PREDICTION OF THE LIQUID CRYSTALLINE BEHAVIOR FOR SOME BIS PHENIL AROMATIC DERIVATIVES WITH ARTIFICIAL NEURAL NETWORKS

Cătălin Lisa*, Victor Bulacovschi

Technical University „Gh. Asachi" Iaşi,
Faculty of Chemical Engineering and Environmental Protection,
Blvd. D. Mangeron 71, 700050 – Iaşi, Romania

*Corresponding author: clisa@ch.tuiasi.ro

Received: 09/02/2007
Accepted after revision: 21/03/2007

Abstract: Artificial neural networks (ANN) are robust and efficient mathematical tools inspired by the biological nervous system, and can be used to simulate a wide variety of complex scientific and engineering problems. A powerful ANN function is determined largely by the interconnections between artificial neurons, similar to those occurring in their natural counterparts of biological systems. Neural network based method proved to be able to appreciate the liquid crystalline behavior with small errors, so it represents an effective tool for structure – properties prediction. The most common type ANN with multiple layers has been used in this work.

Keywords: *liquid crystal properties, prediction, feed forward neural networks*

INTRODUCTION

Over the past few years, liquid crystalline materials have been studied extensively and have gained great importance due to their unique mechanical, rheological and optical properties [1-3]. Their special properties have been successfully employed, for example, in information technology. The use of neural networks to the prediction of compound properties has as main advantage the fact that they can simulate the nonlinear relationship between structural information and properties of compounds during the training process, and generalize the knowledge among homologous series without need for theoretical formulas. The ability of neural network is significant in determination quantitative structure-property relationship, because compounds with known properties can be used to train ANN, so that, subsequently, properties of other compounds that can not be ascertained by experimentation can be determined [4-8].
In this paper we used an organic compounds database [9] (250 in all), which includes a wide variety of compounds: bis aromatic types containing connecting groups in the rigid core as azo, azomethine or double bond.

METHODS

Computational neural networks provide powerful tools for modeling of materials. In the chemical sciences, the use of computational neural networks has rapidly increased over the past 10 years. It is the goal of this paper to presents results on the use artificial neural network (ANN) in material science as a method for making accurate predictions of organic compounds properties based on their molecular structure and for designing molecular – based compounds that have specified properties in future.
An ANN is composed of simple nonlinear elements operating in parallel and interconnected [10-12]. These elements are called artificial neurons, processing elements, nodes or units, and they are inspired by the biological nervous. An ANN is composed of several layers of neurons: an input layer, one or more hidden layer, and an output layer. Each layer of neurons receives its input from the previous layer or from the network input, and the output of each neuron feeds the next layer or the output of the network. The way in which these nodes are interconnected and distributed in several layers determines the network architecture and its applications. For example, figure 1 shows a scheme of an ANN with three layer, four neurons in the input layer, three neurons in the hidden and two output neurons (4-3-2).
As in nature, the network function is determined largely by the connection between elements. The network stores the information in the strength of the neuron interconnection; therefore we can train an ANN to perform a particular function or model by adjusting the values of these connections between elements. So, the architecture of an ANN consists of a description of how many layers the network has, the number of neurons in each layer, each layer's transfer function and how the layers are connected each to other. The best architecture to be use depends on the type of problem represented by the network. A large number of neurons in the hidden layer produces that the network required less iterations in the training, however, each of these steps will be slow. The number of neurons in the hidden layer has not to be the same that input layer, in order to avoid the memorizing of all the inputs data, instead of

generalize from the individual cases. The size of the input and output levels is given for the nature of the problem and for the degree of goodness that is expected for the network [13-15].

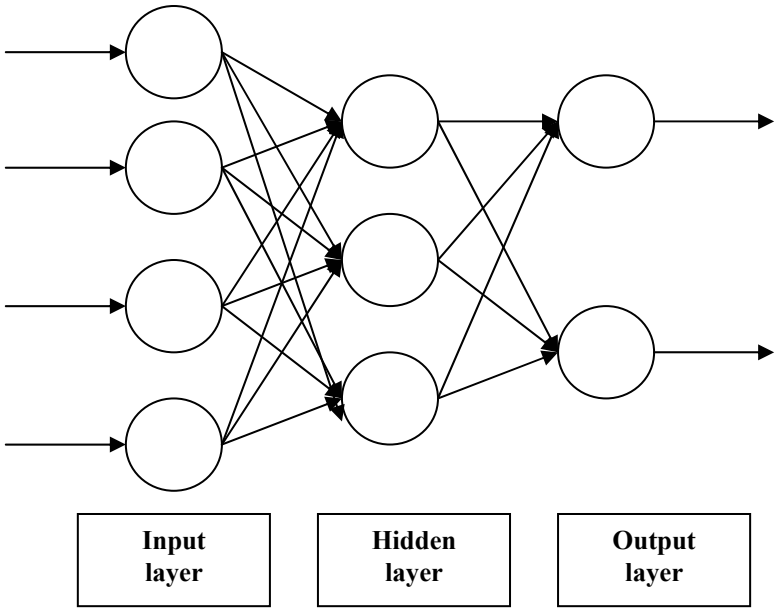


Figure 1. *A simple 4-3-2 neural network*

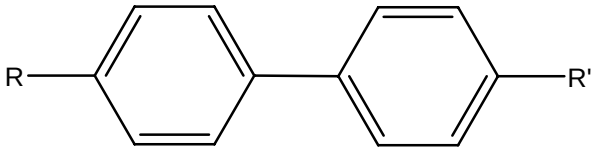
An ANN has to be trained before using it. The learning rule or training algorithm is a procedure for modifying the weights of the connections between the neurons of the network, and applied in order to train the network to perform some particular task. In supervised learning, the learning rule is provided with a set of examples (input and target vectors) of proper network behavior. As the inputs are applied to the network, the networks are compared with the targets by means of a performance function. The learning rule is then used to adjust the weights of the network in order to move the network outputs closer to the targets. In unsupervised learning, the weights are modified in response to network inputs only. There are no target outputs available. Most of these algorithms perform clustering operations, and categorize the input patterns into a finite number of classes. This is especially useful in such application as vector quantizations, recognition of standards such as voice or images, etc.

One of the problems that occur during neural network training is called overfitting, as explained before. The error on the training set is driven to a very small value, but when new data are presented to the network the error is large. The network has memorized the training examples, but it has not learned to generalize to new situations. One method for improving network generalization is to use a network that is just large enough to provide an adequate fit. Unfortunately, it is difficult to know beforehand how large a network should be for a specific application. There are early stopping methods for improving generalization. In this technique, the available data are divided into three subsets. The first subset is the training set, which is used for computing the gradient and updating the network weights and biases. The second subset is the validation set. The error on the validation set is monitored during the training process. The validation error will normally decrease during the initial phase of training, as does the training set error.

However, when the network begins to overfit the data, the error on the validation set will typically begin to rise. When the validation error increases for a specified number of iterations, the training is stopped, and the weight and biases at the minimum of the validation error are returned.
The input sets used for the training are, which determine what the network will be able to solve. It is necessary that input sets were representative for the problem; in this way, the network will be able to answer satisfactorily to any kind of input. Another important fact concerning the ANNs it that the network training process is necessary only once, in such a way that ANN trained could be used to solve a new situation very fast [16].
In this work, we have used the most common type ANN with multiple layers and supervised learning called feed-forward network. In this network, each neuron has several inputs and one output, and uses a nonlinear differentiable transfer function.

RESULTS AND DISCUSSIONS

In this paper we used an organic compounds database (250 in all), which includes a wide variety of bis aromatic types compounds (scheme 1).



Scheme 1. *The general structures of the analyzed compounds*

The organic compounds have similar structures with small structural changes that allow a systematical analysis of the factors that influences liquid crystals properties and determination of some parameters that will be used in prediction with neural networks. Our database contains compounds with different units connected to the aromatic core such as CN, Br, variable length alkyl chains, ketones by means of ester or ether linking group.
The first step of the study was to establish the "input" parameters for the ANN algorithms. The parameters considered as inputs of the neural model were: length of the rigid core ($\mathbf{L_{rig}}$), length of the flexible core ($\mathbf{L_{flex}}$), the asymmetry parameter (**S**) and molecular weight (**M**), evaluated using modeling simulation. The length of the flexible and the rigid core and the asymmetry parameters were estimated by mechanical molecular simulation using Hyperchem program.
"**S**" represents the ratio between the length and the diameter of the polymer single chain. All the above enumerated parameters are considered as inputs of the neural networks because they are the most important structural characteristics that impose a liquid crystalline behavior.
Concerning the liquid crystal behavior (LC), we have coded with "1" the possibility to generate a mesophase and with "0" the crystalline or amorphous phases. This is the output variable for neural model.
Firstly the experimental data were split into training and validation sets, with 198 and 52 experimental points, respectively.

A special software application - *NeuroSolution* - was used in this paper in order to design and obtain predictions of neural networks. In this program, the following specifications are necessary: the network type, the input and desired output values, the stop condition of the training, the number of processing elements in hidden layers, the activation functions, the learning rule, the maximum number of epochs and some configuration parameters to display the neural model development. We developed and trained many networks, changing the above options, and then we selected the best one that balances the size and the performance.

In this study a number of different neural network architecture with **4** inputs, single, two or three hidden layers with **10**…**40** intermediate neurons and **1** output (table 1) were used since a number of different experiments were performed.

Table 1. *Different topologies tested for the feed forward neural networks*

No.	Network topology	MSE	r	E_p(%)
1.	MLP(4:10:1)	0.0033	0.990	0.102
2.	MLP(4:15:1)	0.0049	0.987	0.146
3.	MLP(4:20:1)	0.0051	0.986	0.188
4.	MLP(4:14:1)	0.0055	0.985	0.217
5.	MLP(4:17:1)	0.0050	0.986	0.189
6.	MLP(4:16:1)	0.0050	0.986	0.199
7.	MLP(4:10:10:1)	0.00094	0.996	0.112
8.	MLP(4:15:10:1)	0.0016	0.993	0.144
9.	MLP(4:15:15:1)	0.00090	0.996	0.093
10.	MLP(4:20:20:1)	0.00060	0.997	0.065
11.	MLP(4:25:25:1)	0.0010	0.996	0.059
12	MLP(4:30:30:1)	0.00070	0.9973	0.073
13.	MLP(4:35:30:1)	0.00056	0.9978	0.057
14.	MLP(4:25:20:1)	0.00098	0.9964	0.064
15.	MLP(4:35:35:1)	0.00045	0.9982	0.046
16.	MLP(4:40:35:1)	0.00046	0.982	0.045
17.	MLP(4:40:40:1)	0.000064	0.9997	0.0087
18.	MLP(4:10:10:10:1)	0.00034	0.9985	0.077
19.	MLP(4:20:20:20:1)	0.00035	0.9986	0.045
20.	MLP(4:30:30:30:1)	0.000012	0.99995	0.0036

The best network topology was determined based upon the mean squared errors (MSE) on the training data. The training phase was considered complete when the error of all the training patterns was less than a pre-specified error criterion or a maximum number of epochs (iterations) had been reached. If, after the entire set of training patterns was presented, the overall error was still unacceptable, the neural network would be returned to the beginning of the training patterns and the process would be repeated. So, the training is considered terminated at the point where network error (MSE) becomes sufficiently small (<0.001).

The mean squared error was computed using the following formula [17]:

$$MSE = \frac{\sum_{j=1}^{P}\sum_{i=1}^{N}(d_{ij} - y_{ij})^2}{N \cdot P} \qquad (1)$$

where P is the number of output processing elements (in this case, P = 1), N is the number of exemplars in the data set, y_{ij} is the network output for exemplar I at processing element j, and d_{ij} is the desired output for exemplar I at processing element j.

Good predictions are obtained with the neural models MLP (4:10:1), MLP (4:40:40:1) and MLP (4:30:30:30:1) on training data. Table 2 presents the predictions of MLP(4:40:40:1) on training data. Two columns named "LC net" appear in table 2: the first contains the real predictions of the network and the second the rounded result to correspond to experimental data.

Table 2. *Prediction of MLP(4:40:40:1) on training data*

L_{rig}	L_{flex}	S	M	LC exp	LC net	LC net round
10.0	16.3	0.118	383	0	-0.00061	0
10.0	14.8	0.118	383	0	-0.0012	0
9.9	17.3	0.112	411	0	0.002395	0
11.1	12.8	0.114	350	1	0.999746	1
10.0	22.7	0.093	509	0	-0.00048	0
10.4	5.1	0.172	319	0	0.002685	0
10.0	17.2	0.114	427	0	0.004448	0
10.0	19.2	0.118	453	0	0.00223	0
9.4	19.7	0.143	338	0	0.000649	0
10.0	25.2	0.087	481	0	-0.00103	0
10.0	15.6	0.129	411	1	0.999316	1
10.0	16.8	0.125	425	0	-0.00187	0
10.3	9.5	0.142	314	0	0.002273	0
11.1	11.4	0.126	389	0	0.001033	0
10.0	26.5	0.086	495	0	-0.00036	0
10.0	15.8	0.135	411	0	0.002002	0
10.0	30.4	0.078	537	0	0.000264	0
10.0	24.6	0.101	525	0	-0.000098	0
10.0	11.9	0.154	354	0	-0.00022	0

Table 3 presents some predictions of the neural models to previously unseen data (not used in the training phase, so "unseen" data for the networks). In Table 3 one can also see the nominal and numerical codifications of the inputs and output for the networks. Cells marked in gray represent wrong predictions of the networks. For the MLP(4:10:1) the probability of a correct answer was 92.31 %, for the MLP(4:40:40:1) the probability of a correct answer was 94.23 %, and for MLP(4:30:30:30:1) was 96.15 %, that is a good performance of the designed networks. Consequently, a feed-forward network MLP(4:30:30:30:1) can predict satisfactory the LC behavior of the some bis phenil aromatic derivatives.

Table 3. *Validation of the neural models*

L_{rig}	L_{flex}	S	M	MLP(4:10:1)		MLP(4:40:40:1)		MLP(4:30:30:30:1)	
				LC exp	LC net	LC exp	LC net	LC exp	LC net
10.0	17.6	0.108	397	0	0	0	1	0	0
8.6	10.6	0.191	327	0	0	0	0	0	0
10.0	15.8	0.112	369	1	0	1	0	1	0
10.0	12.5	0.129	340	0	0	0	0	0	0
10.0	26.6	0.086	495	0	0	0	0	0	0
9.7	8.9	0.142	291	1	0	1	1	1	1
10.0	18.9	0.109	411	0	0	0	0	0	0
10.0	15.1	0.121	369	0	0	0	0	0	0
10.0	29.0	0.079	551	0	0	0	0	0	0
10.0	20.9	0.092	481	0	0	0	0	0	0
7.2	19.3	0.118	445	0	0	0	0	0	0
10.0	20.2	0.101	425	0	0	0	0	0	0
10.0	15.6	0.128	397	0	1	0	1	0	1
10.0	20.2	0.103	425	0	0	0	0	0	0
10.0	18.9	0.107	411	0	0	0	0	0	0
10.0	29.1	0.081	579	0	0	0	0	0	0
10.7	2.6	0.196	291	0	0	0	0	0	0
10.0	20.2	0.107	425	0	0	0	0	0	0
10.0	18.9	0.109	411	0	0	0	0	0	0
7.2	21.8	0.107	473	0	0	0	0	0	0
8.6	14.4	0.185	355	0	0	0	0	0	0
10.0	10.0	0.145	312	0	0	0	0	0	0
10.0	20.8	0.152	525	0	0	0	0	0	0
10.0	19.0	0.130	483	0	0	0	0	0	0
7.2	11.6	0.147	294	0	1	0	0	0	0

CONCLUSIONS

Conventional methods to model the properties of the organic compounds involve a step by step procedures based on a complex mathematical model with large time and memory complexity. Hence the application of ANNs offers a simple yet effective route to model complex systems with out much of mathematical complexities and hence large error. Further advantage with ANNs is that is has an automated learning process by the adjustment of the synaptic weights. Hence like the human brain it can learn seeing the trends in the data and hence can find solutions. Hence complex proprieties like creep, fatigue and wear where there is no rigid mathematical model, can be easily modeled without much complication, a further advantage being the saving of valuable experimentation time. It has hence been found out very good results have been obtained using the ANN approach.

The ideal conditions concluded for the use of ANNs in organic compound property prediction are:

- A large training dataset is always useful for good predictions.
- A fast training algorithm is more suited for industrial applications

- The more complex the non-linear relation between the input and output, the larger is the training dataset required.
- The analysis of relationships between simple and complex properties provide additional help in prediction of data.

This work has shown the excellent capability of an ANN approach for the prediction liquid crystalline behavior of some bis phenil aromatic derivatives. Simple architecture neural networks and simple methods of establishing the networks' structure are proposed for process modeling: feed-forward networks with a single, two or three hidden layers. For the MLP(4:30:30:30:1) the probability of a correct answer was 96.15 %, that is a good performance of the designed networks. Consequently, a feed-forward network MLP(4:30:30:30:1) can predict satisfactory the LC behavior of the some bis phenil aromatic derivatives.

A direct neural network modeling that means prediction of proprieties as function of structural parameters will be completed with a inverse modeling procedure in order to appreciate the structures that lead to imposed characteristics.

REFERENCES

1. Imrie C.T., Loubser C., Engelbrecht P., McCleland C.W., Zheng Y.F.: *J. Organomet. Chem*., **2003**, **48**, 665.
2. Kleinoder, T., Spycher, S., Yan, A. *Prediction of properties of compounds,* Wiley-VCH, Weinheim, **2003**.
3. Neubert, M. : Chemical Structure – Property relationships, *in : Liquid crystals experimental study of physical properties and phase transitions* (Satyedra Kumar, ed.), Cambridge University Press, **2001**.
4. Maranas, C.D. : *AIChE Journal*, **1997**, **43**(5), 1250.
5. Landrum J., Penzotti J., Putta S.: *Meas. Sci. Technol.*, **2005**, **16**, 270.
6. Bhadeshia, H.K.D.H. : *ISIJ International*, **1999**, **10**, 966.
7. Zhang, Z., Friedrich, K. : *Composites Science and Technology*, **2003**, **63**, 2029.
8. Zhenyu, J., Zhang, Z., Friedrich, K. : *Composites Science and Technology*, **2006**, in press.
9. * * * LiqCryst Online, Liquid Crystal Group Hamburg.
10. Lisa, G., Curteanu, S.: *Studii si cercetari stiintifice, Univ. Bacau*, **2005**, **VI**(1), 65.
11. Curteanu S., Lisa G., *Buletinul Institutului Politehnic din Iasi,* **2004**, **XL** (3-4), 96.
12. Lisa, C., Apreutesei, D., Lisa, G., Bulacovschi, V., Curteanu, S. : Structural parameters modeling of some ferrocene liquid crystals by using HyperChem software, *The International Conference of Physical Chemistry, ROMPHYSCHEM-12*, Bucuresti, September 6-8, **2006**.
13. Stephanopoulos, G., Han, C.: *Computers & Chemical Engineering*, **1996**, **20**(6-7), 743.
14. Wilson, J.A., Zorzetto,L.F.M.: *Computers & Chemical Engineering*, **1997**, **21**(9) 951.
15. Roy, N.K., Potter, W.D., Landau, D.P.: *IEEE Transactions on Neural Networks*, **2006**, **17**, 1001.
16. Leon, F., Curteanu, S., Lisa, C., Hurduc, N. : *Molecular Crystals and Liquid Crystals*, **2006**, in press.
17. Gaster, J., Zupan. J. : *Chem. Int. Ed. Engl*., **1993**, **32**, 503.