Journal of Applied Polymer Science

Prediction of the Gas Solubility in Polymers by a Radial Basis Function Neural Network Based on Chaotic Self-Adaptive Particle Swarm Optimization and a Clustering Method

Mengshan Li,[1,2] Xingyuan Huang,[1] Hesheng Liu,[1] Bingxiang Liu,[2] Yan Wu[2]
[1]College of Mechanical and Electric Engineering, Nanchang University, Nanchang, China
[2]School of Information Engineering, Jingdezhen Ceramic Institute, Jingdezhen, China
Correspondence to: X. Huang (E-mail: huangxingyuan001@126.com)

ABSTRACT: A novel model based on a radial basis function neural network (RBF NN), chaos theory, self-adaptive particle swarm optimization (PSO), and a clustering method is proposed to predict the gas solubility in polymers; this model is hereafter called CSPSO-C RBF NN. To develop the CSPSO-C RBF NN, the conventional PSO was modified with chaos theory and a self-adaptive inertia weight factor to overcome its premature convergence problem. The classical *k*-means clustering method was used to tune the hidden centers and radial basis function spreads, and the modified PSO algorithm was used to optimize the RBF NN connection weights. Then, the CSPSO-C RBF NN was used to investigate the solubility of N_2 in polystyrene (PS) and CO_2 in PS, polypropylene, poly(butylene succinate), and poly(butylene succinate-*co*-adipate). The results obtained in this study indicate that the CSPSO-C RBF NN was an effective method for predicting the gas solubility in polymers. In addition, compared with conventional RBF NN and PSO neural network, the CSPSO-C RBF NN showed better performance. The values of the average relative deviation, squared correlation coefficient, and standard deviation were 0.1282, 0.9970, and 0.0115, respectively. The statistical data demonstrated that the CSPSO-C RBF NN had excellent prediction capabilities with a high accuracy and a good correlation between the predicted values and the experimental data. © 2013 Wiley Periodicals, Inc. J. Appl. Polym. Sci. 130: 3825–3832, 2013

KEYWORDS: applications; polystyrene; properties and characterization; theory and modeling; thermal properties

Received 6 March 2013; accepted 8 May 2013; Published online 8 July 2013
DOI: 10.1002/app.39525

INTRODUCTION

Over the past 10 years, polymers have become an important part of daily life. In view of their importance in various applications, many researchers have focused considerable attention on the gas solubility of polymers.[1,2] The solubility is one of the most important physicochemical properties of polymeric compounds; it determines the compatibility of blending systems. The solubility data of various gases in polymers provide useful criteria for determining the requisite processing conditions, which are mostly collected from experimental and prediction data.[3–5] Unfortunately, the solubility data of gases in polymers within a wide range of pressures and temperatures are rare because some experimental studies are difficult to implement under many restricted conditions. In addition, experimental studies are very expensive and time-consuming.[3,6] Therefore, many researchers have tried to predict the solubility by theoretical methods, such as perturbed hard-chain theory, lattice–fluid theories, and cubic equations of state.[3,7,8] The theoretical model of the solubility data is as crucial as the experimental measurements for the understanding of the processes; incidentally, theoretical calculation is inexpensive and a timesaver compared to experiments. On the other hand, the most conventional prediction methods cannot predict the solubility of highly polar substances correctly and have some shortcomings, including a large inaccuracy.[1,7,9]

As far as the solubility of gases in polymers within a wide range of temperatures and pressures is concerned, it is affected by many factors, including the temperature, pressure, and interactions with groups of macromolecular chains. Because of the nonlinear mapping in these factors, most conventional methods for the prediction of gas solubility in polymers are often limited, whereas artificial neural networks (ANNs) have

© 2013 Wiley Periodicals, Inc.

found more popularity for the prediction of various thermodynamic properties.[8,10–12] Because of the nonlinear nature of the solubility of gases in polymers,[7,13–16] the ANN method could be considered an alternative research method to the solubility model.[8,17] For this purpose, many types of ANN model have been proposed. Bakhbakhi[7] and Lashkarbolooki et al.[8] presented a comparison between ANNs and equations of state for solubility prediction and demonstrated that the ANN method was a powerful approach with better accuracy. Recently, radial basis function neural networks (RBF NNs) have received considerable attention because of their potential to approximate nonlinear behavior. Li and coworkers[15,18]and Huang et al.[19] presented RBF NNs for melting index prediction. Khajeh and Modarress[20] proposed an adaptive neuro-fuzzy inference system (ANFIS) and RBF NN for the solubility prediction of gases in polystyrene (PS) and indicated that the ANFIS had better accuracy. In other words, RBF NNs without parameter optimization cannot achieve the desired performance. Therefore, a limitation in the RBF NN is that one has to set the controlling parameter beforehand. If the controlling parameter is not set appropriately, the performance of the prediction results will not be satisfactory.

Therefore, to improve the performance of the RBF NN, an optimization of the RBF NN parameters is necessary. Researchers have discovered that many evolutionary algorithms, such as the genetic algorithm (GA),[21,22] simulated annealing algorithm (SA),[23] ant colony algorithm,[18] and particle swarm optimization (PSO) algorithm,[24–27] can be used for this optimization. The PSO algorithm is a global and advanced algorithm with a strong ability to search global optimum values. Compared with the GA and SA, the most important advantages of the PSO algorithm are that the PSO algorithm has few parameters to adjust and is easy to implement.[28] Recently, researchers have developed lots of neural network models based on the PSO algorithm and have demonstrated that the PSO is a powerful approach for ANNs.[29,30] Although the PSO algorithm shows significant performance, the conventional PSO algorithm suffers from a premature convergence problem. Although the mentioned previously studies have achieved a high level of solubility prediction accuracy, a greater performance of the prediction model is still the first-line goal in the academic and industrial communities.

As motivated and inspired by the aforementioned problems, the aim is to overcome these problems and develop an effective and accurate prediction method for gas solubility in polymers. In this study, a novel model based on chaos theory, a self-adaptive PSO algorithm, a clustering method, and RBF NN is proposed; hereafter, this model is called the CSPSO-C RBF NN. In the CSPSO-C RBF NN, the traditional PSO algorithm was modified with chaos theory and a self-adaptive inertia weight factor (ω) to overcome the premature convergence problem. Then, the modified PSO algorithm was used to optimize the RBF NN connection weights, and the clustering method (*k*-means) was used to tune the hidden centers and radial basis function spreads. With the CSPSO-C RBF NN, the solubility of gases in polymers was investigated in a wide range of temperature and pressure. Comparison among different neural networks was carried out in detail to reveal that our proposed CSPSO-C RBF NN outperformed the RBF NN, particle swarm optimization neural network (PSO NN), and ANFIS model reported in related literature.[20]

THEORY

RBF NN

The RBF NN, as a typical feed-forward network, has been found to be very beneficial to many engineering problems. It consists of three layers: the input layer, hidden layer, and output layer. In this study, the activation function ($[f(x)]$), used as a Gaussian type, was as follows:

$$f(x)=\exp\left[-\left(\frac{x_i-c_j}{\sigma_j}\right)\right]$$

where $\mathbf{x_i}$ is the input vector and c_j and σ_j are the center and spread of the *i*th node in the hidden layer, respectively. The output $[O(x)]$ is given by

$$O(x)=\sum_{i=1}^{n} w_m k_m$$

where n is the number of clusters of the network, w_m is the connection weight, and k_m is the output of the hidden layer. The learning procedure of the RBF NN mainly consists of three parts: the connection weights, the hidden centers, and the radial basis function spreads. In this study, the connection weights were optimized by the modified PSO algorithm, called the *chaotic self-adaptive PSO algorithm*, whereas the hidden centers and radial basis function spreads were tuned by the classical *k*-means clustering method.

k-Means Clustering Method

The selection of the hidden centers and the widths was of great significance to the performance of the RBF NN. In this study, the classical *k*-means method was used to tune the hidden centers and radial basis function spreads of the RBF NN. Its implementation was performed as follows:

Step 1: Initialization: We initialized the radial basis function centers to the first training data.
Step 2: Clustering: All of the input data were grouped in different clusters or function centers. The result was that each input datum belonged to a cluster. The clustering formula is as follows:

$$\left\|x_m-c_{j^*}\right\|=\min_j\left\|x_m-c_j\right\|$$

where x_i is the input data and c_j is the function center of cluster *j*.
Step 3: Updating: For each c_j, the function center was updated as follows:

$$c_j=\frac{1}{m_j}\sum_{x_i\in j} x_i$$

where m_j is the number of data of cluster *j*.
Step 4: We repeated steps 2 and 3 until no more changes occurred in any clusters.

Table I. Experimental Data Used in This Study

Gas	Polymer	Temperature (K)	Pressure (MPa)	Solubility (g/g)	Data points	References
CO_2	PBS	323.15-453.15	1.025-20.144	0.0088-0.0176	69	37,38
CO_2	PBSA	323.15-453.15	1.098-20.127	0.0118-0.1741	58	37,38
CO_2	PP	313.20-483.70	2.930-24.910	0.0205-0.2617	92	4,38-40
CO_2	PS	170.00-473.15	2.068-44.410	0.0028-0.1606	104	5,6,38,41,42
N_2	PS	170.00-453.20	2.989-69.470	0.0011-0.0260	89	4-6,42
Total		170.00-483.70	1.025-69.470	0.0011-0.2617	412	

Chaotic Self-Adaptive PSO Algorithm

The PSO algorithm is an evolutionary intelligent algorithm, in contrast to the GA and SA; the most important advantages of the PSO algorithm, including its easy implementation and high performance, make it widely accessible.[24,26] In the PSO algorithm, the population is called a *swarm*, and its individuals are called *particles*. When, in an n-dimensional search space, the total number of particles is assumed to be m, the swarm is denoted as $x=(x_1, x_2, \ldots, x_m)^T$, the position of the ith particle is denoted as $\mathbf{x_i}=(\mathbf{x_{i,1}}, \mathbf{x_{i,2}}, \ldots, \mathbf{x_{i,n}})^T$, the velocity of the ith particle is expressed as vector $\mathbf{v_i}=(\mathbf{v_{i,1}}, \mathbf{v_{i,2}}, \ldots, \mathbf{v_{i,n}})^T$, the best position of the ith particle is $p_i=(p_{i,1}, p_{i,2}, \ldots, p_{i,n})^T$, and the best position of the neighbor particles is $p_g=(p_{g,1}, p_{g,2}, \ldots, p_{g,n})^T$ In the conventional PSO algorithm, the position and velocity are updated as follows:

$$V_{i,d}^{k+1}=\omega V_{i,d}^{k}+C_1\left(P_{i,d}^{k}-X_{i,d}^{k}\right)+C_2\left(P_{g,d}^{k}-X_{i,d}^{k}\right) \quad (1)$$

$$X_{i,d}^{k+1}=X_{i,d}^{k}+V_{i,d}^{k+1} \quad (2)$$

where $i=1, \ldots, m$; $\mathbf{x}_{i,d}^{k}$ and $\mathbf{v}_{i,d}^{k}$ denote the position and velocity of ith particle at d-dimensional and the kth iteration, respectively; c_1 and c_2 are the acceleration coefficients; $p_{i,d}^{k}$ represents the best position of the ith particle in d dimensions; and $p_{g,d}^{k}$ denotes the global best position.

Although conventional PSO algorithm shows significant performance, it has a potentially dangerous property of being premature.[31] In this study, the traditional PSO algorithm was modified with chaos theory and self-adaptive ω with the purpose of avoiding the premature convergence problem and accelerating the converging speed. The self-adaptive ω was proposed to obtain a balance between the exploration and exploitation, whereas chaos theory was used to generate chaotic sequences, which were used to adapt the acceleration coefficients. In eq. (1), ω affects the performance of the algorithm. A larger ω facilitates global exploration, whereas a smaller one facilitates local exploitation.[32,33] To enhance the performance of the PSO algorithm, in this study, the global best fitness and the average of the particles' local best fitness were used to determine ω in each iteration, and a self-adaptive ω was defined as follows:[31–36]

$$\omega=\omega_{\max}-P_{gbest(k)}/P_{lbestave(k)}-(\omega_{\max}-\omega_{\min})\times k/k_{\max}$$

ω was used to monitor the search situation and adapt the ω value based on three feedback parameters, k, $P_{gbest(k)}$, and $P_{lbestave}$. Here, $P_{gbest(k)}$ denotes the global best fitness at the kth iteration, $P_{lbestave}$ denotes the average local best fitness, $k_{\max}$ is the maximum iteration, k denotes the current iteration, and $\omega_{\max}$ and $\omega_{\min}$ are the maximum and minimum inertial weights, respectively.

c_1 and c_2, shown in eq. (1), were generated by a logistic map. The logistic map is a polynomial mapping (equivalently, a recurrence relation) of degree 2, often cited as an archetypal example of how complex chaotic behavior can arise from very simple nonlinear dynamical equations. Mathematically, the logistic map is written as follows:

$$c_{n+1}=4c_n(1-c_n)$$

where c_n is a number between zero and one. The logistic map was used to tune c_1 and c_2.

The modified PSO algorithm, called a *chaotic self-adaptive PSO algorithm*, was used to optimize the RBF NN connection weights. The RBF NN trained by the chaotic self-adaptive PSO and clustering method, called the CSPSO-C RBF NN, was used to investigate the solubility of gases in polymers within a wide range of temperatures and pressures.

EXPERIMENTAL DATA AND ARCHITECTURE

The experimental data used in this study were obtained from the literature. After the comprehensive evaluation of the experimental data, when we removed the redundant and invalid data points, a database containing 412 data points was finally established for the CSPSO-C RBF NN model. The gases were CO_2 and N_2, and the polymers were poly(butylene succinate) (PBS), poly(butylene succinate-*co*-adipate) (PBSA), PS, and polypropylene (PP). Table I shows the sources of statistical data used in this study.

The database was divided into five groups with different gases and polymers; these groups were N_2 in PS and CO_2 in PBS, PBSA, PS, and PP. Then, each group was randomly divided into three subsets including training, validation, and testing sets. To verify the network generalization, 292 data points (ca. 70%)

Table II. Topology Studies of the Optimal ANN Configuration

Hidden neuron	ARD	SD	R^2	Best fitness
3	0.3222	0.0601	0.9694	7.31×10^4
4	0.2853	0.0578	0.9698	2.14×10^4
5	0.2157	0.0489	0.9737	1.42×10^4
6	0.1715	0.0230	0.9772	1.38×10^4
7	0.1726	0.0223	0.9845	9.58×10^5
8	0.1312	0.0118	0.9925	9.80×10^6
9	0.1282	0.0115	0.9970	5.96×10^7
10	0.1425	0.0210	0.9864	9.75×10^5
11	0.1846	0.0347	0.9738	9.42×10^5
12	0.1799	0.0321	0.9739	9.23×10^5
13	0.2018	0.0523	0.9701	2.44×10^4
14	0.3241	0.0628	0.9697	9.18×10^5
15	0.4786	0.0662	0.9688	1.75×10^4

were used to train, validate, and give the test set 60 data points (ca. 15%) each.

The number of neurons in the input and output layers are defined by the system's properties. Therefore, the CSPSO-C RBF NN with one hidden layer, two inputs (temperature and pressure), and one output (solubility of gas in the polymer) is designed. The number of neurons in the hidden layer needed to be optimized. In this study, a heuristic was proposed with a previously prepared procedure; 13 CSPSO-C RBF NN models were generated with the assumption that the number of neurons in the hidden layer varied from 3 to 15. Table I shows the

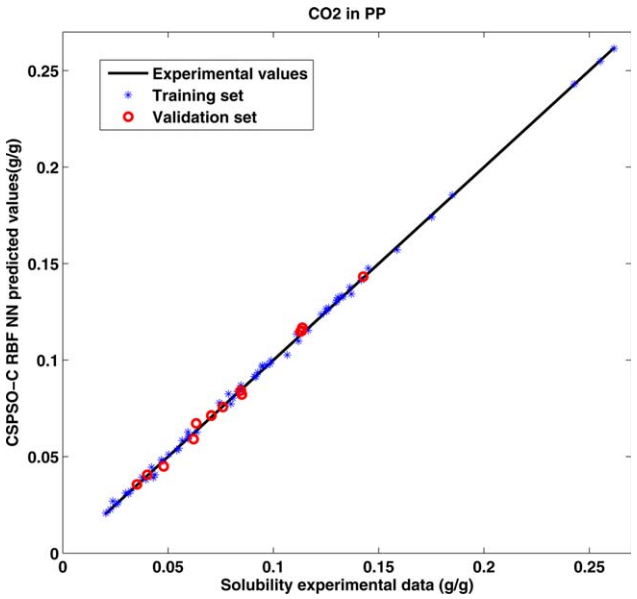

Figure 2. Predicted solubility of CO_2 in PP by the CSPSO-C RBF NN versus the experimental data. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

average relative deviation (ARD), standard deviation (SD), squared correlation coefficient (R^2), and best fitness calculated for the different network configurations. The network with the lowest ARD and SD and highest R^2 was selected. According to Table II, the hidden layer with nine neurons was selected for the CSPSO-C RBF NN.

In this study, the CSPSO-C RBF NN was designed for the solubility prediction of gases in polymers. The predictability was evaluated by the calculation of ARD, SD, and R^2. ARD and SD were defined as follows:

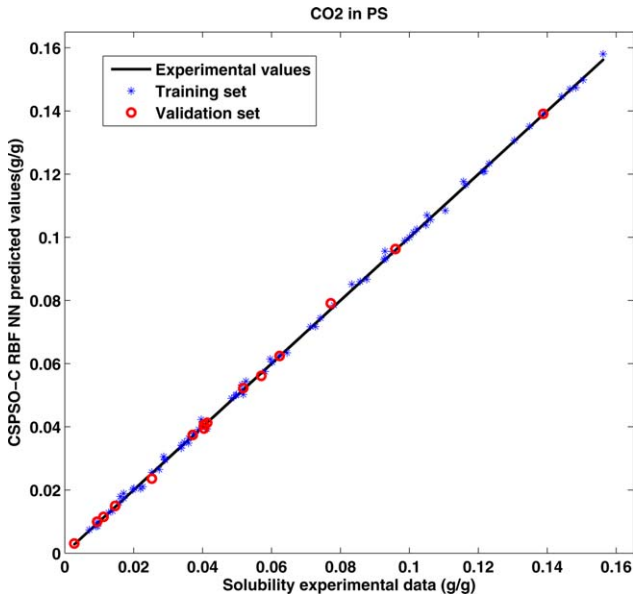

Figure 1. Predicted solubility of CO_2 in PS by the CSPSO-C RBF NN versus the experimental data. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

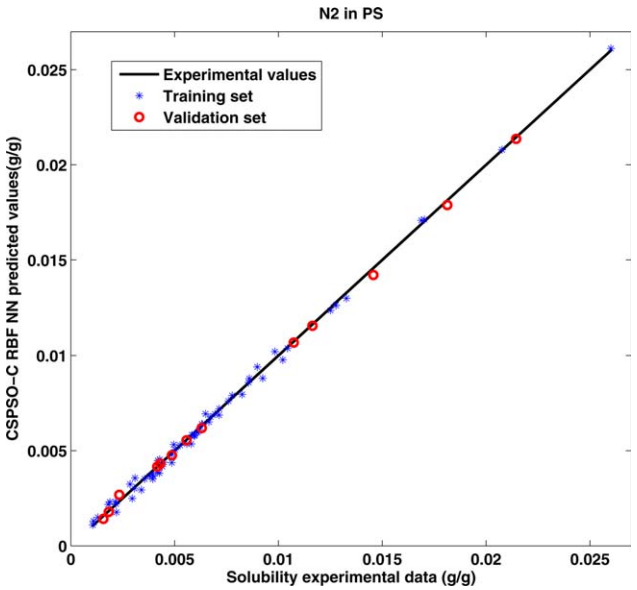

Figure 3. Predicted solubility of N_2 in PS by the CSPSO-C RBF NN versus the experimental data. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

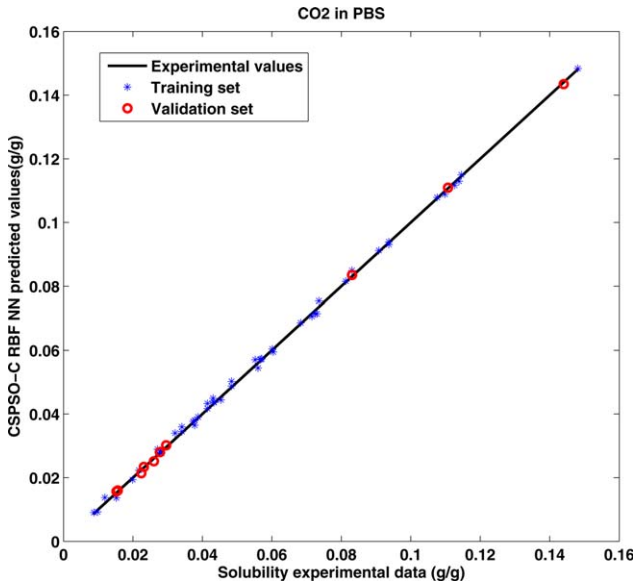

Figure 4. Predicted solubility of CO_2 in PBS by the CSPSO-C RBF NN versus the experimental data. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

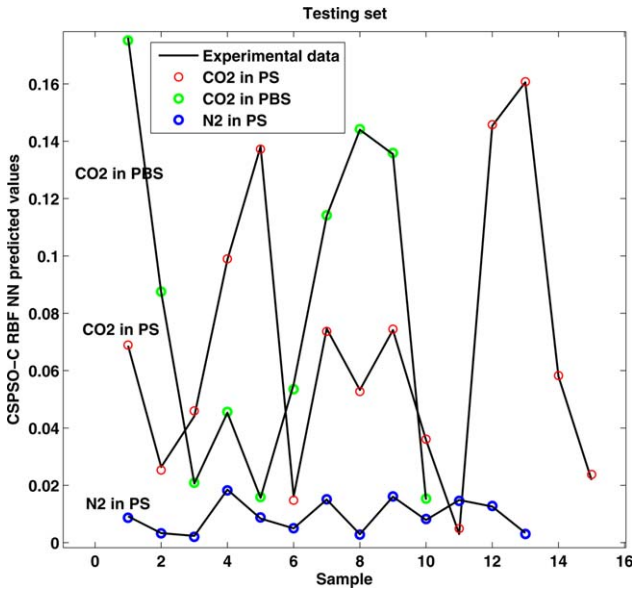

Figure 6. Comparison between the experiment and prediction in the testing set. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

$$\mathrm{ARD}=\frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}\frac{|\mathrm{Pr}\,e(\mathrm{i})-\mathrm{Exp}(\mathrm{i})|}{\mathrm{Exp}(\mathrm{i})}$$

$$\mathrm{SD}=\sqrt{\frac{1}{\mathrm{N}}\sum_{\mathrm{i}=1}^{\mathrm{N}}(x_i-x_o)^2}$$

where N is the number of data points, Pre(i) is the predicted value of the model, Exp(i) is the experimental value, and x_o is the average of N data points.

RESULTS AND DISCUSSION

An RBF NN combined with chaos theory, self-adaptive PSO algorithm, and clustering method, called simply the CSPSO-C RBF NN, was applied to predict the solubility of gases in polymers as a sample test in this study. The best network architecture was 2–9–1 (2 input units, 9 neurons in the hidden layer, and 1 output neuron). The CSPSO-C RBF NN was used to investigate the solubility of N_2 in PS and that of CO_2 in PBS, PBSA, PS, and PP. In Figures 1–5, the

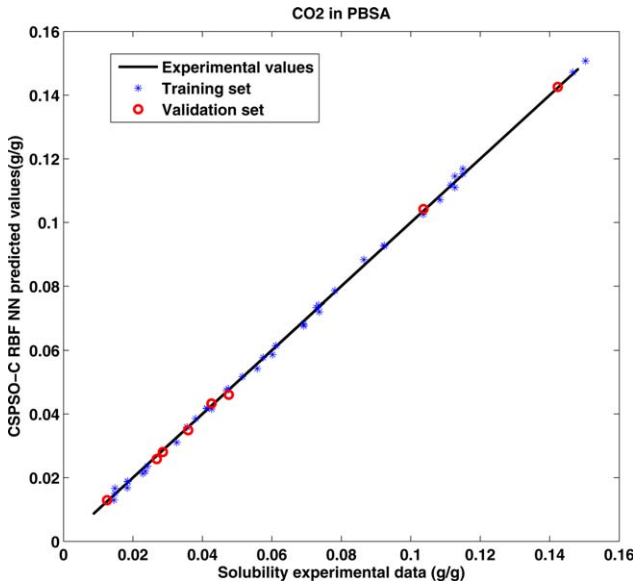

Figure 5. Predicted solubility of CO_2 in PBSA by the CSPSO-C RBF NN versus the experimental data. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

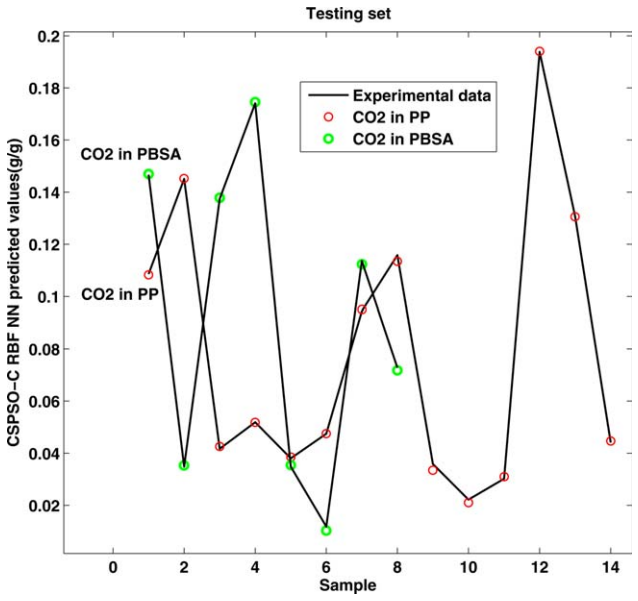

Figure 7. Comparison between the experiment and prediction in the testing set. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

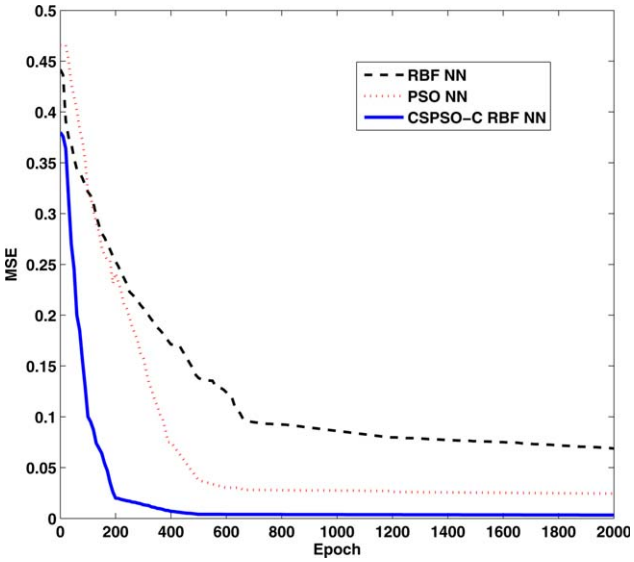

Figure 8. Curve of the mean square error versus the epoch. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

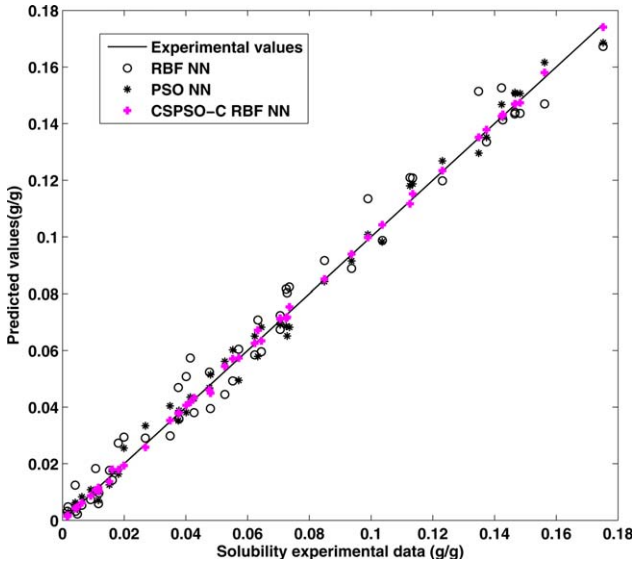

Figure 9. Predicted gas solubility in the polymers by the CSPSO-C RBF NN versus the experimental data. [Color figure can be viewed in the online issue, which is available at www.interscience.wiley.com.]

prediction of the solubility of N_2 in PS and that of CO_2 in PBS, PBSA, PS, and PP are plotted against the experimental data for the training and validation sets. The training set learned to fit the parameters, and the validation set was used to estimate the error rate to tune the model parameters. In these figures, the lines show the ideal modeling in which the prediction values were equal to the experimental data, whereas the asterisk and circle indicate the correlations between the experimental and predicted values in the training and validation sets, respectively.

As observed in these figures, the output of the CSPSO-C RBF NN model showed good agreement with the target, no matter the training set or the validation set. Particularly, for CO_2 in PP, as shown in Figure 2, it showed a better correlation between the predicted and experimental values, with a close proximity of the best linear fit to the perfect fit.

In Figures 6 and 7, the prediction of the solubility of N_2 in PS and that of CO_2 in PBS, PBSA, PS, and PP are plotted against the experimental data for the testing sets. As observed in these figures, the results showed the superiority of the CSPSO-C RBF NN. They could have also meant that the CSPSO-C RBF NN model had excellent prediction capability and good correlation with the experimental data.

In addition, the RBF NN and PSO NN were used as comparative models. The curves of the mean square error versus the epoch of different models are shown in Figure 8. As shown in Figure 8, the convergence speed of the proposed CSPSO-C RBF NN was faster than the others and proved its ability to avoid the premature convergence problem. In addition, to show that our proposed method outperformed the RBF NN and PSO NN, a database containing 50 data points (10 data points in each group) was also established to compare the different neural networks. The correlation between the predicted results and experimental values is illustrated in Figure 9. The simulation performance was evaluated by the calculation of ARD, SD, and R^2. Table III shows the results for these models. In addition, for CO_2 in PS, the ARD for ANFIS and the RBF NN proposed by Khajeh and Modarress[20] were 0.2543 and 0.6498, respectively,

Table III. Statistical Comparison Results from This Study

	RBF NN			PSO NN			CSPSO-C RBF NN		
Compounds	ARD	R^2	SD	ARD	R^2	SD	ARD	R^2	SD
PS/CO_2	0.3830	0.9566	0.0647	0.2520	0.9785	0.0413	0.1300	0.9971	0.0113
PP/CO_2	0.3378	0.9622	0.0523	0.2274	0.9801	0.0404	0.1107	0.9988	0.0107
PS/N_2	0.4012	0.9477	0.0718	0.2960	0.9686	0.0443	0.1370	0.9956	0.0125
PBS/CO_2	0.3976	0.9516	0.0721	0.2710	0.9712	0.0422	0.1340	0.9962	0.0121
$PBSA/CO_2$	0.3863	0.9524	0.0612	0.2680	0.9786	0.0410	0.1292	0.9973	0.0111
Average	0.3812	0.9541	0.0644	0.2629	0.9754	0.0418	0.1282	0.9970	0.0115

whereas in this study, this value was 0.1282. R^2 for the neural network trained by the unified PSO proposed by Ahmadi[24] was 0.99493, whereas in this study, it was 0.9970.

These results indicate that the performance of the CSPSO-C RBF NN was superior to those of the others. The statistical results of these models, shown by ARD and SD in Table III, indicated that the CSPSO-C RBF NN model was more accurate than the RBF NN and PSO NN models. Obviously, the strong correlations shown by R^2 indicated a satisfactory agreement between the experimental and predicted data.

In one word, the experiment results indicated that the CSPSO-C RBF NN model was better, faster, and more accurate and is a practicable method for the analysis and design of polymer process technology. Compared with the existing methods, the major reason for the superiority of the CSPSO-C RBF NN model was due to the hybrid training algorithm based on chaos theory, the self-adaptive PSO algorithm, and the clustering method. The advanced hybrid algorithm relied on the following three points with the purpose of avoiding premature convergence and accelerating the converging speed. One is the *k*-means method proposed to tune the hidden centers and function spreads of the RBF NN, one is the self-adaptive ω proposed for the balance between the exploration and exploitation of PSO, and the other is chaos theory, used to adapt the acceleration coefficients of the PSO algorithm.

CONCLUSIONS

The prediction of gas solubility in polymers by the ANN model has become the most vital subject in polymer processing. In this study, we proposed an RBF NN prediction model trained by a hybrid algorithm combined with a self-adaptive PSO algorithm, chaos theory, and a clustering method to predict the solubility of gases in polymers; we aimed to use this to replace costly and time-consuming measurements in laboratory. In short, this study indicated that the CSPSO-C RBF NN model for predicting the solubility of gases in polymers has good application prospects and is a useful tool for the analysis and design of polymer processes. In the future studies, we will follow up on this subject and focus on how to apply this method to solve more realistic problems all the time.

ACKNOWLEDGMENTS

The authors gratefully acknowledge the support of the National Natural Science Foundation of China (contract grant numbers 51163011, 61202313, and 51263015) and the Graduate Student Innovation Fund of Nanchang University (contract grant number cx2012011).

REFERENCES

1. Eslamimanesh, A.; Gharagheizi, F.; Mohammadi, A. H.; Richon, D. *Chem. Eng. Sci.* **2011**, *66*, 3039.
2. Skerget, M.; Mandzuka, Z.; Aionicesei, E.; Knez, Z.; Jese, R.; Znoj, B.; Venturini, P. *J. Supercrit. Fluids* **2010**, *51*, 306.
3. Nalawade, S. P.; Picchioni, F.; Janssen, L. *Prog. Polym. Sci.* **2006**, *31*, 19.
4. Sato, Y.; Fujiwara, K.; Takikawa, T.; Sumarno; Takishima, S.; Masuoka, H. *Fluid Phase Equilib.* **1999**, *162*, 261.
5. Sato, Y.; Yurugi, M.; Fujiwara, K.; Takishima, S.; Masuoka, H. *Fluid Phase Equilib.* **1996**, *125*, 129.
6. Hilic, S.; Boyer, S.; Padua, A.; Grolier, J. *J. Polym. Sci. Part B: Polym. Phys.* **2001**, *39*, 2063.
7. Bakhbakhi, Y. *Math. Comput. Model.* **2012**, *55*, 1932.
8. Lashkarbolooki, M.; Vaferi, B.; Rahimpour, M. R. *Fluid Phase Equilib.* **2011**, *308*, 35.
9. Gharagheizi, F.; Eslamimanesh, A.; Mohammadi, A. H.; Richon, D. *Ind. Eng. Chem. Res.* **2011**, *50*, 221.
10. Nasouri, K.; Bahrambeygi, H.; Rabbi, A.; Shoushtari, A. M.; Kaflou, A. *J. Appl. Polym. Sci.* **2012**, *126*, 127.
11. Mirzaei, E.; Amani, A.; Sarkar, S.; Saber, R.; Mohammadyani, D.; Faridi-Majidi, R. *J. Appl. Polym. Sci.* **2012**, *125*, 1910.
12. Faridi-Majidi, R.; Ziyadi, H.; Naderi, N.; Amani, A. *J. Appl. Polym. Sci.* **2012**, *124*, 1589.
13. Bakhbakhi, Y. *Expert Syst. Appl.* **2011**, *38*, 11355.
14. Perez-Blanco, M.; Hammons, J. R.; Danner, R. P. *J. Appl. Polym. Sci.* **2010**, *116*, 2359.
15. Li, J. B.; Liu, X. G. *J. Appl. Polym. Sci.* **2011**, *119*, 3093.
16. Chen, Y.; Huang, S. P.; Liu, Q. L.; Broadwell, I.; Zhu, A. M. *J. Appl. Polym. Sci.* **2011**, *120*, 1859.
17. Mehdizadeh, B.; Movagharnejad, K. *Fluid Phase Equilib.* **2011**, *303*, 40.
18. Li, J. B.; Liu, X. G.; Jiang, H. Q.; Xiao, Y. D. *J. Appl. Polym. Sci.* **2012**, *125*, 943.
19. Huang, M.; Liu, X.; Li, J. *J. Appl. Polym. Sci.* **2012**, *126*, 519.
20. Khajeh, A.; Modarress, H. *Expert Syst. Appl.* **2010**, *37*, 3070.
21. Wen-Jiang, X.; Ying-Zhi, G.; Dong-Yuan, G. *J. Comput.* **2012**, *7*, 1116.
22. Vadood, M.; Semnani, D.; Morshed, M. *J. Appl. Polym. Sci.* **2011**, *120*, 735.
23. Niaki, S.; Nasaji, S. A. *Int. J. Adv. Manuf. Tech.* **2011**, *56*, 777.
24. Ahmadi, M. A. *Fluid Phase Equilib.* **2012**, *314*, 46.
25. Lazzus, J. A.; Ponce, A.; Chilla, L. *Fluid Phase Equilib.* **2012**, *317*, 132.
26. Leung, S.; Tang, Y.; Wong, W. K. *Expert Syst. Appl.* **2012**, *39*, 395.
27. Sedki, A.; Ouazar, D. *Math. Model. Nat. Phenom.* **2010**, *5*, 132.
28. Zhang, Y. D.; Wu, L. A. *Sensors* **2011**, *11*, 4721.
29. Wang, H. S.; Wang, Y. N.; Wang, Y. C. *Expert Syst. Appl.* **2013**, *40*, 418.
30. Liu, X. G.; Zhao, C. Y. *AIChE J.* **2012**, *58*, 1194.
31. Xu, G. *Appl. Math. Comput.* **2013**, *219*, 4560.
32. Li, C. H.; Yang, S. X.; Nguyen, T. T. *IEEE Trans. Syst. Man Cybern. B Cybern.* **2012**, *42*, 627.
33. Hu, M. Q.; Wu, T.; Weir, J. D. *Inf. Sci.* **2012**, *213*, 68.

34. Wang, Y.; Li, B.; Weise, T.; Wang, J. Y.; Yuan, B.; Tian, Q. *J. Inf. Sci.* **2011**, *181*, 4515.

35. Nickabadi, A.; Ebadzadeh, M. M.; Safabakhsh, R. *Appl. Soft Comput.* **2011**, *11*, 3658.

36. Sun, C. L.; Zeng, J. C.; Pan, J. S.; Xue, S. D.; Jin, Y. C. *Inf. Sci.* **2013**, *221*, 355.

37. Sato, Y.; Takikawa, T.; Sorakubo, A.; Takishima, S.; Masuoka, H.; Imaizumi, M. *Ind. Eng. Chem. Res.* **2000**, *39*, 4813.

38. Khajeh, A.; Modarress, H.; Mohsen-Nia, M. *Iran. Polym. J.* **2007**, *16*, 759.

39. Lei, Z. G.; Ohyabu, H.; Sato, Y.; Inomata, H.; Smith, R. L. *J. Supercrit. Fluids* **2007**, *40*, 452.

40. Li, D. C.; Liu, T.; Zhao, L.; Yuan, W. K. *Ind. Eng. Chem. Res.* **2009**, *48*, 7117.

41. Sato, Y.; Takikawa, T.; Takishima, S.; Masuoka, H. *J. Supercrit. Fluids* **2001**, *19*, 187.

42. Newitt, D. M.; Weale, K. E. *J. Chem. Soc.* **1948**, *1039*, 1541.