Revue Roumaine de Chimie, **2008**, *53*(12), 1141–1148

OPTIMIZATION OF A POLYSILOXANE SYNTHESIS PROCESS USING ARTIFICIAL INTELLIGENCE METHODS

Silvia CURTEANU[a*] and Maria CAZACU[b]

[a] "Gh. Asachi" Technical University, Faculty of Chemical Engineering
Bd. D. Mangeron No. 71A, 700050, Iaşi, Roumania, silvia_curteanu@yahoo.com
[b] "Petru Poni" Institute of Macromolecular Chemistry,
Aleea Gr.Ghica Voda 41A, Iaşi, 700487, Roumania

Received April, 2008

Neural networks and genetic algorithms were tested as tools for modeling and optimization of a polysiloxanes synthesis process. The heterogeneous cationic polymerization of octamethylcyclotetrasiloxane was approached for this purpose. The dependence between the main parameters of the process (conversion and molecular weight) and working conditions (temperature, reaction time, amounts of catalyst and co-catalyst) was modeled by using feedforward neural networks. The neural network model is then included into an optimizing control scheme, which uses a genetic algorithm solving technique and a multiobjective function in a scalar form. Genetic algorithms based methodology provides accurate results, computing optimal values for decision variables, which lead to the maximum monomer conversion and the desired value for molecular weight.

INTRODUCTION

Artificial neural networks (ANN) have been attracting great interest as predictive models, as well as for pattern recognition. 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.[1]

The open literature presents many attempts concerning neural network applications for polymerization processes: direct modeling with different types of neural networks, neural networks based soft sensors, inferential modeling, inverse neural network modeling, optimization, process control.[2-5] These problems were reviewed in a precedent work.[6] Fernandes and Lona provide a brief tutorial on simple and practical procedures that can help in selecting and training neural networks and address complex cases where the application of neural networks has been successful in the field of polymerization.[7]

Process optimization and control can have a significant strategic impact on polymer plant operability and economics. Polymer production facilities increase pressures for production cost reductions and more stringent quality requirements.[2] Generally, the optimization of a polymerization process is multiobjective in nature since it normally has several objectives, often conflicting and non-commensurable, that must be satisfied at the same time. Therefore, solving such a problem is accompanied by difficulties, starting with the manner of formulating the objective function, continuing with the choice of the working procedure, and the selection of the results from several possible options.

In recent years, there is a growing interest in the optimization techniques based on evolutionary algorithms, especially *genetic algorithms* (GA). Because of their flexibility, ease of operation, minimal requirements and global perspective, these algorithms have been successfully used in a wide variety of multiobjective problems. Regarding the use of genetic algorithms in chemical process engineering, particularly in polymer reaction engineering, two research directions are the most popular within the scientific community:

* Corresponding author: silvia_curteanu@yahoo.com

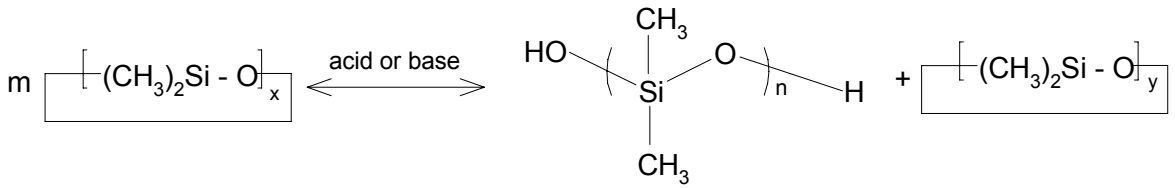
a) increasing the performance of algorithms that are already in use or enhancing their capabilities; b) applying genetic algorithms for different chemical process optimization. All the relevant work in this field has been surveyed the papers of Coello,[8] Deb[9] and Carlos,[10] where both the advantages and disadvantages of different types of GA have been analyzed. A series of applications of genetic algorithms in chemical process optimization were reviewed by Bhaskar,[11] Nandasana,[12] Barbarosa[13] and Brintrup[14].

This paper presents the use of neural networks and genetic algorithms as tools for modeling and optimization applied to the polymerization of octamethylcyclotetrasiloxane. The process of polysiloxane synthesis is very complex, with many reactions occurring concomitantly. The variation in time of the main parameters of the process (conversion and molecular weight) was modeled by using neural networks. The neural network model is then included into an optimizing control scheme, which uses a genetic algorithm solving technique and a multiobjective function in a scalar form. Our approach presents the advantage of computing the optimal values for the weights of the objectives within the genetic algorithm, along with the optimal values for decision variables.

EXPERIMENTAL

Two general methods are well known and widely used for polysiloxane synthesis: polycondensation of bifunctional siloxanes and ring-opening polymerization (ROP) of cyclic oligosiloxanes. ROP is the most traditional and significant way to obtain high molar mass linear polysiloxanes. This polymerization may be carried out either anionically or cationically and can be thermodynamically or kinetically controlled. There is a wide variety of compounds that can initiate the ROP polymerization of cyclosiloxanes, including strong organic and inorganic acids or bases.[15] New catalyst systems continue to be tested for this purpose.[16]

It is well known that, in the presence of the strong acids or bases, the Si-O bonds in both unstrained cyclosiloxanes and linear macromolecules (which have comparable energy) can be broken, and a mixture of cyclic and linear polysiloxanes will be obtained, according to Scheme 1. The siloxane bonds are continuously broken and reformed until the reaction reaches a thermodynamic equilibrium. Thus, the process leading to the polysiloxanes is a complex one, where many concurrent reactions (ring-opening, condensation, back-biting, redistribution etc.) occur.



with: x = 3 or 4, y = mixture of 3,4, 5.....

Scheme 1

In this paper, the synthesis of polydimethylsiloxane-α,ω-diols was performed by ring-opening polymerization of octamethylcyclotetrasiloxane, D_4, in the presence of a solid acid as catalyst and in absence of solvent. Water is used as a co-catalyst.[17] D_4 was chosen as monomer because this is of higher industrial importance towards cyclotrisiloxanes and represents the major component in the cyclosiloxanes mixture resulting from the synthesis.[18,19]

Materials

Octamethylcyclotetrasiloxane, $[(CH_3)SiO]_4$, D_4, was supplied by Fluka AG with the following characteristics: b.p.= 175 ^oC; n_D^{20} = 1.396; d_4^{20} = 0.955, purity > 99 % (GC). Purolite CT175 – a macroporous styrene-divinylbenzene copolymer with sulphonic groups, supplied by Viromet – was utilized after conditioning which consisted of washing with water, washing with a 4 % NaOH solution, regeneration with a 4 % HCl solution, washing with water and subsequent dehydration by azeotrope distillation with toluene and vacuum drying (110^{o}C, 20 mmHg). The cation-exchanger has the following characteristics: active group -SO_3H; volumic exchange capacity - 1.87 meq/ml; gravimetric exchange capacity - 4.5 meq/g; specific surface - 35 m^2/g.

Procedure

Cyclic monomer (D_4) was loaded in a reaction vessel equipped with thermometer, reflux condenser, and mechanical stirrer, and immersed in a thermostated bath at the pre-established temperature, according to the experimental program. The cation-exchanger, Purolite CT-175, as a catalyst (2.5 wt. % related to the reaction mixture) and water, as a co-catalyst (in pre-established percent related to the reaction mixture) were added when the desired temperature was attained in the reaction vessel. This is the initial moment of the reaction. The reaction mixture was stirred in these conditions for a certain time, according to the experimental program, after which the catalyst was removed by filtration. The reaction mixture was devolatilized by heating at 150^{o}C/5 mm Hg. The remained polymer was weighted to calculate yield and analyzed (molecular weight). The molecular weights, M_v, were evaluated on the bases of the viscometric measurements (M_v) performed at 25^{o}C, with an Ubbelohde Suspended Level Viscometer (Table 1).

Table 1

Experimental data for D_4 polymerization

No. crt.	Temp. (°C)	Time (h)	Catalyst (% gr.)	Water (% gr.)	Conversion	M_v
1	50	0.75	2.0	0.3	31.9	68 630
2	90	0.75	2.0	0.3	84.1	61 065
3	50	1.75	2.0	0.3	49.4	75 984
4	90	1.75	2.0	0.3	93.2	74 235
5	50	0.75	5.0	0.3	73.4	102 151
6	90	0.75	5.0	0.3	92.2	85 138
7	50	1.75	5.0	0.3	89.7	115 753
8	90	1.75	5.0	0.3	96.0	100 152
9	50	0.75	2.0	0.7	14.4	47 156
10	90	0.75	2.0	0.7	79.1	57 645
11	50	1.75	2.0	0.7	46.5	67 033
12	90	1.75	2.0	0.7	87.2	62 436
13	50	0.75	5.0	0.7	58.8	83 796
14	90	0.75	5.0	0.7	89.9	72 575
15	50	1.75	5.0	0.7	86.7	108 188
16	90	1.75	5.0	0.7	94.5	89 462
17	30	1.25	3.5	0.5	34.1	96 257
18	110	1.25	3.5	0.5	90.2	62 877
19	70	0.25	3.5	0.5	49.4	70 683
20	70	2.25	3.5	0.5	91.5	86 351
21	70	1.25	0.5	0.5	8.6	31 931
22	70	1.25	6.5	0.5	92.6	135 000
23	70	1.25	3.5	0.1	90.8	105 822
24	70	1.25	3.5	0.9	80.0	68 292
25	70	1.25	3.5	0.5	87.3	87 856
26	70	1.25	3.5	0.5	87.5	87 762
27	70	1.25	3.5	0.5	88.1	87 592
28	70	1.25	3.5	0.5	87.1	87 920
29	70	1.25	3.5	0.5	86.8	88 321
30	70	1.25	3.5	0.5	86.5	89 032
31	70	1.25	3.5	0.5	87.4	87 830

Accurate computational methods are an alternative to experiments since they are less costly and time-consuming and provide greater flexibility in terms of conditions and/or components explored.[20]

NEURAL NETWORK MODELING

The success in obtaining a reliable and robust network depends strongly on the choice of process variables involved, as well as the available set of data and the domain used for training purposes.

The most commonly used ANN is the standard backpropagation network in which every layer is linked or connected to the immediately previous layer. In this network, information propagates in only one direction and is useful for steady-state modeling. It has been shown that this type of network with at most two hidden layers can solve any non-linear problem, provided there is a sufficient numbers of hidden nodes. [21, 22]

Experimental data from Table 1 were used to design the neural network which has four inputs (temperature – T, reaction time – t, amount of catalyst – c_{cat} and amount of water – c_{water}) and two outputs (monomer conversion, x, and viscometric molecular weight, M_v). 10 % of these data represent the validation data set and the remaining data is the training data set.

One major problem in the development of neural network model is determining the network architecture, *i.e.* the number of hidden layers and the number of neurons in each hidden layer. No good theory or rule accompanies the neural network topology that should be used and trial-and-error procedure is still required. This is done by testing several topologies and comparing the prediction errors. Smaller errors indicate potentially good topologies, *i.e.* neural network topologies with chances to train well and to output good results. Fernandes and Lona[23] offer some practical recommendations for searching for potentially good topologies.

We propose a genetic algorithm based method for detecting the optimal topology for a neural network that should approximate the test data as well as possible. The representation of solutions in

chromosomes must simultaneously take into account two problems: including the information on network topology (number of hidden layers, number of neurons in these layers) and including actually the connection weights and biases of the neurons, with the purpose of verifying the network training errors. All this information is coded by real numbers, that is why we use the real encoding for the chromosome genes. A neural network with 2 hidden layers has the capability of learning convex regions in the solution space, that is why it has been considered that a network with 1 or 2 hidden layers is sufficiently complex for the majority of the test data.

The chosen representation has both advantages and disadvantages. The advantage is represented by the simplicity of the approach, as the genetic algorithm also accomplishes the finding of the optimum topology and the training of the neural network (determining the connection weights that allow approximating the test data). As a disadvantage can be mentioned a long training time because of the big number of chromosome genes (both information regarding the topology, and the connection weights and biases of the neurons).

A MLP(4:42:14:2) is obtained with this procedure applied to the polymerization process modeling; that means a feed-forward neural network with 4 neurons in the input layer (corresponding to the input variables *T, t,* c_{cat}, c_{water}), two hidden layers with 42 and 14 hidden neurons, respectively, and 2 neurons in output layer (for the output variables *x*, and M_v). The performance registered in the training phase for this network were: *MSE* (mean squared error) = 0.000452, *cor* (correlation between training data and neural model predictions) = 0.9985, and E_p (percent error) = 0.8755 %.

It is important to emphasize the fact that the method based on genetic algorithms is fast, secure and with high probability to lead a suitable network architecture. Moreover, the created program also accomplishes the training and predictions of the determined network.

GENETIC ALGORITHM OPTIMIZATION

Process optimization involves the minimization (or maximization) of an objective function, that can be established from a technical and/or economical view-point. In general, the decision variables are subjected to constraints such as valid ranges (max and min limits), as well as constraints related to safety considerations, and those that arise from the process model equations.[24]

Generally, in chemical engineering problems, both the objective function and the constraints are non-linear. Computational methods of non-linear programming with constraints usually have to cope with problems such as numerical evaluation of derivatives and feasibility issues. On the other hand, derivative-free methods, also called direct search methods, are usually less efficient and more time-consuming, as they usually require a higher number of iterations. This lower efficiency of direct search methods results from the necessity of solving the non-linear model equations in each iteration. Another important difficulty in optimization problems is the presence of multiple (local) optima, that is usually overcome by running the optimization program several times with different starting guesses of the decision variables.[24] All these drawbacks can be overcome by a correct formulation of the optimization problem and by using a reliable process model and an adequate solving method.

In our approach, the control variable vector, *u*, has as components:

$$u = [T, t, c_{cat}, c_{water}] \quad (1)$$

An admissible control input *u** should be formed in such a way that the performance index, *J*, defined by the following equation, is minimized:

$$\min J = w_x (1 - x) + w_{Mv} \left(1 - \frac{M_v}{M_{vd}} \right)^2 \quad (2)$$

subject to:

$$u_{min} \le u \le u_{max}; \quad 0 \le x \le 1; \quad M_v \ge 0 \quad (3)$$

In the equation (2), *J* is the objective function to be minimized, *w* are the weighting factors and M_{vd} is the desired value of the viscometric molecular weight.

The objective function includes the maximization of monomer conversion, which leads to higher productivity. The endpoint requirement on M_v leads to the production of the polymer having desired properties, because several physical properties of polymers are related to their values of molecular weight.

Genetic algorithms are among the most widely used stochastic search algorithms and represent a promising alternative to gradient-based optimization techniques.[11,12,,25,26] GAs have been applied to a

variety of different optimization problems because of their flexibility, but also due to the recent progress in the theory behind these algorithms.[27]

GAs are based on the mechanism of natural selection and genetics. They start with an initial set of solutions, called population, each solution in the population being called a chromosome. The chromosomes are evolved through successive iterations, called generations, by genetic operators (selection, crossover and mutation) that mimic the principle of natural evolution.

One obvious problem of the optimization that combines many objectives into a single function is that it may be difficult to generate a set of weights (namely w in equation 2) that properly scales the objectives when little is known about the problem. A single weighted sum approach requires a priori knowledge of the weights to vary the emphasis given to each objective. The present approach has the advantage of computing the optimal values for the weight coefficients of the objectives within the genetic algorithm, along with the optimal values for decision variables.

The optimization procedure includes the neural model, MLP(4:42:14:2) and is solved with a genetic algorithm. The fitness function of the GA is the scalar objective function (2). Genetic algorithm provides, after an iterative calculus, the optimal values for decision variables (T, t, c_{cat}, c_{water}), which are the inputs for the neural network model and, also, the weights for the objective function. With these inputs, the neural network computes the parameters x and M_v and the last one will by compared with M_{vd}. If the two values are identical or the difference between them is very small, we can conclude that the task of the optimization, represented by minimum of the objective function, is achieved.

RESULTS AND DISCUSSION

Table 2 presents a comparison between the experimental data and the predictions of the MLP(4:42:14:2) network on the validation data set (previously unseen data), demonstrating that the model included in the optimization procedure offers accurate results. Average relative errors for x and M_v are 2.8593 and 4.7481, and the correlation is 0.9986 and 0.9573, respectively.

Table 2

Experimental data and predictions of MLP(4:42:14:2) in the validation phase

T	t	c_{cat}	c_{water}	x experim	M_v experim	x network	M_v network	x error	M_v error
50	0.75	2.5	0.4	33.3	75630	34.17	75873	2.5986	0.32144
50	1.75	2	0.7	46.5	67033	45.50	61482	2.1505	8.28097
70	0.75	2.5	0.4	70.2	68000	68.71	66422	2.1225	2.32088
70	1.25	3.5	0.5	86.5	89032	87.34	87914	0.9732	1.25621
80	0.75	2.5	0.2	58.6	64200	60.61	64896	3.4386	1.08447
80	0.75	2.5	1.5	16.8	58600	15.40	54050	8.3333	7.76451
90	1.75	2	0.7	87.2	62436	90.20	57076	3.4404	8.58549
110	0.75	2.5	0.5	77.5	67000	79.56	69539	2.6581	3.79004
110	1.5	2.5	0.5	91.3	66600	91.28	60387	0.0189	9.32919

In this article, a simple genetic algorithm with real value encoding for the chromosomes was used. The stop GA condition corresponds to the point where the maximum number of generations has been executed. Population size, number of generations, crossover probability and mutation probability are known as the control parameters of genetic algorithm. The values of these parameters must be specified before the execution of GA and they depend on the nature of the objective function.

The optimization procedure as well as the neural network modeling are implemented in *Matlab 7.5* with original software, as specific functions were programmed for each phase of the genetic algorithm.

The optimization results are presented in two tables with the following structure: column 1 contains

the identification number used to refer the optimization in the discussions; columns 2, 3 and 4 contain the parameters of the genetic algorithm: the size of the initial population, *pop_size*, the number of generations, *gen_no* and the values for mutation and crossover rates, *mut_rate*, *cross_rate*; column 5 – the weights of the objectives computed within the genetic algorithm (w_x and w_{Mv}); column 6 – the optimal values of the decision variables provided by the GA (T, t, c_{cat}, c_{water}); column 7 – monomer conversion (x) and molecular weight (M_v) obtained as predictions of the neural model; column 8 – the value of the objective function (J) and the imposed value for molecular weight (M_{vd}).

Since GA is a stochastic algorithm, we ran it 3-4 times, for each situation (each row in the tables of results) to get statistically meaningful values of the computing time. One of the solutions (the best solution) is chosen and inserted into the tables.

Table 3 contains optimizations performed with different values for GA parameters in order to determine the best set for the actual polymerization process. The imposed value for M_{vd} is 80 000, thus we keep in mind the objective of the optimization: to obtain a greater conversion and $M_v = M_{vd}$. Only several examples are selected in Table 3 from approximately 50 tests.

Table 3

Optimizations with different values for the GA parameters.

No.	pop size	gen no	cross_rate mut_rate	Weights calculated by GA	Control variables	Outputs ANN	J, M_{vd}
1.	30	100	cross_rate = 0.9 mut_rate = 0.03	w_x = 14.59 w_{Mv} = 1.76	T = 65 t = 1.93 c_{cat} = 3.9 c_{water} = 0.6	x = 99.87 M_v = 96390	J = 0.000146 M_{vd} = 80000
2.	50	100	cross_rate = 0.9 mut_rate = 0.03	w_x = 67.56 w_{Mv} = 42.76	T = 65.4 t = 1.94 c_{cat} = 3.57 c_{water} = 0.65	x = 99.99 M_v = 80330	J = 0.000002 M_{vd} = 80000
3.	300	100	cross_rate = 0.9 mut_rate = 0.03	w_x = 0.4167 w_{Mv} = 22.2697	T = 67.3 t = 1.72 c_{cat} = 3.65 c_{water} = 0.62	x = 98.56 M_v = 79988	J = 0.005974 M_{vd} = 80000
4.	400	200	cross_rate = 0.9 mut_rate = 0.03	w_x = 0.5357 w_{Mv} = 31.5268	T = 66.8 t = 1.76 c_{cat} = 3.61 c_{water} = 0.63	x = 98.94 M_v = 80088	J = 0.0052728 M_{vd} = 80000
5.	50	100	cross_rate = 0.6 mut_rate = 0.03	w_x = 2.7595 w_{Mv} = 2.6700	T = 98.39 t = 2.18 c_{cat} = 3.50 c_{water} = 0.60	x = 98.72 M_v = 78308	J = 0.004583 M_{vd} = 80000
6.	50	100	cross_rate = 0.3 mut_rate = 0.03	w_x = 54.7391 w_{Mv} = 7.8565	T = 96.82 t = 2.11 c_{cat} = 4.19 c_{water} = 0.78	x = 97.92 M_v = 83253	J = 0.002368 M_{vd} = 80000
7.	50	100	cross_rate = 0.9 mut_rate = 0.1	w_x = 40.0468 w_{Mv} = 27.5685	T = 66.05 t = 1.94 c_{cat} = 3.81 c_{water} = 0.77	x = 99.53 M_v = 82361	J = 0.00626 M_{vd} = 80000
8.	50	100	cross_rate = 0.9 mut_rate = 0.5	w_x = 0.0117 w_{Mv} = 52.5536	T = 60.57 t = 0.88 c_{cat} = 2.38 c_{water} = 0.22	x = 76.70 M_v = 79756	J = 0.032126 M_{vd} = 80000

Therefore, the appropriate parameters of GA used to solve the proposed optimization problem are: pop_size = 50, gen_no = 100, cross_rate = 0.9 and mut_rate = 0.03. But, for our example, the influence of the GA parameters is not very significant. There are several results in Table 3 that represent acceptable solutions.

Table 4 presents supplementary optimization results obtained for different values imposed to viscometric molecular weight, M_{vd}. In most of the runs, M_v calculated with optimal decision values is very close to M_{vd}, which means that one of the optimization goals is reached. Also, monomer conversion has high values.

Table 4

Optimizations with different values for M_{vd}.

No.	pop size	gen no	cross_rate mut_rate	Weights calculated by GA	Control variables	Outputs ANN	J, M_{vd}
1.	50	100	cross_rate = 0.9 mut_rate = 0.03	w_x = 0.0301 w_{Mv} = 0.0477	T = 83.92 t = 2.24 c_{cat} = 2.39 c_{water} = 0.99	x = 90.67 M_v = 29259	J = 0.013028 M_{vd} = 20000
2.	50	100	cross_rate = 0.9 mut_rate = 0.03	w_x = 0.2260 w_{Mv} = 6.5373	T = 81.91 t = 1.36 c_{cat} = 1.85 c_{water} = 0.58	x = 91.84 M_v = 40039	J = 0.018454 M_{vd} = 40000
3.	50	100	cross_rate = 0.9 mut_rate = 0.03	w_x = 0.0455 w_{Mv} = 4.1720	T = 78.99 t = 1.45 c_{cat} = 1.65 c_{water} = 0.32	x = 97.48 M_v = 60001	J = 0.001145 M_{vd} = 60000
4.	50	100	cross_rate = 0.9 mut_rate = 0.03	w_x = 0.1139 w_{Mv} = 15.5583	T = 101.01 t = 2.14 c_{cat} = 3.30 c_{water} = 0.60	x = 99.98 M_v = 70164	J = 0.000055 M_{vd} = 70000
5.	50	100	cross_rate = 0.9 mut_rate = 0.03	w_x = 0.0228 w_{Mv} = 1.1973	T = 65.94 t = 1.90 c_{cat} = 3.68 c_{water} = 0.57	x = 99.98 M_v = 90067	J = 0.000004 M_{vd} = 90000
6.	50	100	cross_rate = 0.9 mut_rate = 0.03	w_x = 25.6667 w_{Mv} = 36.0647	T = 64.41 t = 1.81 c_{cat} = 3.62 c_{water} = 0.44	x = 99.99 M_v = 102830	J = 0.000001 M_{vd} = 100000

Run 1 in Table 4 does not represent an acceptable result. Tests with different values of GA parameters do not improve it; a better value for monomer conversion is obtained along with a worse value for M_v. This is the general rule of the multiobjective optimization with conflicting partial objectives, where a compromise between the possible solutions is realized.

The use of a scalar objective function, with weightly combined partial objectives has as main disadvantage the fact that the weight coefficients have to be previously known. In this work, the disadvantage of the scalarization is counteracted by the fact that GA computes the weight coefficients along with the optimal values of the control variables. This is possible because the GA considers the optimization problem globally, irrespective of the particular meaning given to the chromosomes.

The optimization procedure based on a simple genetic algorithm and a neural network model applied in this paper is easy to manipulate and provides satisfactory results. The theoretical approach of the synthesis of siloxane polymers performed here leads to useful information for practical purposes.

The modeling and optimization procedures based on neural networks and genetic algorithm can be also applied to other processes for which the amount of knowledge is limited.

CONCLUSIONS

In this paper we developed and tested the modeling and optimization capacities of simple topologies and simple working strategies for neural networks and genetic algorithms. For the polysiloxane synthesis, these techniques provide a useful tool to easy find the reaction conditions to obtain a polymer with desired molecular weight and high yield. The genetic algorithm solves the optimization problem and computes the weights attached to the partial objectives, while the neural network constitutes the model included in the optimization procedure. The neural model, a multilayer perceptron with 2 hidden layers, computes monomer conversion and molecular weight that are used in the objective function of the optimization procedure, with the goal to maximize monomer conversion and to obtain a desired value for the molecular weight.

The procedure developed herein is quite general and it can easily be used for other optimization problems.

Acknowledgements: This research occurred in the framework of the Project PN II PC 71-006/18.09.2007 for which the authors acknowledge financing authority (NASR = National Authority for Scientific Research).

REFERENCES

1. C. A. O. Nascimento, R. Giudici and R. Guardani, *Comput. Chem. Eng.*, **2000**, *24*, 2303-2314.
2. Y. Tian, J. Zhang and J. Morris, *Chem. Eng. Proc.*, **2000**, *41*, 531-538.
3. J. Zhang, *Ind. Eng. Chem. Res.*, **2004**, *43*, 1030-1038.
4. C. W. Ng and M. A. Hussain, *Chem. Eng. Proc.*, **2004**, *43*, 559-570.
5. C. Kuroda and J. Kim, *Neurocomputing*, **2002**, *43*, 77-89.
6. S. Curteanu and A. M. Popa, *Rev. Roum. Chim.*, **2004**, *49*, 3-23.
7. F. A. N. Fernandes, L. M. F. Lona, *Polym. React. Eng.*, **2002**, *10*, 181-192.
8. C. A. Coello, "Handling Preferences in Evolutionary-Based Multiobjective Optimization: A survey." In 2000 Congress on Evolutionary Computation, Piscataway, New Jersey, IEEE Service Center, **2000,** *1*, 30-37.
9. K. Deb, "Multi-objective Optimization using Evolutionary Algorithms", Wiley, Chichester UK, 2001.
10. A. C. C.Carlos, D. A. van Veldhuizen and G. B. Lamont, "Evolutionary algorithms for solving multi-objective problems", Kluwer Academic, New York, **2002**.
11. V. Bhaskar, S. K. Gupta and A. K. Ray, *Rev. Chem. Eng.*, **2000,** *16*, 1-54.
12. A. D. Nandasana, A. K. Ray and S. K. Gupta, *J. Chem.Reactor Eng.*, **2003,** *1*, R2.
13. A. P. Barbosa-Póvoa, *Comput. Chem. Eng.*, **2007**, *31*, 833-855.
14. A. M. Brintrup, J. Ramsden and A. Tiwari, *Comput. Ind.*, **2007,** 58, 279-291.
15. M. L. Vadala, M. Rutnakornpituk, M. A. Zalich, T. G. St Piere and J. S. Riffle, *Polymer,* **2004**, *45*, 7449-7461.
16. J. Chojnowski, W. Fortuniak, J. Kurjata, S. Rubinsztajn and J. A. Cella, *Macromolecules,* **2006**, *39*, 3802-3807.
17. M. Cazacu and M. Marcu, *Macromol. Rep*. A, **1995**, *32*, 1019-1029.
18. M. Cazacu, M. Marcu, S. Drăgan, C. Matricala, M. Simionescu and M. Holerca, *Polymer,* **1997**, *38,* 3967-3971.
19. M. Cazacu, M. Marcu, S. Drăgan, C. Matricala, *J. Appl. Polym. Sci.,* **1996**, *60*, 731-734.
20. Z. A Makrodimitri, R. Dohrn and I.G. Economou, *Macromolecules,* **2007**, *40*, 1720-1729.
21. G. Cybenko, *Math. Control Signal. Syst*., **1989**, *2*, 303-314.
22. N. K. Roy, W.D.Potter and D.P. Landau, *IEEE T. Neural Networ*., **2006**, *17*, 1001 – 1014.
23. F. A. N. Fernandes and L.M.F. Lona Braz. *J. Chem. Eng*., **2005**, *22*, 323-330.
24. C. A. O. Nascimento, R. Giudici and R. Guardani, *Comput. Chem. Eng*., **2000**, *24*, 2303-2314.
25. B. V. Babu and R. Angira, *Comp. Chem. Eng.*, **2006**, *30*, 989-1002.
26. E. G. Shopova and N.G. Vaklieva-Bancheva, *Comp. Chem. Eng.* **2006**, *30*, 1293-1309.
27. X. F. Yan, D. Z. Chen and S. X. Hu, *Comp. Chem. Eng.* **2003**, *27*, 1393-1404.