepsilon})\right] = 0 \quad \forall \quad \tau$$

$$\Psi_{u^2{}'\varepsilon^2}(\tau) = E\left[(u(t)^2-\overline{u^2})(\varepsilon^2(t+\tau)-\overline{\varepsilon^2})\right] = 0 \quad \forall \quad \tau$$

where $^-$ denotes the time average and the $'$ in the subscripts indicates that the mean has been removed. The above tests often give the experimenter a great deal of information regarding the deficiencies in the fitted model and can indicate which terms should be included in the model to improve the fit. All the functions are usually

computed in normalized form

$$\Psi_{a'b'}(\tau) = \frac{\frac{1}{n}\sum_{t=1}^{n-\tau}(a(t)-\bar{a})(b(t+\tau)-\bar{b})}{\sqrt{\Psi_{a'a'}(0)\Psi_{b'b'}(0)}} \qquad -1\leq\Psi_{a'b'}\leq1$$

If the number of the data points is large the standard deviation of the correlation is $1/\sqrt{n}$ and the 95% confidence intervals are approximately $\pm1.96/\sqrt{n}$ (Billings and Voon 1983, Billings and Voon 1986).

A new set of model validity tests, called on-line validity tests to distinguish them from the off-line tests, have be derived which can be used in on-line computations with exponential windowing. This approach has also been used for performance monitoring in nonlinear adaptive noise cancellation (Billings and Alturki 1990).

Suppose $t+\tau+1$ samples are available up to the current time, where τ is the number of the lags, the newest sample is defined as the measurement at the t' th time instant and earlier measurements are at the i' th time instant, $i=-\tau-1, ..., t-1$. Taking the average value of the correlation function associated with finite lags at the time instant t

$$\rho_{ab}(\tau,t) = \frac{1}{\tau+1}\sum_{k=0}^{\tau}\Psi_{ab}(k,t)$$

This represents a compromise between completeness of the validity tests and complexity of the computation. As the number of the samples increases, the values at the i'th time instant, $i=t+1, ...,$ can be successively obtained. From these values, it is possible to observe if the fitted model is adequate or not in on-line identification. The normalized functions corresponding to the five validity tests above are given as (the subscript s denotes normalization)

$$\rho_{\varepsilon\varepsilon_s}(\tau,t) = \frac{\rho_{\varepsilon\varepsilon}(\tau,t)}{\Psi_{\varepsilon\varepsilon}(0,t)} = \frac{\eta\rho_{\varepsilon\varepsilon}(\tau,t-1)+(1-\eta)\bar{\varepsilon}_\tau(t)\varepsilon(t)}{\Psi_{\varepsilon\varepsilon}(0,t)}$$

$$\rho_{u\varepsilon_s}(\tau,t) = \frac{\rho_{u\varepsilon}(\tau,t)}{\sqrt{\Psi_{uu}(0,t)\Psi_{\varepsilon\varepsilon}(0,t)}} = \frac{\eta\rho_{u\varepsilon}(\tau,t-1)+(1-\eta)\bar{u}_\tau(t)\varepsilon(t)}{\sqrt{\Psi_{uu}(0,t)\Psi_{\varepsilon\varepsilon}(0,t)}}$$

$$\rho_{\varepsilon(\varepsilon u)_s}(\tau,t) = \frac{\rho_{\varepsilon(\varepsilon u)}(\tau,t)}{\Psi_{\varepsilon\varepsilon}(0,t)\sqrt{\Psi_{uu}(0,t)}} = \frac{\eta\rho_{\varepsilon(\varepsilon u)}(\tau,t-1)+(1-\eta)\varepsilon(t)\bar{D}_\tau(t)}{\Psi_{\varepsilon\varepsilon}(0,t)\sqrt{\Psi_{uu}(0,t)}}$$

$$\hat{\rho}_{u^2\varepsilon_s}(\tau,t) = \frac{\rho_{u^2\varepsilon}(\tau,t)}{\sqrt{\Psi_{u^2u^2}(0,t)\,\Psi_{\varepsilon\varepsilon}(0,t)}} = \frac{\eta\rho_{u^2\varepsilon}(\tau,t-1)+(1-\eta)\bar{C}_\tau(t)\varepsilon(t)}{\sqrt{\Psi_{u^2u^2}(0,t)\,\Psi_{\varepsilon\varepsilon}(0,t)}}$$

$$\hat{\rho}_{u^2\varepsilon^2_s}(\tau,t) = \frac{\rho_{u^2\varepsilon^2}(\tau,t)}{\sqrt{\Psi_{u^2u^2}(0,t)\,\Psi_{\varepsilon^2\varepsilon^2}(0,t)}} = \frac{\eta\rho_{u^2\varepsilon^2}(\tau,t-1)+(1-\eta)\bar{C}_\tau(t)[\varepsilon^2(t)-\overline{\varepsilon^2_{p_2}(t)}]}{\sqrt{\Psi_{u^2u^2}(0,t)\,\Psi_{\varepsilon^2\varepsilon^2}(0,t)}}$$

where

$$\eta = \frac{t}{t+1} \ , \quad \bar{\varepsilon}_\tau(t) = \frac{1}{\tau+1} \sum_{k=0}^{\tau} \varepsilon(t-k) \ , \quad \bar{u}_\tau(t) = \frac{1}{\tau+1} \sum_{k=0}^{\tau} u(t-k) \ ,$$

$$\bar{D}_\tau(t) = \frac{1}{\tau+1} \sum_{k=0}^{\tau} \varepsilon(t-k-1) \, u(t-k-1) \ , \quad \bar{C}_\tau(t) = \frac{1}{\tau+1} \sum_{k=0}^{\tau} [u^2(t-k)-\overline{u_{p_1}^2}(t)]$$

$$\Psi_{\varepsilon\varepsilon}(0,t) = \eta\Psi_{\varepsilon\varepsilon}(0,t-1) + (1-\eta)\varepsilon^2(t)$$

$$\Psi_{uu}(0,t) = \eta\Psi_{uu}(0,t-1) + (1-\eta)u^2(t)$$

$$\Psi_{u^2u^2}(0,t) = \eta\Psi_{u^2u^2}(0,t-1) + (1-\eta)[u^2(t)-\overline{u_{p_1}^2}(t)]^2$$

$$\Psi_{\varepsilon^2\varepsilon^2}(0,t) = \eta\Psi_{\varepsilon^2\varepsilon^2}(0,t-1) + (1-\eta)[\varepsilon^2(t)-\overline{\varepsilon_{p_2}^2}(t)]^2$$

$$\overline{u_{p_1}^2}(t) = \frac{1}{p_1} \sum_{i=t-p_1+1}^{t} u^2(i) \ , \quad \overline{\varepsilon_{p_2}^2}(t) = \frac{1}{p_2} \sum_{i=t-p_2+1}^{t} \varepsilon^2(i) \ ,$$

Here $p_1 \le t$, $p_2 \le t$, represent the number of the finite samples of the input $u(i)$ and the residual $\varepsilon(i)$ which are kept in the memory of the computer. When $t < 0$, $u(i)$ and $\varepsilon(i)$ are defined as zero.

The values of all these ρ functions are within the range from -1 to 1. According to the results from the off-line model validity tests (Billings and Voon 1983, Billings and Voon 1986), if the fitted model produced by the on-line estimator is correct, $\rho_{\varepsilon\varepsilon_s}(\tau,t)$ should be near zero and all the other functions should be zero. Considering the asymptotic memory length N_a $(=\frac{1}{1-\lambda})$ as the length of data to be tested, if N_a is large the standard deviation of the correlation is $1/\sqrt{N_a}$ and the 95% confidence intervals are defined as approximately $\pm 1.96/\sqrt{N_a}$. Replacing t with N_a in the equation $(\eta = \frac{t}{t+1})$ yields the definition

$$N_a = \frac{\eta}{1-\eta}$$

The assignment of the forgetting factor η should be co-ordinated with the forgetting factor λ in the on-line estimator, because the available length of data to be tested, N_a, has been defined by λ in (5-1). With the definitions, (5-1) and the above equation, the selection of η obeys the equation, $\eta = 1/(2-\lambda)$.

These test functions are related to the selection of τ (the number of lags). Considering the time for the on-line computation and the time-availability of the models, τ may be designed as typically $5 - 20$.

References

Ardalan, S.H. (1986): Floating-point roundoff error analysis of the RLS and LMS adaptive algorithms. IEEE Trans. Vol. CAS-33, No. 12, pp. 1192-1208, Dec. 1986.

Ardalan, S.H. and S.T. Alexander (1987): Fixed-point roundoff error analysis of the exponentially windowed RLS algorithm for time-varying systems. IEEE Trans. Acoust., Speech and Signal Processing, Vol. ASSP-35, No. 6, pp. 770-783, June 1987.

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

Billings, S.A. and W.S.F. Voon (1983): Structure detection and model validity tests in the identification of nonlinear systems, IEE Proc., 1983, Pt.D, Vol. 130, pp. 193-199.

Billings, S.A. (1985): an overview of nonlinear systems identification, 7th IFAC, Identification and System Parameter Estimation, York, UK.

Billings, S.A. and W.S.F. Voon (1986): Correlation based model validity tests for nonlinear models, Int.J.Systems Sci., 1986, Vol. 15, pp. 601-615.

Billings,S.A. and W.S.F. Voon (1987): Piecewise linear identification of nonlinear systems, Int. J. Control, Vol. 46, No. 1, pp. 215-235, 1987.

Billings, B.A., S. Chen (1989) : Extended model set, global data and threshed model identification of severely nonlinear systems, Int. J. Control. 1989, Vol. 50, No. 5, pp. 1897-1923.

Billings, S.A. and F.A. Alturki (1990): Performance monitoring in nonlinear adaptive noise cancellation. Research Report No.409, Department of Control engineering, University of Sheffield, UK.

Chen S. and S.A. Billings (1988): A recursive prediction error parameter estimator for nonlinear models. Research Report No.326, Dept. of Control Engineering, University of Sheffield, UK.

Chen, S., S.A. Billings, W. Luo (1989): Orthogonal least squares methods and their application to non-linear system identification, Int.J.Control, Vol. 50, pp. 1873-1896, 1989.

Chen S. and S.A. Billings (1989): Representations of nonlinear systems: the NARMAX model. Int. J. Control, Vol. 48, 1989.

Clarke D. and P.J. Gawthrop (1975): Self-tuning controller. Proc. Inst. Elec. Emg., Vol. 122, pp. 929-934, 1975.

Fnaiech, F. and L. Ljung (1987): Recursive identification of bilinear systems. Int. J. Control, Vol. 45, No. 2, pp. 453-470, 1987.

Gentleman, W.M. (1973): Least squares computation by Givens transformation without square roots. J. Inst. Math. and Appl., Vol. 12, pp. 329-336, 1973.

Gentleman, W.M. (1974): Regression problems and the QR decomposition. Bulletin Inst. Math. and Appl., Vol. 10, pp. 195-197, 1974.

Golub, G.H. and G.P.H. Styan (1973): Numerical computations for univariate linear models. J. Statist. Comput. Simul., Vol. 2, pp. 253-274, 1973.

Leontaritis, I.J. and S.A. Billings (1985): Input-output parametric models for nonlinear systems, Part I: Deterministic nonlinear systems; Part II: Stochastic nonlinear systems; Int. J. Control, Vol. 41, pp. 303-344, 1985.

Ling, F., D. Manolakis and J.G. Proakis (1986): A recursive modified Gram-Schmidt algorithm for least-squares estimation. IEEE Trans. Acoust., Speech and Signal Processing, Vol. ASSP-34, No. 4, pp. 829-836, Aug. 1986.

Liu, Y.P., M.J. Korenberg, S.A Billings. and M.B. Fadzil (1987): The nonlinear identification of a heat exchanger. 26th IEEE Conference on Decision and Control, Dec.9-11, 1987, Los Angeles, U.S.A.

Ljung, L. and T. Soderstrom (1983): Theory and Practice of Recursive Identification. MIT Press, Cambridge, Mass., 1983.

Fig. 1a Input (Noise Free Case)

Fig. 1b Output (Noise Free Case)

Fig. 2a predicted Output (Global Model m=28)

Fig. 2b residuals (Global Model m=28)

Fig. 3a predicted Output (Global Model m=84)

Fig. 3b residuals (Global Model m=84)
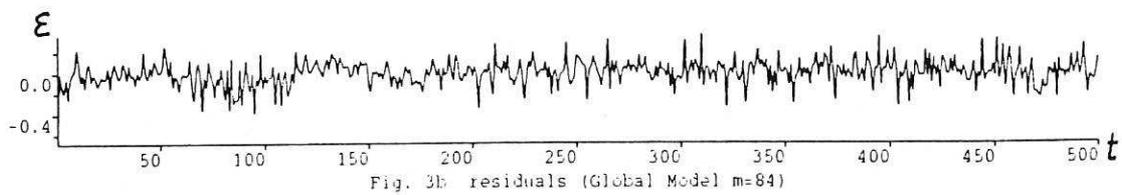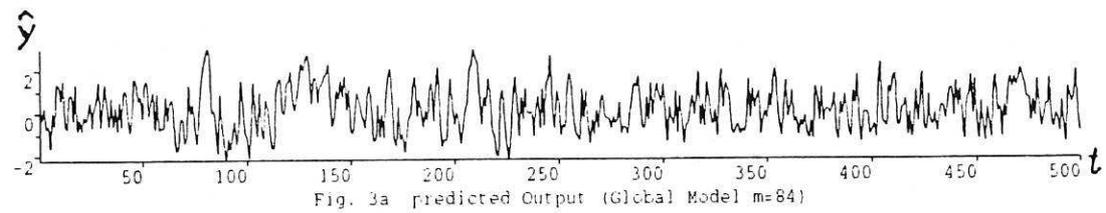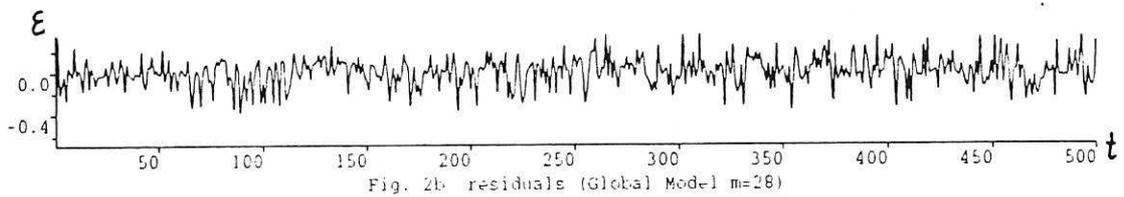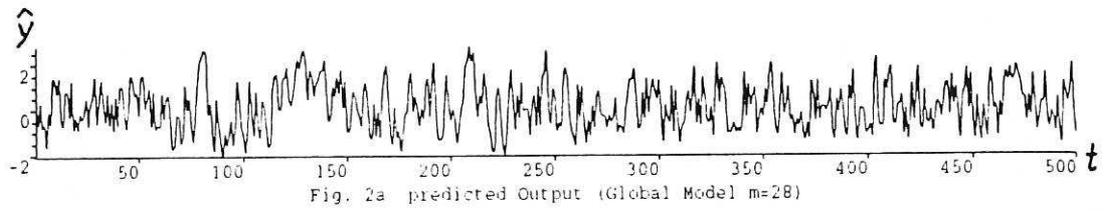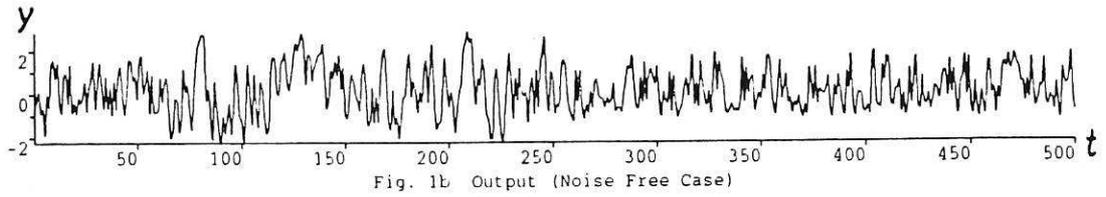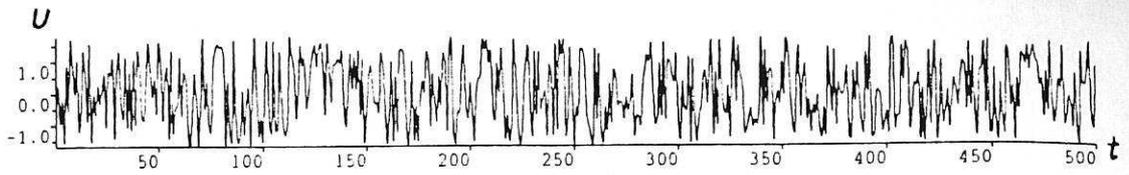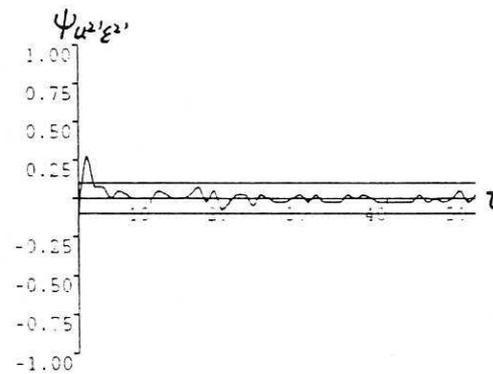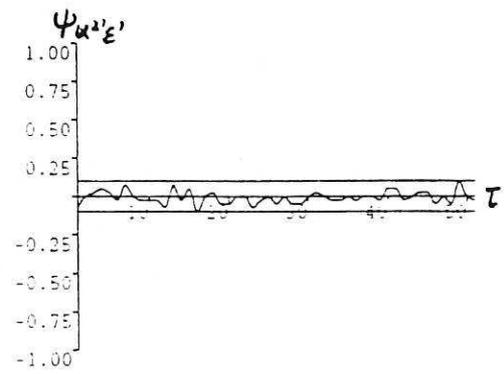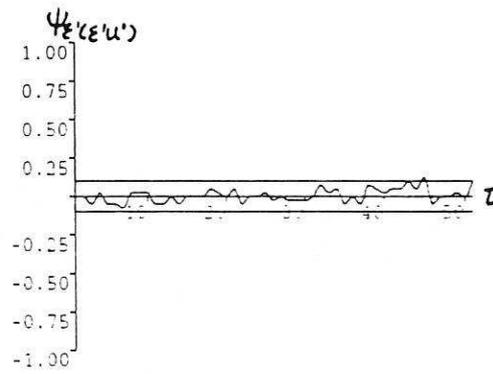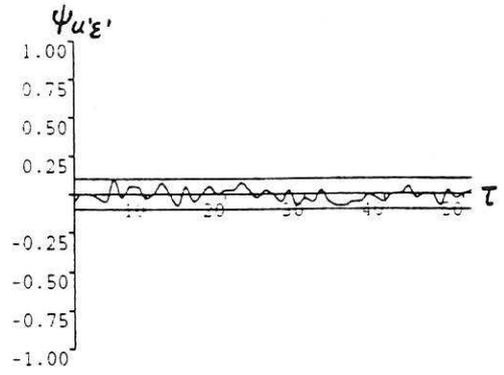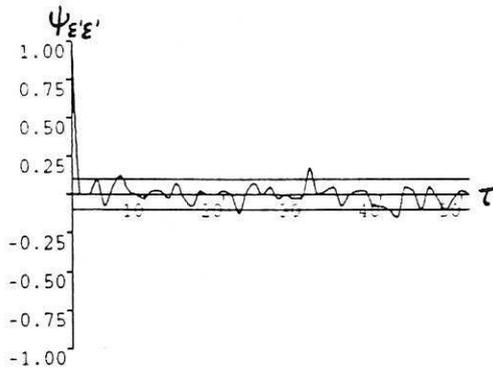
Fig. 4 Off-line validity tests (Global Model m=29)
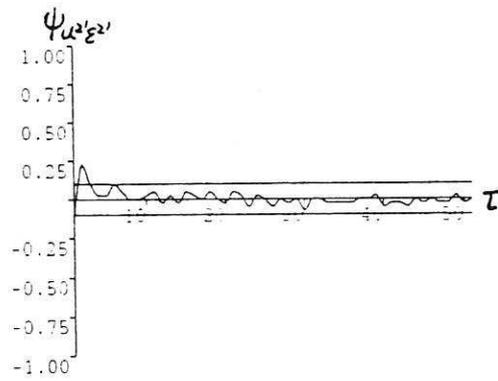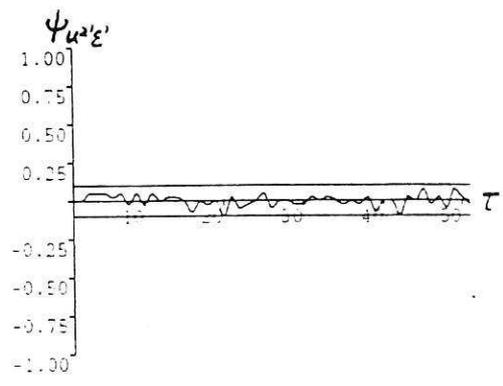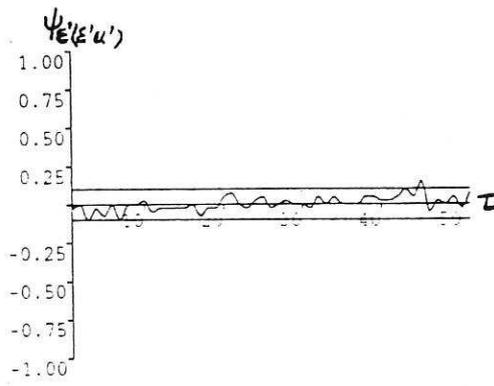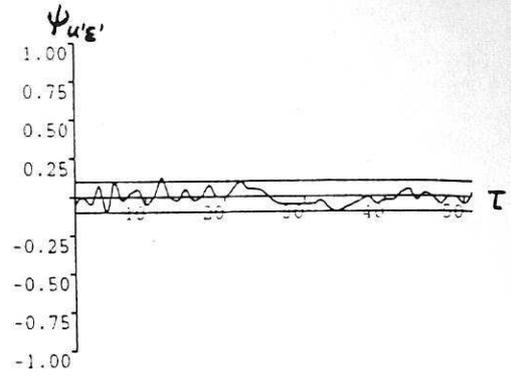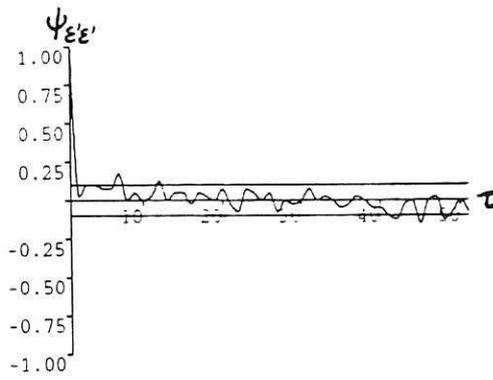
Fig. 5 Off-line validity tests (Global Model m=84)
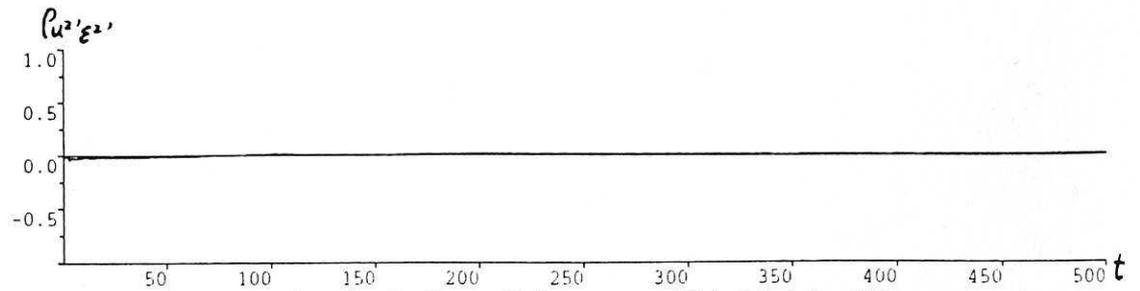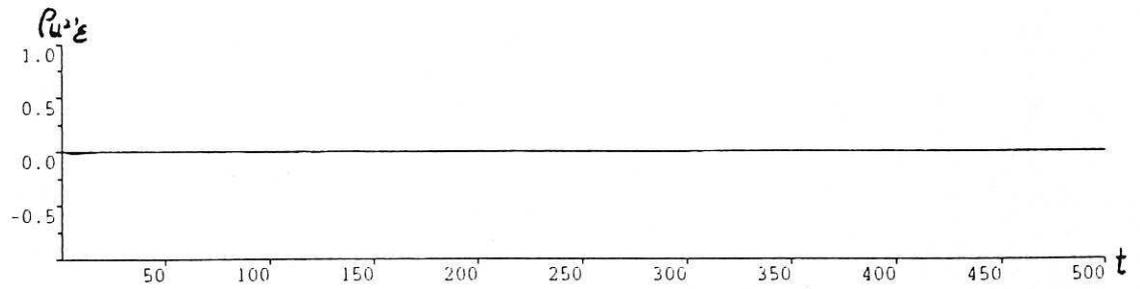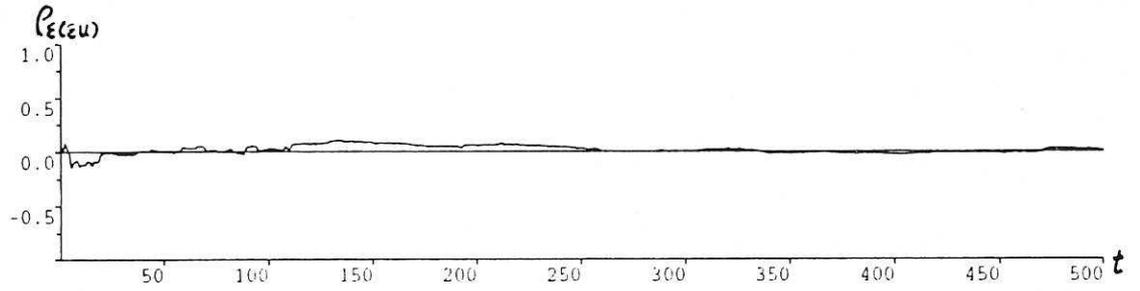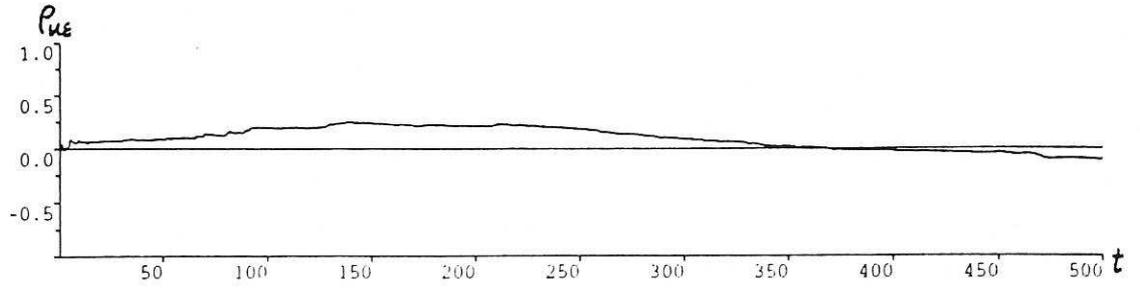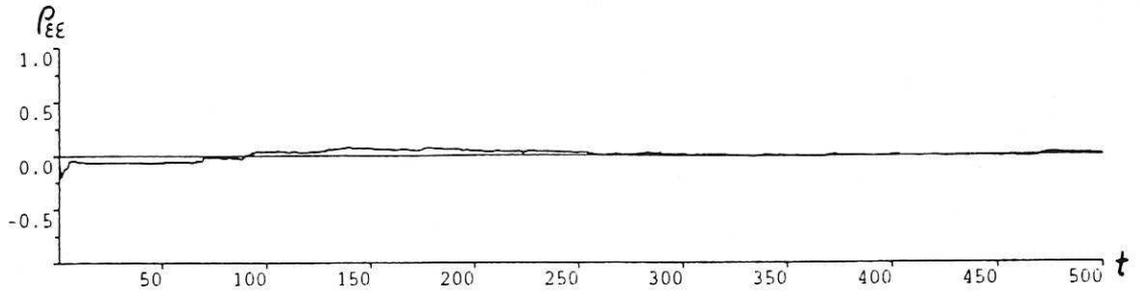
Fig. 6   On-line Validity Tests   (Global Model m=28)

Fig. 7   On-line Validity Tests (Global Model m=84)

Fig. 8a   Predicted Output   (Noise Free Case, 0.97)

Fig. 8b   Residuals   (Noise Free Case, 0.97)

Fig. 9a   Parameter of u(t-1)
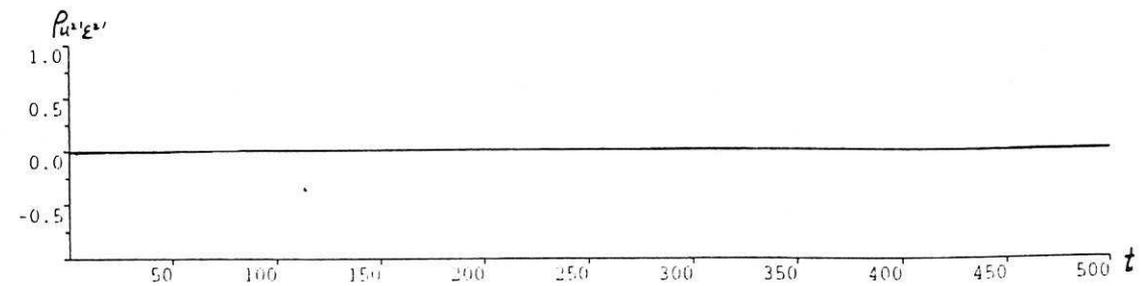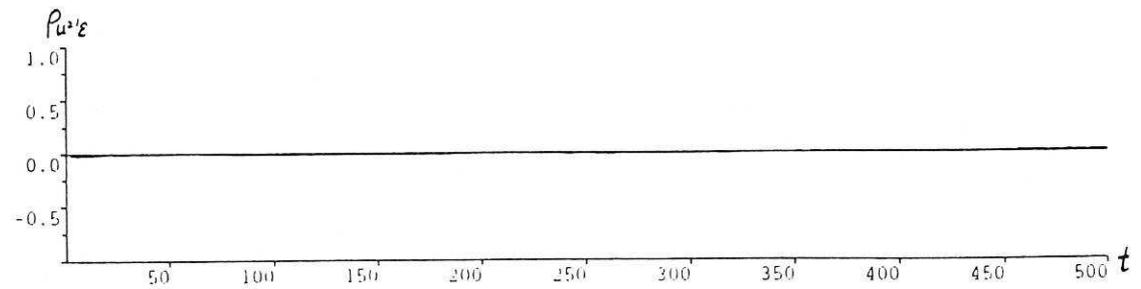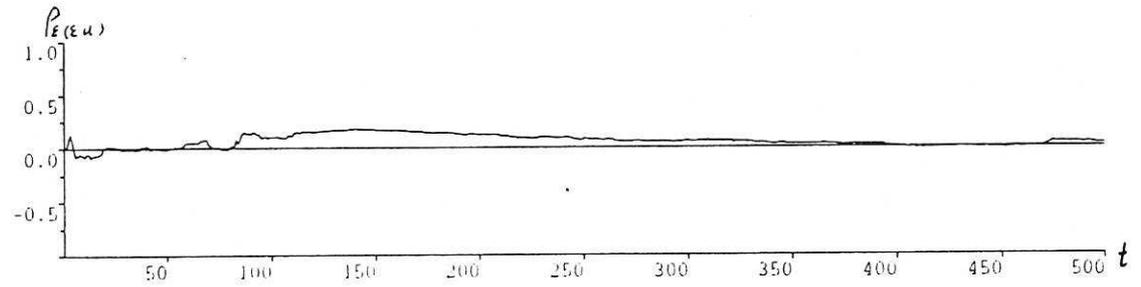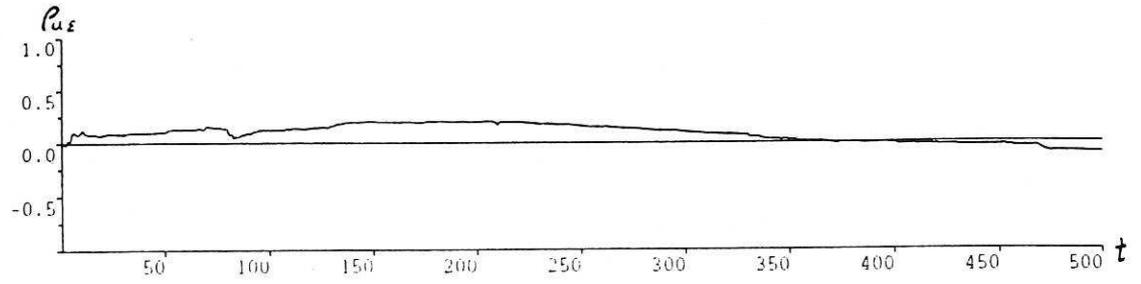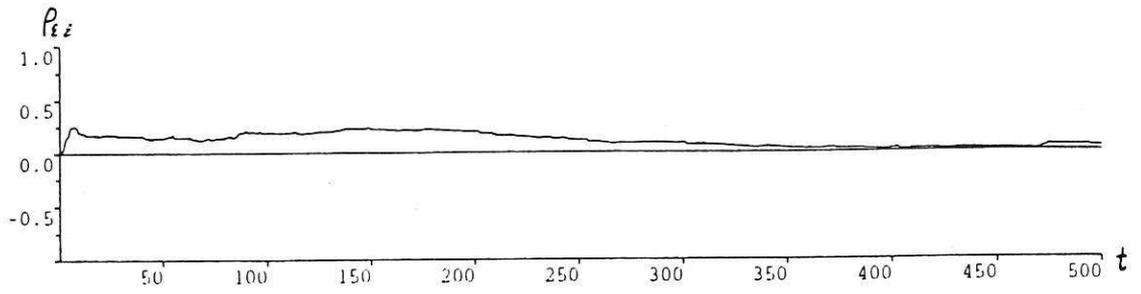
Fig. 9b   Parameter of y(t-1)

Fig. 9c   Parameter of u(t-1)u(t-1)
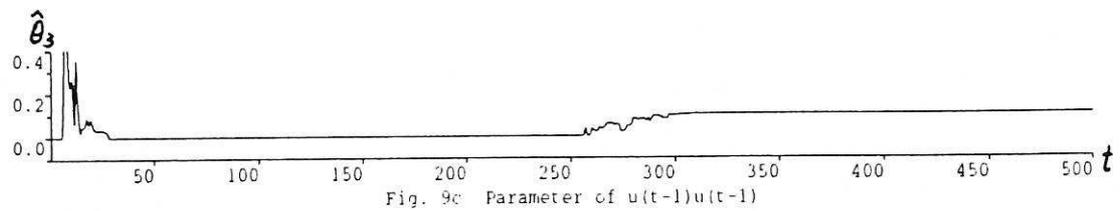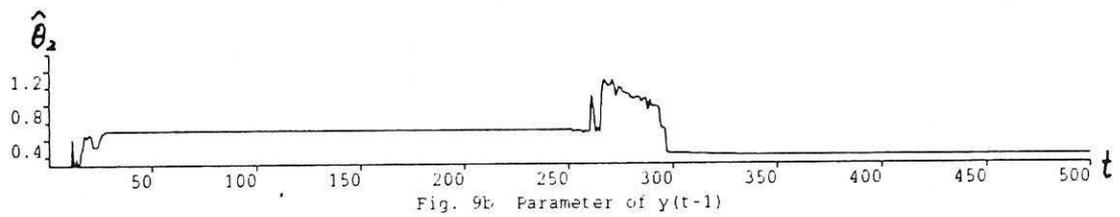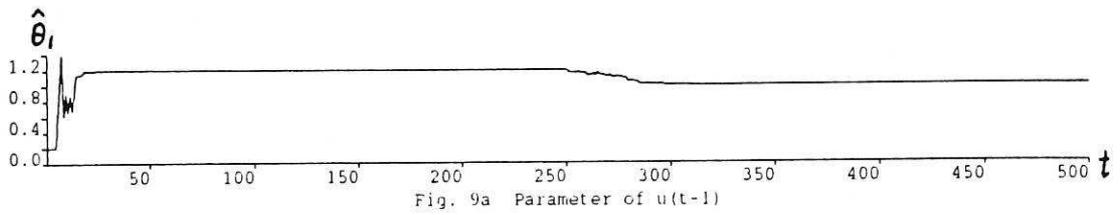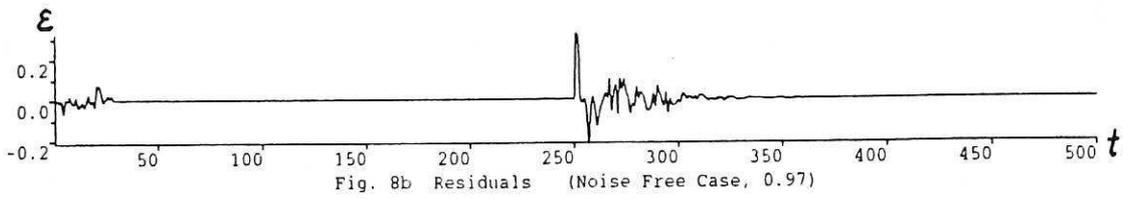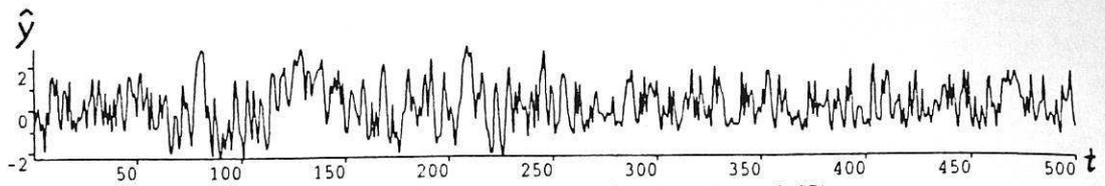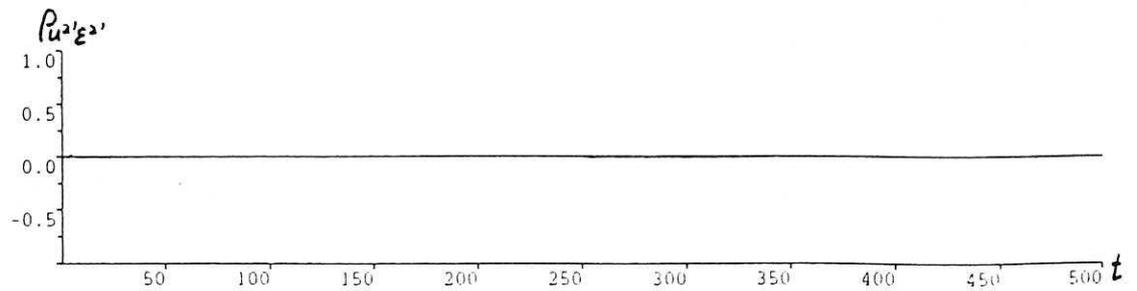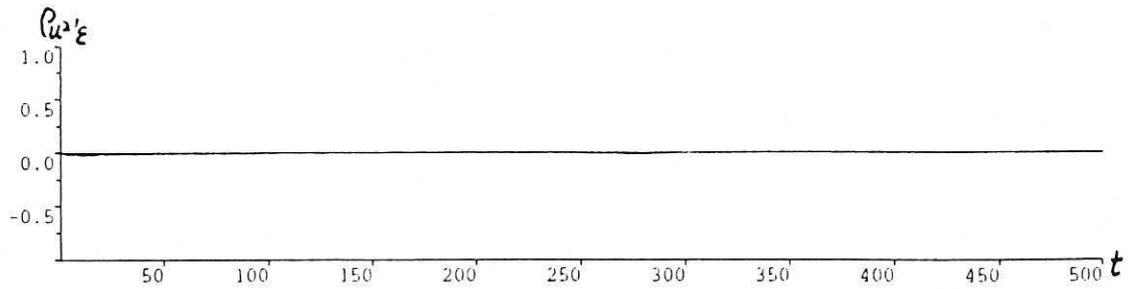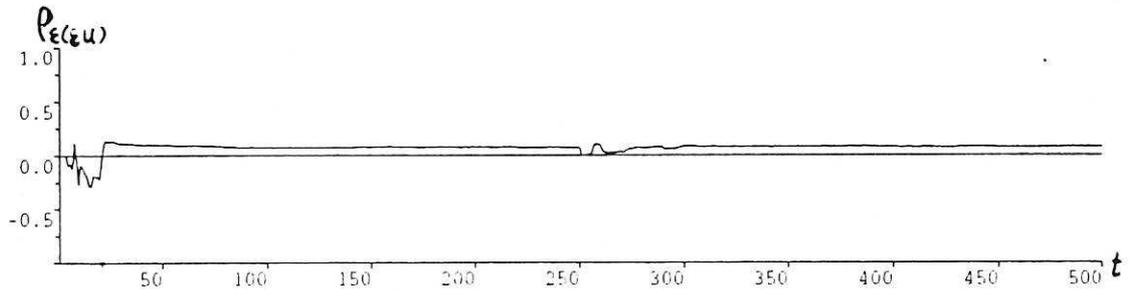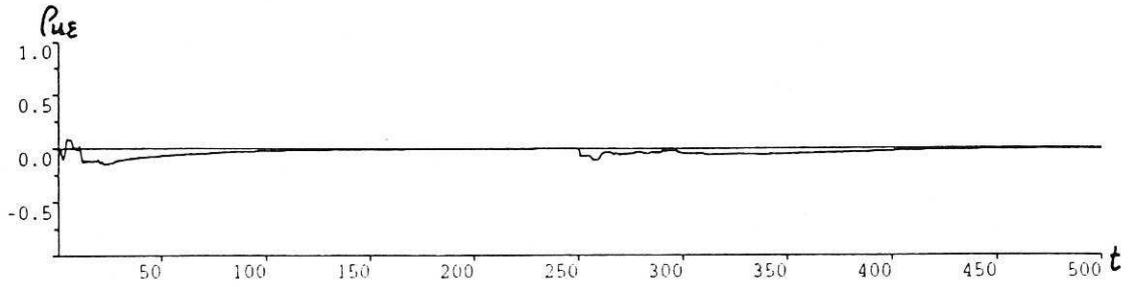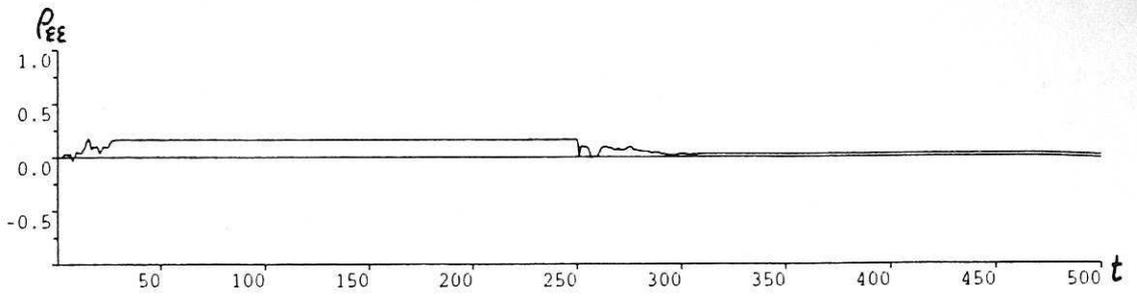
Fig. 10 On-line Validity Tests (Noise Free Case, 0.97)

Fig. 11a   Input   (Noise Case)

Fig. 11b   Output   (Noise Case)

Fig. 12a   Predicted Output   (Noise Case, 0.97)

Fig. 12b   Residuals   (Noise Case, 0.97)

Fig. 13a   Parameter of u(t-1)

Fig. 13b   Parameter of y(t-1)

Fig. 13c   Parameter of u(t-1)u(t-1)

$\rho_{\epsilon\epsilon}$

$\rho_{u\epsilon}$

$\rho_{\epsilon(\epsilon u)}$

$\rho_{u^2{}'\epsilon}$

$\rho_{u^2{}'\epsilon^2{}'}$
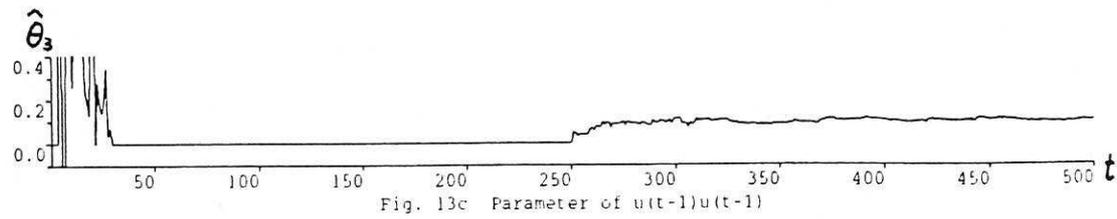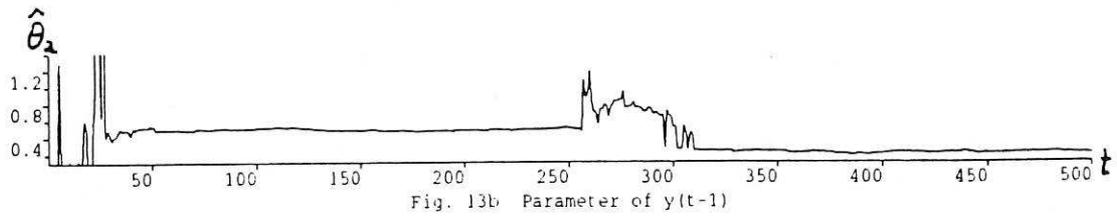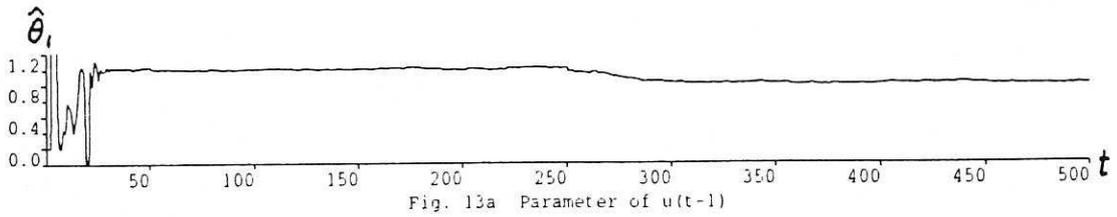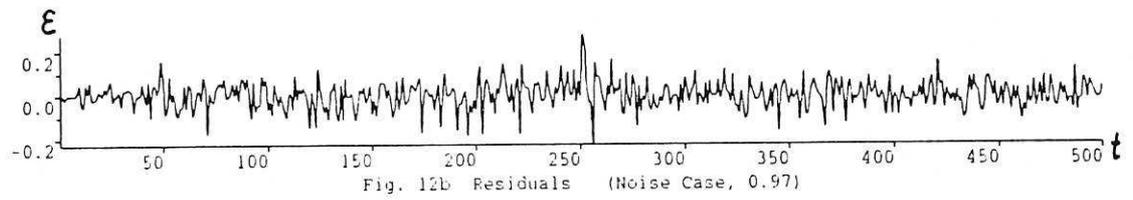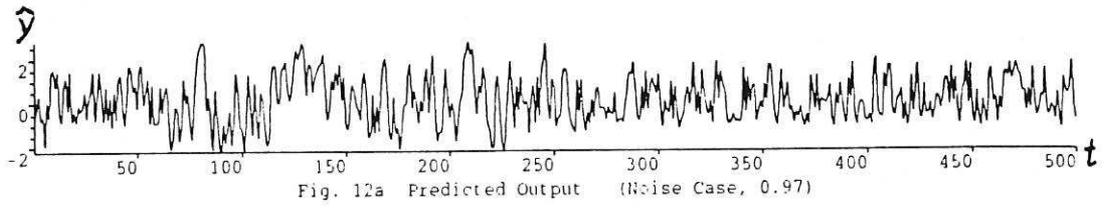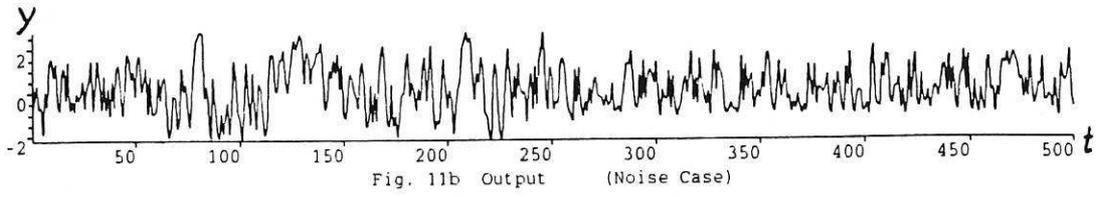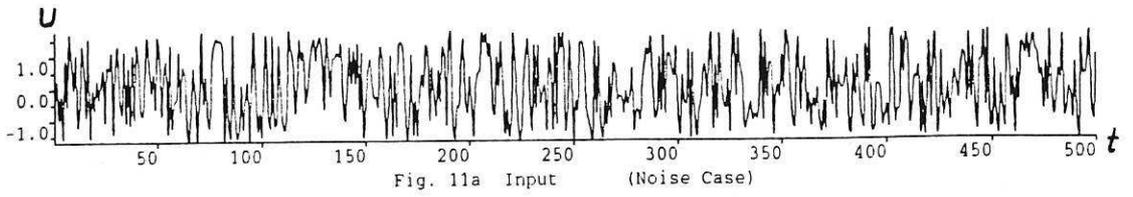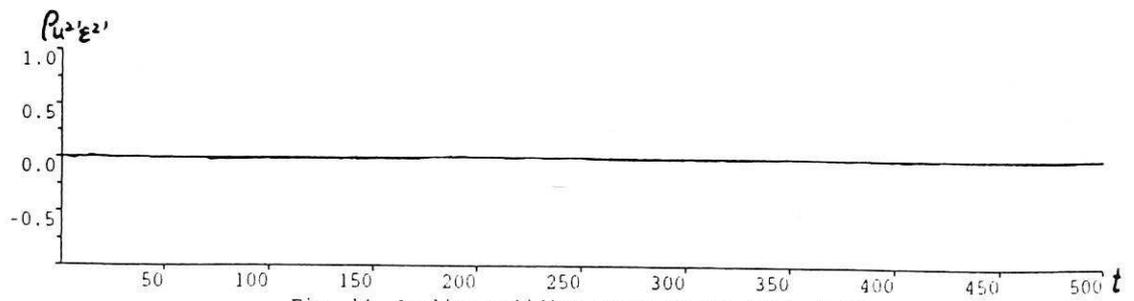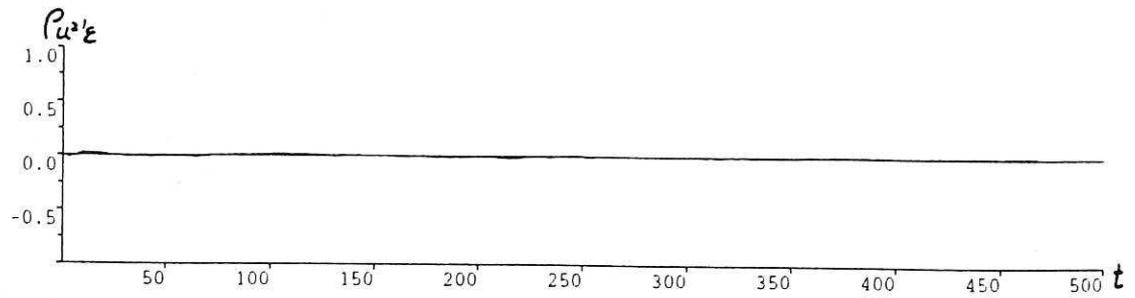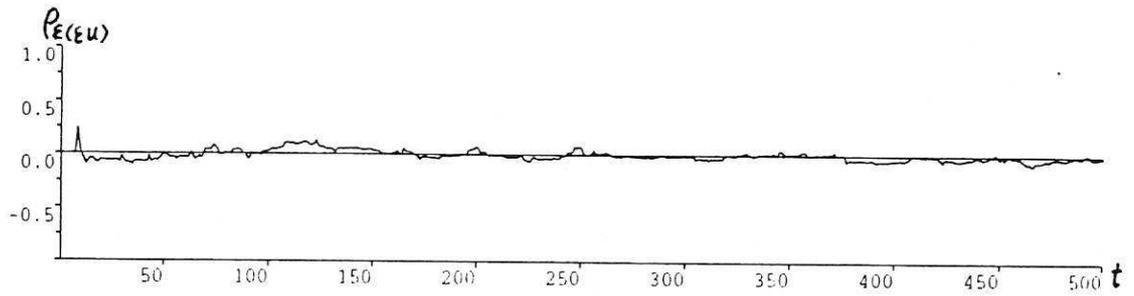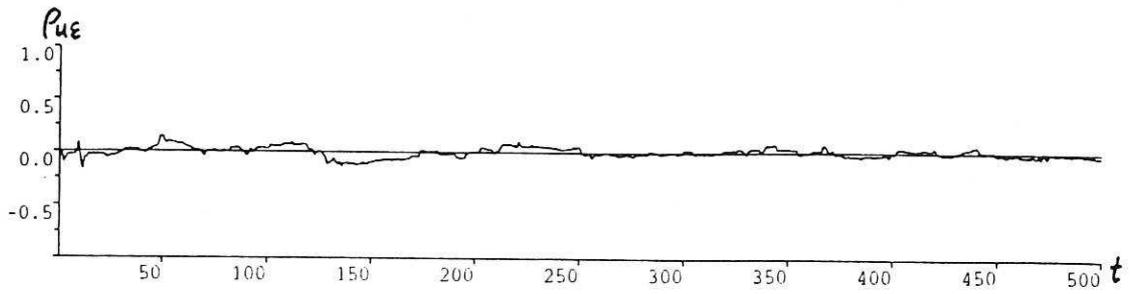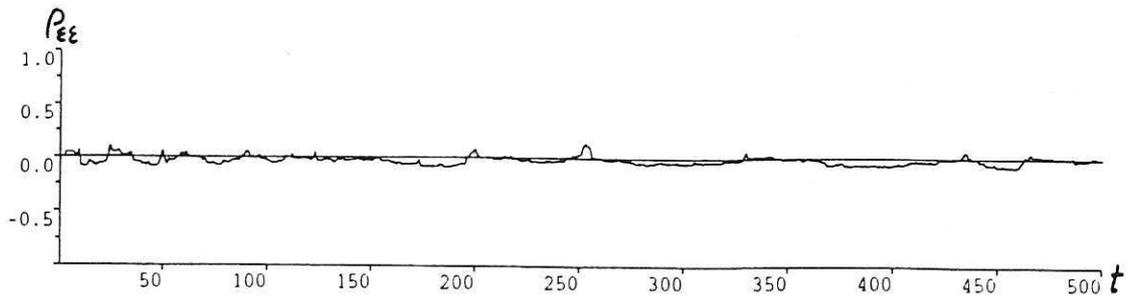
Fig. 14 On-line Validity Tests (Noise Case, 0.97)

Fig. 15a   Predicted Output (Noise Case, 0.90)

Fig. 15b   Residuals   (Noise Case, 0.90)

Fig. 16a   Parameter of u(t-1)   (Noise Case, 0.90)

Fig. 16b   Parameter of y(t-1)   (Noise Case, 0.90)

Fig. 16c   Parameter of u(t-1)u(t-1)   (Noise Case, 0.90)

Fig. 17a   predicted Output (Noise Case, 0.99)

Fig. 17b   residuals   (Noise Case, 0.99)
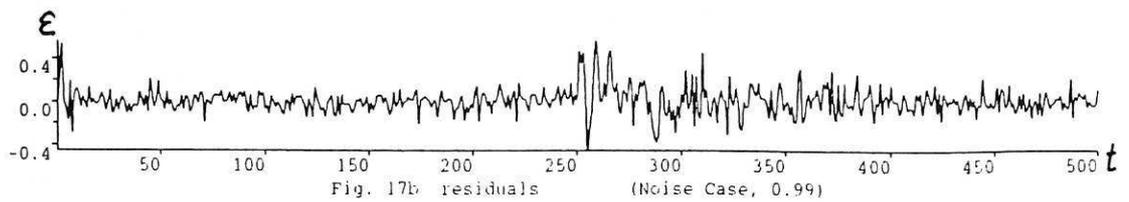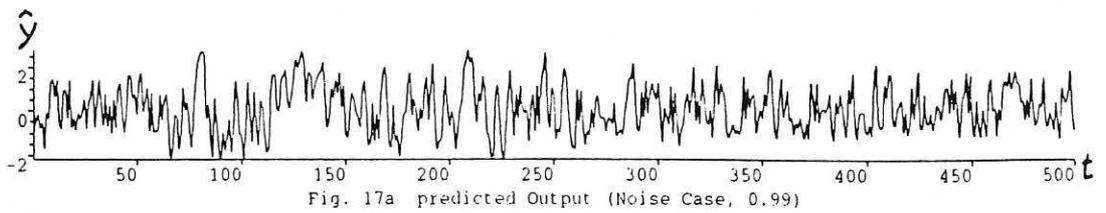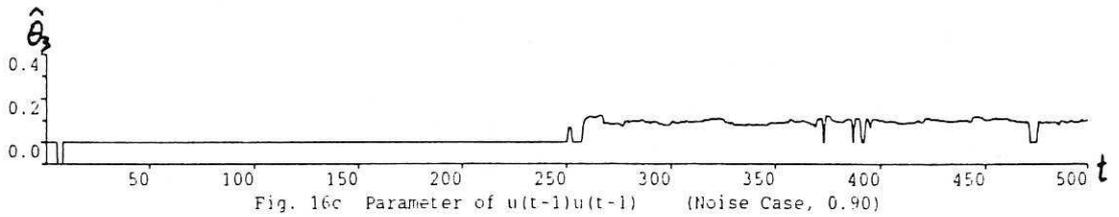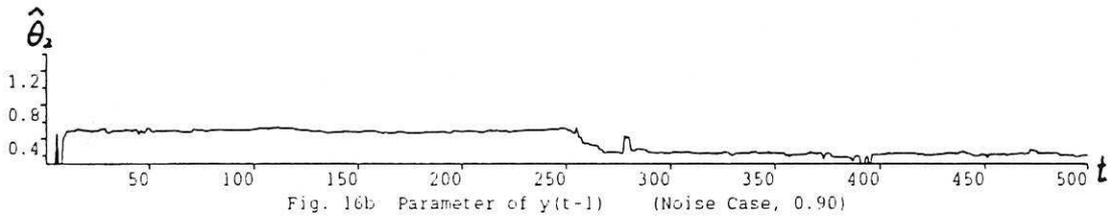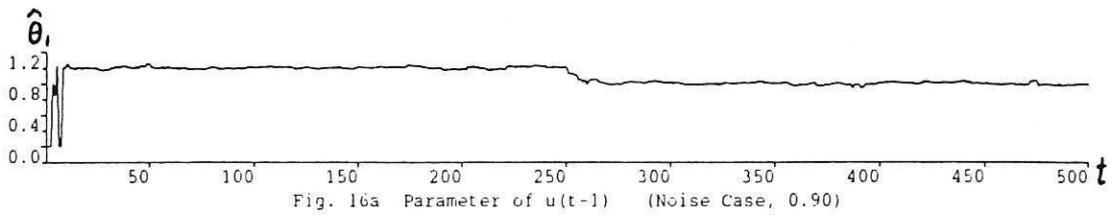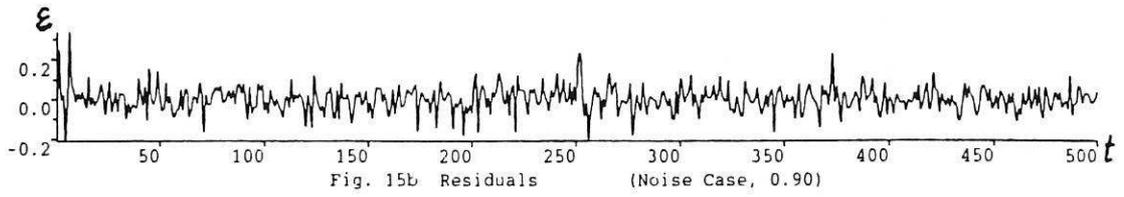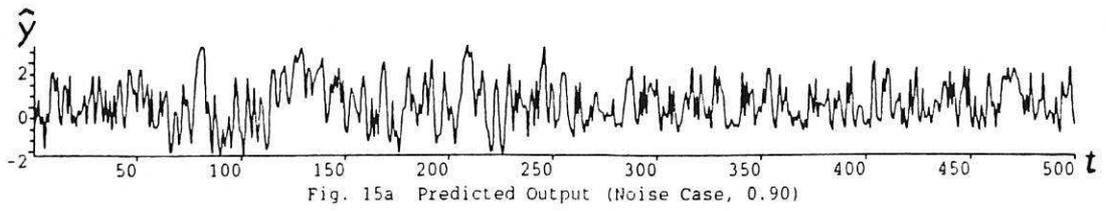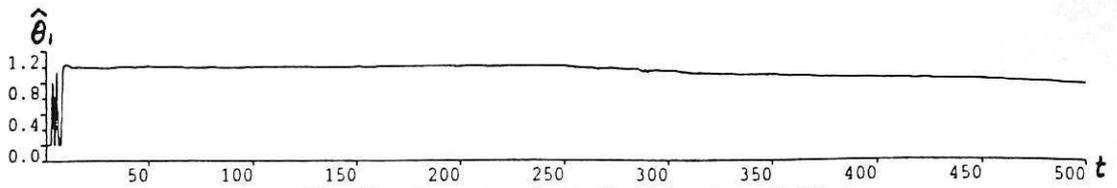
Fig. 18a Parameter of u(t-1) (Noise Case, 0.99)
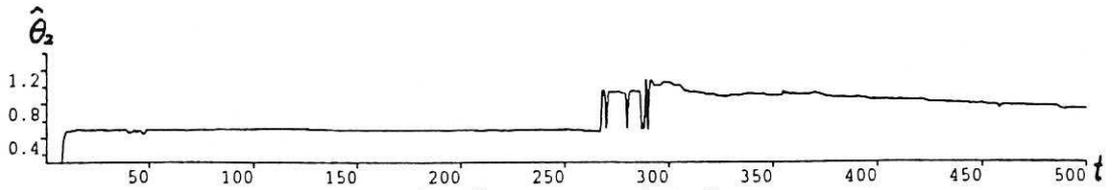
Fig. 18b Parameter of y(t-1)
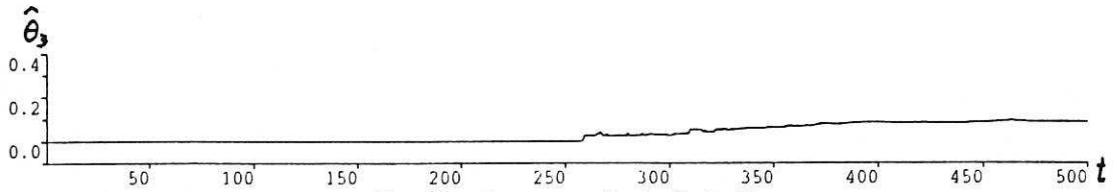
Fig. 18c Parameter of u(t-1)u(t-1)
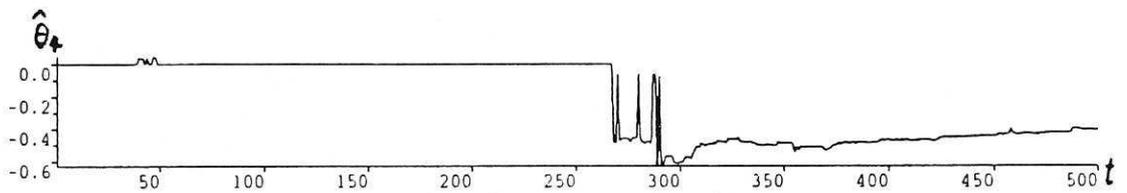
Fig. 18d Parameter of u(t-2)
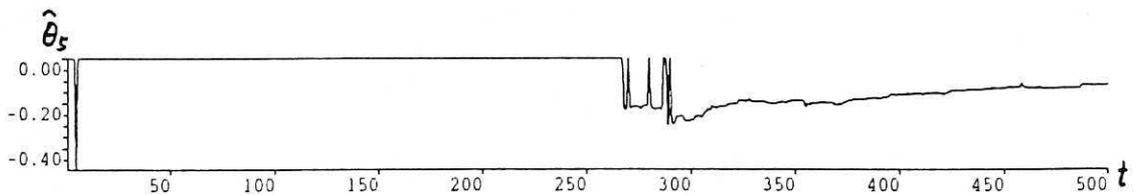
Fig. 18e Parameter of y(t-2)
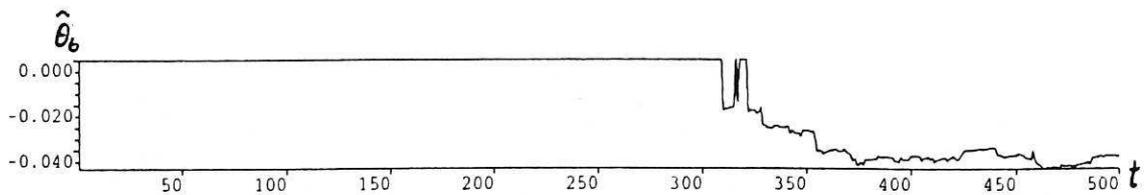
Fig. 18f u(t-2)u(t-2)

Fig. 18g Parameter of e(t-1)