International Journal of Natural and Engineering Sciences 2 (2): 105-109, 2008
ISSN: 1307-1149 www.nobel.gen.tr

Optimization of Fermentation Medium for the Production of Lipopeptide Using Artificial Neural Networks and Genetic Algorithms

Sarat Babu IMANDI[1*] Sita Kumari KARANAM[2] Hanumantha Rao GARAPATI[3]

[1] Department of Biotechnology, ANITS, Sangivalasa, Bheemunipatnam, Visakhapatnam – 531 162, India

[2] M.R.College of Pharmacy, M. R. P. G. College, Phool Baugh, Vizianagaram – 535 002, India

[3] Center for Biotechnology, Department of Chemical Engineering, Andhra University, Visakhapatnam – 530 003, India

[*]Corresponding Author
e-mail : saratbabu_imandi@yahoo.com

Received: January 15,2008
Accepted: March 30, 2008

Abstract

Artificial neural networks and genetic algorithms were used to model and optimize fermentation conditions for the production of a novel lipopeptide by *Bacillus subtilis* MO–01. Experimental data reported in the literature were used to build the neural network model. Four process variables viz., sucrose (g/l), ammonium chloride (g/l), ferrous sulphate (μM), zinc sulphate (mM) served as inputs to the neural network model, and lipopeptide yield (g/l) served as an output of the model. Genetic algorithm was used to optimize the input space of the neural network model to find the optimum values of the variables for maximum lipopeptide yield. Using this procedure, artificial intelligence techniques have been effectively integrated to create a powerful tool for process modeling and optimization.

Key words: Artificial neural networks; genetic algorithm; Response Surface Methodology; lipopeptide.

INTRODUCTION

The performance of a fermentation processes is affected by numerous factors, which includes pH, temperature, time of fermentation, inoculum level, and the concentrations of medium components. Since the effects of these factors are very complex with possible interactions among them, they are often characterized through experimentation. To account for the interactive influences of different factors and to reduce the number of laborious experiments, statistical techniques such as Response Surface Methodology (RSM) are increasingly being used [1–3]. RSM seeks to identify and optimize significant factors with the purpose of determining what levels of the factors maximize the response (product yield or productivity). It uses statistical experimental designs to develop empirical models that relate a response (dependent variable) to some factors (independent variables). The literature is replete with studies that demonstrate the effectiveness of RSM which is essentially a collection of statistical and regression techniques.

In recent years, a limited number of researchers have investigated the possibility of using non-statistical techniques, such as artificial intelligence, to optimize fermentation processes [4, 5]. Among the various artificial intelligence techniques, genetic algorithms, a powerful stochastic search and optimization technique, have received considerable attention. Genetic algorithms can be used to optimize fermentation conditions without the need of statistical designs and empirical models. Such an approach has recently been used to optimize the production of polyols [6], the production of xylitols [7], and a culture medium for fed-batch culture of insect cells [8]. Although the use of genetic algorithms for fermentation optimization has proven to be effective, the methodology does not store the information generated at each stage of the optimization process. In contrast, RSM produces a model, although empirical, that mathematically describes the relationship existing between the independent and dependent variables of the process under consideration. The resulting model can be used for optimization as well as analysis of the sensitivity of the model output against each input variable. The most widely used approximating functions in the model building stage of RSM are quadratic polynomials.

From the perspective of process modeling, neural networks provide a mathematical alternative to the quadratic polynomial for representing data derived from statistically designed experiments. Neural networks are universal function approximators under certain general conditions [9]. This ability to approximate functions to any desired degree of accuracy makes them attractive for use as empirical models in response surface analysis. The input space of neural network models may be optimized using genetic algorithms. An attractive feature of the genetic algorithm is that it does not require continuity or differentiability of the objective function. A recent study has investigated the use of neural network and genetic algorithm to model and optimize the production of gluconic acid from glucose [10]. However, no comparison with RSM was made as the experiments were not based on statistical design. Liu et al., [11] found that neural networks outperformed quadratic polynomials in the modeling of a fermentation process. However, neural networks were not used in the optimization step. In this paper, study of the use of neural network and genetic algorithm was reported to accomplish objectives similar to those of RSM. A comparison of the hybrid approach and the standard RSM approach and their application to predict optimum conditions for a fermentation process reported by Gu et al., [12] is presented.

METHODOLOGY

Response Surface Methodology

Response surface methodology combines statistical experimental designs and empirical model building by regression for the purpose of process or product optimization. Statistical experimental design is a powerful method for accumulating information about a process rapidly and efficiently from a small number of experiments, there by minimizing experimental costs. An empirical model is then used to relate the response of the process to some independent variables. This usually entails fitting a quadratic polynomial to the available data by regression analysis. The general form of the quadratic polynomial is:

$$Y = b_0 + \Sigma\, b_i X_i + \Sigma\, b_i X_i^2 + \Sigma\, b_{ij} X_i X_j + e, \quad (1)$$

where Y is the predicted response, the X_i and X_j terms stand for independent variables, b_0 is the intercept, the b_i and b_{ij} terms are regression coefficients, and e is a random error component.

A near-optimum point can then be deduced by calculating the derivatives of Eq (1) or by mapping the response of the model onto a surface contour plot. There are numerous commercial software packages that facilitate the use of the quadratic polynomial for process modeling and optimization.

Gu et al., [12] fitted Eq (1) to their experimental data obtained from a Central Composite Design (CCD) for the production of the lipopeptide by *Bacillus subtilis* MO–01. The independent variables are sucrose (g/l), ammonium chloride (g/l), ferrous sulphate (μM), zinc sulphate (mM) (X_1, X_2, X_3 and X_4). The experimental design levels and the ranges of the four independent variables are listed in Table 1.

Neural Network–Genetic Algorithm Approach

A neural network is a mathematical representation of the neurological functioning of a brain. It simulates the brain's learning process by mathematically modeling the network structure of interconnected nerve cells. Because neural networks operate directly on input–output data, the essential requirement of neural network modeling is sufficient numbers of data. A neural network is thus a purely data driven model made up of interconnected processing elements called neurons that are organized in layers. A typical neural network has an input layer, one or more hidden layer, and an output layer. The neurons in the hidden layer, which are linked to the neurons in the input and output layers by adjustable weights, enable the network to compute complex associations between the input and output variables. The inputs of each neuron in the hidden and output layers are summed and the resulting summation is processed by an activation function. The most common choice of activation function is the sigmoid function. The process of determining the adjustable weights is known as training and it is analogous to the process of determining the coefficients of a polynomial by regression. The weights are initially selected in random and an iterative algorithm is then used to find the weights that minimize differences between the network–calculated and actual outputs.

The most commonly used algorithm is the back–propagation algorithm. In this training algorithm, the error between the results of the output neurons and the actual outputs is calculated and propagated backward through the network. The algorithm adjusts the weights in each successive layer to reduce the error. This procedure is repeated until the error between the actual and network-calculated outputs satisfies a pre-specified error criterion. Thus, neural network modeling is essentially a curve fit in multidimensional space. The text by Baughman and Liu, [13] provides a comprehensive description of the neural network modeling approach and its applications in bioprocessing. In this study a neural network model was constructed to model the fermentation process reported by Gu et al., [12]. The neural network consisted of a single output neuron i.e. lipopeptide yield (g/l) and four input neurons viz., sucrose (g/l), ammonium chloride (g/l), ferrous sulphate (μM), zinc sulphate (mM).

The neural network models can be considered as objective functions for the purpose of optimization. However, using conventional optimization techniques such as gradient-based methods to optimize a neural network model is not a simple task because it is difficult to calculate the derivatives of the model. Genetic Algorithms (GA), which are based on the principles of evolution through natural selection, i.e., the survival of the fittest strategy, have established themselves as a powerful search and optimization technique to solve problems with objective functions that are not continuous or differentiable. The genetic algorithm explores all regions of the solution space using a population of individuals (solutions). Each individual represents a set of independent variables. Initially, a population of individuals is formed randomly. The fitness of each individual is evaluated using an objective function. In this work the objective function is the neural network model. Upon completion of the fitness evaluation, genetic operations such as mutation and crossover are applied to individuals selected according to their fitness to produce the next generation of individuals for fitness evaluation. This process continues until a near optimum solution is found. A complete description of the implementation of genetic algorithms and their use as a problem-solving and function optimization technique can be found in the books by Holland, [14] and Goldberg, [15]. All of the neural network models and genetic algorithms described in this study were implemented in Matlab version 7.0. A modified version of the genetic algorithm of was used Houck et al., [16].

RESULTS AND DISCUSSION

Response Surface Methodology

Gu et al., [12] fitted Eq (1) to their experimental data obtained from a Central Composite Design (CCD) for the production of the lipopeptide by *Bacillus subtilis* MO–01. The independent variables are sucrose (g/l), ammonium chloride (g/l), ferrous sulphate (μM), zinc sulphate (mM) (X_1, X_2, X_3 and X_4). The experimental design levels and the ranges of the four independent variables are listed in Table 1. The dependent variable is lipopeptide yield (mg/l) (Y). The best fit regression equation obtained for the dependent variable is given.

$$Y = 1614.55 + 153.53X_1 - 17.97X_2 - 41.65X_3 + 43.28X_4 - 104.23X_1^2 - 47.26X_2^2 - 49.1X_3^2 - 67.48X_4^2 + 9.19X_1X_2 - 57.57X_1X_3 - 88.2X_1X_4 + 16.54X_2X_3 + 3.06X_2X_4 - 41.65X_3X_4 \quad (2)$$

The goodness of fit of the quadratic polynomial is expressed by the coefficient of determination, R^2 which is found to be 0.9233. The closer the value of R^2 is to 1, the

better is the correlation between the observed and predicted values. The value of R^2 indicates a fair degree of correlation between the observed and predicted values; about 92.33% of the variability in the response can be explained by the quadratic polynomial model. Contour plots obtained from the regression equation indicate a local optimum exists for each response in the area experimentally investigated; a set of values on the three independent variables that leads to maximum lipopeptide yield. The location of this optimum can be obtained by differentiating Eq (2) with respect to $X_1 - X_4$ and solving the resulting sets of algebraic equations. The maximum lipopeptide yield reported by Gu et al., [12] is 1.712 g/l. The combinations of the four independent variables giving the maximum lipopeptide yield are listed in Table 2. Also shown in Table 2 are the optimum conditions identified by the proposed neural network-genetic algorithm approach using the same data set reported by Gu et al., [12], and these are discussed in the next sections.

Table 1. Independent variables used and experimental design levels

Variables	Coded levels				
	–2	–1	0	+1	+2
X_1: Sucrose (g/l)	15	17.5	20	22.5	25
X_2 : Ammonium chloride (g/l)	1.8	2.4	3	3.6	4.2
X_3 : Ferrous sulphate (µM)	5.5	7	8.5	10	11.5
X_4 : Zinc sulphate (mM)	0.02	0.03	0.04	0.05	0.06

Neural Network Modeling

The first step in implementing a neural network modeling approach is to design the topology of the network (Fig.1).

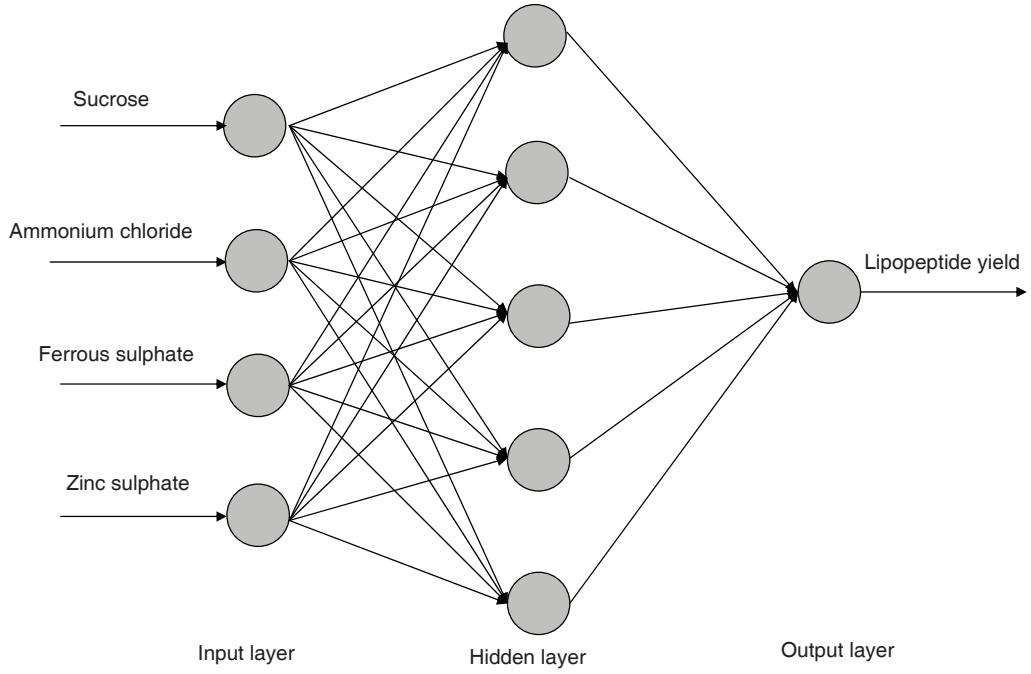

Figure 1. The Neural network topology with single hidden layer

A number of design parameters affect performance. These parameters include the choice of activation function and training algorithm, training parameters such as learning rate and momentum, number of hidden layers, number of neurons in each hidden layer, initial weights, and training duration. In general, feed-forward neural networks with one hidden layer containing a sufficiently large number of hidden neurons have been shown to be capable of providing accurate approximations to any continuous nonlinear function [9]. Unfortunately, no specific guidelines exist for the remaining design parameters because the topology of a neural network is likely to be problem-specific. The choice of design parameters for a neural network is thus often the result of empirical rules combined with trial and error. The configuration of the neural network developed in this work (a 4–5–1 structure: four input neurons–five neurons in one hidden layer–one output neuron) was determined after brief experimentation. To avoid the problem of overtraining, the data set comprising 30 experimental runs reported by Gu et al., [12] was split into two categories: a 'training set' of 27 experimental runs for optimizing the weights of the networks and a 'testing set' of 3 experimental runs to evaluate their predictive capability. Because empirical models like neural networks do not extrapolate data well, data for network training should be selected carefully if the best results are to be achieved. In this study the data selected for network training covered the lower and upper bounds of the output neuron (Y).

The transfer functions used in the neural networks were tansig and purelin at the hidden layer and outer layer respectively while newff function was used for the training of the neural networks which involve the evaluation of weights and biases based on a chosen optimization algorithm. The MATLAB built in function, trainbr, was utilized in this work which is based on Bayesian regularization back propagation method coupled with Levenberg-Marquardt optimization algorithm. The following equation is the outcome of the neural network training:

$$Y = W2*(2./(1.0+\exp(-2*(W1*X1+b1)))-1)+b2 \quad (3)$$

where $\boldsymbol{W^1}$ and $\boldsymbol{W^2}$ are the weights, $\boldsymbol{b^1}$ and b^2 are the biases, '*Y* ' is the predicted value from the neural network and **X** is the row vector of 4 independent variables, and $\mathbf{X^1}$ represents the transpose of this vector with a dimension of (5x1).

The equation (3) represents the output, Y (lipopeptide yield), for the given set of independent variables represented in 'X' when 'tansig' was used as the transfer function in the hidden layer and 'purelin' was used as the transfer function in the outer layer. The input data of the independent variable were transformed between -1 and +1 using the built in function 'premmnf' while 'postmmx' was used to transform back the optimized set of independent variables into the original scale. The simulated values of lipopeptide yield as given by equation (3) are in close agreement with those of experimental values and hence equation (3) would be considered as an adequate mathematical model for representing the lipopeptide production with the chosen process variables. A global optimization technique Genetic algorithm as implemented in Genetic Algorithm and Direct Search Toolbox of MATLAB 7.0 was utilized to optimize the above equation to obtain the optimum values of the process variables (X vector).

the network–trained outputs while the open circles denote the network–predicted outputs for input variables belonging to the testing set. The network models not only fit the training data very well but also provide predictions of the testing data very close to those measured experimentally. For comparison, lipopeptide yield calculated from the polynomial regression equation (Eq (2)) is also shown in Fig. 2 (triangles). It is obvious that the neural network predictions are much closer to the line of perfect prediction than those of the quadratic polynomial equations, confirming the usefulness of the neural networks as empirical model in response surface analysis. Once a satisfactory neural network model is created over the ranges of independent variables of interest, it can be used for optimization. For the fermentation example examined in this work, the optimum values of lipopeptide yield were obtained by using a genetic algorithm to optimize the input space of the neural network model developed. The optimum input conditions that result in the maximum output value were shown in Table 2. The maximum achievable lipopeptide yield for the fermentation is 1.908 g/l, according to the neural network model. This maximum yield identified by the neural network model is 11% higher than that identified by the polynomial equation.

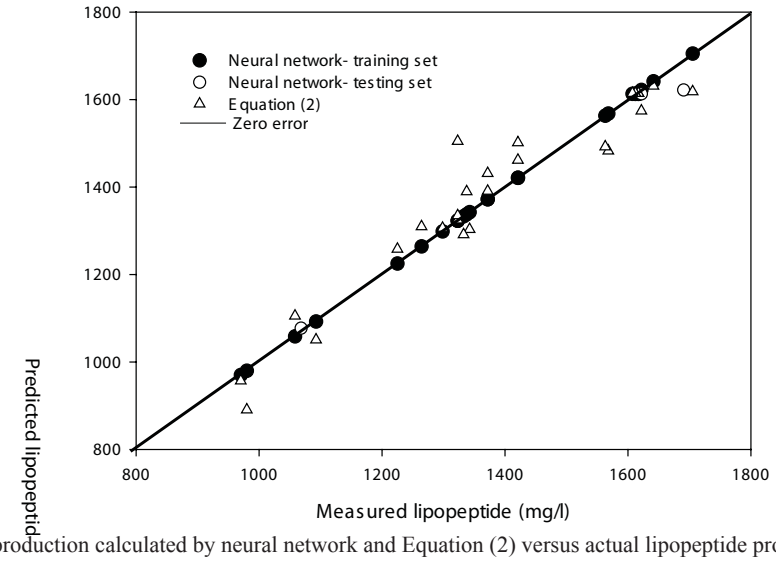

Figure 2. Lipopeptide production calculated by neural network and Equation (2) versus actual lipopeptide production.

Table 2. Maximum lipopeptide concentrations identified by quadratic polynomial and neural network models and the optimum input sets that result in the maximum output values.

Model	Dependent variable lipopeptide yield (g/l) (Y)	Independent variables			
		Sucrose (g/l) (X1)	Ammonium chloride (g/l) (X2)	Ferrous sulphate (μM) (X3)	Zinc sulphate (mM) (X4)
Quadratic polynomial	1.712	22.431	2.781	6.7879	0.0377
Neural network	1.908	22.5144	2.62677	8.31416	0.0289826

Fig. 2 shows the network–calculated lipopeptide yield for the training and testing data sets plotted against the corresponding experimental data. The solid circles represent

CONCLUSION

Empirical model building in the standard RSM approach often entails fitting quadratic polynomials to data derived from statistically designed experiments. In some cases like fermentation optimization problems, the ability of the quadratic polynomial to approximate the true response surface of a process may not be adequate. Hence in order to build a good response surface model, higher order polynomials or other models such as neural networks are required. Back propagation–ANN modeling and GA was successfully employed in this optimization study. The ANN model was found to be highly predictive of the system compared to a simple quadratic regression model. This work found that neural networks provided better fits to experimental data than conventional quadratic polynomials. The input space of a neural network model can be optimized using genetic algorithm. Thus the optimum values of the fermentation conditions obtained from neural network model are, at sucrose of 22.51 g/l, ammonium chloride of 2.63 g/l, ferrous sulphate of 8.31μM, zinc sulphate of 0.029 mM. Using the ANN-GA method, a maximum lipopeptide yield of 1.908 g/l of culture was obtained. This value was 11 % higher than the maximum output values obtained by optimizing the media constituents using response surface methodology indicating superior performance of ANN–GA method in optimizing the media constituents for enhancing lipopeptide yield by the *Bacillus subtilis* MO–01.

Acknowledgement

This project was financed by Research and Development Cell, ANITS, Sangivalasa, Bheemunipatnam, Visakhapatnam – 531 162, India.

REFERENCES

[1] Imandi, S.B, Bandaru, V.V.R, Somalanka, S.R, Bandaru, S.R and Garapati, H.R. (2008), Application of statistical experimental designs for the optimization of medium constituents for the production of citric acid from pineapple waste. Bioresour. Technol. 99, 4445–4450, 2008.

[2] Imandi, S.B, Bandaru, V.V.R, Somalanka, S.R. and Garapati, H.R. Optimization of medium constituents for the production of citric acid from byproduct glycerol using Doehlert experimental design. Enzyme. Microbial Technol. 40, 1367–1372, 2007.

[3] Dutra, R.L, Maltez, H.F.and Carasek, E. **Development of an on-line preconcentration system for zinc determination in biological samples.** Talanta 69,488–493, 2006.

[4] Kennedy M, Krouse D. Strategies for improving fermentation medium performance: a review. J Ind Microbiol Biotechnol 23, 456–475,1999.

[5] Weuster-Botz D. Experimental design for fermentation media development: statistical design or global random search? J Biosci Bioeng 90, 473–483,2000.

[6] Patil SV, Jayaraman VK, Kulkarni BD. Optimization of media by evolutionary algorithms for production of polyols. Appl Biochem Biotechnol 102, 119–128, 2002.

[7] Baishan F, Hongwen C, Xiaolan X, Ning W, Zongding H. Using genetic algorithms coupling neural networks in a study of xylitol production: medium optimization. Proc Biochem 38, 979–985, 2003.

[8] Marteijn RCL, Jurrius O, Dhont J, de Gooijer CD, Tramper J, Martens DE. Optimization of a feed culture medium for fed-batch culture of insect cells using a genetic algorithm. Biotechnol Bioeng 81, 269–278, 2003.

[9] Hornik K, Stinchcombe M, White H. Multilayer feedforward networks are universal approximators. Neural Networks 2, 359–366, 1989.

[10] Cheema JJS, Sankpal NV, Tambe SS, Kulkarni BD. Genetic programming assisted stochastic optimization strategies for optimization of glucose to gluconic acid fermentation. Biotechnol Prog 18, 1356–1365, 2002.

[11] Liu CH, Hwang CF, Liao CC. Medium optimization for glutathione production by Saccharomyces cerevisiae. Proc Biochem 34, 17–23, 1999.

[12] Gu XB, Zheng ZM , Yu HQ, Wang J, Liang FL, Liu RL . Optimization of medium constituents for a novel lipopeptide production by *Bacillus subtilis* MO-01 by a response surface method. Proc Biochem 40, 3196–3201, 2005.

[13] Baughman DR, Liu YA. Neural Networks in Bioprocessing and Chemical Engineering. San Diego: Academic Press 1995.

[14] Holland J. Adaptation in Natural and Artificial Systems. Ann Arbor: University of Michigan Press 1975.

[15] Goldberg D. Genetic Algorithms in Search, Optimization, and Machine Learning.

Reading: Addison-Wesley 1989.

[16] Houck CR, Joines JA, Kay MG. A Genetic Algorithm for Function Optimization:

A Matlab Implementation. Technical Report NCSU-IE TR 95-09. Raleigh, NC: North Carolina State University 1995.