Monograph:
Billings, S.A. and Mao, K.Z. (1998) Model Identification and Assessment Based on Model Predicted Output. Research Report. ACSE Research Report 714 . Department of Automatic Control and Systems Engineering

---

Model Identification and Assessment Based on Model Predicted Output

S.A.Billings and K.Z.Mao
Department of Automatic Control and Systems Engineering
University of Sheffield
Mappin Street, Sheffield S1 3JD
United Kingdom

Research Report No. 714

1 May 1998

University of Sheffield

Model Identification and Assessment Based on Model Predicted Output

S.A.Billings and K.Z.Mao

Department of Automatic Control and Systems Engineering

University of Sheffield

Sheffield S1 3JD, UK

Abstract

Conventional system identification algorithms are based on the minimisation of the one step ahead prediction errors. In this study it is shown that one step ahead predictions do not always provide a good assessment of model quality. The model predicted output which can be considered as the long range prediction is suggested as an alternative criterion for model assessment. Based on this criterion a new system identification algorithm is developed.

1 Introduction

Conventional system identification algorithms are based on the minimisation of the one step ahead prediction errors. Typically the data set is split into an estimation set which is used to identify the model and a test set which is used to assess the predictive capacity of the model obtained. If the one step ahead predictions over both the estimation data and the test data are good the model is considered as acceptable. This procedure is regarded as universally correct and has been used in system modelling and identification for various model forms. These include the linear ARMAX model (Autoregressive Moving Average model with eXogenous input) and the NARMAX model (Nonlinear ARMAX) and in neural network training for the multilayer perceptron (MLP) and radial basis function (RBF) architectures.

The efficiency of one step ahead predictions for model assessment is examined in the present study. A theoretical analysis shows that one step ahead predictions over a test data set may not always provide a good assessment of model quality if the input and output are stationary signals. Assessing identified models based on one step ahead predictions can therefore produce incorrect results. This is demonstrated using an example where the model which provides almost perfect one step ahead predictions over the estimation and test data sets has incorrect step and frequency responses.

An alternative criterion for model assessment is the model predicted output. Although not often addressed, model predicted output has long been used for model assessment (for example Ljung 1987, Billings *et al* 1989, 1992). Model predicted outputs can be considered as multistep predictions with prediction horizons from 1 to $N - d$, where N is the data length, d is the maximum value of the input and output lags, and these are therefore

more sensitive to unmodelled dynamics than one step ahead predictions. However, until recently model predicted output has not been used as an optimisation index in system identification (Berger 1995, 1997). This is possibly because the model predicted output is usually a high order nonlinear function of the model parameters. In Berger (1995, 1997) the model structure was assumed to be known *a priori* and the minimisation of model predicted outputs was achieved through optimising parameter estimation. But the model predicted output mainly depends on the model structure. If a model structure is insufficient to describe the system, optimising the parameter estimates will achieve a very limited improvement in the model predicted output. Therefore, when the model structure is unknown *a priori*, the optimisation of model predicted outputs should be done by selecting a correct model structure while the parameters can be estimated using a least squares algorithm if the model can be expressed in a linear-in-the-parameters form.

Motivated by the above observations a new system identification algorithm based on the minimisation of model predicted output errors is developed in the present study. Actually this algorithm minimises three performance indices: one step ahead prediction errors, model predicted output errors and model size. This multi-objective optimisation problem can be solved by optimising a combined cost function which is a weighted sum of these three performance indices. But there exist at least two difficulties associated with the minimisation of this combined cost function. First, weighting elements are not easy to assign in practice. Although it is often claimed that weighting elements can be decided in terms of the designers preferences it is very difficult to establish a quantitative relationship between values of the weighting elements and objectives such as approximation accuracy. Second, the cost function is an $(N-d)^{th}$ order nonlinear function of the model parameters. Even if the model structure is known *a priori*, estimating the parameters by directly minimising the high order nonlinear function is impractical if the prediction horizon is extended to cover the data length. In the present study, the three performance indices are minimised separately but simultaneously using a multi-objective genetic algorithm (GA). Because model size and model predicted output are both dependent on the model structure employed, minimisation of these three performance indices can be done by selecting the optimal model structure from a set of candidate models using genetic algorithms, while the parameters can be estimated using a least squares routine. The advantages of the new algorithm are that it is not necessary to solve a high order nonlinear optimisation problem for parameter estimation, and objectives such as approximation accuracy are easier to realise.

The present study is organised as follows. In §2, the one step ahead prediction for model identification and assessment are briefly reviewed, and the efficiency of one step ahead prediction is analysed and illustrated. In §3, a new system identification algorithm optimising model predicted outputs is developed. Numerical examples are presented in §4.

2 One step ahead predictions for model assessment

2.1 A brief review of the one step ahead prediction error for model assessment

An important aspect of system modelling and identification is the problem of model structure determination, that is determining the model form, size and complexity. An undersized model will provide an inadequate representation but an oversized model will have a tendency to overfit the given data and will often perform badly on unseen data. Determining an appropriate model structure is therefore important and has received considerable attention in the literature. There are two main kinds of model selection methods; the backward elimination method and the forward inclusion method. The backward elimination method starts with a large model and unimportant model elements are dynamically removed. The model elements could for example be terms in a NARMAX model or neurons in a neural network model. In the forward inclusion method important model elements are dynamically added according to a selection criterion. The forward regression orthogonal algorithm (Billings *et al* 1988a) belongs to the inclusion method and has been widely used in dynamic nonlinear system identification of various model forms. These include the NARMAX model (Billings *et al* 1988b, Zhu and Billings 1993, Billings and Mao 1997, and many others), radial basis function (RBF) neural networks (Chen *et al* 1990b, 1991, Brouwn *et al* 1994, Arciniegus *et al* 1994) and fuzzy systems (Wang and Mendel 1992, Jang and Sun 1993, Wang and Langari 1995, and others).

Whatever method is employed assessing model quality is an important step in any system identification procedure. Because of the ease of implementation the one step ahead prediction is one of the most commonly used assessment indices. Typically the data is divided into an estimation set which is used for model identification and a test set which is used to test the model predictive capacity. If the model selected provides good predictions over the test data the model is considered as acceptable (see for example Nahas *et al* 1992, Ligntbody *et al* 1997). But recent research suggests that this procedure may not always be an acceptable test approach. This was illustrated by, for example Zhu and Rohwer (1996) who demonstrated that using test data can cause incorrect estimation of the mean and variance of a group of data. It is therefore instructive to examine the efficiency of one step ahead predictions for model assessment.

2.2 Numerical illustration of one step ahead predictions in model assessment

The efficiency of one step ahead predictions for model assessment will be investigated in this section where it is shown that one step ahead predictions over a test data set do not always provide more insight regarding model inadequencies than predictions over just the estimation data set. This is best illustrated using a simple example.

Example

Consider the model of a heat exchanger (Smith and Corripio 1997)

$$G(s) = \frac{0.8}{(30s+1)(10s+1)(3s+1)} \tag{1}$$

Using a PRBS sequence with amplitude ± 3 as the input and a sampling period of 2 seconds 800 data samples were generated. The first 400 data samples were used for model estimation and the last 400 samples were used for model testing. The maximum lags of the input and output were all set to 3, and the approximation accuracy was set to 99.9%. The discrete time linear ARX model was identified

$$\hat{G}(z^{-1}) = \frac{0.0002749 + 0.002305z^{-1}}{1 - 1.9335z^{-1} + 0.9435z^{-2}} \tag{2}$$

The one step ahead predicted output over the estimation and test sets are shown in Figure 1 (a) and (b) and clearly indicate that the model eqn (2) is adequate. Indeed the model appears to fit almost perfectly. However the different unit step responses of the original system eqn (1) and the identified model eqn (2) shown in Figure 2 (a) and (b) reveal that the identified model is incorrect. This is conformed by the different frequency responses illustrated in Figure 3 (a) and (b).

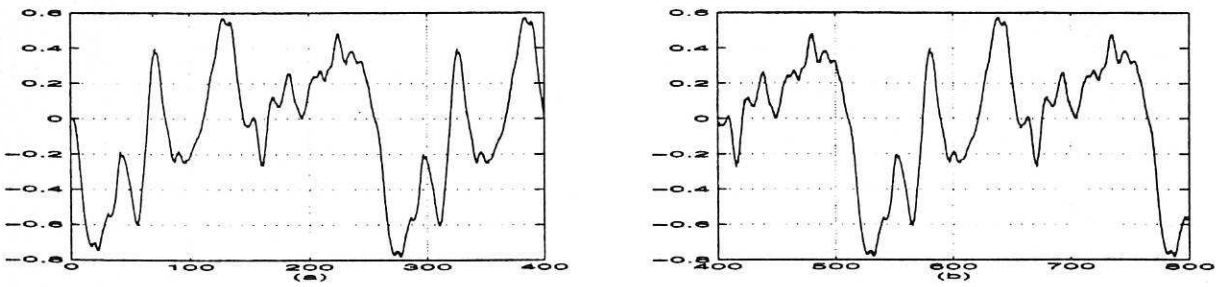

Figure 1: One step predictions for model eqn (2) (a) the estimation data, (b) the test data (solid-measurement, dashed-prediction)

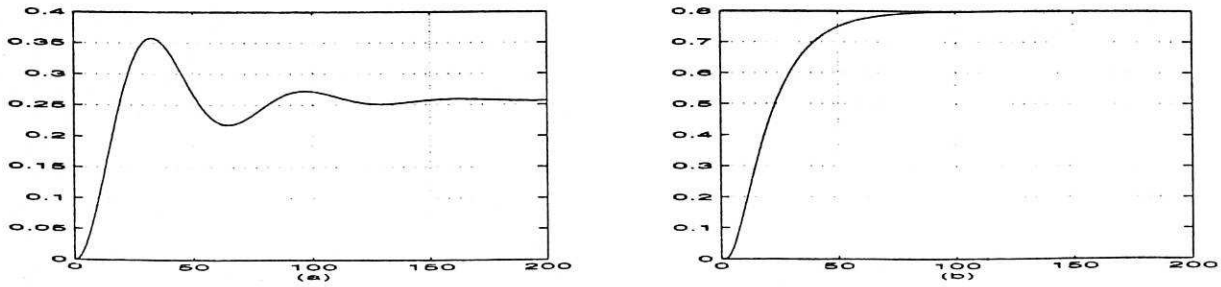

Figure 2: Unit step responses for models eqn (1) and eqn (2) (a) the identified model eqn (2), (b) the model eqn (1)

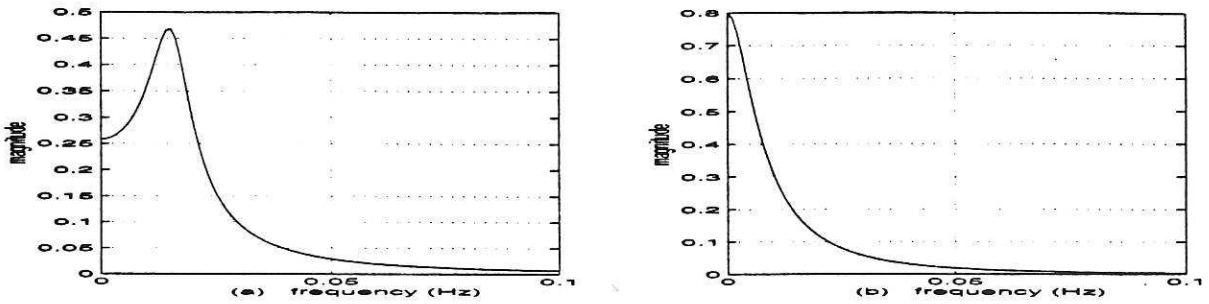

Figure 3: Frequency responses for models eqn (1) and eqn (2) (a) the identified model eqn (2), (b) the model eqn (1)

The input for the identification was a PRBS sequence and the system was therefore persistently excited. But why was an insufficient model selected and why did the model produce almost perfect one step ahead predictions? This is because the identification is based on the contribution of candidate terms to the one step ahead output prediction which is proportional to the correlation between the output and the candidate terms. When the output signal changes slowly as in Figure 1, the correlations between the output $y(k)$ and the candidate terms $y(k-1)$, $y(k-2)$ *etc* are large. This can occur for example when the signals are over sampled. Thus, a model with only a few terms such as $y(k-1)$, $y(k-2)$ will meet the approximation accuracy requirement. In this example, the omitted model terms make little energy contribution to the output, in fact only 0.06%. The correct model in this case is

$$G(z^{-1}) = \frac{0.000248 + 0.00226z^{-1} + 0.00179z^{-2}}{1 - 2.309z^{-1} + 1.667z^{-2} - 0.429z^{-3}} \tag{3}$$

A comparison between models (2) and (3) shows that the structure of the identified model eqn (2) was insufficient, and severe bias was induced due to the omitted model terms. As a consequence the dynamic characteristics and and the static gain of the identified model (2) are all quite different from the true model. This means that although the omitted terms are trivial in terms of the energy of the output, these are important to the system characteristics. Determining model structure according to one step ahead predictions may therefore produce an incorrect result especially if the sample rate is inappropriate. The one step ahead prediction over the test data set is expected to detect the inadequacies of model (2), however the test failed. The failure is analysed in the next subsection.

2.3 A theoretical analysis of the failure of one step ahead predictions for model assessment

Consider the nonlinear dynamic model

$$y(k) = \sum_{i=1}^{n} \varphi_i(k)\theta_i = \phi_1(k)\Theta_1 + \phi_2(k)\Theta_2 \tag{4}$$

where the regressors $\varphi_i(k)$ are some nonlinear function of the input and output, and θ_i represent the parameters. $\phi_1(k) = [\varphi_1(k), \ldots, \varphi_m(k)]$, $\phi_2(k) = [\varphi_{m+1}(k), \ldots, \varphi_n(k)]$, $\Theta_1 = [\theta_1, \ldots, \theta_m]^T$, $\Theta_2 = [\theta_{m+1}, \ldots, \theta_n]^T$. Notice that eqn (4) can be used to represent a wide class of model types including radial basis function (RBF) neural network architectures, fuzzy logic models *etc.*

Consider an insufficient model where the last a few terms on the rhs of eqn (4) have been deliberately omitted

$$y(k) = \phi_1(k)\Theta_1 \tag{5}$$

and compute the least squares parameter estimate

$$\begin{aligned}
\hat{\Theta}_1 &= (\Phi_{E1}^T\Phi_{E1})^{-1}\Phi_{E1}^T Y_E \\
&= (\Phi_{E1}^T\Phi_{E1})^{-1}\Phi_{E1}^T(\Phi_{E1}\Theta_1 + \Phi_{E2}\Theta_2) \\
&= (\Phi_{E1}^T\Phi_{E1})^{-1}\Phi_{E1}^T\Phi_{E1}\Theta_1 + (\Phi_{E1}^T\Phi_{E1})^{-1}\Phi_{E1}^T\Phi_{E2}\Theta_2 \\
&= \Theta_1 + (\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{E1}^T\Phi_{E2}}{N})\Theta_2
\end{aligned} \tag{6}$$

where $\Phi_{Ei} = [\phi_i^T(1), \ldots, \phi_i^T(N)]^T$ $(i = 1, 2)$, $Y_E = [y(1), \ldots, y(N)]^T$ and the subscript E denotes the estimation data set.

Consider the one step ahead prediction errors over the estimation data

$$\begin{aligned}
\epsilon(k) &= y(k) - \hat{y}(k) = \phi_1(k)\Theta_1 + \phi_2(k)\Theta_2 - \phi_1(k)\hat{\Theta}_1 \\
&= \phi_2(k)\Theta_2 - \phi_1(k)(\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{E1}^T\Phi_{E2}}{N})\Theta_2 \quad (k = 1, 2, \ldots, N)
\end{aligned} \tag{7}$$

The variance of the one step prediction errors is

$$\begin{aligned}
\sigma_{\epsilon_E}^2 &= E[\epsilon^2(k)] \\
&= E[\Theta_2^T(\frac{\Phi_{E2}^T\Phi_{E2}}{N})\Theta_2 + \Theta_2^T(\frac{\Phi_{E2}^T\Phi_{E1}}{N})(\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{E1}^T\Phi_{E1}}{N})(\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{E1}^T\Phi_{E2}}{N})\Theta_2 \\
&\quad -\Theta_2^T(\frac{\Phi_{E2}^T\Phi_{E1}}{N})(\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{E1}^T\Phi_{E2}}{N})\Theta_2 - \Theta_2^T(\frac{\Phi_{E2}^T\Phi_{E1}}{N})(\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{E1}^T\Phi_{E2}}{N})\Theta_2] \\
&= E[\Theta_2^T(\frac{\Phi_{E2}^T\Phi_{E2}}{N})\Theta_2 - \Theta_2^T(\frac{\Phi_{E2}^T\Phi_{E1}}{N})(\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{E1}^T\Phi_{E2}}{N})\Theta_2]
\end{aligned} \tag{8}$$

Now consider the one step ahead prediction errors over the test data set

$$\begin{aligned}
\epsilon(k) &= y(k) - \hat{y}(k) = \phi_1(k)\Theta_1 + \phi_2(k)\Theta_2 - \phi_1(k)\hat{\Theta}_1 \\
&= \phi_2(k)\Theta_2 - \phi_1(k)(\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{E1}^T\Phi_{E2}}{N})\Theta_2 \quad k = N+1, \ldots, 2N.
\end{aligned} \tag{9}$$

Define $\Phi_{Ti} = [\phi_i^T(N+1), \ldots, \phi_i^T(2N)]^T$ $(i = 1, 2)$, $Y_T = [y(N+1), \ldots, y(2N)]^T$, where the subscript T denotes the test data, to yield the variance of the one step ahead prediction

errors over the test data

$$\sigma^2_{\epsilon_T} = E[\Theta_2^T(\frac{\Phi_{T2}^T\Phi_{T2}}{N})\Theta_2 + \Theta_2^T(\frac{\Phi_{E2}^T\Phi_{E1}}{N})(\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{T1}^T\Phi_{T1}}{N})(\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{E1}^T\Phi_{E2}}{N})\Theta_2$$
$$-\Theta_2^T(\frac{\Phi_{T2}^T\Phi_{T1}}{N})(\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{E1}^T\Phi_{E2}}{N})\Theta_2 - \Theta_2^T(\frac{\Phi_{E2}^T\Phi_{E1}}{N})(\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{T1}^T\Phi_{T2}}{N})\Theta_2] \tag{10}$$

Assuming the input and output are stationary signals, then for large N

$$\frac{\Phi_{E1}^T\Phi_{E1}}{N} \approx \frac{\Phi_{T1}^T\Phi_{T1}}{N} \longrightarrow \text{constant matrix}$$

$$\frac{\Phi_{E2}^T\Phi_{E2}}{N} \approx \frac{\Phi_{T2}^T\Phi_{T2}}{N} \longrightarrow \text{constant matrix}$$

$$\frac{\Phi_{E1}^T\Phi_{E2}}{N} \approx \frac{\Phi_{T1}^T\Phi_{T2}}{N} \longrightarrow \text{constant matrix}$$

$$\frac{\Phi_{E2}^T\Phi_{E1}}{N} \approx \frac{\Phi_{T2}^T\Phi_{T1}}{N} \longrightarrow \text{constant matrix}$$

Eqn (10) becomes

$$\sigma^2_{\epsilon_T} \approx E[\Theta_2^T(\frac{\Phi_{E2}^T\Phi_{E2}}{N})\Theta_2 - \Theta_2^T(\frac{\Phi_{E2}^T\Phi_{E1}}{N})(\frac{\Phi_{E1}^T\Phi_{E1}}{N})^{-1}(\frac{\Phi_{E1}^T\Phi_{E2}}{N})\Theta_2] = \sigma^2_{\epsilon_E} \tag{11}$$

Eqn (11) shows that the variances of the one step ahead prediction errors over the test data and estimation data are approximately equal if the input and output signals are stationary. For example, the variances of the one step prediction error over the estimation and the test data sets were 7.9605×10^{-5} and 8.3867×10^{-5} for the example in §2.2. This implies that if the one step predictions over the estimation data can not detect the presence of unmodelled dynamics, the unmodelled terms may not be detected by the one step ahead prediction over the test data either. Eqns (8) and (10) suggest that if the parameters of the omitted model terms are small, the test can fail even if the input and output are nonstationary. Indeed, experience in practical system identification shows that it is not unusual for the model to have terms that are dynamically important but which have relatively small parameters and make a relatively small contribution to the one step ahead output prediction.

Notice that the problem discussed above is the model underfitting problem. The above illustration and analysis does not contradict the well known result that predictions over the test data may degrade if the model is overfitted.

3 A new identification algorithm based on the model predicted output

3.1 Using model predicted output for model testing and system identification

Consider a nonlinear polynomial model

$$\begin{aligned} y(k) &= F[y(k-1),\ldots,y(k-n_y),u(k-1),\ldots,u(k-n_u), \\ &\quad e(k-1),\ldots,e(k-n_e)] + e(k) \\ &= \sum_{i=1}^{n} \varphi_i(k)\theta_i + e(k) \end{aligned} \tag{12}$$

where $y(k)$ and $u(k)$ denote the output and input at time instant k, $\{e(k)\}$ is an independently and identically distributed random noise sequence with zero mean and finite variance. Many algorithms can be used to identify the model eqn (12), for example the forward regression orthogonal algorithm (Billings *et al* 1988a), prediction error algorithm (Soderstrom and Stoica 1989) and others. But almost all these identification algorithms employ the following cost function or the variants of it

$$J = \sum_{k=1}^{N} [y(k) - \hat{y}(k)]^2 \tag{13}$$

where $\hat{y}(k)$ denotes the one-step ahead prediction at time instant k

$$\hat{y}(k) = \hat{F}[y(k-1),\ldots,y(k-n_y),u(k-1),\ldots,u(k-n_u),\hat{e}(k-1),\ldots,\hat{e}(k-n_e)] \tag{14}$$

and $\hat{F}$ denotes the estimation of the function F, $\hat{e}(k) = y(k) - \hat{y}(k)$ denotes the estimation of the prediction errors. In §2 it was shown that the one step ahead prediction may not be sensitive to the omitted terms which are dynamically important but which have small parameters and which make a relatively small contribution to the one step ahead output prediction.

An alternative criterion for model assessment is the model predicted output. The model predicted output at time instant k denoted by $\overline{y}(k)$ is defined as follows

$$\overline{y}(k) = \hat{F}[\overline{y}(k-1),\ldots,\overline{y}(k-n_y),u(k-1),\ldots,u(k-n_u),0,\ldots,0)] \tag{15}$$

Notice the difference between $\overline{y}(k)$ and $\hat{y}(k)$. In the computation of model predicted output, the prediction errors at previous time instants are inherited by the predictions at later time instants. Thus, the model predicted output is more sensitive to the unmodelled terms and model predicted output is a more severe requirement on the model accuracy than one step ahead predictions. Indeed, The inadequency of model (2) becomes obvious if model predicted outputs are used to assess the model, as shown in Figure 4.

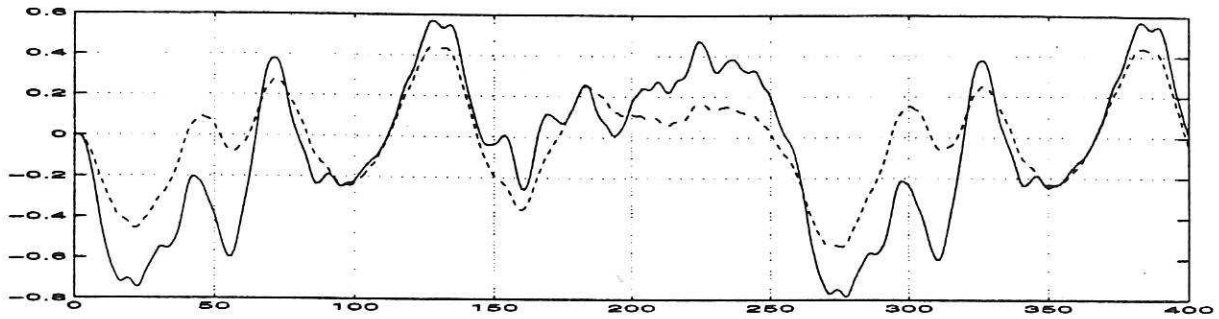

Figure 4: Model predicted outputs for model eqn (2) (solid–measurement, dashed–model predicted output)

Model predicted output has long been used for model assessment (see for example Ljung 1987, Billings *et al* 1989, Billings *et al* 1992). But until recently the model predicted output has not been used as an optimisation index in system identification. Berger (1995 and 1997) proposed the cost function

$$J = \sum_{k=1}^{N} [y(k) - \overline{y}(k)]^2 \tag{16}$$

but assumed the model structure was known *a priori*. Optimisation of the model predicted outputs were achieved through an optimising parameter estimation. But when the model structure is unknown, optimisation of the model predicted output should be done by selecting the correct model structure because model predicted outputs mainly depend on the model structure. If a model structure is inadequate to describe the system, parameter optimisation can achieve very limited improvements on the model predicted output.

3.2 The basic idea for a new algorithm

Motivated by the above analysis a new model structure detection algorithm using the model predicted output is developed in the present study. Actually the new algorithm minimises three performance indices: one step ahead prediction errors, model predicted output errors and model size. This multi-objective optimisation problem can be summarised as follows

$$J_1 = \min \left\{ \sum_{k=1}^{N} [y(k) - \overline{y}(k)]^2 \right\}$$

$$J_2 = \min \{n\}$$

$$J_3 = \min \left\{ \sum_{k=1}^{N} [y(k) - \hat{y}(k)]^2 \right\}$$

where n denotes the number of terms in the model. This multi-objective optimisation problem can be solved by optimising a combined cost function which is the weighted sum of the three performance indices

$$J = \lambda_1 \sum_{k=1}^{N} [y(k) - \hat{y}(k)]^2 + \lambda_2 \sum_{k=1}^{N} [y(k) - \overline{y}(k)]^2 + \lambda_3 n \tag{17}$$

where λ_i $(i = 1, 2, 3)$ are weighting elements. Minimising this combined cost function has, however, at least two difficulties. First, the weighting elements are very difficult to select in practice. Although it is often claimed that weighting elements can be assigned in terms of the designers preferences it is very difficult to establish a quantitative relationship between values of the weighting elements and model properties such as approximation accuracy. Second, the combined cost function is a high order nonlinear function of the unknown model parameters. Even if the model structure is known *a priori* directly optimising the high order cost function is impractical even if genetic algorithms are employed.

In this study the three performance indices are minimised separately using a multi-objective genetic algorithm. Because an approximation accuracy requirement is often set for the model to be identified, the three-objective optimisation problem can be converted into a constrained double-objective optimisation problem

$$J_1 = \min \left\{ \sum_{k=1}^{N} [y(k) - \overline{y}(k)]^2 \right\}$$

$$J_2 = \min \{n\}$$

subject to

$$1 - \frac{\sum_{k=1}^{N} [y(k) - \hat{y}(k)]^2}{\sum_{k=1}^{N} y^2(k)} > cutoff_1$$

where $cutoff_1$ is the required approximation accuracy. In this study the system will be limited to linear-in-the-parameter models including polynomial models and rational models, so that the orthogonal algorithm (Korenberg *et al* 1988) can be used to select model terms. The orthogonal algorithm selects model terms according to the contributions to the one step ahead predicted output using the error reduction ratios. Because the error reduction ratios depend on the order in which the candidate terms are orthogonalised, employing different orthogonalisation paths which specify the orthogonalisation order for all candidate terms, might yield different model structures. The forward regression orthogonal algorithm (Billings *et al* 1988a) and minimal model structure detection algorithms (Mao and Billings 1997) have been developed as possible solutions to this problem. However, this property of the standard orthogonal algorithm can be made full use of in this study to produce a set of candidate models, the optimal model can then be selected from the set according to the model predicted output and model size.

Genetic algorithms (GAs) have been successfully applied for model structure detection

and were implemented as a constrained single-objective optimisation problem (Mao and Billings 1997). The above constrained double-objective optimisation problem can be solved in a similar way using multi-objective GAs. The difference between single-objective GAs and multi-objective GAs is that the latter involve a ranking process which converts performance index vectors into scalars. Details of the new algorithm using multi-objective GAs are described below.

3.3 Details of the identification algorithm

Once they are incorporated into the framework of a genetic algorithm different optimisation problems are solved in a similar way. Typically, a genetic algorithm consists of the following operations; encoding, fitness value assignment, reproduction, crossover and mutation. Some of the operations in the multi-objective genetic algorithm are similar to those used in the single-objective genetic algorithm (Mao and Billings 1997), and are therefore summarised in Appendix I. Only the ranking and fitness operation will be desceribed here.

3.3.1 Ranking and fitness assignment

Inspired by the principle of natural evolution that individuals which adapt to the environment will survive and hand down chromosomes to descendants, optimal search procedure of GAs is guided by the fitness of individuals (Goldberg 1989). In single objective genetic algorithms, fitness can easily be assigned as proportional or inversely proportional to the value of the performance index that is to be optimised. In the present multi-objective optimisation problem, the performance index of each individual is a vector which consists of the model predicted output errors and model size. The performance index vectors need to be transformed into scalars in order to assign fitness values in a similar way to the single-objective case. The transformation process is called ranking, and the transformed scalars are called ranks.

Each individual represents an orthogonalisation path. By applying a standard orthogonal algorithm using the order defined by the individuals, a set of candidate models can be obtained. Model predicted outputs corresponding to each model can be computed using eqn (15) and ranked (Fonesca and Fleming 1998). Assuming that the performance index vectors for individuals i and j are respectively $[J_{1i}\ J_{2i}]$, $[J_{1j}\ J_{2j}]$. There are 9 relations between the two vectors

(1) $J_{2j} < J_{2i},\ J_{1j} < J_{1i}$ (2) $J_{2j} = J_{2i},\ J_{1j} < J_{1i}$

(3) $J_{2j} < J_{2i},\ J_{1j} = J_{1i}$ (4) $J_{2j} > J_{2i},\ J_{1j} > J_{1i}$

(5) $J_{2j} = J_{2i},\ J_{1j} > J_{1i}$ (6) $J_{2j} > J_{2i},\ J_{1j} = J_{1i}$

(7) $J_{2j} = J_{2i},\ J_{1j} = J_{1i}$ (8) $J_{2j} < J_{2i},\ J_{1j} > J_{1i}$

(9) $J_{2j} > J_{2i},\ J_{1j} < J_{1i}$

The physical interpretation of cases (1) and (2) is that the candidate model j has smaller or equal model predicted output errors than model i, but the number of terms in model j is less. In case (3), the model j has the same prediction error as model i, but the number of terms in model j is less. In these three cases model j is considered as better than model i. Following a similar analysis, it can be seen that model j is no better than model i in cases (4)-(7). In case (8), model j has less terms than model i, if the model predicted output error of model j is slightly larger, that is

$$\frac{1}{\sum_{k=1}^{N} y^2(k)} \frac{J_{1j} - J_{1i}}{J_{2i} - J_{2j}} < cutoff_2$$

then model j can be considered as a better model than model i, where $cutoff_2$ is the threshold which is typically set to $0.01\% - 0.05\%$.

In case (9), model j has more terms than model i, if it has a significant reduction of the model predicted output error compared with model i, that is

$$\frac{1}{\sum_{k=1}^{N} y^2(k)} \frac{J_{1i} - J_{1j}}{J_{2j} - J_{2i}} > cutoff_2$$

then model j can be considered as better than model i.

Comparing the performance index vector of each individual with the others in the current population, the rank of each individual can be determined.

Once a ranking is obtained, the fitness can be computed using the following mapping scheme

$$f_i = f_{max} - \frac{f_{max} - f_{min}}{r_{max} - r_{min}}(r_i - r_{min}) \tag{18}$$

where r_i denotes the rank of the i^{th} individual. r_{min}, r_{max}, and f_{min}, f_{max} are the minimum and maximum ranks in the current population, and the minimum and maximum fitness values respectively. In this study f_{min} and f_{max} are set to 0.5 and 1 respectively.

Typically, a genetic algorithm consists of encoding, fitness assignment, reproduction, crossover and mutation. The successive application of these operations in the new identification procedure is described in the following subsection.

3.3.2 Summary of the new algorithm

The new algorithm can be summarised as follows

(I) Generate an initial population set $\mathcal{P}$ consisting of l individuals, each individual represents an orthogonalization path. Set the current generation number $i = 1$.

(II) Apply the standard orthogonal algorithm using the orthogonalization path represented by each individual to obtain the corresponding model structure, compute the

model predicted output, corresponding model size, rank and fitness value. Form a mating pool $\mathcal{M}$ using all individuals in the population set $\mathcal{P}$ at the probabilities assigned to each individual according to the corresponding fitness value.

(III) Randomly select a pair of parent strings from the mating pool $\mathcal{M}$. Choose a random crossover point and exchange the parent string bits to produce two offsprings and put the offsprings in the offspring set $\mathcal{O}$. Repeat this procedure $l/2$ times.

(IV) Mutate each bit of each offspring in the set $\mathcal{O}$ with a pre-specified mutation rate and calculate the fitness value of each mutated offspring using the procedure summarised in step (II).

(V) Select the l fittest individuals from sets $\mathcal{P}$ and $\mathcal{O}$ by comparing fitness values.

(VI) Reset the set $\mathcal{P}$ with the newly selected l individuals, reset the number of generations $i = i + 1$, and nullify the offspring set $\mathcal{O}$.

(VII) Steps (II)-(VI) are repeated until a pre-specified number of generations arrives.

Setting $cutoff_1$ and $cutoff_2$ to 99.9% and 0.02% respectively. Applying the above procedure to the example in §2, produced the following model

$$\hat{G}(z^{-1}) = \frac{0.000248 + 0.00226z^{-1} + 0.0018z^{-2}}{1 - 2.309z^{-1} + 1.744z^{-2} - 0.429z^{-3}} \tag{19}$$

The model predicted output of model eqn (19) are shown in Figure 5 which indicates that the model is adequate to represent the original system eqn (1). This is confirmed by the frequency and unit step responses of the model shown in Figure 6 which are almost identical to those of the original system shown in Figures 3(b) and 2(b).

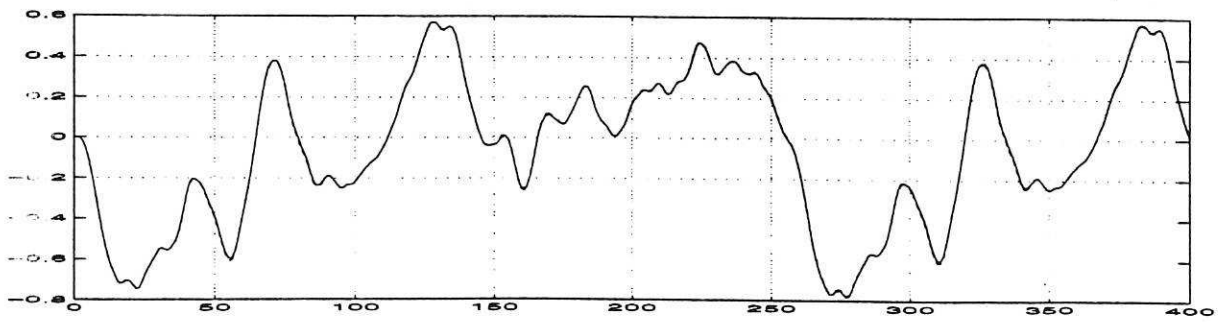

Figure 5: Model predicted outputs for model eqn (19) (solid–measurement, dashed–model predicted output)

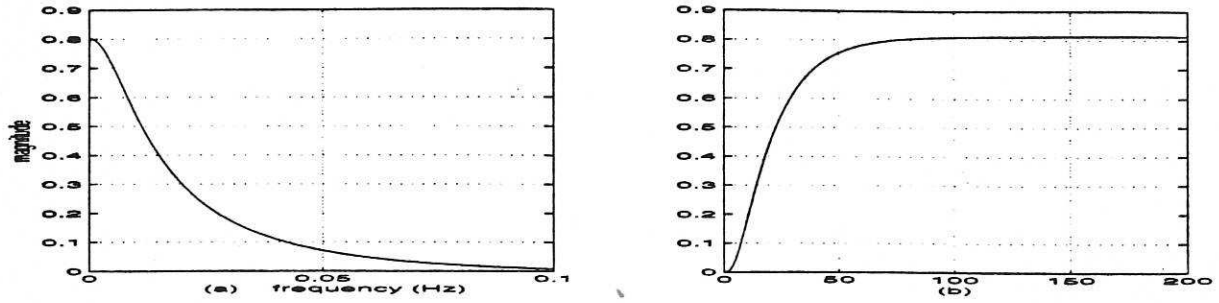

Figure 6: Frequency and step responses for model eqn (19) (a) frequency responses, (b) unit step responses

4 Examples

4.1 Example 1

Consider the following model

$$G(s) = \frac{2 \times 229}{(s+1)(s^2+30s+229)} \tag{20}$$

Using a PRBS sequence with amplitude ± 3 as the input and a sampling period of 0.05 seconds 800 data samples were generated. The first 400 data samples were used for parameter estimation, the remaining 400 data samples were used for testing. The maximum lags of the input and output were set to be 4, and the one step ahead prediction approximation accuracy was set to 99.98%. Fitting the data using a conventional algorithm produced the following model

$$\hat{G}(z^{-1}) = \frac{0.0144z^{-1} + 0.0099z^{-2}}{1 - 1.939z^{-1} + 1.198z^{-2} - 0.245z^{-3}} \tag{21}$$

The one step ahead predictions over both the estimation and test data are perfect. But the model predicted outputs, shown in Figure 7, indicates that the model is inadequate.

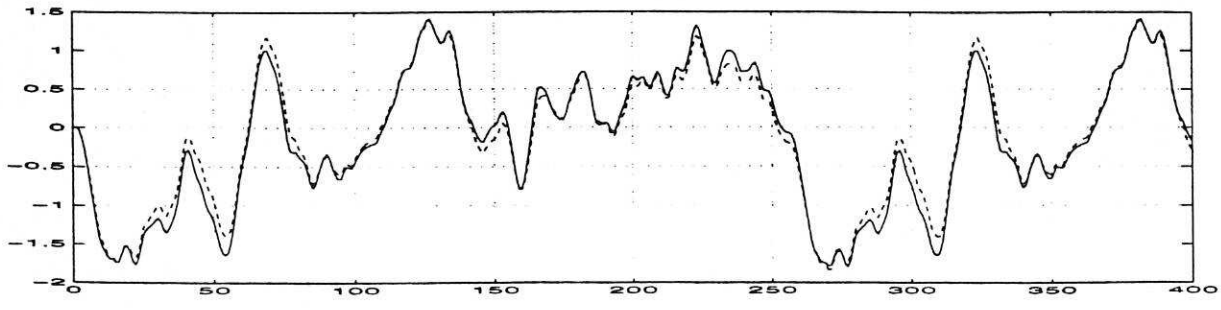

Figure 7: Model predicted outputs for model eqn (21) (solid-measurement, dashed-model predicted output)

This is confirmed by the frequency and step responses of model (20) as shown in Figures 8-9.

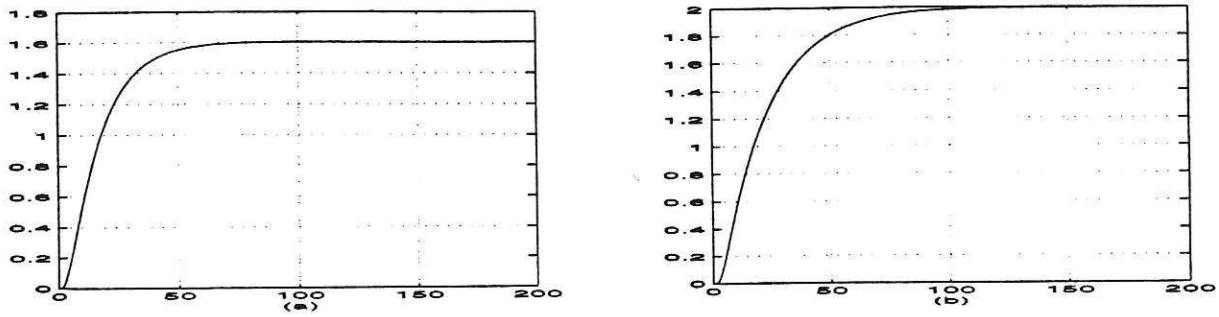

Figure 8: Unit step responses for models eqn (20) and eqn (21) (a) the identified model eqn (21), (b) the original model eqn (20)

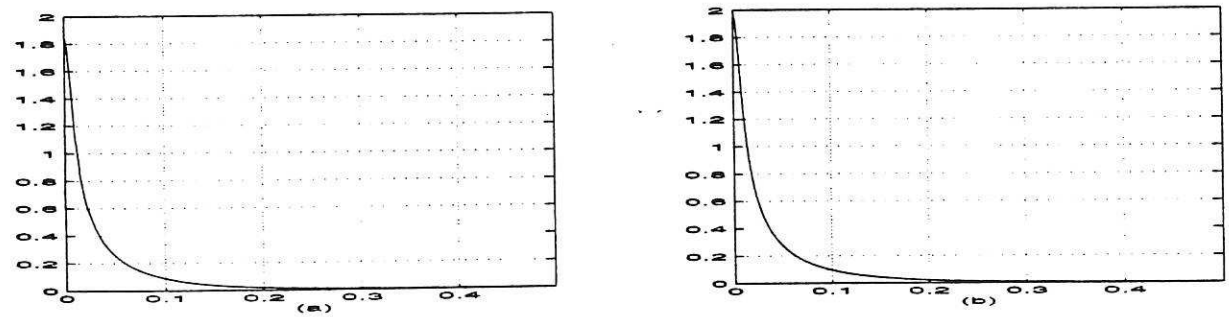

Figure 9: Frequency responses for models eqn (20) and eqn (21) (a) the identified model eqn (21), (b) the model eqn (20)

Setting the approximation accuracy of the one step ahead prediction $cutoff_1$ to 99.98%, $cutoff_2$ to 0.02%, and applying the procedure summarised in §3.3.2, produced the following model

$$\hat{G}(z^{-1}) = \frac{0.0018 + 0.0144z^{-2} + 0.0099z^{-2}}{1 - 1.939z^{-1} + 1.198z^{-2} - 0.245z^{-3}} \tag{22}$$

The model predicted output shown in Figure 10 indicates that the model eqn (22) is adequate to represent the original system. This is confirmed by the step and frequency responses shown in Figure 11.

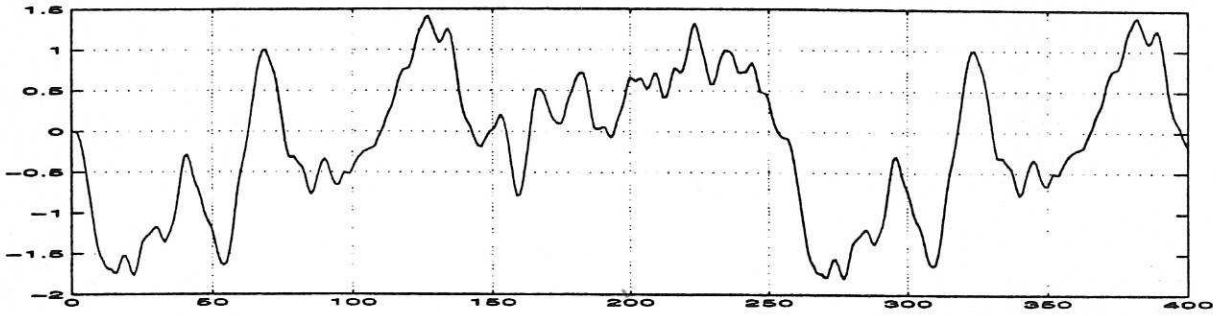

Figure 10: Model predicted outputs for model eqn (22) (solid-measurement, dashed-model predicted output)

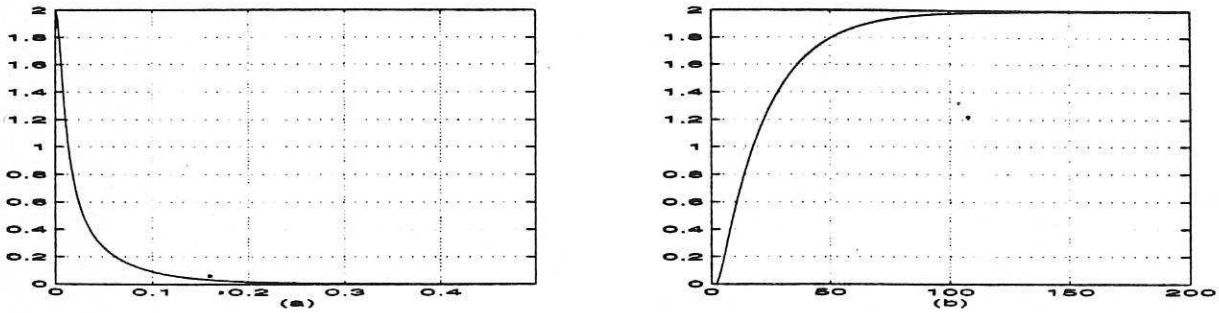

Figure 11: Frequency and step responses for model eqn (22) (a) frequency responses, (b) unit step responses

4.2 Example 2

Consider the nonlinear Hammerstein system shown in Figure 12

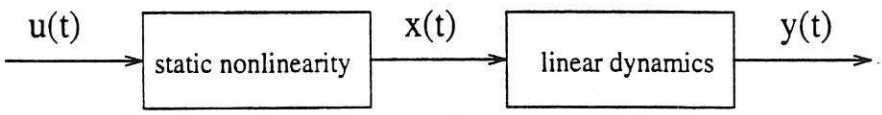

Figure 12: A nonlinear Hammerstein system

The static nonlinear element scales the input $u(t)$ to produce $x(t)$, the linear dynamics are represented by the transfer function $G(s)$, whose output is $y(t)$. In this example the static nonlinear element and the linear dynamics are as follows

$$x(t) = 0.5u^2(t) + u(t) \tag{23}$$

$$G(s) = \frac{256(0.976s + 1)}{26.27s^3 + 36.31s^2 + 10.14s + 1} \tag{24}$$

The input was a uniformly distributed random sequence with zero mean and amplitude ± 1. A total of 800 data points were obtained by sampling at an interval of 1 second. The first 400 data samples were used for parameter estimation, the remaining 400 data samples were used for testing.

Assuming that the maximum lags of the input and output of the linear block are 3, the approximation accuracy $cutoff_1$ was set to 99.98%. Fitting the data using a linear ARX model and a conventional algorithm produced the model

$$\hat{G}(z^{-1}) = \frac{1.5311 + 4.3351z^{-1} - 2.1735z^{-2}}{1 - 2.3188z^{-1} + 1.8247z^{-2} - 0.5028z^{-3}} \tag{25}$$

The one step ahead predicted output over the estimation and test data were perfect. This suggests that the identified model in eqn (25) is an excellent representation of the system. However this is not true because the original system is nonlinear. The different unit step responses of the identified model and the true system, illustrated in Figure 13 (a) and (b), clearly show that the model eqn (25) is incorrect. The insufficiency of model eqn (25) was successfully detected by the model predicted outputs as shown in Figure 14.

Now setting $cutoff_2$ to 0.02% and identifying the system using the procedure summarised in §3.3.2 produced the following model

$$\begin{aligned} y(k) &= 1.67y(k-1) - 0.688y(k-2) - 0.012y(k-3) + 1.45u(k) + 5.33u(k-1) \\ &\quad +1.26u(k-2) + 0.724u^2(k) + 2.66u^2(k-1) + 0.63u^2(k-2) \end{aligned} \tag{26}$$

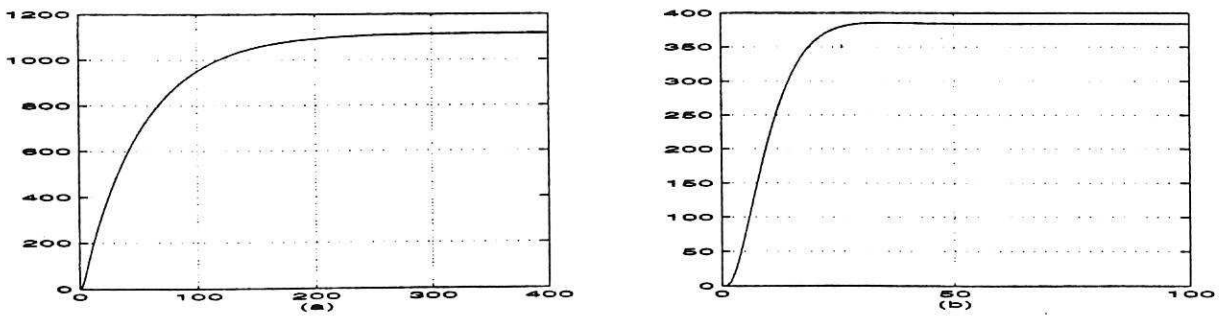

Figure 13: Unit step responses for models eqns (23)-(24) and eqn (25) (a) the identified model eqn (25), (b) the model eqns (23)-(24)

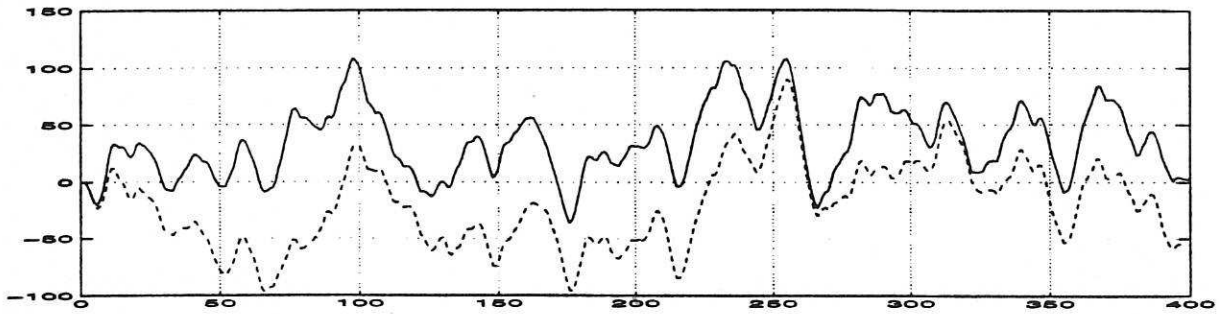

Figure 14: Model predicted outputs for model eqn (25) (solid-measurement, dashed-model predicted output)

The model predicted output and the unit step responses are shown in Figure 15 (a)-(b). Clearly the model has almost perfect model predicted outputs and almost identical unit step responses with the original system model eqns (23)-(24).

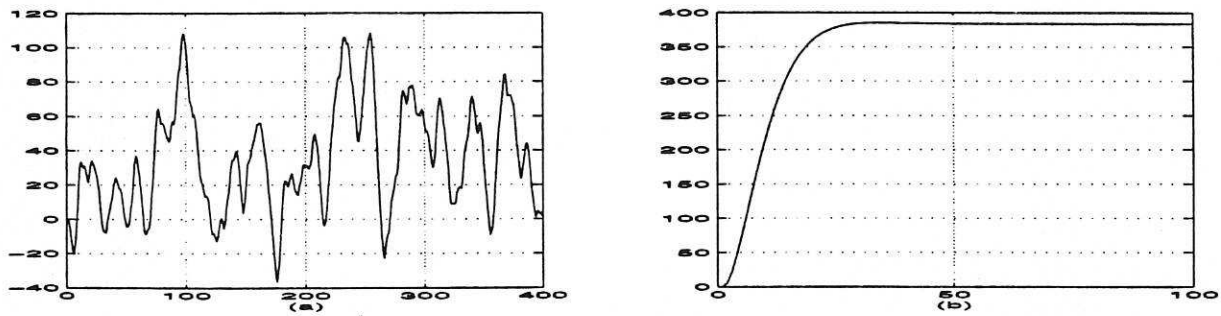

Figure 15: Model predicted output and step responses for model eqn (26) (a) model predicted output (solid-measurement, dashed-model predicted output), (b) unit step responses

5 Conclusions

Conventional system identification and model assessment are based on the one-step ahead predictions. In this study the efficiency of one step ahead prediction has been examined. Theoretical analysis and numerical examples have demonstrated that one step ahead predictions may not always be sufficient to measure model quality. The efficiency of model predicted output for model assessment has been established and a new system identification algorithm based on the minimisation of model predicted outputs has been developed.

6 Acknowledgements

SAB gratefully acknowledges that part of this work was supported by EPSRC.

References

[1] Berger, C.S., 1995, Recursive single-layer nets for output error dynamic models, *IEEE Transaction on Neural Networks*, 6(2), 508-511.

[2] Berger, C.S., 1997, Modeling dynamic systems using finite elements, *International Journal of Control*, 68(3), 431-448.

[3] Billings, S.A., M.J.Korenberg and S.Chen, 1988a, Identification of output-affine systems using an orthogonal least squares algorithm, *International Journal of Systems Science*, 19, 1559-1568.

[4] Billings, S.A., S.Chen and M.J.Korenberg, 1988b, Identification of MIMO non-linear systems using a forward-regression orthogonal estimator, *International Journal of Control*, 49, 2157-2189.

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

[6] Billings, S.A., H.B.Jamaluddin and S.Chen, 1992, Properties of neural networks with applications to modelling nonlinear dynamic-systems, *International Journal of Control*, 55 (1), 193-224.

[7] Billings, S.A. and K.Z.Mao, 1997, Rational model data smoothers and identification algorithms, *International Journal of Control* , 68(2), 297-310.

[8] Brouwn, G.G., A.Krigsman, H.B.Verbruggen, and P.M.Bruijn, 1994, Single layer networks for nonlinear system identification, *Engineering Applications of Artificial Intelligence*, 7(3), 227-243.

[9] Chen, S., C.F.N.Cowan, and P.M.Grant, 1991, Orthogonal least squares learning algorithm for radial basis function networks, *IEEE Transaction on Neural Networks*, 2, 302-309.

[10] Chen, S. and S.A.Billings, 1992, Neural networks for nonlinear dynamic system modelling and identification, *International Journal of Control*, 56, 319-346.

[11] Fonseca, C.M. and P.J.Fleming, 1998, Multiobjective optimisation and multiple constraint handling with evolutionary algorithms, Part I: a united formulation, *IEEE Transactions on Systems, Man and Cybernetics Part A-Systems and Humans*, 28 (1), 38-47.

[12] Ligntbody, G., P.O'Reilly, G.W.Irwin, K.Kelly and J.McCcormick, 1997, Neural modelling of chemical plant using MLP and B-spline networks, *Control Engineering Practice*, 5(11), 1501-1515.

[13] Goldberg, D.E., *Genetic Algorithms in Search, Optimization and Machine Learning*, Reading, Massachusetts: Addison-Wesley, 1989.

[14] Jang, J.S.R. and C.T.Sun, 1993, Functional equivalence between radial basis function networks and fuzzy inference systems, *IEEE Transaction on Neural Networks*, 4, 156-159.

[15] Korenberg, M.J., S.A.Billings, Y.P.Liu and P.J.McIlroy, 1988, Orthogonal paramter estimation algorithm for nonlinear stochastic systems, *International Journal of Control*, 48, 193-210.

[16] Ljung, L. 1987, *System Identification: Theory For the User*, London: Prentice-Hall.

[17] Mao, K.Z. and S.A.Billings, 1997, Algorithms for minimal model structure detecion in nonlinear dynamic system identification, *International Journal of Control*, 67(2), 311-330.

[18] Nahas, E.P., M.A.Henson and D.E.Seborg, 1992, Nonlinear internal model control strategy for neural network models, *Computer & Chemical Engineering*, 16(12), 1039-1057.

[19] Smith, C.A. and A.B.Corripio, 1997, *Principles and Practice of Automatic Process Control*, second edition, John Wiley and Sons, Inc.

[20] Soderstrom, T. and P.Stoica, 1989, *System Identification*, Prentice Hall.

[21] Wang, L.X. and J.M.Mendel, 1992, Fuzzy basis functions universal approximation, and orthogonal least squares learning, *IEEE Transaction on Neural Networks*, 3, 807-814.

[22] Wang, L. and R.Langari, 1995, Building sugeno-type models using fuzzy discretization and orthogonal parameter estimation techniques, *IEEE Transaction on Fuzzy System*, 3(4), 454-458.

[23] Zhu, H. and R.Rohwer, 1996, No free lunch for cross test, *Neural Computation*, 8(7), 1421-1427.

[24] Zhu, Q.M. and S.A.Billings, 1993, Parameter estimation for stochastic nonlinear rational models, *International Journal of Control* , 57, 309-333.

Appendix I

A.1 Encoding

The bits of each individual represent the order in which candidate terms are orthogonalised. For example the code of an individual of a model with 8 candidate terms is of

the following form

$$\overbrace{1}^{P_1}\ \overbrace{6}^{P_2}\ \overbrace{3}^{P_3}\ \overbrace{7}^{P_4}\ \overbrace{2}^{P_5}\ \overbrace{4}^{P_6}\ \overbrace{8}^{P_7}\ \overbrace{5}^{P_8}$$

where P_i denotes the i^{th} orthogonalised term. The physical interpretation of the above string is that the first term orthogonalised in eqn (12) is $\varphi_1(k)$, the second term orthogonalised is $\varphi_6(k)$, ..., the last term orthogonalised is $\varphi_5(k)$.

A.2 Reproduction

The *roulette wheel* approach was employed to implement the reproduction procedure in this study. Each string is allocated a slot of the roulette wheel subtending an angle proportional to its fitness at the center of the wheel. A random number in the range of 0 to 2π is generated. A copy of a string goes to the mating pool if the random number falls in the slot corresponding to the string. For a population with size l, the reproduction process is repeated l times and l strings go into the mating pool.

A.3 Crossover

The purpose of the crossover operation is to generate new individuals by exchanging bits. Assuming that two randomly selected parent strings are given by

$$\overbrace{1}^{P_1}\ \overbrace{6}^{P_2}\ \overbrace{3}^{P_3}\ \overbrace{7}^{P_4}\ \overbrace{2}^{P_5}\ \overbrace{4}^{P_6}\ \overbrace{8}^{P_7}\ \overbrace{5}^{P_8}$$

$$\overbrace{1}^{P_1}\ \overbrace{3}^{P_2}\ \overbrace{8}^{P_3}\ \overbrace{5}^{P_4}\ \overbrace{4}^{P_5}\ \overbrace{7}^{P_6}\ \overbrace{2}^{P_7}\ \overbrace{6}^{P_8}$$

First randomly select the bit at which orthogonalised term will be changed, for example P_4. Then detect terms at this position in the parent strings, *i.e.*, $\varphi_7(k)$ and $\varphi_5(k)$. Exchanging positions of $\varphi_5(k)$ and $\varphi_7(k)$ in each string, yields two offsprings

$$\overbrace{1}^{P_1}\ \overbrace{6}^{P_2}\ \overbrace{3}^{P_3}\ \overbrace{5}^{P_4}\ \overbrace{2}^{P_5}\ \overbrace{4}^{P_6}\ \overbrace{8}^{P_7}\ \overbrace{7}^{P_8}$$

$$\overbrace{1}^{P_1}\ \overbrace{3}^{P_2}\ \overbrace{8}^{P_3}\ \overbrace{7}^{P_4}\ \overbrace{4}^{P_5}\ \overbrace{5}^{P_6}\ \overbrace{2}^{P_7}\ \overbrace{6}^{P_8}$$

A.4 Mutation

The purpose of mutation is to generate a different individual which is not easy to achieve by the crossover operation. In this study mutation is achieved by exchanging the selected bit with a random selected bit of the same string. Consider for example, the following string

$$\overbrace{1}^{P_1}\ \overbrace{6}^{P_2}\ \overbrace{3}^{P_3}\ \overbrace{7}^{P_4}\ \overbrace{2}^{P_5}\ \overbrace{4}^{P_6}\ \overbrace{8}^{P_7}\ \overbrace{5}^{P_8}$$

If bit P_1 is supposed to mutate, exchanging this bit with a randomly selected bit from $P_2 - P_8$, for example P_6, yields the mutated string

$$\overbrace{4}^{P_1}\ \overbrace{6}^{P_2}\ \overbrace{3}^{P_3}\ \overbrace{7}^{P_4}\ \overbrace{2}^{P_5}\ \overbrace{1}^{P_6}\ \overbrace{8}^{P_7}\ \overbrace{5}^{P_8}$$