Author is presently with National Institute of Technology, Rourkela
mondal_kp@yahoo.co.in

Use of Artificial Neural Networks for Prediction of Cell Mass and Ethanol Concentration in Batch Fermentation using *Saccharomyces cerevisiae* Yeast

Dr (Ms) K Pramanik, *Member*

The application of neural network (ANN) for the prediction of fermentation variables in batch fermenter for the production of ethanol from grape waste using Saccharomyces cerevisiae yeast has been discussed in this article. Artificial neural network model, based on feed forward architecture and back propagation as training algorithm, is applied in this study. The Levenberg- Marquardt optimization technique has been used to upgrade the network by minimizing the sum square error (SSE). The performance of the network for predicting cell mass and ethanol concentration is found to be very effective. The best prediction is obtained using a neural network with two hidden layers consisting of 15 and 16 neurons, respectively.

Keywords: Neural network; ANN model; Simulation; *Saccharomyces cerevisiae* yeast

NOTATION

ANN	:	artificial neural network
BP	:	back propagation
NN	:	neural network
SSE	:	sum square error
w_{ij}, w_{jk}	:	connection weights
w_{0j}, w_{0k}	:	bias values
α	:	learning rate

INTRODUCTION

The control of the key fermentation variables is of paramount importance for a successful fermentation process. For this, there is need for on-line monitoring of variables. But to obtain a reliable on-line estimates is very difficult task, especially in fermentation and other biological systems as these are non-linear, complex and time varying in nature and hence calls for suitable simulation and prediction method. Considerable efforts have been made by several researchers to develop a methodology based on various mathematical models so as to use the same as prediction tool[1-4]. The major drawbacks of these models are that they require a large number of experiments and often the models are very complex to explain the experimental observations[5]. In this context, the application of neural networks (ANN) has been considered as a promising tool because of its simplicity for simulating the main state variables in fermentation processes as a function of the progress of the process[6-13]. Other advantages of ANN are that they require less time for development than the traditional mathematical models. They also require minimum number of experiments and have the ability to learn the complex relationships without requiring the knowledge of model structure[14-17].

In this study, efforts have therefore been made to investigate the use of neural network techniques for the prediction of cell mass and ethanol concentration under varying fermentation conditions and to compare the experimental results with those obtained by neural network (NN) simulation.

MATERIALS AND METHODS

The *Saccharomyces cerevisiae* yeast strain extracted from toddy using series dilution technique was used in this study. The strain was maintained on solid nutrient medium containing 1% glucose, 0.5% peptone, 0.3% beef extract, 3% malt extract and 2% agar-agar. After the colonies were observed, the slants were kept in the refrigerator at 4°C until further use. The yeast inoculum was prepared by growing the culture in grape juice medium at pH 4.25 without addition of any nutrient. The temperature and the speed of the agitator was maintained at 30°C and 110 rpm in the incubator, respectively.

Equipment and Experimental Procedure

Fermentation experiments were conducted in a 1l-batch fermenter. The fermenter was equipped with agitator and temperature control systems. The fermenter and the medium were sterilized. The pH was adjusted by the addition of H_2SO_4 prior to inoculation. The fermentation experiments were carried out under anaerobic condition. The production medium was used with different initial sugar concentrations for various batches of experiments. The agitator speed was maintained constant throughout the experiment at 200 rpm.

Analysis of Biomass, Sugar and Ethanol

The concentrations of ethanol and sugar were measured spectrophotometrically. Ethanol was determined by measuring

Dr (Ms) K Pramanik is with the Department of Chemical Engineering, National Institute of Technology, Warangal 506 004.

This paper was received on September 3, 2003. Written discussion on the paper will be received until November 30, 2004.

optical density at 600nm after standard distillation using dichromate solution. The sugar was estimated by DNS method. To estimate the biomass concentration, the cells were separated by centrifugation at 14 000 rpm for 5-min and washed with distilled water. The cells were then dried at 80°C until constant weight was achieved and the cell concentration was measured based on dry weight of cells.

Data Generation

Experimental data were obtained by conducting a series of batch experiments in fermenter under varying conditions of pH (4, 4.5, 4.75, 5.0), temperature (30°C and 35°C) and initial sugar concentration (5%, 10%, 15% and 20% w/v). For each set of parameters selected, data were collected as a function of time. The experimental data (Number 70) were then smoothened using the polynomial regression equations of order 2 to 5 in MS Excel Pentium III Computer to generate a large number of data sets.

NEURAL NETWORKS

The foundation of a neural network is the neuron which is also called as node or neurode. Each neuron is a processing element which performs a weighted sum of all inputs variables that feed it. Depending on the value of weighted sum of the variables, the neuron gives a signal to the neurons in the adjacent layer through a non-linear transfer function (sigmoid function in this case). The choice of the architecture of the network depends on the task to be performed and the architecture of the model is specified by the node characteristics, network topology and learning algorithm. In standard architecture, neurons are grouped into different layers like-input, output and hidden layers. Generally, for modelling of physical systems, three-layered, feed-forward network is normally used. But in the present study, a four-layered feed-forward network as shown in Figure 1 has been used which consists of a layer of input neurons, two layers of hidden neurons and one layer of output neurons. The ANN configuration is represented as 5:15:16:1, that is, the input layer consists of five inputs, each hidden layer consists of fifteen and sixteen neurons, respectively and the output layer consists of one output.

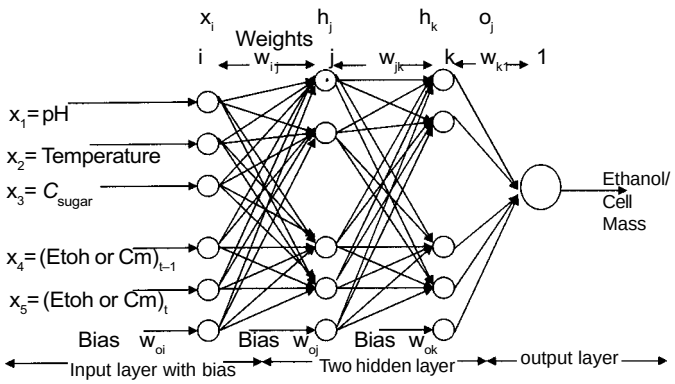

Figure 1 Four-layered feed-forward neural network

Back Propagation (BP) Training Algorithm

The back propagation of error algorithm, based on multi-layered feed-forward net and considered to be the most versatile algorithm[17], was used to train the network for predicting correct outputs those obtained from experiments and generated one. The BP algorithm adjusts the network weights and bias values to minimize the square sum of the difference between the given output (X) and output values calculated by the net (X') using gradient decent method as follows

$$SSE = 1/2\ N\sum (X - X')^2 \qquad (1)$$

where N is the number of experimental data points used for the training.

Levenburg-Marquardt technique is used to improve the learning rate and stability of the BP algorithm for searching minimum error. The algorithm and data processing were performed in MS Excel Pentium III Computer using MATLAB package.

Training and Testing Procedure

The entire data were divided into two sets. The larger set consisting of 405 data was used for training and the rest 97data were reserved for use in testing and validation of the ANN predicted output values. As the ANN inputs should be in the range of 0 to1, so all inputs are normalized by using the formula as follows

$$V_{\text{normalized}} = (V - V_{\min})/(V_{\max} - V_{\min}) \qquad (2)$$

The following steps were then followed for training the network

Step 0 : Initialize weights (set random values between 0 and 1).

Step 1 : While stopping condition is false, do steps 2-9.

Step 2 : For each training pair of set, do steps 3-8.

Step 3 : Each input unit (x_i, i = 1, 2, ... , 5) receives input signals x_i and sends this signal to all the nodes in the next hidden layer.

Step 4(a) : Each hidden unit (h_j, j = 1, 2, ... , 15) sums its weighted input signals and bias is added to this weighted sum to compute its output signal as

$$O_{oj} = f\left(w_{0j} + \sum w_{ij}\, x_i\right), \quad \text{for } i = 1, 2, \ldots, 5$$

and sends this signal to all the units in the following layer (h_k, hidden layer).

Step 4(b) : Each hidden unit (h_k, k = 1, 2, ... , 16) sums its weighted input signals and bias is added to this weighted sum to compute its output signal as

$$O_{ok} = f\left(w_{0j} + \sum w_{jk}\, O_{oj}\right), \quad \text{for } j = 1, 2, \ldots, 15$$

and sends this signal to all the units in the following layer (O_1, output layer).

Step 5 : The output unit (O_l, l = 1) in the output layer sums its weighted input signals and applies its activation function to compute its output as

$$O_{ol} = f\left(w_{0i} + \sum w_{kl}\, O_{ok}\right), \quad \text{for } k = 1, 2, \ldots, 16$$

Step 6 : The BP of errors starts at output layer 'O_l' to compute its error information term as

$$\delta_l = (t_l - O_{ol})\, f\left(w_{0l} + \sum w_{kl}\, O_{ok}\right), \quad (t_l = \text{target value})$$

It calculates weights correction and bias correction terms as

$$\Delta w_{kl} = \partial\, \delta_l\, O_{ok}, \text{ and }, \Delta w_{0l} = \alpha\, \delta_l, \text{ respectively.}$$

where δ_l is the error sent to nodes in the previous layer.

Step 7(a): Each hidden unit (h_k, k = 1, 2, ... , 16) sums its delta inputs from units in the next layer and calculate its error information term as

$$\delta_{ok} = \delta_k\, f^1\left(O_{ok}\right) \text{ where } \delta_k = \sum \delta_l\, w_{kl} \text{ (for l = 1)}$$

It calculates its weights correction and bias correction term as

$$\Delta w_{jk} = \partial\, \delta_k\, O_k \text{ and } \Delta w_{ok} = \alpha\, \delta_k, \text{ respectively.}$$

Step 7(b) : As step 7(a), each hidden unit (h_j, j = 1, 2, ... ,15) sums its delta inputs from units in the next layer and calculate its error information term as

$$\delta_{oj} = \delta_j\, f^1\left(O_{oj}\right) \text{ where } \delta_j = \sum \delta_k\, w_{jk} \text{ for } k = 1, 2, \ldots, 16.$$

It calculates its weights correction term and bias correction term as

$$\Delta w_{jk} = \partial\, \delta_k\, O_j \text{ and } \Delta w_{0j} = \alpha\, \delta_j$$

Step 8 : Each output node (O_l, l = 1) updates its weights and bias as

$$w_{jk}\,(\text{new}) = w_{kl}\,(\text{old}) + \Delta w_{kl} \text{ for } k = 1, 2, \ldots, 16.$$

$$w_{0k}\,(\text{new}) = w_{0l}\,(\text{old}) + \Delta w_{0l}$$

Each hidden node (h_k, k = 1, 2, ... , 16) updates its weights and bias as

$$w_{ij}\,(\text{new}) = w_{jk}\,(\text{old}) + \Delta w_{jk} \text{ for } j = 1, 2, \ldots, 15.$$

$$w_{0j}\,(\text{new}) = w_{0k}\,(\text{old}) + \Delta w_{0k}$$

Step 9 : Test of stopping condition.

As noted as earlier, the Levenberg-Marquardt variation of non-linear least squares optimization technique is used to upgrade the back propagation algorithm. It involves the following additional computations in the step 6 to calculate its weight correction term as

$$\Delta w_{jk} = \left(H + \lambda \times I\right)^{-1}_{kL}\, \partial \delta_k\, Z_j$$

where k is the row number; L, the number of neurons in that layer and I is the identity matrix

$$(H)_{kL} = \left[J'\, J\right]_{kL} \text{ and } J_{kL} = \partial(t_k - y_L)\,\partial w, \text{ respectively}$$

where, I is the identity matrix of the function; H, the Hesian matrix of the function; J, the Jacobian matrix of the function; and λ is step length, (the parameter for Levenberg- Marquardt method).

After training, the network is tested by introducing the testing input data sets. Experimental data are then compared with simulated data. If the network predictions are in close agreement with the experimental data, then network topology is accepted. Else the training process is repeated with new parameters.

RESULTS AND DISCUSSION

In this investigation, the ANN simulations are performed two times, one for predicting ethanol concentration and other for cell mass concentration. In the first case, the neural network employed has five input nodes corresponding to the five fermenter variables, namely, pH, temperature, sugar concentration, ethanol concentration at time (t – 1), and ethanol concentration at time t and one output node corresponding to the ethanol concentration at time (t + 1). Whereas for predicting cell mass concentration, pH, temperature, sugar concentration, biomass concentration at time t and (t – 1) as the input nodes inputs and cell mass concentration at time (t + 1) as the output nodes are used for simulation purpose. Out of 502 experimental data, first 405 data sets are used to train the network and the last 97 data are used for testing and validation of the NN model. During training the network, SSE has been kept at 0.001 and the frequency of progress displays (in epochs) is set at 50 with maximum epochs of 1000 to train the network.

Neural network models corresponding to different numbers of hidden layers (1 and 2 in each case) and number of neurons in the hidden layers are tried to find the network architecture that provides the least error. A neuron architecture with 2 hidden layers containing 15 neurons and 16 neurons is found to be optimum in both cases of ethanol and cell mass predictions. The ethanol results produced in training and testing to assess the ethanol concentration at any desired time interval-using ANN are shown in Figure 2–Figure 5. There is very little difference between the ethanol concentrations predicted by ANN model and training and testing data, as observed in Figure 2–Figure 4. A BP neural network is found to be very efficient in predicting ethanol concentration in the range of data contained in the learning set, that is, 405 data sets. The simulation results show that introduction of two hidden layers improve the forecasting performance of ANN compare to the use of single hidden layer. The average SSE is observed to be reached the

error goal of 0.001, when two hidden layers are used instead of 0.05 reached with only one hidden layer containing 16 neurons. It is then decided to evaluate the network model using 97 testing data sets. The results show that the predicted ethanol concentrations of ANN results obtained are as close as with training and testing data given to the network. Figure 5 shows the percentage relative error between ANN predicted values and the training/testing data sets. As it is observed that the percentage relative error lies in the range — 1.649 to 1.9. The NN explored in the first case is employed as a module in the second case for predicting cell mass concentration.

The results of training and testing conducted on 502 data sets for cell mass prediction are shown in Figure 6–Figure 9.

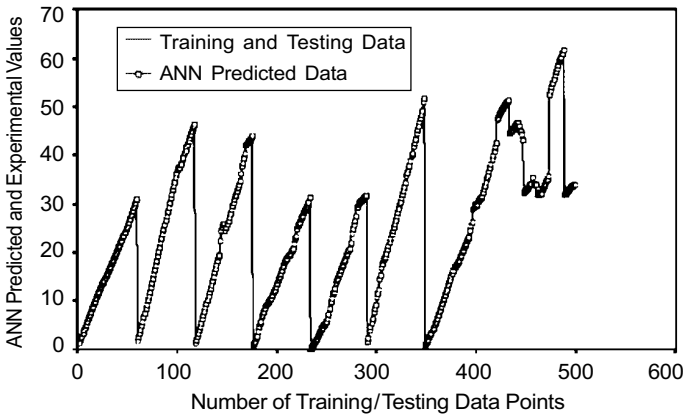

Figure 2 Comparison between outputs of ANN predicted and training and testing data of ethanol concentration (g/l) as a function of time

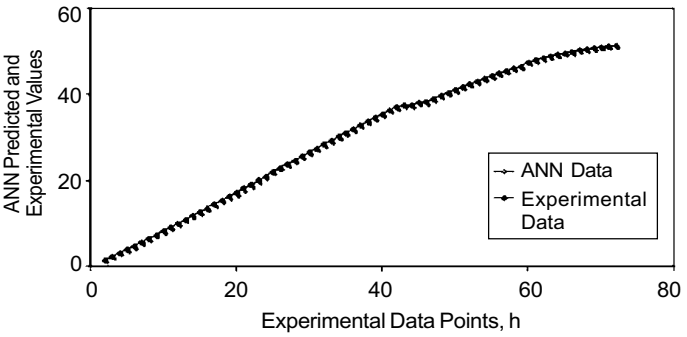

Figure 3 Comparison between ANN simulated and experimental output data of ethanol concentration (g/l) as a function of time

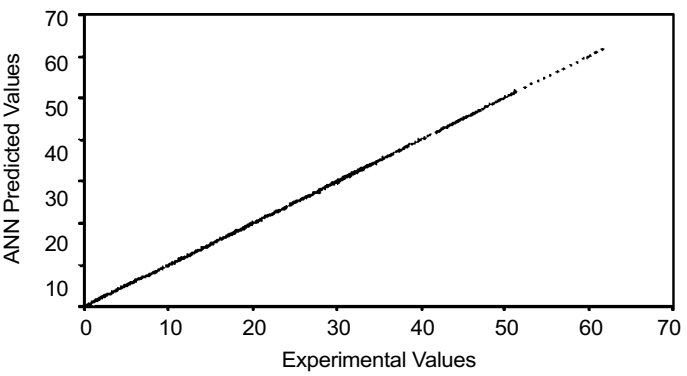

Figure 4 Comparison between ANN simulated and experimental data of ethanol concentration (g/l)

Figure 6, compares the ANN simulated value of cell mass concentration with the training and testing data sets, whereas Figure 7 and Figure 8 show the comparison between ANN simulated output data and cell mass concentration data obtained by experiments. The percentage relative error at each sampling interval as shown in Figure 9 lies in the range of –0.9926 to 1.9. The maximum error as 1.9 is observed at 118 data

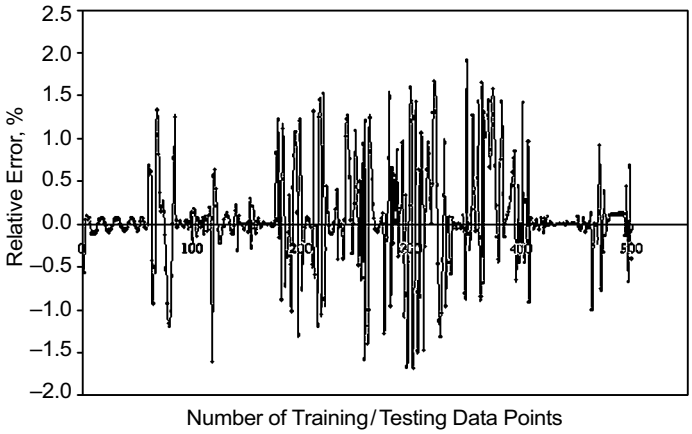

Figure 5 Percentage relative error between ANN simulated and training and testing output data of ethanol concentration (g/l) as a function of time

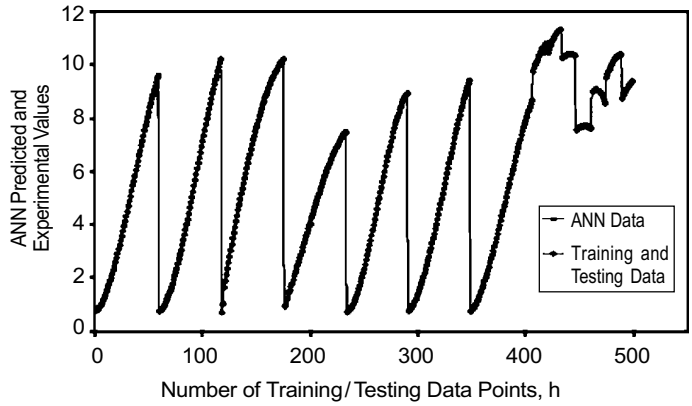

Figure 6 Comparison between outputs of ANN predicted and training and testing data of cell mass concentration (g/l) as a function of time

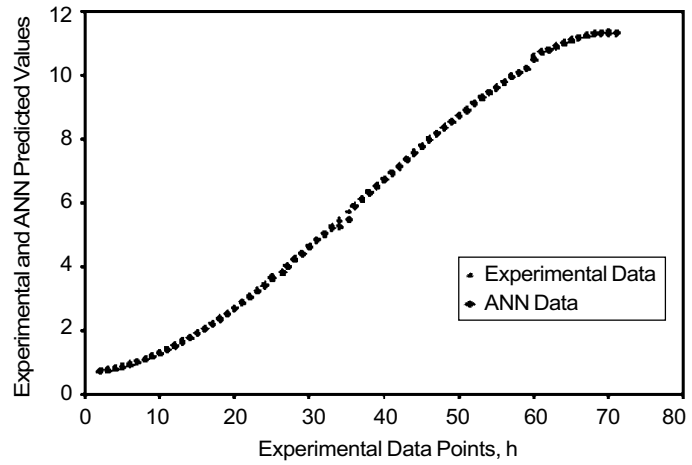

Figure 7 Comparison between ANN simulated and experimental cell mass concentration (g/l) as a function of time

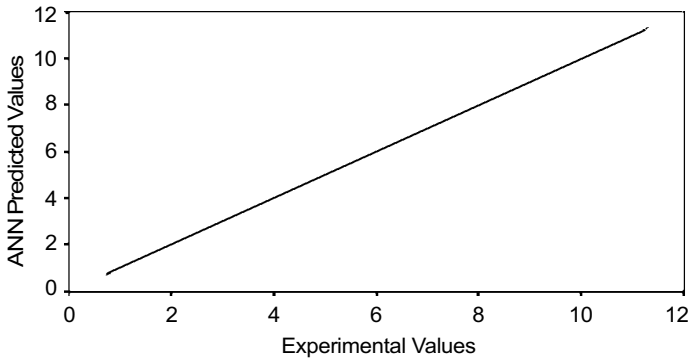

Figure 8 Comparison between ANN simulated and experimental data of cell mass concentration (g/l)

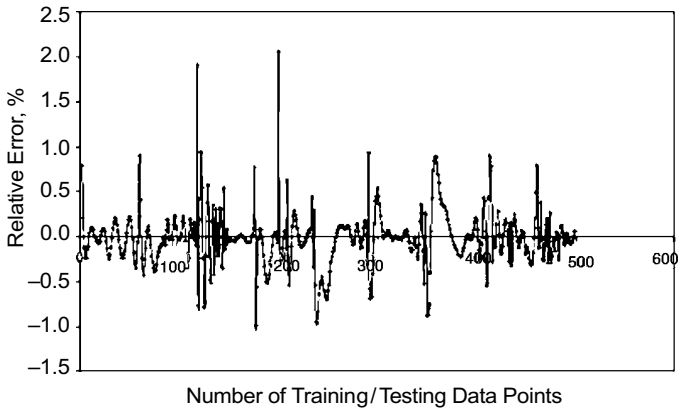

Figure 9 Percentage relative error between ANN simulated and training and testing output data of cell mass concentration as a function of time

point. The results show that the performance index of cell mass concentration obtained from ANN simulation is close to the training and testing data sets.

CONCLUSION

In the present investigation, neural networks have been designed and demonstrated to predict the state of batch fermentations with grape juice extracted from grape waste by taking into account the effect of pH, temperature and initial sugar concentration as a function of time. A simple propagation network using the Levenberg-Marquardt for training the network is found to be very effective to generalize and predict the cell mass and ethanol concentration during batch fermentation. The configuration of the back propagation neural network that gives the best prediction is the one with two hidden layers consisting of 15 neurons and 16 neurons in each layer. ANN predicted results are very close to the experimental values. The average SSE is observed to be reached the error goal of 0.001 and the maximum percentage relative errors are found to be 1.9077 and 1.9 for ethanol and cell mass, respectively. Therefore, the prediction capability of neural networks can be utilized as a promising technique for modelling, estimating and predicting bio-processes which are non-linear in nature and whose dynamics are poorly known.

REFERENCES

1. P M Schlosser and J E Bailey. 'An Integrated Modelling — Experimental Strategy for the Analysis of Matabolic Pathways.' *Mathematical Bio-sciences,* vol 65, 1990, p 87.

2. J Albiol, J Robuste, C Casas and M Poch. 'Biomass Estimation in Plant Cell Cultures using Extended Kalman Filter.' *Bio-technology Progreses,* vol 9, 1993, p 174.

3. F Valero, J Lafuente, M Poch and C Sola. 'Biomass Estimation using On-line Glucose Monitoring by Flow Injection Analysis.' *Application of Bio-chemistry and Bio-technology,* vol 24/25, 1990, p 591.

4. G Stephanopoulos and K San. 'Studies on On-line Bioreactor Identification I: Theory.' *Bio-technology and Bio-engineering,* vol 26, 1984, p 1176.

5. J Albiol, C Campmajo, C Casas and M Poch. 'Biomass Estimation in Plant Cell Cultures: A Neural Network Approach.' *Bio-technology Progress,* vol 11, 1995, p 88.

6. M Syu and G T Tsao. 'Neural Network Modelling of Batch Growth Rate Pattern.' *Bio-technology and Bio-engineering,* vol 42, 1993, p 376.

7. P Linko and Y H Zhu. 'Neural Network Modelling for Real Time Variable Estimation and Prediction in the Control of Glucoamylase Fermentation.' *Process Bio-chemistry,* vol 27, 1992, p 275.

8. V Thibault, A Van Breusegem and A Cheruy. 'On-line Prediction of Fermentation Variables using Neural Networks.' *Bio-technology and Bio-engineering,* vol 36, 1990, p 1041.

9. J Glassy, G A Montague, A C Ward and V B Kara. 'Artificial Neural Networks based Experimental Design Procedure for Enhancing Fermentation Development.' *Bio-technology and Bio-engineering*, vol 44, 1994, p 397.

10. C Campmajo, M Poch, J Robuste, F Valero and J Lafuente. 'Evaluation of Artificial Neural Networks for Modelling Enzymatic Glucose Determination by Flow Injection Analysis.' *Analysis,* vol 20, 1992, p 127.

11. A Tholudur and W F Ramirez. 'Optimization of Fed-batch Bio-reactors using Neural Network Parameter Function Models.' *Bio-technology Progress,* vol 12, 1996, p 302.

12. P Mhaskar, M A Hjorts and M A Henson. 'Cell Population Modelling and Parameter Estimation of Continuous Cultures of *Saccharomyces cerevisiae*.' *Bio-technology Progress,* vol 18, 2002, p 1010.

13. M J Willis, G A Montague, C Di Massimo, M T Tham and A J Morris. 'Artificial Neural Networks in Process Estimation and Control.' *Automatica,* vol 28, no 6, 1992, p 1097.

14. C Shene, B Andrews and A J Asenjo. 'Fed-batch Optimizations of *Bacillus subtilis* Fed-batch ToC46 (pPFF1) for the Synthesis of a Recombinant β-1, 3-glucanase: Experimental Study and Modelling.' *Enzyme and Microbial Technology,* vol 24, 1999, p 247.

15. C Shene, B Andrews and A J Asenjo. 'Optimization of *Bacillus subtilis* for the Fed-batch Fermentations for the Maximization of the Synthesis of a Recombinant β-1, 3-glucanase.' *Computer Applications in Bio-technology,* Japan, vol CAB 7, 1998, p 219.

16. T Hill, L Marquez, M Oconnor, and W Remus. 'Artificial Neural Network Models for Forecasting and Decision Making.' *International Journal of Forecasting,* vol 10, 1994, p 5.

17. J Savkovic-Stevenovic. 'Neural Networks for Process Analysis and Optimization: Modelling and Applications.' *Computers in Chemical Engineering,* vol 18, no 11/12, 1994, p 1149.

18. H El Mounayni, H Kishawy and V Tandon. 'Optimized CNC End Milling: a Practical Approach.' *International Journal of Computer-Integrated Manufacturing,* vol 15, no 5, 2002, p 453.