DESIGN ISSUES IN APPLYING NEURAL NETWORKS TO MODEL HIGHLY NON-LINEAR PROCESSES

S.K. Doherty, J.B. Gomm, D. Williams and D.C. Eardley

Control Systems Research Group, Liverpool John Moores University, U.K.

INTRODUCTION

Over the past decade, there has been a resurgence of research interest into the field of Artificial Neural Networks (ANNs) in such diverse areas as signal processing, pattern recognition and financial forecasting. In control engineering, this research has been particularly intense and has focused on the modelling of dynamic systems. A valid process model is the keystone of modern control systems design, the quality of control being determined by the validity of the model. In fact, many control strategies (e.g. Model Predictive Control) incorporate the model directly into the control system.

There are two different approaches to model development. If the system's dynamics can be described analytically then a first principle model can be derived. This approach, relies on good process information being available and even for relatively simple and well understood processes, can be a difficult and time consuming task.

Alternatively, a parametric model can be obtained using system identification techniques. These methods are usually based on several assumptions about the process, the most important being linearity and time-invariance. The model structure must first be specified. Process input/output data is then used to calculate the model parameters by linear regression. If the process is time-variant, these parameters can be periodically updated. If the process is non-linear, an array of linear models can be used to span the operating range.

While these methodologies usually provide acceptable solutions, there are many cases where they are prone to failure. Often, it is not possible to adequately represent system characteristics such as non-linearity, time delay, saturation, time-varying parameters and overall complexity. Such is the case with many chemical processes due to their intrinsic uncertainties and strong non-linearities. ANNs offer the promise of a modelling tool capable of emulating such complex systems, which is why they are being vigorously investigated by the control systems community.

Indeed, the modelling abilities of ANNs have been demonstrated (e.g. Bhat and McAvoy (1), Bhat et al. (2), Chen et al. (6, 7)), and they have been used successfully to control non-linear processes, both in simulation (e.g. Saint Donat et al. (3, 4), Hernández and Arkun (10), Chen and Khalil (11)) and on-line (e.g. Evans et al. (14)).

ANNs are not a panacea for all the problems of process modelling, as they do have their limitations. For instance, the successful training of an ANN model often requires considerable amounts of experimental input/output data, and the resulting models are unreliable outside the operating regimes of this training data. Also, it is not possible to directly incorporate any plant knowledge, which may exist, into an ANN model. However, there are many plants for which there is an abundance of data and a scarcity of plant information. It is in this area that ANNs offer a cost effective, efficient modelling tool.

Before an ANN model can be trained, several parameters must be selected. This is the design stage. Choices are made concerning the identification experiment, the type of ANN used and its training algorithm, the network topology and how the validity of the model is assessed. This is the scope of the engineer's input to what is otherwise a black box approach. Hence, these decisions are extremely important as they ultimately determine the validity of the resulting model.

This paper addresses these decisions and their consequences. This study is part of a research program whose ultimate objective is to control a pilot scale in-line pH neutralisation plant using an ANN approach. Conventional methods of pH control usually use a series of Continuous Stirred Tank Reactors (CSTRs) whose capacity serves to dampen pH fluctuations. A practical pH process cannot be adequately represented or controlled using linear approximations. PID feedback control, gain scheduling and linear adaptive control have all been tried and rejected as likely control strategies (Choi and Rhinehart(17)). pH control remains the ultimate example of a highly non-linear process, and is therefore ideally suited to appraising the non-linear modelling capabilities of ANNs.

This paper looks at the development of an ANN model of a CSTR. The modelling of the in-line pH plant will involve additional problems (see Rhinehart(15), Williams et al. (16)). However, the point at which reagent is injected can be modelled as a CSTR (Jacobs et al. (19)), and was therefore a logical starting point for the project.

While other workers (Bhat and McAvoy (1),Bhat et al.(2), Saint Donat et al.(3,4)) have used ANNs to model pH in a CSTR, the difficulties in developing, and the limitations of, such models have not been sufficiently addressed. In particular, the choices relating to the excitation signal used to train the ANN, the data sample time, the topology of the network and validation of the models have not be satisfactorily analyzed. While these design issues may not be crucial for processes with mild non-linearities, they are of paramount importance for systems with severe non-linearities, such as the pH process, and if not properly addressed the resulting ANN model prediction errors can be totally unacceptable. Studies have been conducted examining all of these choices and this paper focuses on two issues, namely selecting an optimum sample time and network topology.

AN ANN MODEL OF THE CSTR PROCESS

In the process shown in fig. 1, acetic acid (CH_3COOH) of concentration C_A flows into the tank at flow rate F_A, and is neutralised by sodium hydroxide (NaOH) of concentration C_B which flows into the tank at rate F_B. The volume of the tank is constant and its contents are assumed to be perfectly mixed. This chemical system, which is weak acid/strong base, is characterised by the steady state (titration) curve shown in Fig. 3. A first principle model of the process is obtained by making material balances on acetate and sodium, using the acetic acid and water equilibrium relationships and the fact that the solution must be electrically neutral. This modelling approach was introduced by McAvoy et al. (5) for the single acid/single base process and generalised by Gustaffson and Waller (18).

A simulation of the process using this first principle model was implemented in the Advanced Continuous Simulation Language (ACSL) using the model parameters detailed in Bhat and McAvoy (1).

CONTROL '94, 21–24 March 1994. Conference Publication No. 389, © IEE 1994.

F_A was held constant and a 10% Random Amplitude Signal (RAS) was superimposed on a steady state, F_B, to provide dynamic input/output data (F_B and pH) for ANN training and model validation. The ANNs incorporated Multi-Layered Perceptrons (MLPs) with two layers of sigmoidal activation functions, trained with the backpropagation algorithm (Werbos (20)). The MLP is widely used in process control applications mainly because its non-linear approximation properties have been mathematically proven (Cybenko(21)).

Process dynamics were incorporated by using delayed values of process input and output as the ANN inputs. The NARX (Non-linear Auto Regressive eXogenous) training structure was adopted because of its ease of implementation. It assumes that for a SISO system the process dynamics can be represented as..

$$\begin{aligned} y(t) = f[&u(t-k),..,u(t-k-n_u+1), \\ &y(t-1),..,y(t-n_y)] + e(t) \end{aligned} \tag{1}$$

where u is the process input, y is the process output, e is process noise and f is some non-linear function. The ANN is trained to predict the process output, y, based on past values of u and y. Commonly, n_u and n_y are chosen such that $n_u = n_y = n$, and n is termed the order of the model. k is the model deadtime (k ≥ 1). Hence, for a 5th order NARX model structure, the ANN inputs consist of five past values of u and five past values of y, and its output is the present value of y. During training, the ANN is required to approximate the non-linear function, f, by means of adjusting its weights so as to minimize the RMS of the prediction errors (the difference between the network and the target outputs) for the training data set.

An ANN model, in a 5th order NARX configuration with k=1, was trained with data generated using a RAS excitation signal with a 240 sec. clock period and a sample time of 24 secs. Bhat et al. (1, 2) and Saint Donat et al. (4) have studied a similarly structured network, using these parameters, training it to predict 5 future values of pH.

The Mean Square Error (MSE) between the ANN prediction and the target output was evaluated for six disparate test signals, which were selected so as to comprehensively test the ANN model. They included data generated with RASs of different frequencies and different amplitude ranges and steady state data. Fig. 2a shows a typical one step-ahead prediction for this ANN. The combined MSE for the six test signals was 0.47 pH units, which could be interpreted as an acceptable modelling error. However, as fig. 2b illustrates, the magnitude of some of these prediction errors is in excess of 2 pH units. This was not the worst case. Prediction on a high frequency RAS resulted in errors of 3.4 pH units.

The largest prediction errors occurred near the equivalence point of the acid/base system, which is where the process gain is a maximum. For this acid/base system the equivalence point is at pH 9.2. Unfortunately, for many neutralisation processes, the control setpoint is often in the vicinity of the equivalence point, making the use of this ANN in a model based control strategy questionable.

This ANN does not faithfully emulate the dynamics of the CSTR pH process. On examining the training MSE it indicated that the ANN was not converging acceptably. The ineptness of this model was confirmed by applying the correlation model validity tests (Chen et al.(8, 9)). These tests are one of the few proposed methods of verifying the adequacy of a non-linear model, other than cross validation on unseen test signals. The procedure is to assess the correlation between the model's prediction errors (ϵ) and its input (u). ϵ should be unpredictable from all linear and non-linear combinations of ϵ and u. Specifically, if five correlation functions fall within the 95% confidence intervals, the model can be regarded as adequate.

This ANN model failed four of the five tests. The results of these tests exceeded the 95% confidence limits at seemingly random values of lag. Increasing the order of the ANN model did not improve the correlation results. This suggested that the ANN failed to accurately emulate the process not because it lacked information at a particular lag, nor because it was under-parameterised.

IDENTIFICATION OF LOCAL LINEAR MODELS

The above ANN failed to accurately model the process. Likely causes were thought to include inappropriate model structure and data sample time. The suggested correlation tests for assessing the adequacy of non-linear models had proved fruitless, and it was therefore decided to study local linear models to see if useful information could be obtained. It was argued that since non-linear processes can often be represented by an array of linearised models, the structure and the parameters of local models may be relevant to their non-linear counterparts.

Fig. 3 shows seven operating points about which local ARX models were identified. The ARX model was selected since the ANNs were trained in the non-linear configuration of this model, namely NARX. The magnitude of the RAS input was set to give approximately ±0.1 pH units maximum output swing, over which range the process characteristic could be approximated by a linear model. The severe process non-linearity was illustrated by the gain variation of the identified models, changing by a factor of 150 between operating points 4 and 7.

Optimum Local Model Structure

The ARX model is expressed by a difference equation of the form..

$$\begin{aligned} y(t) = &-a_1y(t-1)-...-a_{n_a}y(t-n_a) \\ &+b_0u(t-n_k)+...+b_{n_b}u(t-n_k-n_b+1) \\ &+e(t) \end{aligned} \tag{2}$$

ARX models were identified using different model orders for fixed data sample time. For each identified model the model validity functions defined in table 1 were noted. The Loss Function (LF) is simply the MSE of the prediction errors. Akaike's Final Prediction Error (AFPE) and Akaike's Information Criterion (AIC) are weighted functions of the LF which penalise for reductions in the prediction errors at the expense of increasing model complexity (i.e. model order and number of parameters). The model validity functions were found with fixed model order ($n = n_a = n_b$) for different dead times (n_k) and this was repeated for various model orders (n = 1..10). In all cases a dead time of $n_k = 1$ yielded the minimum value of all three validity functions. As expected a dead time of one sample time was appropriate for this process, since the CSTR was assumed to be perfectly mixed. Table 3 shows the loss function for increasing model orders (with $n_k = 1$) identified at various operating points. The general trend of negligible reduction in the LF for higher model orders than 2 was consistently observed. Analogous results were noted for the AIC and the AFPE. Consequently, a 2nd order structure would certainly be selected as the most parsimonious local linear model of this CSTR pH process.

Local Model Sample Time Selection

ARX models were identified using different sample times for fixed model orders, and this was repeated for different model orders and at the seven operating points. Fig. 4 shows the LF for a 2nd order model structure identified at operating point 3 for sample times ranging from 24 to 0.5 seconds. The relative decrease in the LF for faster sample times is representative of results at the other operating points.

If the loss function was the sole criteria for sample time selection, then 6 or 3 seconds would probably be selected as the best compromise. However, for a robust linear model, the position of the dominant pole in the z-plane is critical and should be within the unit circle stability boundary. Table 2 details the magnitude of the dominant poles for the local models identified at different sample times. Poles of magnitude greater than 0.9 would normally be regarded as too close to the stability boundary ($z=1$). The identified models with these poles would be highly sensitive to noise and would therefore be rejected. On this basis a sample time of at least 12 seconds would be preferential.

APPLICATION OF THE LINEAR MODELLING RESULTS TO AN ANN MODEL

The linear systems investigation had proposed a 2nd order model structure and a sample time of no faster than 12 seconds for a robust local model. A corresponding study was now performed to establish the optimum ANN model structure and data sample time.

ANN model structure is used here in reference to the NARX model structure and does not encompass the topology of the hidden layer. For networks with 10 or less inputs 15 hidden layer nodes were used. For networks with more than 10 inputs the size of the hidden layer was increased. No attempt was made to optimise the number of hidden layer nodes as, provided that the MLP ANN incorporated sufficient hidden layer nodes (at least as many as it has inputs), the ANNs mapping accuracy was relatively insensitive to the number of hidden layer nodes. This observation has been reported elsewhere (e.g. Pollard et al.(13)).

Optimum ANN Model Structure

ANN models of different orders (1st to 5th and 10th) with $k=1$ were trained with identical training data sampled at a fixed rate. So as the performance of the ANNs could be fairly compared, they were all tested using the same six unseen test signals, namely, 4 different RAS clock periods (30, 60, 120 and 240 seconds), a 'modified' RAS whose amplitude was greatly reduced thus, testing the model predictions solely in the region of maximum process non-linearity, and steady state data. These disparate test signals were selected to comprehensively test the adequacies of the ANN models. Additionally, the ANNs were all trained for 1000 epochs, with each epoch containing 1000 data vectors.

All of the MSEs were normalised so that the sum of the MSEs for all networks when tested on any one signal was unity, thus weighting the MSEs so that the predictions for each of the six test signals is equally important. These normalised MSEs were then summed for each model order giving a single performance index (PI) for each model order, which is displayed in fig. 5. Although the 10th order model has a marginally smaller PI than the 2nd order one, the latter would be chosen on the parsimony principle. The AIC, shown in fig. 6, and AFPE both reinforced this choice. Recall that they are weighted functions of the LF. For the linear system identification, the number of model parameters (n) is very small compared with the number of data vectors (N), giving a minute n/N ratio which barely weights the LF. However, for ANN models the number of parameters (i.e. weights) is significant, and increases rapidly with model order. For example a 5th order ANN model with 15 hidden layer nodes has 181 parameters, whereas the same order ARX model has only 10 parameters. The AIC is negative because of its logarithmic definition. The best AIC is thus the largest negative number, whereas the best AFPE (always positive) is the smallest number. Hence, both these indices strongly supported the choice of a 2nd order ANN model structure. The model structure suggested in the linear systems investigations therefore appertained directly to the most suitable ANN model structure.

Optimum ANN Model Data Sample Time

Fig. 7 shows the evolution of the training MSE for 2nd order ANNs with identical topology trained with data generated from the same RAS, but sampled at different rates. The networks converged faster and to a lower training MSE for faster sample times. Also, there does not appear to be a limit to this trend, with a sample time of 0.5 seconds exhibiting an improvement over the 1.0 second case.

While there is no reason to expect the optimum sample time for a robust linear model to perform similarly in the non-linear case, one would expect a minimum sample time, below which the ANN model performance deteriorates. One possible explanation for the consistent reduction in training MSE with faster sample time is the corresponding reduction in clock pulses in the input data. Since all training files contained the same number of data vectors, the number of clock pulses per epoch is directly proportional to the sample time. Hence, the input is changing more frequently and the incremental change in output is larger for slower sample times. Reducing the sample time appears to have the effect of making the input/output mapping less complex.

Whilst the training MSE is an important metric of model adequacy, it should not be compared to the test MSE, since during training all input/output data was scaled to between 0.1 and 0.9 to avoid saturation of the MLP sigmoid activation functions. More importantly, the training MSE is only an indication of how well the ANN can predict on the training data. It gives no indication of prediction accuracy on unseen data, which is an essential requirement of a good model.

To see if faster sample times resulted in better models, ANNs were trained using sample times ranging from 0.5 to 12 seconds, and for each sample time 3 different RAS clock periods were used. Then, by following a similar procedure to that described above for the model order analysis, a single normalised loss function was obtained for each data sample time. This performance index is shown in fig. 8 and it demonstrates that there is a lower limit on optimum sample time selection. A sample time of 1.0 second is conclusively the best for the modelling of this process. Reducing the sample time to 0.5 results in a dramatic increase in the MSE performance index.

The ideal sample time for the identified local linear models, perhaps not surprisingly, was not directly applicable to the non-linear modelling sample time. However, if the z-plane pole positions are neglected, the linear model study did indicate that a 24 second sample time was unsuitably slow.

MULTI STEP-AHEAD PREDICTION

Some neural predictive control strategies minimize a cost function over a moving horizon of several steps-ahead (e.g. neural predictive control, see Evans et al. (14)). When predicting more than one step-ahead, any prediction errors in the ANNs output are fed back to its inputs. This can cause an accumulation of errors resulting in large prediction errors. A significant improvement in the multi step-ahead predictions of ANN models can be achieved by encoding the input/output data using the spread encoding technique (Evans et al. (14)). Instead of applying an input value to a single ANN node, it is spread over several nodes using a sliding Gaussian probability distribution function.

A 2nd order ANN was trained using the spread encoding technique, with each input/output value spread over six nodes. Hence, the network had 24 input layer nodes and 6 output layer nodes. The one step-ahead prediction MSEs for this ANN show little improvement over the conventionally trained ANN where each element of a data vector is directly mapped to a single input node (single node encoding). However, the multi step-ahead prediction MSEs are dramatically improved as shown in fig. 9.

The ANN model finally selected was trained with data sample every 1 sec. The network had a 2nd order NARX topology with 15 hidden layer nodes. For one step-ahead prediction, this model had a combined MSE of 0.011, with a maximum prediction error of 0.84, for the six test signals. The validity of this model was reinforced by the correlation tests, where it satisfied four of the five tests, and scarcely exceeded the confidence limits for the fifth test. This is a dramatic improvement over the first ANN model investigated, where the data was sampled every 24 secs, and the network had considerably more parameters having a 5th order NARX configuration with 15 hidden layer nodes. Recall that this ANN had a combined MSE of 0.47 and a maximum prediction error of 3.4 pH units when predicting one step-ahead on the same six test signals. Furthermore, the finally selected model predicting recursively 24 steps-ahead, and hence at 24 sec intervals, still significantly outperformed the original ANN model predicting one step-ahead.

CONCLUSIONS

This paper has looked at the selection of some of the design parameters which are crucially important for the training of a valid ANN model of processes with strong non-linearities. Arbitrary selection of data sample time and network structure can result in an ANN model with unacceptable prediction errors. Useful guidelines concerning data sample time and model structure can be obtained by studying local linear models. The AFPE and AIC penalise over-parameterised networks and are therefore useful indicators of model parsimony. They can be used in conjunction with correlation analysis for model selection and validation.

ACKNOWLEDGEMENT

The authors would like to thank BNFL at Sellafield for their sponsorship of this work.

REFERENCES

1. Bhat, N. and McAvoy, T.J., 1990, Computers Chem. Engng., 14, 573-583

2. Bhat, N., Minderman, P.A., McAvoy, T.J. and Wang, N.S., 1990, IEEE Control Sys. Magazine, April, 24-30

3. Saint Donat, J., Bhat, N. and McAvoy, T.J., 1990, American Control Conf., 2466-2471

4. Saint Donat, J., Bhat, N. and McAvoy, T.J., 1991, Int. J. Control, 54, 1453-1468

5. McAvoy, T.J., Hsu, E. and Lowenthal, S., 1972, Ind. Engng. Chem. Process Des. Dev., 11, 68-70

6. Chen, S., Cowan, C.F., Billings S.A. and Grant, P.M., 1990, Int. J. Control, 51 1215-1228

7. Chen, S., Billings, S.A. and Grant, P.M., 1990, Int. J. Control, 51 1191-1214

8. Chen, S., Billings, S.A. and Grant, P.M., 1992, Int. J. Control, 55 1051-1070

9. Chen, S. and Billings, 1992, Int. J. Control, 56 1319-1346

10. Hernández, E. and Arkun, Y., 1992, Computers Chem. Engng. 16, 227-240

11. Chen, F.C. and Khalil, H.K., 1992, Int. J. Control, 55, 1299-1317

12. Hunt, K.J. and Sbarbaro, D., 1991, IEE Proc.-D, 138, 431-438

13. Pollard, J.F., Broussard, M.R., Garrison, D.B. and San, K.Y., 1992, Computers Chem. Engng, 16, 253-270

14. Evans, J.T., Gomm, J.B., Williams, D., Lisboa, P.J.G. and To, Q.S., 1993, "Application of Neural Networks to Modelling and Control", 74-88, Chapman and Hall, U.K.

15. Rhinehart, R.R., 1990, Intech, July, 40-42

16. Williams, G.L., Rhinehart, R.R. and Riggs, J.B., 1990, Ind. Engng. Chem. Res., 29, 1254-1259

17. Choi, J.Y. and Rhinehart, R.R., 1987, IEEE Am. Control Conf. Proc., 2084-2087

18. Gustaffson, T.K. and Waller, K.V., 1983, Chem. Engng. Sci., 38, 389-398

19. Jacobs, O.L.R., Badran, W.A., Proudfoot, C.G. and White, C., 1987, IEE Proc-D, 134, 196-199

20. Werbos, P., 1974, Ph.D. thesis, Harvard Univ.

21. Cybenko, G., 1989, Maths, Contr. Signal and Sys., 2 , 303-314

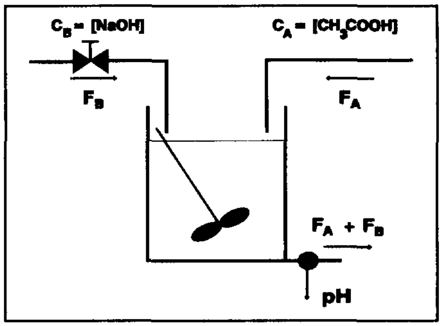

Fig. 1 The CSTR pH Process

Function	*Definition*
LF	$\frac{1}{N}\sum \epsilon^2(t)$
AFPE	$\frac{1 + \frac{n}{N}}{1 - \frac{n}{N}} LF$
AIC	$\ln\{\frac{1}{2}\sum \epsilon^2(t)\} + \frac{2n}{N}$

where $\epsilon(t)$ - *prediction errors*
n - *number of model parameters*
N - *number of data vectors*

TABLE 1. - Model Validity Functions

Δt (secs)	Dominant Pole
24	0.782
12	0.888
6	0.944
3	0.971
1.5	0.986
1.0	0.990
0.5	0.995

TABLE 2. - Local Model Poles

Op. Point	1st Order	2nd Order	3rd Order	4th Order	5th Order	6th Order
1	9.01	8.45	8.40	8.40	8.39	8.38
3	9.93	8.75	8.72	8.70	8.71	8.71
4	8.51	7.82	7.82	7.77	7.77	7.77
6	12.69	10.78	10.75	10.67	10.68	10.66

TABLE 3. - Loss Functions (x 10^{-6}) for ARX models identified at different operating points

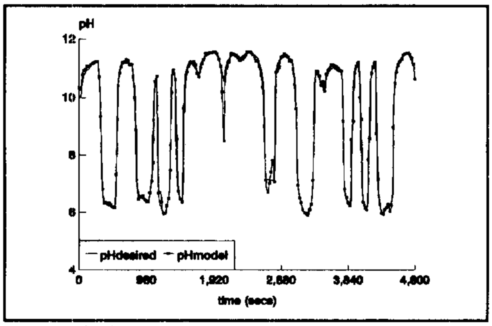

Fig. 2a Typical one step-ahead prediction for initial ANN model

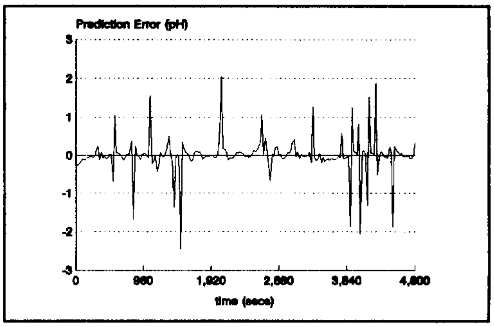

Fig. 2b Prediction errors for Fig. 2a

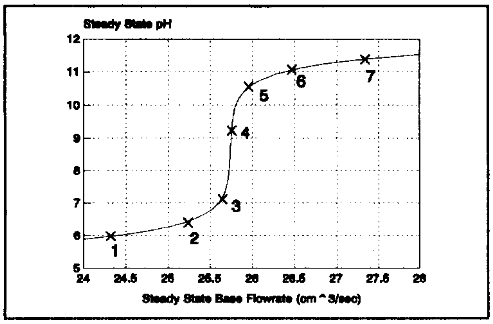

Fig. 3 Local model operating points

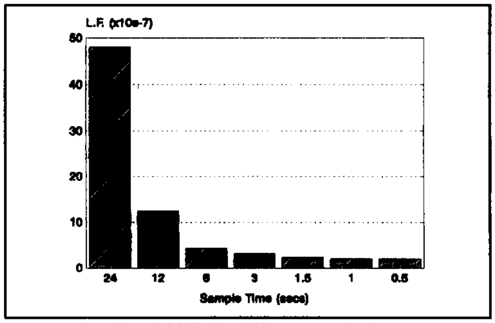

Fig. 4 Local model LFs for different sample times

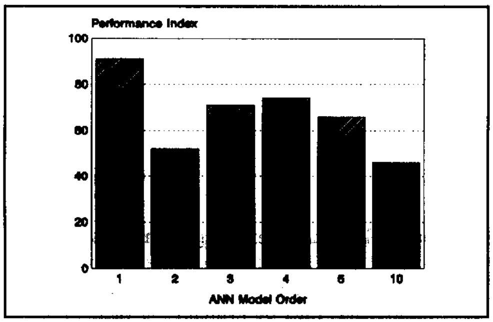

Fig. 5 ANN model performance index for different model orders

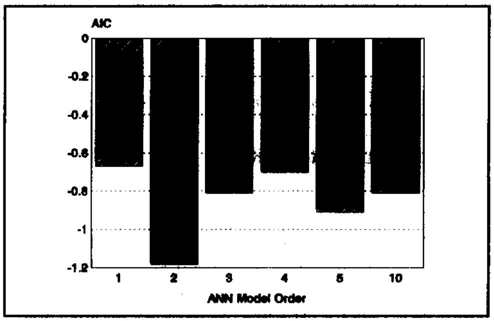

Fig. 6 ANN model AICs for different model orders

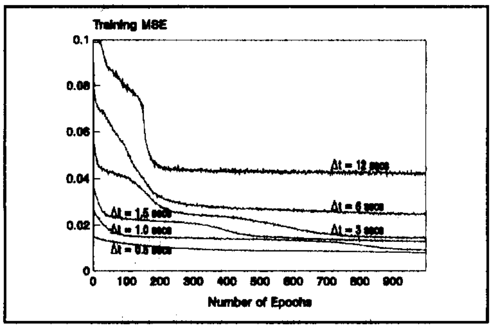

Fig. 7 ANN training MSEs for different sample times

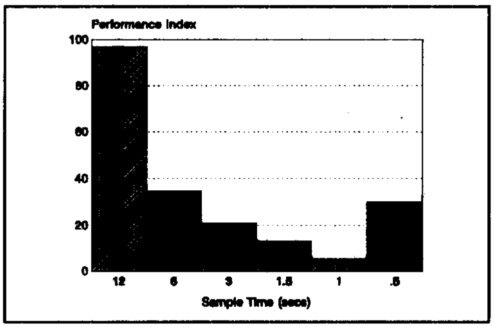

Fig. 8 ANN model performance index for different sample times

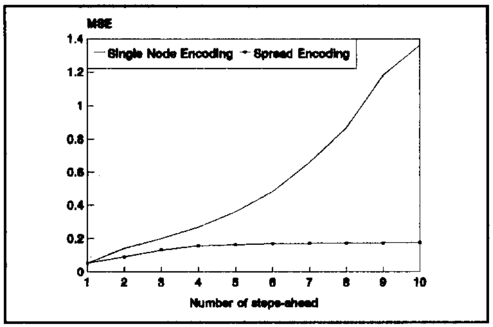

Fig. 9 ANN multi-step ahead prediction MSEs