Brazilian Journal
of Chemical
Engineering

ISSN 0104-6632
Printed in Brazil
www.abeq.org.br/bjche

Vol. 25, No. 03, pp. 515 - 522, July - September, 2008

MODELING OF AN INDUSTRIAL DRYING PROCESS BY ARTIFICIAL NEURAL NETWORKS

E. Assidjo, B. Yao*, K. Kisselmina and D. Amané

Institut National Polytechnique Houphouët-Boigny, Département Génie Chimique et Agroalimentaire,
Laboratoire de Procédés Industriels de Synthèse et de l'Environnement, Phone : +(225) 07 80 05 19,
Fax +(225) 30 64 04 06, BP 1093 Yamoussoukro, Côte d'Ivoire.
E-mail: beyao@yahoo.fr

(Received: March 3, 2007 ; Accepted: April 23, 2008)

Abstract - A suitable method is needed to solve the nonquality problem in the grated coconut industry due to the poor control of product humidity during the process. In this study the possibility of using an artificial neural network (ANN), precisely a Multilayer Perceptron, for modeling the drying step of the production of grated coconut process is highlighted. Drying must confer to the product a final moisture of 3%. Unfortunately, under industrial conditions, this moisture varies from 1.9 to 4.8 %. In order to control this parameter and consequently reduce the proportion of the product that does not meet the humidity specification, a 9-4-1 neural network architecture was established using data gathered from an industrial plant. This Multilayer Perceptron can satisfactorily model the process with less bias, ranging from -0.35 to 0.34%, and can reduce the rate of rejected products from 92% to 3% during the first cycle of drying.
Keywords: Neural network; Grated coconut drying; Modeling.

INTRODUCTION

Grated coconut is a product obtained from the transformation of coconuts through a long and complex process. In summary, the coconuts are shelled before their skin is peeled off. Then they are crushed. The crushed product is pasteurized and dried to give fine coconut particles with a size of about 3 mm (Ahoulé, 2004).

Drying must confer to particles a final moisture of 3 percent (Jend and Das, 2007). However, under industrial conditions, it is very difficult to maintain this final moisture, which varies from 1.9 to 4.8 % (Ahoulé, 2004). Indeed, a percentage of moisture higher than this value (3%) jeopardizes preservation of the product because of the probable proliferation of microorganisms. Therefore redrying is necessary to overcome this problem. On the other hand, when the moisture is less than 3%, the weight of the grated coconut is reduced causing economic waste. Indeed, in the case studied, the rate of product rejection due to the failure to achieve stipulated humidity (3%) after the first drying cycle is greater than 90%. So a second draying cycle and sometimes a third is necessary for the same sample, involving supplementary costs and a decrease in productivity.

The objective of this study is to mitigate this insufficiency by modelling the manufacturing process in order to reduce the nonquality as much as possible during the first drying cycle.

Generally, modelling takes into account mass and energy conservation to obtain equations for the process studied (Barreto, 1997; Jarvensivu and Seaworth, 1998). Nevertheless, it is sometimes impossible to establish these equations due to the complexity of the phenomenon. Therefore, "black-box" models based on input-output patterns like

*To whom correspondence should be addressed

artificial neural networks are useful (Haykin, 1994; Fujiwara, 1995; Desheng et al., 2006; Tai-Yue and Shih-Chien, 2006).

Artificial neural networks are mathematical tools whose functioning is inspired by that of the human brain (Grossberg, 1982; Kohonen, 1987). They are a promising tool for simulating variables of processes because of their simplicity. They have the ability to learn the complex relationships without a priori knowledge of model structure (Shene et al., 1998, 1999; Hill et al., 1994; Savkovic-Stevenovic, 1994).

Like their biological counterpart, the neurons in layers receive, treat (by weighted summation) and transfer information generally via a nonlinear function. They have been intensively used in different activity domains: banking, insurance, defence, industry, etc (Desheng et al., 2006; Tai-Yue and Shih-Chien, 2006; Pramanik, 2004; Shene et al., 1998, 1999; Tseng-Chung and Li-Chiu, 2005). In industry, their applications are innumerable and exponentially increasing (Baughman and Liu, 1995); those specifically related to the drying process are mentioned by Jinescu and Lavric (1994, 1995). Nevertheless, no work has been reported concerning application of artificial neural networks to the modelling of the industrial process of drying grated coconut. We, therefore, explore their flexibility to estimate the final moisture of this product.

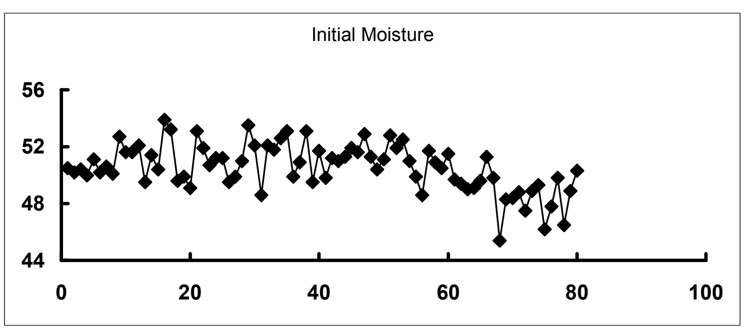
MATERIALS AND METHOD

Data Set

The data used in this study were provided by an Ivorian company of coconut processing. Its principal product is grated coconut in different thicknesses (medium, fine). This product is obtained via a process which can be divided into unitary operations. The most important of them is the drying step operated in a Proctor dryer (Proctor Inc., Glasgow, Scotland, UK) composed of three compartments (A, B and C). This dryer is a parallelepiped fluidised bed dryer with a size of 20 x 4 x 3 m. Its different compartments contain one or two fans each, which blow hot air onto a conveyor where the products are circulated to dry. The compartments are subdivided into seven subcompartments (i.e., A1, A2, A3, B1, B2, C1 and C2). Thus, the process depends on many variables. But, in this study, the variables of concern were initial moisture of the crushed and pasteurized grated coconut, the seven Proctor dryer temperatures and the final product temperature. The response of interest was the final moisture of the dried grated coconut.

Table 1: A summary of the data set

	IM (%)	TA1 (°C)	TA2 (°C)	TA3 (°C)	TB1 (°C)	TB2 (°C)	TC1 (°C)	TC2 (°C)	FT (°C)	FM (%)
Min	45.4	115.6	104.7	85.0	81.2	63.8	55.2	67.6	37.0	1.0
Max	53.9	130.1	130.6	118.4	120.7	121.2	100.6	96.3	45.8	4.8
Mean	50.5	124.8	122.8	107.7	103.6	89.2	83.0	82.9	42.0	2.1
SD	1.7	2.58	3.49	5.11	5.71	10.6	8.4	6.81	2.1	0.8
RSD (%)	3.3	2.1	2.8	4.8	5.5	11.9	10.2	8.2	5.1	38.3



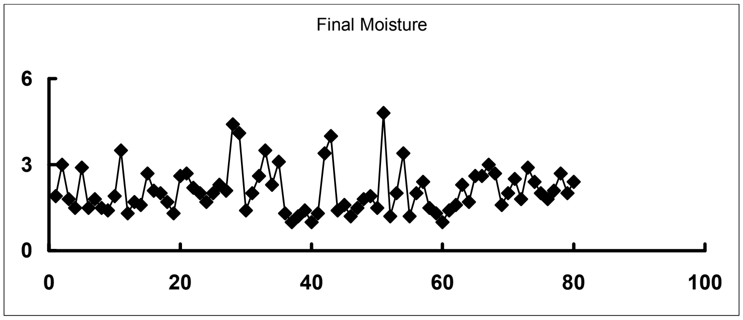


Figure 1: Synthetic presentation of records for initial and final moistures

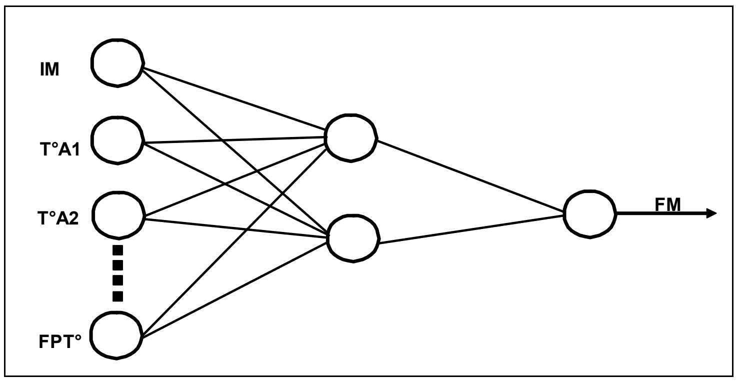


Figure 2: A Multilayer Perceptron

Data were randomly collected from this essential step in the process of drying grated coconut, resulting in 159 records. In table 1 the minimum, maximum, mean, standard deviation (SD) and the relative standard deviation (RSD) values of the variables are presented. Figure 1 is a synthetic presentation of records for initial and final moistures before the modeling.

Neural Network

The neural network is constituted of simple elements, each calculating a weighted sum of all input variables that feed it. Different types of artificial neural network are available (e.g. support vector machine (SVM), self-organisation map (SOM), multilayer perceptron (MLP)) (Haykin, 1994; Fujiwara, 1995; Karim and Rivera, 1992). The third (MLP) is the most widely used (Fujiwara, 1995). Generally, the neurons are grouped into three different types of layers, as shown in Figure 2:

- Input layer: whose number of nodes depends on input variables
- Output layer: whose number of nodes is equal to the number of predicted variables
- Hidden layer: situated between the first two layers.

Finding neural network models consists of computing the appropriate weight and biases that minimize the discrepancy between observed and simulated data.

For training the network, the steps were as follows:

1. data were divided into three subsets: training (80 data), validation (40 data) and testing (39 data) subsets;
2. all inputs were normalized using the formula

$$xn_i = \frac{2(x_i - x_{min})}{(x_{max} - x_{min})} - 1 \qquad (1)$$

where xn_i is the normalized data ranging between -1 and 1, x_i is the initial data and x_{min} and x_{max} are the minimum and maximum values of the data set;

3. the weights were initialized setting random values;
4. each input neuron (*i*) received input patterns and sent this signal to all nodes in the next layer (hidden layer);

5. each hidden unit (*j*) summed its weighted input signals and added the bias before computing its output as follows:

$$y_j = \tanh(\sum w_{ij}x_i + b_j) \quad (2)$$

The result obtained was sent to the nodes of the output layer;
6. the output units, just as the former (hidden units), calculated a weighted sum of their input signals and applied an activation function;
7. the output layer unit computed the error term and calculated weights and bias correction terms;
8. the error terms were propagated in inverse direction to the signal and different weights were adjusted.

The choice of neural architecture is related to the task to be performed and the model architecture is specified by neuron characteristics, network topology and training algorithm (Syu and Tsao, 1993). The choice of a good network topology is not a straightforward task. There are no hard rules or theorems for finding an optimal topology for a given set of input- output data. However, an appropriate topology may be found by performing network pruning or network growing (Kadhir et al., 2000). Starting with a sufficiently big topology, the neural network is pruned by eliminating the links containing insignificant weights using a weight elimination method, for example the optimal brain damage (OBD) method developed by Le Cun et al. (1989). Alternatively, starting with a small architecture, the network is grown until reaching a size that gives a good prediction model. Another approach is determination of Schwarz's Bayesian information criterion (BIC) (Schwarz, 1978) obtained as follows:

$$BIC = \log\left(\frac{V}{n}\right) + p\frac{\log(n)}{n} \quad (3)$$

where V is the sum of squares approximation errors, n the number of training patterns and p the total number of network weights.

Degree of freedom can also enable to identify the topology best adapted for the phenomenon studied (Khamis et al., 2006). It is, by definition, the number of observations minus the number of parameters that are free to vary. If N is the number of observations and k that of estimated parameters, the degree of freedom (df) is as follows:

$$df = N - k \quad (4)$$

In the case of a multilayer perceptron with one output, the estimated parameters are concerned with not only the connection weights and bias of the output layer, but also the connection weights and bias that interconnect the hidden layers. If k is assumed to be the number of these estimated parameters, then it is determined using the following equation, for the architecture of a hidden layer:

$$k = n_j(n_i + 2) + 1 \quad (5)$$

where n_j and n_i are the number of nodes in the hidden and the input layers, respectively.

A degree of freedom must be a positive value; this fact imposes an upper limit on the size of the network:

$$n_j(\max) = \frac{N-1}{n_i + 2} \quad (6)$$

This methodology using degree of freedom is rarely utilized because of the fact that the artificial neural network was formerly developed for nonparametric phenomena modelling (Grossberg, 1982; Ripley, 1994).

In this work, the neural networks were performed using Matlab neural network toolbox Release 14. The fully connected feedforward was considered. First, a small network topology was chosen (e.g. 9-1-1). Second, the number of hidden nodes was grown and an arbitrary number, 15, was chosen. The training algorithm in this work was the versatile Leverberg-Marquardt technique used to improve the learning rate and the stability of the back-propagation algorithm in searching for minimum error. It is also the fastest method for training a feed-forward neural network.

The back-propagation algorithm is a gradient descent method that adjusts the network weights and bias values to minimize the square sum of the difference between the given output (y_o) and the one calculated (y_c) by the net (Werbos, 1990), which can be obtained by

$$e = \frac{1}{2}\sum (y_c - y_o)^2 \quad (7)$$

The training subset is used for computing and updating the network weights and biases. The error in the validation set is monitored during the training process. The validation error will normally decrease during the initial phase of training, as does the training set error. However, when the network begins to overfit the data, the error in the validation set will typically begin to increase. When the validation error increases for a specified number of iterations, the training is stopped and the weights and biases of the minimum validation error are returned. This method (early stopping) enables improvement of the generalization (Ungar et al., 1996). The test set error is not used during training, but it is used to compare different models.

The transfer function of hidden layer nodes was the *Tanh* function and the linear function for the output layer.

RESULTS AND DISCUSSION

As neural network weights are initialized before their modification during the training process in order to obtain the smallest possible predicting error, simulations were performed 1000 times. The best result was then recorded. The objective of this work was to determine the black box model for the process of drying grated coconut. During training, the frequency of progress display (in epochs) was set at 50 with a maximum of 500 to train the networks.

Firstly, the number of the hidden layer was set at 1. The number of neurons in this hidden layer was varied to find the architecture that provides the least error. During this step, the model obtained was simulated to find the calculated responses. These values were compared to the observed ones. The regression line obtained from this comparison was characterized by its slope (m), its intercept (b) and its correlation coefficient (r). The different results of the comparison are presented in Table 2. Analyzing this table, it appears that when the number of nodes in the hidden layer increases, slope, intercept and correlation coefficient values vary for the training subset as well as the test one. But are these variations significant enough to establish discrepancies between these values?

In order to answer to this question, an ANOVA test was performed. The Fisher coefficient obtained (0.72) is lower than its limiting value (2.46) at the 0.05 level, consequently showing that the number of neurons in the hidden layer significantly affected the behaviour of the neural network.

Table 2: Characteristics of the regression plot of observed to calculated outputs for training and test sets

	Training			Test		
Nodes in hidden layer	**m**	**b**	**r**	**m**	**b**	**r**
1	0.489	-0.217	0.774	0.738	-0.026	0.699
2	0.656	-0.138	0.814	0.866	-0.017	0.750
3	0.736	-0.101	0.857	0.764	-0.035	0.695
4	0.745	-0.082	**0.865**	0.783	0.032	**0.783**
5	0.815	-0.065	0.909	0.575	-0.184	0.635
6	0.836	-0.070	0.919	0.690	-0.131	0.649
7	0.938	0.002	0.980	0.847	-0.102	0.598
8	0.883	-0.035	0.954	0.643	-0.195	0.530
9	0.993	0.004	**1.000**	1.154	0.119	**0.725**
10	0.982	-0.008	0.996	0.291	-0.456	0.228
11	0.971	-0.015	0.990	0.737	-0.086	0.490
12	0.993	-0.005	1.000	0.319	-0.150	0.294
13	1.000	0.000	1.000	0.649	-0.280	0.405
14	1.000	0.000	1.000	0.840	0.031	0.461
15	1.000	0.000	1.000	0.585	-0.094	0.300

The correlation coefficient r varied from 0.774 to 1.000 for the training subset and from 0.229 to 0.783 for the test one. The higher its absolute value (i.e. 1.000), the better is the adequacy between predicted values and observed ones. Thus, it can be observed that when the number of nodes in the hidden layer increased, the correlation coefficient increased too up to 1.000 for the training subset. The models are therefore improved. Moreover, when considering the test subset, it appears that the correlation coefficients did not evolve any more in the same direction than the increase in number of neurons in the hidden layer. Its value is higher for an architecture with four nodes (r = 0.783) or nine nodes (r = 0.723) in the hidden layer. But, as the best network is a compromise between the results obtained during training and those with the generalization (test), the networks that characterize the data of the process of drying grated coconut with the least error are the 9-4-1 and 9-9-1 networks. While the former gives a positive degree of freedom (df = 35), the latter has a negative calculated degree of freedom (df = -20). Due to the fact that the degree of freedom must be positive, the maximum acceptable number of neurons was calculated using equation 6. The number obtained, 7, enables the elimination of the 9-9-1 network. In addition, it can be observed (Table 3) that a network with four hidden layer neurons has the smallest generalization error and Schwarz's Bayesian information criterion (BIC).

Table 4 clearly shows that the addition of a second hidden layer does not improve the performance of the network.

The model of the 9-4-1 network was diagnosed using error analysis. The scattered error plot (Figure 3) shows that errors are uniformly and randomly distributed around the mean value (0) in the range of -0.35 to 0.34 %. As the errors have homogeneous variance and there is no observed systematic trend in residual values, the model obtained from the 9-4-1 architecture is adequate for modelling the patterns.

Table 3: Calculated Bayesian information criterion

Nodes in hidden layer	Mean error	BIC
1	0.164	-2.473
2	0.139	-2.331
3	0.136	-2.128
4	0.085	**-2.752**
5	0.101	-1.830
6	0.097	-1.631
7	0.162	-1.194
8	0.157	-1.918
9	0.153	-0.790
10	0.274	-1.023
11	0.221	-0.203
12	0.211	-0.010
13	0.295	0.349
14	0.334	0.619
15	0.454	0.965

Table 4: Characteristics of the regression plot when a second hidden layer was added

	Training			Test		
Nodes in second hidden layer	**m**	**b**	**r**	**m**	**b**	**r**
1	0.663	-0.124	0.866	0.878	0.077	0.743
2	0.757	-0.092	0.877	0.651	-0.138	0.742
3	0.750	-0.103	0.881	0.365	-0.308	0.428
4	0.863	-0.052	0.934	0.631	-0.072	0.495
5	0.816	-0.075	0.904	0.173	-0.541	0.057
6	0.948	-0.028	0.982	0.192	-0.561	0.091
7	0.913	-0.063	0.955	0.762	-0.205	0.676
8	0.903	-0.047	0.965	0.588	-0.136	0.623
9	0.950	-0.024	0.982	0.506	-0.145	0.339
10	0.996	-0.002	0.999	0.435	-0.194	0.391

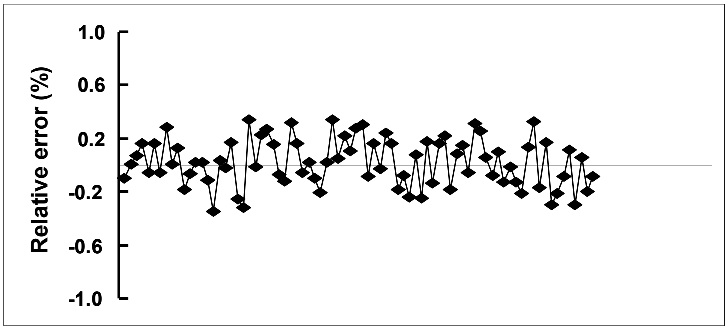


Figure 3: Scatter plot of residuals between observed and calculated outputs

CONCLUSION

In the present study, neural networks were designed and demonstrated to predict the moisture of dried grated coconut. A simple propagation network using the Levenberg-Marquardt for training the network was found to be very effective to generalize and predict the moisture of the final dried product. By using the analysis of variance test (ANOVA), it was found that the number of hidden layer nodes significantly affected (at the 0.05 level) the performance of the neural network. The configuration of the back propagation neural network that gave the best prediction was the one with one hidden layer consisting of four neurons. ANN predicted results were very close to the experimental values. The average MSE was observed to have reached the error goal of 0.01 and the maximum percentage relative errors were found to be between -0.35 and 0.34 %. Therefore, the predictive capability of neural networks can be utilized as a promising technique for modelling, estimating and predicting the process of drying grated coconut, whose dynamics are poorly known.

NOMENCLATURE

TAi	Temperatures of section Ai	°C
TBi	Temperatures of section Bi	°C
B	Intercept of regression line	%
TCi	Temperatures of section Ci	°C
FM	Final moisture	%
FT	Final temperature	°C
IM	Initial moisture	%
M	Slope of regression line	dimensionless
R	Correlation coefficient	dimensionless
SD	Standard deviation	(-)
RSD	Relative standard deviation	(-)
xn_i	Normalized data ranging between -1 and 1	(-)
x_i	Initial data	(-)
x_{min}	Minimum value of the data set	(-)
x_{max}	Maximum value of the data set	(-)
y_j	j^{th} hidden neuron output	(-)
w_{ij}	Weights of hidden layer	(-)
b_j	j^{th} hidden unit bias	(-)
BIC	Schwarz's Bayesian information criterion	(-)
V	Sum of squares approximation errors	(-)
n	Number of training patterns	(-)
p	Total number of network weights	(-)
N	Number of observations	(-)
k	Number of estimated parameters	(-)
df	Degree of freedom	(-)
e	Sum of squares error	(-)
y_c	Calculated output	(-)
y_o	Given output	(-)
m	Slope of regression line	(-)
b	Intercept of regression line at the origin	(-)
r	Correlation coefficient of the regression	(-)
MSE	Mean square error	(-)

REFERENCES

Ahoulé A. G., Optimisation de la production de coco râpé, Thèse, Département Génie Chimique et Agro-alimentaire, Institut National Polytechnique Houphouët-Boigny, Yamoussoukro (Ivory Coast), pp 1-40 (2004).

Barreto, G. A., Lime kiln hybrid control system. In Proceedings of Dynamic Modelling, Control, and

Application for Industry Workshop, Vancouver. IEEE Industry Applications Society, 44-50 (1997).

Baughman, D. R. and Liu, Y. A., Neural networks in bioprocessing and chemical engineering, San Diego: Academic Press, p. 5 (1995).

Desheng (Dash), Wu., Zijiang, Y. and Liang L., Using DEA-neural network approach to evaluate branch efficiency of a large canadian bank, Expert Systems with Applications, 31 (1), 108-115 (2006).

Fujiwara, T., in A. B. Bulsari (ed.), Neural networks for chemical engineers, Elsevier-Publishers, Amsterdam, 283-295 (1995).

Grossberg, S., Studies of the Mind and Brain, Reidel Press, Drodrecht, Holland (1982).

Haykin, S., Neural Networks, A Comprehensive Foundation, Macmillan College Publishing Company Inc., New York (1994).

Hill, T., Marquez, L., Oconnor, M. and Remus, W., Artificial neural network models for forecasting and decision making, International Journal of Forecasting, 10, p. 5 (1994).

Jarvensivu, M. and Seaworth, B., Neural network model use for quality prediction and control. In: Proceedings of International Symposium on Artificial Intelligence in Real Time Control. Arizona, IFAC, 3, 6-10 (1998).

Jend, S. and Das, H., Modelling for vacuum drying characteristics of coconut presscake, J. Food Eng., 79 (1), 92-99 (2007).

Jinescu, G. and Lavric, V., The artificial neural networks and the drying process. In Proceedings of International Drying Symposium, Vol. B, 1379-1385 (1994).

Jinescu, G. and Lavric, V., The artificial neural networks and the drying process modeling, Drying Technology, 13, no. 57, 1579-1586 (1995).

Kadhir, M. T., Richalet, J., Ringwood, J. and O'Connor, B., Modelling and predictive control of milk pasteurisation in a plate heat exchanger. In Proceedings Foodsim, Nantes, France (2000).

Karim, M. N. and Rivera, S. L., Artificial neural networks in bioprocess state estimation, Advances in Biochemical Engineering, 46, 1-33 (1992).

Khamis, A., Ismail, Z., Haron, K. and Tarmizi, A., Neural network model for oil palm yield modelling, J. Applied Sciences, 6 (2), 391-399 (2006).

Kohonen, T., Self-organization and associative memory, Second Edition, Springer Series in Information Sciences, Vol. 8, Springer Verlag, Berlin (1987).

Le Cun, Y., Denker, J.S. and Solla, S.A., Optimal brain damage; Advances in Neural Information, 2, 126-142 (1989).

Pramanik, K., Use of artificial neural networks for prediction of cell mass and ethanol concentration in batch fermentation using *saccharomyces cerevisiae* yeast, IE (I) Journal.CH, 85, 31-35 (2004).

Ripley, B. D., Neural networks and flexible regression and discrimination, Advances in Applied Statistics, 2, 37-56 (1994).

Savkovic-Stevenovic, J., Neural networks for process analysis and optimization: modelling and applications, Computers in Chemical Engineering, 18, (11-12), p 1149 (1994).

Schwarz, G., Estimating the dimension of a model, The Annals of Statistics, 6 (2), 461-464 (1978).

Shene, C., Andrews, B. and Asenjo, A. J., Optimization of *Bacillus subtilis* for the fed-batch fermentations for the maximization of the synthesis of a recombinant b -1, 3-glucanase, Computer Applications in Bio-technology, Japan, vol. CAB 7, p. 219 (1998).

Shene, C., Andrews, B. and Asenjo, A. J., Fed-batch optimizations of *Bacillus subtilis* fed-batch ToC46 (pPFF1) for the synthesis of a recombinant b -1, 3-glucanase: Experimental study and modelling, Enzyme and Microbial Technology, 24, p. 247 (1999).

Syu, M. J. and Tsao, G. T., Neural network modelling of batch cell growth pattern, Biotechnol. Bioeng., 42, 376-380 (1993).

Tai-Yue, W. and Shih-Chien, C., Forecasting innovation performance via neural networks - A case of Taiwanese manufacturing industry, Technovation, 26 (5-6), 635-643 (2006).

Tseng-Chung, T. and Li-Chiu, C., Neural networks analysis in business failure prediction of Chinese importers: A between-countries approach, Expert Systems with Applications, 29 (2), 244-255 (2005).

Ungar, L. H., Hartman, J. E., Keele, J. D. and Martin, G. D., Process modelling and control using neural networks, Int. Conf. on Intelligent Systems in Process Engineering, AIChE Symp. Series, 312-318 (1996).

Werbos, P. J., Backpropagation through time: What it does and how to do it, Proc. of the IEEE., 78, 1550-1560 (1990).