Research Journal of Microbiology

ISSN 1816-4935

Academic
Journals Inc.

www.academicjournals.com

Research Journal of Microbiology 5 (12): 1275-1281, 2010
ISSN 1816-4935 / DOI: 10.3923/jm.2006.520.526

Optimization of Culture Medium for the Production of Poly-γ-glutamic Acid Using Artificial Neural Networks and Genetic Algorithms*

Imandi Sarat Babu and Garapati Hanumantha Rao
Department of Chemical Engineering, Center for Biotechnology,
Andhra University, Visakhapatnam-530 003, India

Abstract: Artificial neural networks and genetic algorithms are used to model and optimize a culture medium for the production of a Poly-γ-Glutamic Acid (PGA) by *Bacillus licheniformis* CCRC 12826. Experimental data reported in the literature was used to build neural network model. The concentrations of four medium components served as inputs to the neural network model and the amount of poly-γ-glutamic acid concentration served as a single output of model. Genetic algorithm was used to optimize the input space of the neural network model to find the optimum settings for maximum poly-γ-glutamic acid concentration. Using this procedure, artificial intelligence techniques have been effectively integrated to create a powerful tool for process modeling and optimization.

Key words: Artificial neural network, genetic algorithm, medium optimization, response surface methodology

Introduction

The performance of fermentation processes is affected by numerous factors, including pH, temperature, inoculum level and the concentrations of medium components. Since the effects of these factors are very complex with possible interactions among the various factors, 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. 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 studies have investigated the possibility of using non-statistical techniques, such as artificial intelligence, to optimize fermentation processes. These studies have been reviewed by Kennedy and Krouse (1999) and Weuster-Botz (2000). 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 (Patil *et al*., 2002), the production of xylitols (Baishan *et al*., 2003) and a culture medium for fed-batch culture of insect cells (Marteijn *et al*., 2003). Although the use of genetic algorithms for fermentation optimization has

Corresponding Author: Imandi Sarat Babu, Department of Chemical Engineering, Center for Biotechnology, Andhra University, Visakhapatnam-530 003, India Tel: +91-891-2844878
Fax: +91-891-2747969

**Originally Published in Research Journal of Microbiology, 2006* 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, albeit 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 (Hornik *et al*., 1989). 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 a neural network model 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 (Cheema *et al*., 2002). However, no comparison with RSM was made as the experiments were not based on statistical design. Liu *et al*. (1999) found that neural networks outperformed quadratic polynomials in the modeling of a fermentation process. However, the neural networks were not used in the optimization step. In this research we report on a study of the use of neural network and genetic algorithm 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 Lu *et al*. (2004) is presented.

Materials and Methods

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:

$$y = b_0 + \Sigma\, b_i x_i + \Sigma\, b_i x_i^2 + \Sigma\, b_{ij} x_i x_j + e, \tag{1}$$

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.

Lu *et al*. (2004) fitted Eq. 1 to their experimental data obtained from a central composite design for the production of the poly-γ-glutamic acid (PGA) by *Bacillus licheniformis* CCRC 12826. The independent variables are the concentrations of four medium components (x_1, x_2, x_3 and x_4). The experimental design levels and concentration ranges of the four independent variables are listed in Table 1. The dependent variable is poly-γ-glutamic acid (PGA) yield (y_1). The best fit regression equation for the dependent variable is listed in Table 2. The goodness of fit of the quadratic polynomial is expressed by the coefficient of determination, R^2. The closer the value of R^2 is to 1, the better is the correlation between the observed and predicted values. The values of R^2 listed in Table 2 indicate a fair

Table 1: Independent variables used and experimental design levels

	Coded levels				
Variables	-2	-1	0	+1	+2
x_1: Glutamic acid (g L^{-1})	28	31	34	37	40
x_2 : Citric acid (g L^{-1})	10	18	26	34	42
x_3 : Glycerol (g L^{-1})	134	140	146	152	158
x_4 : NH_4Cl (g L^{-1})	1	6	11	16	21

Table 2: Quadratic regression equations obtained for dependent variables (Lu *et al*., 2004)

Dependent variable	Best fit equation	Eq.	R^2
Poly-γ-glutamic acid yield (g L^{-1})	$y_1 = 35.34 -3.82x_1^2-4.34x_2^2-3.83x_3^2-7.20x_4^2$	2	0.707

Table 3: Maximum poly-γ-glutamic acid concentrations identified by quadratic polynomial and neural network models and the optimum input sets that result in the maximum output values

	Dependent variable	Independent variable			
Models	Poly-γ-glutamic acid yield (g L^{-1}) (y_1)	Glutamic acid (g L^{-1}) (x_1)	Citric acid (g L^{-1}) (x_2)	Glycerol (g L^{-1}) (x_3)	NH_4Cl (g L^{-1}) (x_4)
Quadratic polynomial	35.52	34.70	26.97	145.45	10.49
Neural network	38.2407	30.9321	35.6784	148.301	9.44703

degree of correlation between the observed and predicted values; about 70.7% 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 four independent variables that leads to maximum PGA 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 PGA yield reported by Lu *et al.* (2004) is 35.52 g L^{-1}. The combinations of the four independent variables giving these maximum concentrations are listed in Table 3. Also shown in Table 3 are the optimum conditions identified by the proposed neural network-genetic algorithm approach using the same data set reported by Lu *et al.* (2004).

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 (1995) 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 Lu *et al*. (2004). The neural network consisted of a single output neuron (poly-γ-glutamic acid (PGA) yield) and four input neurons (concentrations of the four medium components).

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, 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 models. 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 (1975) and Goldberg (1989). 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 Houck *et al*. (1995) was used.

Results and Discussion

Neural Network Modeling

The first step in implementing a neural network modeling approach is to design the topology of the network (Fig. 1). 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 (Hornik *et al*., 1989). 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 4-6-1 structure: four input neurons-six neurons in one hidden layer-one output neuron) was determined after brief experimentation. To avoid the problem of overtraining, the data set comprising 28 experimental runs reported by Lu *et al*. (2004) was split into two categories: a training set comprising 25 experimental runs was used to optimize the weights of the neural network and a testing set comprising 3 experimental runs was used 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_1).

Figure 2 shows the network-calculated poly-γ-glutamic acid (PGA) yield for the training and testing data sets plotted against the corresponding experimental data. The solid circles represent 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

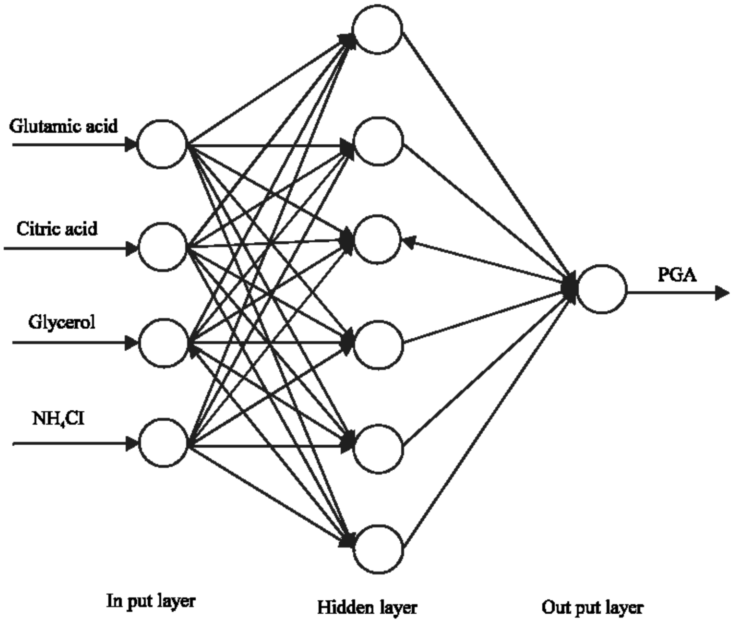

Fig. 1: The Neural network topology with three layers

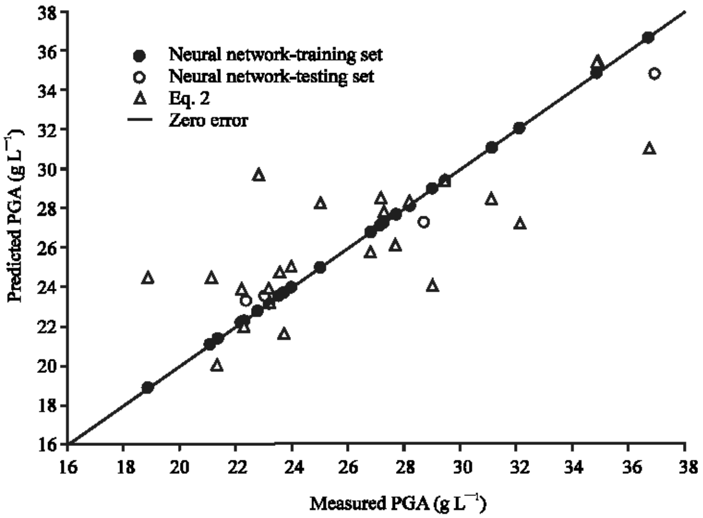

Fig. 2: Poly-γ-glutamic acid production calculated by neural network and Eq. 2 versus actual poly-γ-glutamic acid production

comparison, PGA yield calculated from the polynomial regression equation (Eq. 2) are 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 models in response surface analysis.

Optimization by Genetic Algorithms

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 PGA concentration may be obtained by using a genetic algorithm to optimize the input space of the neural network models developed. As with other artificial intelligence techniques, performance of the genetic algorithm is affected by a number of design parameters. These parameters include the initial population size, parent selection, crossover rate, mutation rate and number of generations. The results obtained are shown in Table 3 together with the input conditions that result in the maximum output values. The maximum achievable PGA concentration for this fermentation is 35.52 g L^{-1}, according to the neural network model. These maximum concentration identified by the neural network model is 7.6596% higher than those identified by the polynomial equation. This difference indicates that solution obtained from a polynomial model with poor modeling capability is not guaranteed to be optimum. The ability of a model to approximate the true response surface of a process with a high degree of accuracy is therefore of key importance in the optimization step.

The results in Table 3 reveal that different optimum conditions are found from models that have different modeling capabilities. To demonstrate that the differences are not due to the type of optimization procedure used (derivative estimation or genetic algorithm), the genetic algorithm was used to optimize the polynomial equations. The results obtained are very similar to the optimum solutions obtained by calculating the derivatives of Eq. 1 and 2. This agreement thus confirms that the optimum conditions identified by the polynomial equation are not dependent upon the method of optimization.

Often only quadratic polynomial models are chosen for a wide variety of fermentation optimization problems. We have shown that the quadratic polynomial is not always accurate enough. Clearly, in order to build a good response surface model for the example examined in this study, higher order polynomials or other models such as neural networks are required. In the model building stage of any RSM, it is of great importance to use an appropriate model to approximate the true response surface of a fermentation process in order to avoid arriving at suboptimal conditions.

Conclusions

Empirical model building in the standard RSM approach often entails fitting quadratic polynomials to data derived from statistically designed experiments. In some cases the ability of the quadratic polynomial to approximate the true response surface of a process may not be adequate. 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 algorithms which do not require the objective function to be continuous or differentiable. The hybrid neural network-genetic algorithm approach described in this work serves as a viable alternative to the standard RSM approach for the modeling and optimization of fermentation processes.

Acknowledgment

The project was financed by University Grants Commission (SAP-III), New Delhi, India.

References

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

Baughman, D.R. and Y.A. Liu, 1995. Neural Networks in Bioprocessing and Chemical Engineering. San Diego: Academic Press.

Cheema, J.J.S., N.V. Sankpal, S.S. Tambe and B.D. Kulkarni, 2002. Genetic programming assisted stochastic optimization strategies for optimization of glucose to gluconic acid fermentation. Biotechnol. Prog., 18: 1356-1365.

Goldberg, D., 1989. Genetic Algorithms in Search, Optimization and Machine Learning. Reading, Addison-Wesley.

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

Hornik, K., M. Stinchcombe and H. White, 1989. Multilayer feedforward networks are universal approximators. Neural Networks, 2: 359-366.

Houck, C.R., J.A. Joines and M.G. Kay, 1995. A Genetic Algorithm for Function Optimization: A Matlab Implementation. Technical Report NCSU-IE TR 95-09. Raleigh, NC: North Carolina State University.

Kennedy, M. and D. Krouse, 1999. Strategies for improving fermentation medium performance: A review. J. Ind. Microbiol. Biotechnol., 23: 456-475.

Liu, C.H., C.F. Hwang and C.C. Liao, 1999. Medium optimization for glutathione production by *Saccharomyces cerevisiae*. Proc. Biochem., 34: 17-23.

Lu, W.K., T.Y. Chiu, S.H. Hung, I.L. Shih and Y.N. Chang, 2004. Use of response surface methodology to optimize culture medium for production of poly-γ-glutamic acid by *Bacillus licheniformis*. Int. J. Applied Sci. Eng., 2: 49-58.

Marteijn, R.C.L., O. Jurrius, J. Dhont, C.D. de Gooijer, J. Tramper and D.E. Martens, 2003. Optimization of a feed culture medium for fed-batch culture of insect cells using a genetic algorithm. Biotechnol. Bioeng., 81: 269-278.

Patil, S.V., V.K. Jayaraman and B.D. Kulkarni, 2002. Optimization of media by evolutionary algorithms for production of polyols. Applied Biochem. Biotechnol., 102: 119-128.

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