Use of Neural Networks for Modeling of Olefin Polymerization in High Pressure Tubular Reactors

WAI-MAN CHAN[1] and CLÁUDIO A. O. NASCIMENTO[2,*]

[1]Technology Center, Poliolefinas S. A., Av. Presidente Costa e Silva, 400, Santo André, São Paulo 09270-000 and [2]Department of Chemical Engineering, Polytechnic School of University of São Paulo, Cx. Postal 61548, São Paulo, SP 05424-970, Brazil

SYNOPSIS

Neural network computing is one of the fastest growing fields of artificial intelligence due to its ability to "learn" nonlinear relationships. This article presents the approach of back propagation neural networks for modeling of free radical polymerization in high pressure tubular reactors. Industrial data were used to train the network for prediction of the temperature profile along the reactor, as well as polymer properties such as density, melt flow index, and molecular weight averages. Comparisons were made between the neural network and mechanistic model predictions published in the literature. Results showed the promising capability of a neural network as an alternative approach to model polymeric systems.

INTRODUCTION

The development of kinetic models for free radical polymerization is not an easy task, especially when the process is carried out under conditions of high pressure and temperature. The difficulties lie with the complex reactions occurring simultaneously inside the reactor, as well as the poor understanding of flow patterns and of heat and mass transfer for mixtures involving polymers. Furthermore, the huge number of kinetic parameters that need to be estimated also constitutes an additional obstacle for sophisticated models.

In light of this difficulty of developing mechanistic models, we explored an alternative way to model polymeric processes based on neural network simulation. The neural network algorithm consists of a set of processing units, called "neurons," connected to one another. By adjusting parameters in the coupling between neurons, the network is capable of learning from a set of numerical data corresponding to the input variables and the desired outputs. In this article, three sets of industrial data were used for the network learning process. The prediction obtained from the trained network for a different set of input variables was found to compare favorably with the actual output values for the system.

REVIEW

High Pressure Process for Olefin Polymerization

Several high pressure tubular reactor models have been published in the literature,[1-7] one of the most comprehensive being that of Zabisky et al.[7] In their model, a kinetic mechanism was employed to describe the polymerization rate and polymer properties. Multiple feeding points for monomers, initiators (oxygen and organic peroxides), and chain transfer agents along the reactor were also taken into account. Figures 1 and 2 show, respectively, a typical industrial high pressure process diagram and a schematic of a tubular reactor for low-density polyethylene production. In this process, monomers are compressed through two stages (primary and secondary compressors) up to the pressure of synthesis (2000–3000 kgf/cm^2) inside the reactor. Oxygen and organic peroxides are used as free radical generators during the polymerization reaction. Side feeds of monomers, chain transfer agents, and ini-

* To whom correspondence should be addressed.

Journal of Applied Polymer Science, Vol. 53, 1277–1289 (1994)
 CCC 0021-8995/94/101277-13

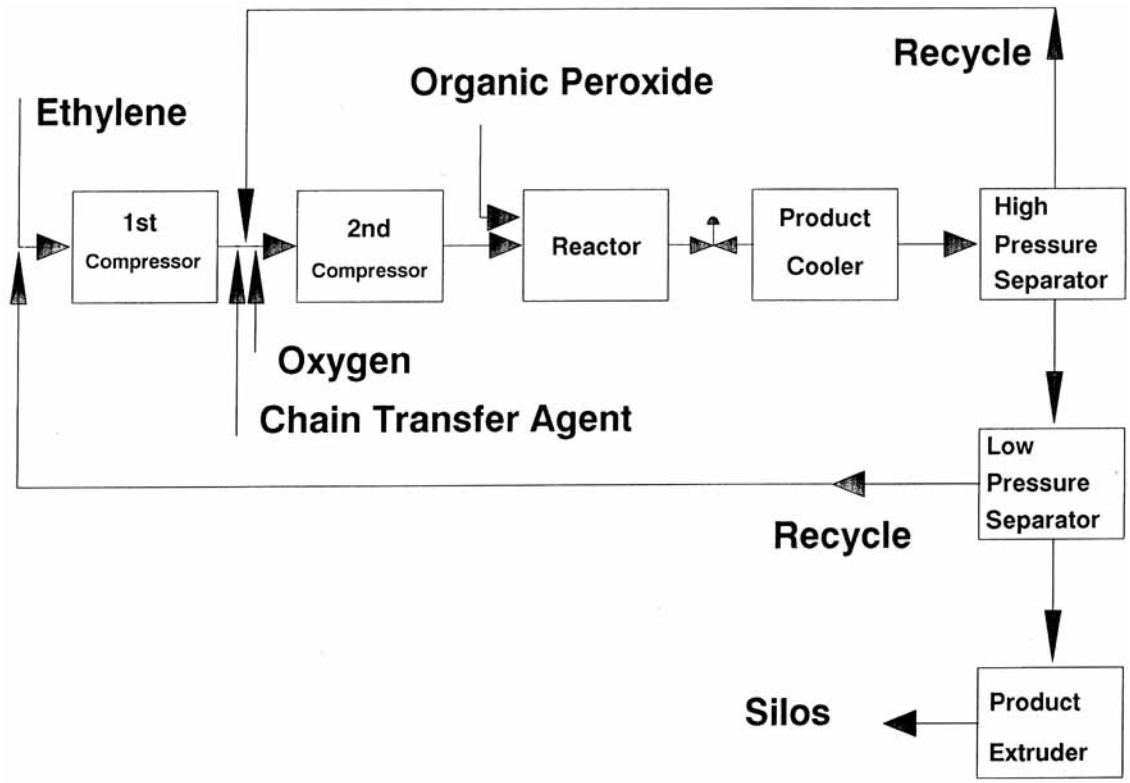

Figure 1 Simplified process flow sheet for synthesis of low-density polyethylene and copolymers.

tiators are possible along the reactor in order to produce a large variety of product grades with different molecular and rheological characteristics. Because the polymerization reaction is highly exothermic, the heat released is removed by cooling water in the jacket around the reactor. In order to increase the heat transfer capacity, the jacket is divided into several heating or cooling sections. Insights about the mechanistic modeling techniques generally used in the literature will be discussed later.

Neural Networks

Neural networks have been attracting great interest as predictive models, as well as for pattern recognition. According to Hernández and Arkun[8] (1992), "there has been a recent explosion of applications of neural networks to areas relevant to chemical engineers," following the works of others.[9,10]

The potential for employing neural networks in the chemical industry is tremendous, because nonlinearity in chemical processes constitutes the general rule. Neural networks possess the ability to "learn" what happens in the process without actually modeling the physical and chemical laws that govern the system. The success in obtaining a reliable and robust network depends strongly on the choice of the process variables involved, as well as the available set of data and its domain used for training purposes.

There are two main structures of neural networks commonly employed: feedforward networks (Fig. 3), in which information propagates only in one direction,[11] are useful for steady-state modeling; and recurrent networks (Fig. 4) are employed when long-term prediction is required[12] and are therefore more appropriate for dynamic models.

In general, the networks consist of processing neurons (represented by circles) and information flow channels between the neurons, usually denominated interconnects. The boxes represent neurons where the inputs to the network are stored. Each processing neuron first calculates the weighted sum of all interconnected signals from the previous layer plus a bias term, Eq. (1), and then generates an output through its activation function, Eq. (2). The most widely used networks are made up of three layers,[11] the input, hidden, and output layers.

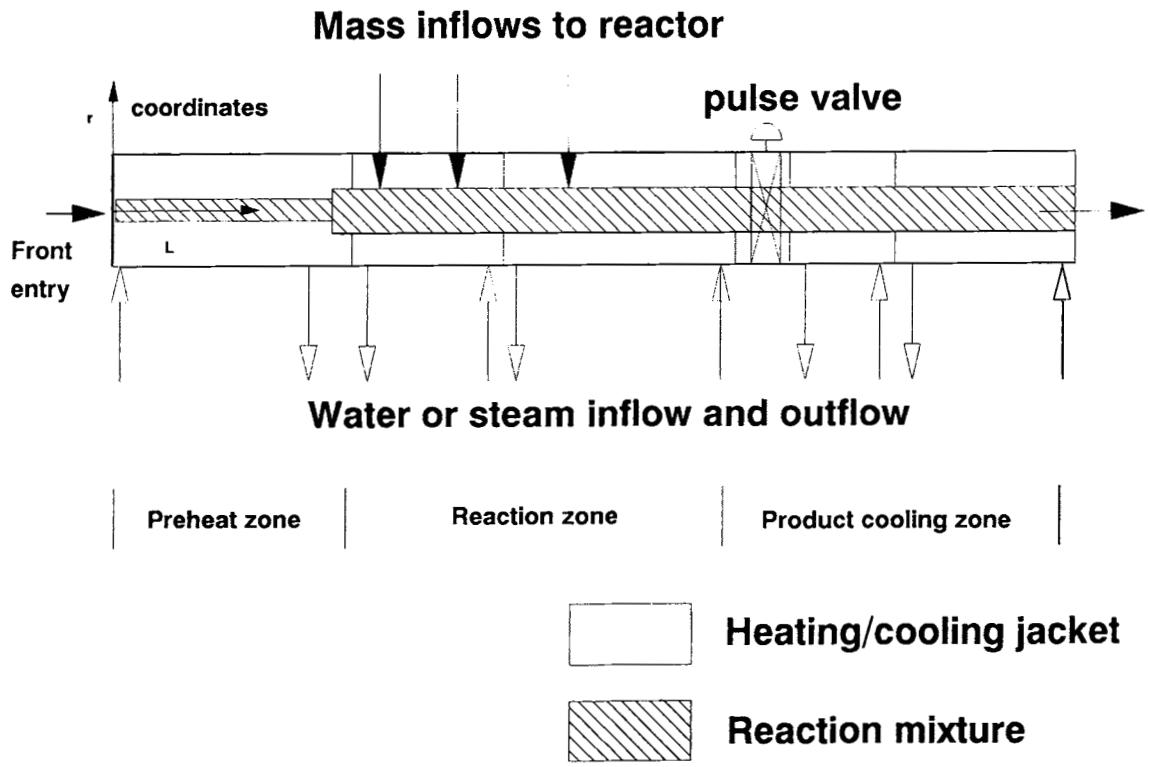

Figure 2 Schematic of high pressure tubular reactor with multiple feed points.

$$S_v = \sum_{u=1}^{n} W_{u,v} x_u + W_{n+1,v} \qquad (1)$$

$$f(S_v) = \frac{1}{1 + e^{-S_v}}. \qquad (2)$$

The system learns by making changes in its weights ($W_{u,v}$) and the input and output variables chosen for the network training are normalized. At present, the most extensively adopted algorithm for the learning phase is the back propagation algo-

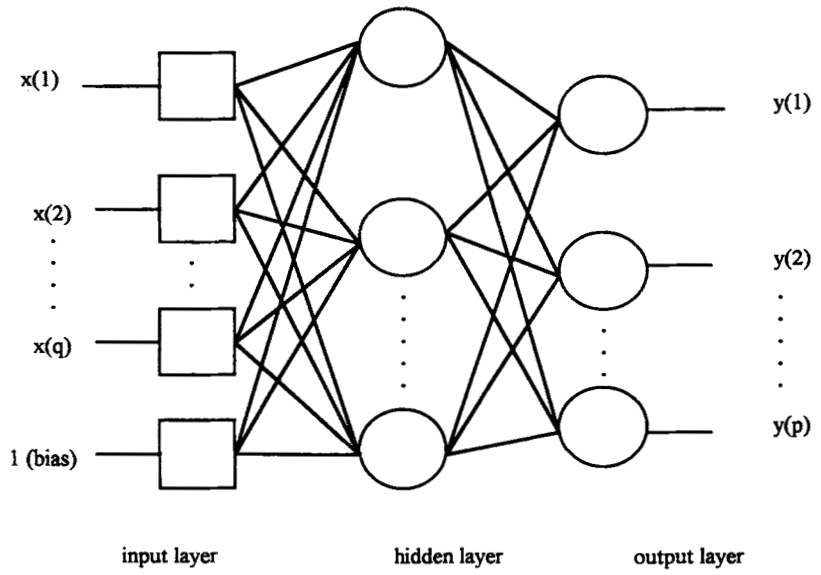

Figure 3 Multilayer feedforward neural network.

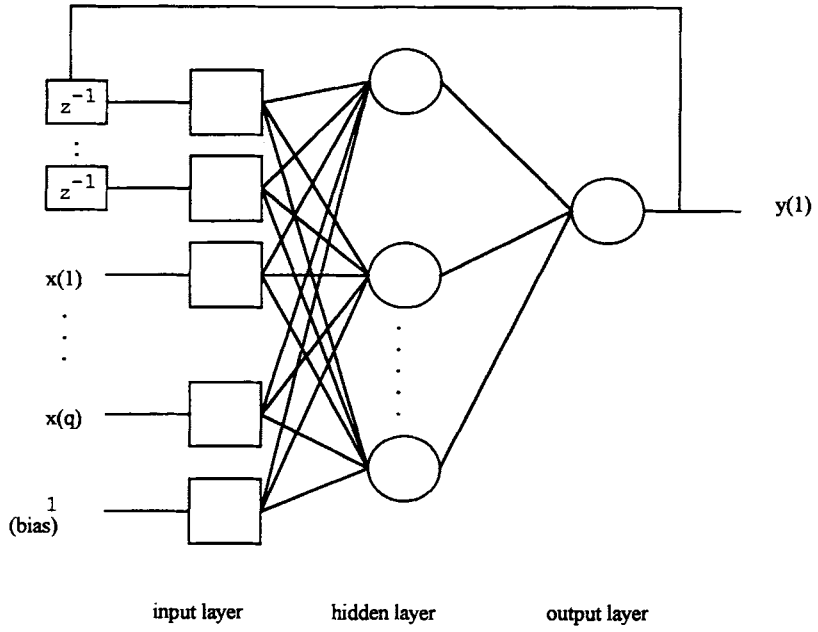

Figure 4 Recurrent neural network.

rithm,[13] which is a generalization of the steepest descent method. It consists of minimizing the mean square error (E), defined as:

$$E = \sum_{m=1}^{r} \sum_{k=1}^{p} [y_k^{(m)} - O_k^{(m)}]^2 \qquad (3)$$

where y_k comes from the r input–output pairs of data (x, y) available for training the network and O_k is obtained from the output layer signal, calculated by the following expression:

$$O_k = f(S_k). \qquad (4)$$

In the traditional gradient approach for minimizing the mean square error E with respect to the weights $W_{u,v}$, one calculates the derivatives $\partial E/\partial W_{u,v}$ and then moves in the steepest descent direction. This technique requires using all the input–output pairs to determine the gradient. The back propagation algorithm also uses gradient information to change the weights; however, it is calculated with respect to only one input–output pair at a time.[9] This input–output pair is introduced to the network and the weights are changed according to the following expression for the output layer:

$$\Delta W_{v,k}^{(m)} = \eta f'(S_k)[y_k^{(m)} - O_k^{(m)}]O_v^{(m)} \qquad (5)$$

where η represents a damping or accelerating factor. For the hidden layer, the expression below is used:

$$\Delta W_{u,v}^{(m)} = \eta f'(S_v)\left[\sum_{k=1}^{p} f'(S_k)[y_k^{(m)} - O_k^{(m)}] \times W_{v,k}^{(m-1)}\right] X_u^{(m)}. \qquad (6)$$

At each iteration, the weights between the hidden and output layers are adjusted first; subsequently, the weights between the input and hidden layers are changed. After presentation of the first input–output pair, one proceeds with the second pair, and so on.

FREE RADICAL COPOLYMERIZATION: MECHANISTIC MODELING OVERVIEW

Most of the mechanistic models in the literature employ a proposed kinetic mechanism and knowledge of flow patterns and, heat and mass transfer to describe all of the important chemical and physical phenomena that occur in the reactor. The reactions outlined below are widely employed to model free radical copolymerization kinetics.[14–18] The terminal model assumption is also frequently considered to be adequate, that is, the rate of reaction de-

pends only on the terminal unit on the chain and, thus, penultimate effects are neglected.

Organic peroxide initiation: the decomposition of an organic peroxide initiator, I, to form free radicals R,

$$I \xrightarrow{kd} n \cdot R$$

where n is the number of radicals formed per initiator molecule, normally two.

Oxygen initiation: oxygen may react with monomer to form peroxides, which then decompose to initiate the polymerization.

$$O_2 + M \xrightarrow{kd_{O_2}} 2R.$$

Propagation: reaction of monomer of type i with radicals of length r terminating in a monomer unit of type j,

$$R_j(r) + M_i \xrightarrow{kp_{ji}} R_i(r+1).$$

Termination by combination: bimolecular reaction between two radicals to form one dead polymer chain,

$$R_i(r) + R_j(s) \xrightarrow{ktc_{ji}} P(r+s).$$

Termination by disproportionation: bimolecular reaction between two radicals to form two dead polymer chains,

$$R_i(r) + R_j(s) \xrightarrow{ktd_{ji}} P^{=}(r) + P(s) \quad \text{or} \quad P^{=}(s) + P(r).$$

The disproportionation reaction forms a dead chain with a terminal double bond, denoted by the superscript $=$.

Transfer to chain transfer agent, modifier, or solvent: transfer of reactivity from radical type j to chain transfer agent TS, to form a dead polymer chain and a transfer radical TS^*,

$$R_j(r) + TS \xrightarrow{kfts_j} TS^* + P(r).$$

Transfer to polymer reaction: involves the transfer of reactivity from a radical of type j to an i type monomer unit in a dead polymer chain to form a radical with the active center located somewhere along the chain. If monomer is present, its propagation will lead to long-chain branching (LCB),

$$R_j(r) + P_i(s) \xrightarrow{kfp_{ji}} R'_i(s) + P(r).$$

Followed by propagation

$$R'_i(s) + M_j \xrightarrow{kp_{ij}} R_j(s+1) \quad \text{(LCB)}$$

where R' denotes an internal radical.

Backbiting reaction: the radical center is transferred to a site along the same chain, leading to short-chain branching (SCB),

$$R_j(r) \xrightarrow{kb_{ji}} R'_i(r).$$

Followed by propagation

$$R'_i(r) + M_j \xrightarrow{kp_{ij}} R_j(r+1) \quad \text{(SCB)}.$$

β-scission of internal radical centers: besides the propagation reactions, internal radicals may undergo β-scission to form a smaller radical and a dead polymer chain with a terminal double bond,

$$R'_j(r) \xrightarrow{k\beta_j} P^{=}(s) + R_{j\,\text{or}\,i}(r-s) \quad \text{or} \quad P^{=}(r-s) + R_{j\,\text{or}\,i}(s).$$

Depending on the complexity of the model being developed, additional reactions can be included in the kinetic mechanism. For more details see Zabisky et al.[7]

Method of Pseudokinetic Rate Constants

In order to simplify the mathematical equations for copolymerization, the method of pseudokinetic rate constants is frequently used.[7,18,19] Pseudokinetic rate constants are the sum of individual rate constants for each elementary reaction weighed by the fraction of monomer or radical type in the reactor. Thus, instead of writing down all the reactions between all monomer or radical types, one formulates the equations in terms of their overall concentrations,

$$k_{\text{global}} = \sum_{i,j} k_{ij}\omega_i\lambda_j \tag{7}$$

where ω_i, λ_j represent either monomer or radical mole fractions in the reacting mixture.

It should be noted that the use of pseudokinetic rate constants when branching reactions are involved is not strictly correct, because the composition of the chains will vary with monomer conversion. In the case of a tubular reactor, the conversion is relatively low ($\sim$ 20%) and the comonomer composition small, making compositional drift virtually insignificant; therefore, pseudokinetic rate constants should be a valid approximation.

Mass Balances

For a plug flow tubular reactor at steady state, the mass balance of species X_i can be represented by the following expression:

$$\frac{dF_{x_i}}{dL} = A_c R_{x_i} \tag{8}$$

where A_c is the cross section area of the reactor, F_{x_i} represents the molar flow rate of species X_i, and R_{x_i} denotes the rate of reaction of species X_i.

Because elementary reactions are assumed, the rate of reaction for all species is first order with respect to each reactant involved.

Energy Balance

Heat transfer along the reactor is an important parameter for predicting the rate of polymerization inside the reactor. For every 10°C rise in temperature, the rate of polymerization may roughly double. The resulting energy balance on the reactor side is:

$$\frac{dT}{dL} = \left(\frac{\overline{\Delta H}}{WC_p}\right)\left(\frac{dP_1}{dL} + \frac{dP_2}{dL}\right) + \left(\frac{2U\pi r_{\text{int}}}{WC_p}\right)(T_c - T) \tag{9}$$

where C_p is the heat capacity of the reaction mixture; $\overline{\Delta H}$ denotes the average heat of polymerization (energy released per mole of monomer reacted, defined as a positive number); P_i indicates moles of monomer i that have reacted to form polymer; r_{int} represents the internal radius of reactor; T_c is the jacket temperature; U indicates the overall heat transfer coefficient; and W denotes the mass flow of the reaction mixture.

Pressure Profiles

Because the rate of reaction and the thermodynamic properties of the reaction mixture are functions of the pressure, it is important to take into account the pressure change down the reactor length. The differential equation for the pressure drop down a tube in the turbulent regime is described by:

$$\frac{dP}{dL} = -\frac{f_f \rho u^2}{g_c r_{\text{int}}} \tag{10}$$

where f_f is the fanning friction factor, ρ the solution density, u the linear velocity, and g_c the gravitational constant, used to convert kgm (kilograms mass) to kgf (kilograms force).

Molecular Weight Averages

In order to predict physical and rheological properties of the polymer produced, a knowledge of the molecular weight distribution is of great importance. For such complex polymerization kinetics, calculation of the entire molecular weight distribution is rather difficult. Fortunately, most polymer properties can be correlated with molecular weight averages, reducing the number of equations to be solved. Among the different techniques available,[7,21] the method of moments provides a relatively simple way of calculating the important averages. The moments of the molecular weight distribution are obtained by writing mass balances for the radical and polymer molecules of chain length r, multiplying each term by the appropriate power of r and summing the terms from $r = 1$ to ∞.

The moments of the polymer radical size distribution are defined as:

$$Y_i = \sum_{r=1}^{\infty} r^i R(r) \tag{11}$$

and similarly, the moments of the polymer size distribution are defined by

$$Q_i = \sum_{r=2}^{\infty} r^i P(r). \tag{12}$$

One of the disadvantages of using the method of moments is the closure problem. Depending on the kinetics adopted, the moments of the polymer size distribution may be a function of the higher moments. When this occurs, one must arrive at a closure technique that will adequately predict the higher moments as a function of the lower ones. For a more detailed discussion concerning the moment closure problem, see Zabisky et al.[7]

The number and weight average molecular weights of a polymer can be calculated, respectively, by:

$$\overline{M_n} = m\frac{Q_1}{Q_0} \quad \overline{M_w} = m\frac{Q_2}{Q_1} \tag{13}$$

where m represents the molecular weight of the monomer.

The parameter that indicates the width of the molecular weight distribution is the polydispersity index, defined as:

$$\mathrm{PI} = \frac{\overline{M_w}}{\overline{M_n}}. \tag{14}$$

Notice that the lower limit of the polydispersity index is unity. In this case, the polymer is said to be monodisperse, that is, all the molecules have the same length.

Branching Frequencies

The SCB and LCB may have a considerable effect on polymer properties. The more SCBs incorporated along the polymer chain, the lower the polymer density will be; LCBs affect the optical and rheological properties. The SCBs are produced by the backbiting reaction; therefore, the number of SCBs produced is:

$$\frac{d(\mathrm{SCB})}{dL} = A_c K_{\mathrm{SCB}}[Y_0]. \tag{15}$$

LCBs are produced by the transfer to polymer reaction. Thus, the total number of LCBs produced is given by:

$$\frac{d(\mathrm{LCB})}{dL} = A_c K_{\mathrm{LCB}}[Y_0][Q_1] \tag{16}$$

where K_{SCB} and K_{LCB} are overall reaction rate constants.

The number of SCBs and LCBs per 1000 carbon atoms is given by (assuming two backbone carbon atoms per monomer unit):

$$\lambda_S = 500\frac{(\mathrm{SCB})}{Q_1} \quad \lambda_L = 500\frac{(\mathrm{LCB})}{Q_1}. \tag{17}$$

Given the mechanistic model presented above, the resulting number of coupled ordinary differential equations is quite large. Additionally, attempts to validate the model require that a substantial number of kinetic parameters be estimated from experimental data under conditions of high pressure and temperature. In light of these difficulties, a different way of modeling the system is presented hereafter.

NEURAL NETWORK APPROACH

As an alternative way of modeling the same polymerization system, the technique of neural networks is proposed. This method requires reliable data to train the network, but once the learning process is completed, the network will be capable of making predictions in a much faster manner than any realistic mechanistic scheme.

Neural networks are characterized by the large number of parameters involved (weights) due to the high connectivity among the neurons. In training a network, we intend to find an optimum set of weights that minimize the mean square error. For small data training sets, the number of weights to be estimated can be higher than the number of available data, and the error in fitting the nontrained data will decrease initially, but then will increase as the network becomes overtrained. In this case, one must check the results for the nontrained data sets. In contrast, when the number of weights is smaller, the overfitting problem is not crucial.[22]

Experimental Data Acquisition

The data used for the training process were collected from an actual industrial tubular reactor for low-density polyethylene production. The reactor, about 1300-m long, is surrounded with heating or cooling jackets. In order to monitor the temperature profile constantly along the reactor, an array of 116 thermocouples placed at strategic points was installed and their values registered in the control room. The presence of 116 thermocouples along the reactor provides operational flexibility, especially in the case when modifications in side feed positions are necessary. For a fixed set of side feed configurations, the temperature control can be carried out by only 60 thermocouples. The input and output variables chosen for network training are specified as follows:

Input Variables

1. Reactor pressure;
2. mass flow rate of monomer at the front feed;
3. mass flow rate of monomer at the first side feed;

4. mass flow rate of monomer at the second side feed;
5. mass flow rate of monomer at the third side feed;
6. mass flow rate of oxygen at the front feed;
7. mass flow rate of oxygen at the first side feed;
8. mass flow rate of oxygen at the second side feed;
9. mass flow rate of oxygen at the third side feed;
10. mass flow rate of chain transfer agent at the front feed;
11. mass flow rate of chain transfer agent at the first side feed;
12. mass flow rate of chain transfer agent at the second side feed;
13. mass flow rate of chain transfer agent at the third side feed;
14. mass flow rate of organic peroxide;
15. mass flow rate of steam in the jacket;
16. temperature of monomer feeds.

Output Variables

1. Density of the polymer produced (ρ);
2. melt flow index of the polymer produced (MI);
3. number average molecular weight of the polymer produced ($\overline{M_n}$);
4. weight average molecular weight of the polymer produced ($\overline{M_w}$);
5. monomer conversion at the end of the reactor (Conv.);
6. to 65. selected temperatures along the reactor.

For each input/output variable, four sets of data corresponding to different grades of commercial low-density polyethylene with varying operating conditions were prepared. The data acquisition was programmed according to a preset schedule while the reactor was operating under quasisteady-state conditions (all resin properties within the desired product specifications). In order to reduce random process fluctuations further, each data set is actually the mean value of three instantaneous readings during the same production campaign. Density, melt flow index, and molecular weight average properties were obtained by using a gradient column, a plastometer, and gel permeation chromatography (GPC), respectively.

Neural Network Training

After collecting the data, the input/output variables were normalized. In our network training process, only three data sets were used, the remaining one being employed to check the validity of the trained network. In this work, we adopt the back propagation algorithm with only one hidden layer in our network. A back propagation network learns by making changes in its weights. In order to select the optimum number of neurons to be used, a sensitivity analysis between the mean square error and the number of neurons was carried out for 10,000 iter-

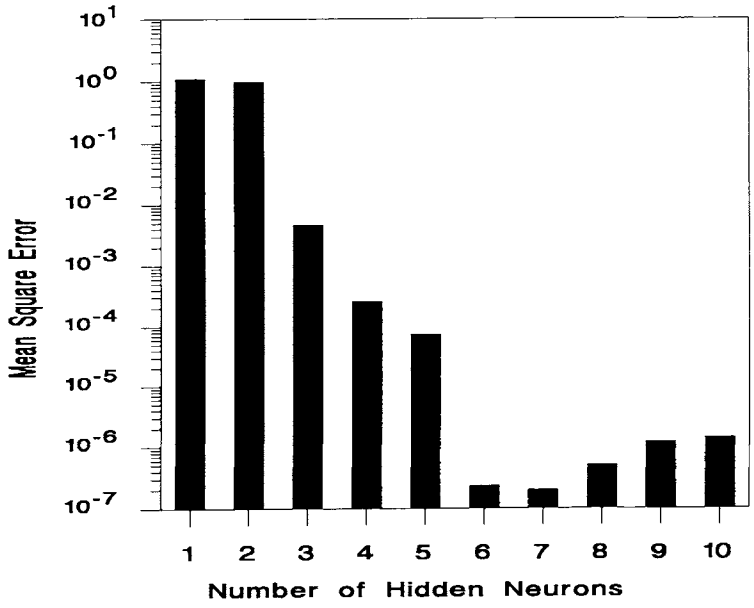

Figure 5 Residual sensitivity analysis after 10,000 iterations.

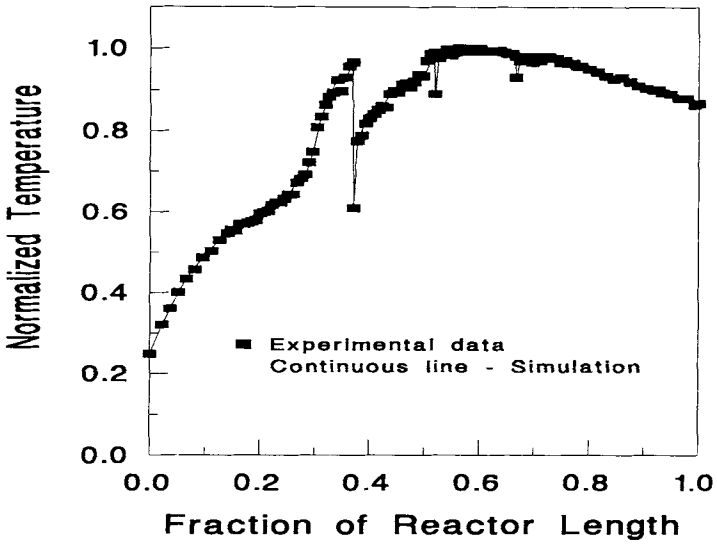

Figure 6 Neural network predictions for reactor temperature profile : training data set × experimental data.

ations. The results are shown in Figure 5. Note that the number of neurons which provides the smallest mean square error is seven, but we chose to use six rather than seven neurons because the results for the nontraining set were satisfactory compared to a seven neuron network, thus reducing the danger of overfitting.[23]

The network training was performed by using software developed in our laboratory for implementing the back propagation algorithm with a dynamic damping or accelerating factor.

RESULTS AND DISCUSSION

After obtaining the trained network with appropriate weights that minimize the mean square error, evaluations were made of the network performance. Figure 6 shows the fit of the reactor temperature profile for one of the training data sets against the actual plant temperatures. Note that the learning of the network is quite good for temperature responses, as well as for the other properties indicated in Table I. Figure 7 illustrates the reactor temperatures predicted by the network for the set of data not used in the training process.

Analyzing the results provided by the network for the nontraining data set, one finds that both the temperature profile and the polymer properties are within acceptable margins of error. Because molecular weight average data obtained from high temperature GPC may have an uncertainty of up to 20%, the network predictions of $\overline{M_n}$ and $\overline{M_w}$ in Table I fall within the confidence interval of the experimental data.

Table I Network Predictions vs. Plant Data

	Training Set						Nontraining Set	
	1st		2nd		3rd		4th	
	Network	Actual	Network	Actual	Network	Actual	Network	Actual
$\rho\left(\frac{g}{cm^3}\right)$	0.9204	0.9204	0.9235	0.9235	0.9215	0.9215	0.9227	0.9228
MI $\left(\frac{g}{10\ min}\right)$	0.4796	0.4800	0.3099	0.3100	0.8399	0.8400	0.2988	0.2500
$\overline{M_n}\left(\frac{g}{mol}\right)$	19005	19000	21000	21000	16000	16000	20827	23000
$\overline{M_w}\left(\frac{g}{mol}\right)$	170041	170000	190003	190000	150006	150000	187320	200000
Conv. (%)	22.3	22.3	22.9	22.9	22.8	22.8	22.7	22.8

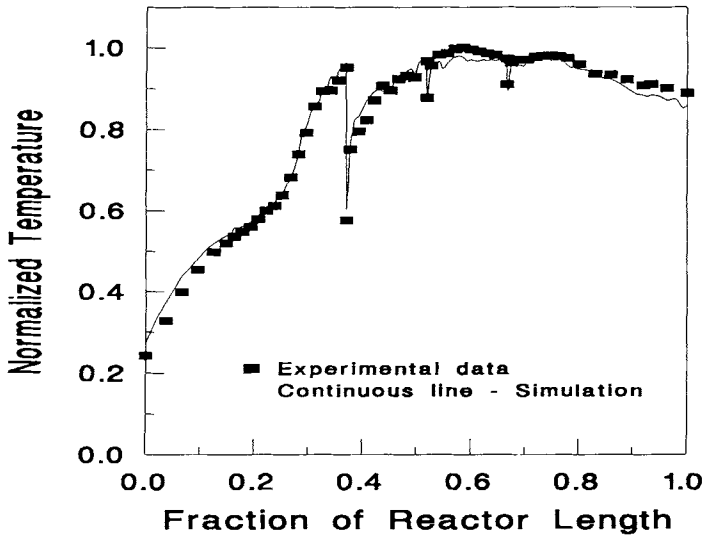

Figure 7 Neural network predictions for reactor temperature profile : nontraining data set × experimental data.

Predictions Between Mechanistic Model and Neural Network

As described in previous sections, a mechanistic model is composed of a set of mass and energy balances that represent the physical and chemical phenomena occurring in the system under study. Generally speaking, a mechanistic model can be a powerful tool for making predictions due to its broad range of applicability. However, the necessity of estimating a large number of parameters and solving a complex set of differential and algebraic equations constitutes an undisputable disadvantage. In this section, we compare the results predicted from the neural network with those obtained by using the mechanistic model proposed by Zabisky et al.[7] for the nontraining data set. Figures 7 and 8 represent the simulations of temperature profiles using the neural network and mechanistic models, respectively, against actual plant data. Table II shows comparisons of the other output variables, as well as their relative errors.

Analyzing the results in Table II, one finds that, for almost all output variables, the neural network provides better predictions than the mechanistic model. Special attention should be given to the re-

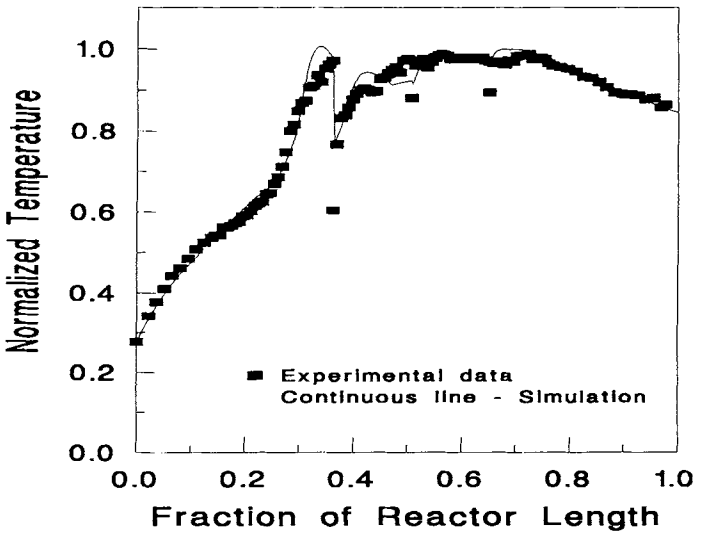

Figure 8 Reactor temperature profile : mechanistic model × experimental data.

Table II Mechanistic Model × Neural Network × Plant Data

	Actual	Mechanistic Model	Rel. Error (%)	Neural Network	Rel. Error (%)
$\rho\left(\frac{g}{cm^3}\right)$	0.9228	0.9223	−0.05	0.9227	−0.01
MI $\left(\frac{g}{10\ min}\right)$	0.25	0.39	56	0.30	20
$\overline{M_n}\left(\frac{g}{mol}\right)$	23000	17900	−22.2	20827	−9.5
$\overline{M_w}\left(\frac{g}{mol}\right)$	200000	212000	6.0	187320	−6.3
Conv. (%)	22.8	21.1	−7.5	22.7	−0.44

sults for the melt flow index (MI). In the mechanistic model, the MI value is not calculated directly from process variables, but obtained from an empirical correlation with the weight average molecular weight ($\overline{M_w}$), resulting in a larger intrinsic error. In contrast, the melt flow index is a direct output variable in our neural network training. For this reason, it is more valid to compare the $\overline{M_w}$ values than MI.

Despite the fact that both the mechanistic and neural network models provide reasonable predictions with respect to the experimental data, it is of fundamental importance to know whether they exhibit similar trends for the different input values. The sensitivity analyses we performed show the same tendency for all output variables; and Figure 9 illustrates the result for the weight average molecular weight versus pressure over a typical range of industrial operating conditions. It is important to point out that the sensitivity of the outcome depends essentially on the domain of the data sets used in the network training process. If the training data are restricted to a narrow range, the output may be fairly insensitive to perturbations and hence might eventually give unrealistic predictions.

CONCLUSIONS

In this article, we have successfully modelled a free radical polymerization in a tubular reactor under

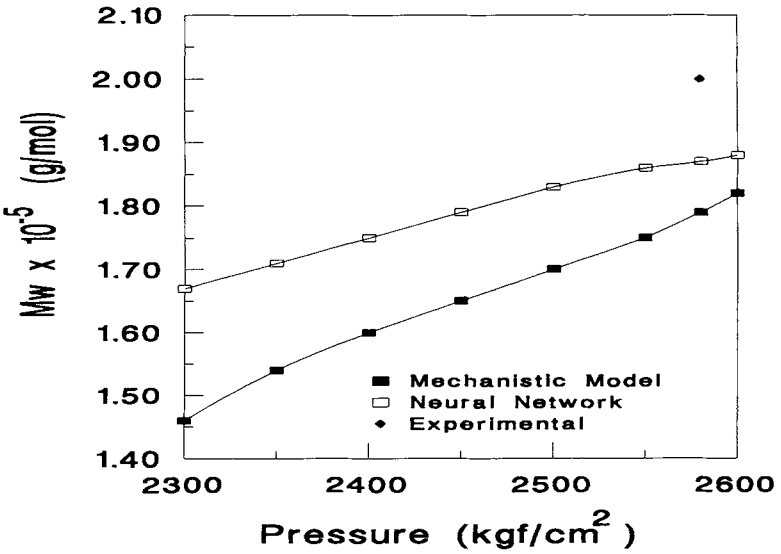

Figure 9 Behavior of weight average molecular weight with respect to pressure : mechanistic model × neural network.

high pressure via the neural network technique. The back propagation algorithm with only one hidden layer (six neurons) was adopted in the network. The training of the network was carried out by using software developed in our laboratory for minimizing the mean square error with respect to the weights. The outputs of the trained network are comparable to or better than those obtained from a mechanistic model published recently in the literature, indicating the potential of network predictions.

Although neural networks are relatively easy to develop and employ to model and simulate systems, there are several difficulties which should not be overlooked: the reliability of the network depends on the quality and range of the training data; the network training process can sometimes be tedious and time consuming; and in some cases, convergence can be slow and difficult, requiring the use of more efficient optimization techniques.

The authors would like to thank Poliolefinas S. A. for providing the industrial data used in this work and Prof. Frank Quina for revising the manuscript.

NOMENCLATURE

A_c	cross section area
C_p	heat capacity of the reaction mixture
f_f	fanning friction factor
F_{x_i}	molar flow rate of species X_i
g_c	gravitational constant used to convert $kg_{(mass)}$ to $kg_{(force)}$
k	rate constant with appropriate units for the order of reaction
k_{global}	overall pseudokinetic constants
L	length of the reactor
LCB	number of long-chain branches in the reaction volume
m	molecular weight of monomer
MI	melt flow index
$\overline{M_n}$	number average molecular weight
$\overline{M_w}$	weight average molecular weight
n	number of radicals formed per peroxide molecule
O_k	output layer signal from the neural network
P	reactor pressure
P_i	moles of monomer i bound as polymer
PI	polydispersity index
$P(r)$	polymer molecule of chain length r
$P^{=}(r)$	polymer molecule of chain length r with a terminal double bond
Q_i	ith moment of the polymer size distribution
r	monomer units
r_{int}	internal reactor radius
$R(r)$	radicals of chain length r
$R'(r)$	internal radical of length r
R_{x_i}	rate of reaction of specie X_i
SCB	number of short-chain branches in the reaction volume
S_v	sum of all interconnected signals from the previous layer in the neural network
T	reactor temperature at length L
T_c	jacket temperature at length L
u	linear velocity
U	overall heat transfer coefficient
W	mass flow rate of the reaction mixture
$W_{u,v}$	weights of neural network
x_u	input variables to the neural network
X_i	chemical species
$[X]$	concentration of any species X
y_k	desired output variables
Y_i	ith moment of the radical size distribution

Greek Letters

$\overline{\Delta H}$	average heat of polymerization
η	damping or accelerating factor for neural network training
λj	monomer or radical mole fractions in the reacting mixture
λ_L	number of long-chain branches per 1000 carbon atoms
λ_S	number of short-chain branches per 1000 carbon atoms
ρ	density
ω_i	monomer or radical mole fractions in the reacting mixture

Subscripts and Superscripts

i, j	monomer type i or j repeat units
s, L	short- and long-chain branches
u, v	indices of the neural network weights
$'$	internal radical
$=$	terminal double bond

REFERENCES

1. S. C. Agrawal and C. D. Han, *AIChE Journal*, **21**(3), 449 (1975).
2. J. K. Beasley, *Comprehensive Polymer Science*, G. Allen and J. C. Bevington, Eds., Pergamon Press, Oxford, 1989.
3. G. Donati, L. Marini, G. Marziano, C. Mazzateri, M.

Sampitano, and E. Langianni, *Proc. 7th Int. Symp. Chem. React. Eng.*, Boston, 1982.
4. P. Ehrlich and G. A. Mortimer, *Adv. Polymer. Sci.*, **1,** 386 (1970).
5. P. Ehrlich and R. N. Pittilo, *J. Polym. Sci.*, **43,** 389 (1960).
6. C. Kiparissides and H. Mavridis, *Chemical Reactor Design and Technology,* H. de Lasa, Ed., NATO ASI Ser. E: Appl. Sci., No. 110, 1985.
7. R. C. M. Zabisky, W. M. Chan, P. E. Gloor, and A. E. Hamielec, *Polymer,* **33,** 2243 (1992).
8. E. Hernández and Y. Arkun, *Computers & Chemical Engineering,* **16**(4), 227 (1992).
9. N. V. Bhat and T. McAvoy, *Computers & Chemical Engineering,* **14**(4/5), 573 (1990).
10. J. C. Hoskins and D. M. Himmelblau, *Computers & Chemical Engineering,* **12**(9/19), 881 (1988).
11. G. Cybenko, *Math. Control Signal Syst.*, **2,** 303 (1989).
12. H. T. Su, T. McAvoy, and P. Werbos, *Ind. Eng. Chem. Res.*, **31,** 1338 (1992).
13. D. Rumelhart and J. McClelland, *Parallel Distributed Processing Explorations in the Micro Structure of Cognition,* MIT, Cambridge, MA, 1986.
14. J. P. Marano, *Polymer Reaction Engineering—An Intensive Short Course on Polymer Production Technology,* School of Chemical Technology, University of New South Wales, Sydney, Australia, 1979.
15. H. Mavridis and C. Kiparissides, *Polym. Proc. Eng.*, **3,** 263 (1985).
16. R. V. Mullikin and G. A. Mortimer, *J. Macromol. Sci., Chem. (A),* **4**(7), 1495 (1970).
17. R. V. Mullikin and G. A. Mortimer, *J. Macromol. Sci., Chem. (A),* **6**(7), 1301 (1972).
18. H. Tobita and A. E. Hamielec, *Makromol. Chem., Macromol. Symp.,* **20/21,** 501 (1988).
19. W. M. Chan, P. E. Gloor, and A. E. Hamielec, *AIChE Journal,* **39**(1), 111 (1993).
20. A. E. Hamielec and J. F. MacGregor, *Polymer Reaction Engineering,* K. H. Reichert and W. Geiseler, Eds., Hanser Publishers, New York, 1983.
21. H. M. Hulburt and S. Katz, *Chem. Eng. Sci.*, **19,** 555 (1964).
22. R. D. De Veaux, D. C. Psichogios, and L. H. Ungar, *Computers Chem. Eng.,* **17**(8), 819 (1993).
23. J. F. Pollard, M. R. Broussard, D. B. Garrison, and K. Y. San, *Computers Chem. Eng.*, **16**(4), 253 (1992).

Received November 1, 1993
Accepted March 2, 1994