Predicting the transition temperature of smectic liquid crystalline compounds from their structure using artificial neural networks

Rachel Schröder, Helge Kränz, Volkmar Vill and Bernd Meyer *

Institute for Organic Chemistry, University of Hamburg, Martin-Luther-King-Platz 6, 20146 Hamburg, Germany

The derivation of material properties of chemical compounds directly from their chemical structure can be used to plan chemical syntheses more efficiently. Here, feed forward–back propagation neural networks are devised to predict the transition temperatures of smectic liquid crystalline compounds based on a set of data that contains 6304 different structural patterns. The trained networks were tested with 1575 smectic liquid crystalline compounds that the networks had not seen before. Four different network architectures were trained to predict the transition temperatures. All networks had the capability to predict a significant portion of the transition temperatures with small deviations. The network with 10 hidden neurons and one output neuron has a high recognition rate and predicts the transition temperatures of about 85% of the structures unknown to the network with an error of ≤20 °C. In contrast, the network with 100 hidden neurons and 370 output neurons makes more precise predictions of the transition temperatures indicated by a low standard deviation of 14.3 °C and by the fact that only 8.3% of the tested structures produced an error of more than 20 °C. However, the latter network gives answers only for 79.4% of the structures in the test set.

Introduction

There are two main approaches to predicting unknown data from a set of known data. First, classical statistical methods (extrapolation, *etc.*) can be used and, second, artificial intelligence (neural networks, *etc.*) can be employed to predict the unknown data. Neural networks have been used to solve a great variety of chemical tasks,[1] such as recognition of NMR[2–4] and mass spectra,[5] computation of electrostatic charges on a molecular surface,[6] and deriving material properties directly from their chemical structures.[7,8] Gakh *et al.* used neural networks to predict material properties using six thermodynamic parameters from alkanes ranging from 6 to 10 C-atoms[7] based on a graph theoretical approach to encode the structures. Using descriptors accessible through molecular modelling of each compound a neural network was shown to predict boiling points and critical temperatures.[8] Here we show that neural networks can be used to predict the transition temperatures of liquid crystalline compounds that form a smectic A phase directly from their chemical structure.

Neural networks are well suited to predict non-linearly dependent relationships between input and output data. Because there is no linear relationship between chemical structures and their associated transition temperatures a feed forward–back propagation neural network was chosen for this kind of task.[9] Such a network consists of several neurons that are grouped into usually three hierarchical layers (Fig. 1). The first layer is the input layer on which the information is presented to the network, *i.e.* in our case the chemical structure, the second is the hidden layer that is used for computing results from the information in the first layer, reformulating it and passing it on to the third layer. The third layer is the output layer that represents these results, *i.e.* the transition temperature. Each neuron of a given layer is connected with each neuron of the hierarchically higher and lower layers but never with neurons within its own layer. Also, no direct connection between the input and the output layer is encoded. The information presented to the input layer is passed along the connections, *i.e.* the weights, to the hidden layer and then further to the output layer. This type of neural network can be trained by comparing the computed results with the target results and adjusting all connections accordingly. Due to this adjustment the network will be able to recognize the information better the next time it is presented to the input layer. This procedure must be repeated for each pattern of the set used to train the network. One presentation of all patterns of the training set to the network is called a training cycle.

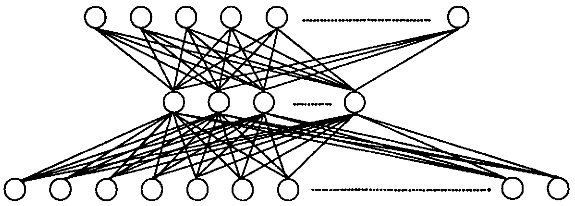

Fig. 1 The top layer is the input layer each neuron of which is connected to each neuron of the middle, the hidden, layer. Each neuron of that layer is connected to each of the bottom, the output, layer. The connections between the neurons, *i.e.* the weights, contain the information of a trained neural network. Each neuron of the hidden and output layer is a simple processing unit that sums up its input and processes this sum through a sigmoid function to produce an output.

Methods

Before the network can start to learn, a way must be found to encode the chemical structure into the input layer. The representation of a chemical structure has to be mapped onto the linear array of input neurons. An encoding of a chemical structure must have two properties: first the encoding must be translationally invariant and, second, it has to be unambiguous in two ways: two structures must not have the same representation and each structure must have only one representation. Therefore, we encoded the chemical structures in the following way: each structure is composed of nine fragments (see Fig. 2). Each fragment has a fixed number of possible substructures and is represented by a corresponding number of input neurons (see Table 1). So, if a fragment is present, the neuron corresponding to this substructure in the given fragment is turned 'on' (= 1) while all other neurons corresponding to this fragment are set to 'off' (= 0). The combination of the encoding of all fragments present in the chemical structure results in a binary pattern that represents the

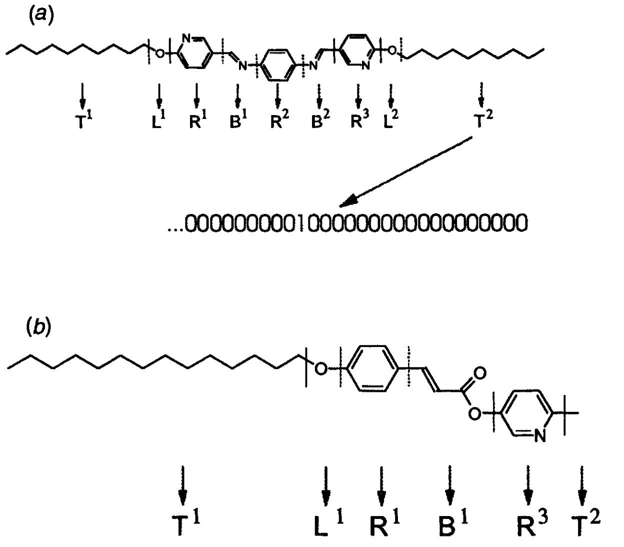

Fig. 2 Panel (*a*) shows an example of a smectic liquid crystalline compound. The compound is composed of nine fragments (T^1, L^1 *etc.*). In this structure each fragment is occupied by a substructure. The structure encoding is exemplified by the fragment T^2. The decyl residue in the T^2 fragment is the tenth out of 28 possible substructures occurring at this position in the molecule. The rightmost 28 neurons of the input layer encode this part, *i.e.* T^2, of the molecule. 27 of these neurons hold a value of 0 except for the tenth neuron that is set to 1. Panel (*b*) shows an example for a structure that does not contain all fragments; the fragments R^2, B^2 and L^2 are not used and, therefore, the corresponding input neurons are set to 0.

Table 1 Division of the molecules into nine fragments. If a fragment is occupied by a substructure the neuron corresponding to this substructure in this fragment will be set to 'on', while all other neurons representing this fragment are 'off'. If a fragment is not present in a given molecule all neurons of the input layer corresponding to this fragment will be 'off'. The description of the fragments is depicted in Fig. 2

Long fragment name	Fragment abbreviation	Number of substructures represented in data set[a]
First terminal	T^1	28
First linking	L^1	6
First ring	R^1	33
First bridge	B^1	19
Second ring	R^2	33
Second bridge	B^2	19
Third ring	R^3	33
Second linking	L^2	6
Second terminal	T^2	28

[a] Number of different fragments used in the training and testing set of the neural networks.

Table 2 Architecture and training parameters for all networks

	Number of neurons in the layer				Computation time[a] *t*/min	
Name	Input	Hidden	Output	No. of cycles[b]	Per 1000 cycles[c]	Total[d]
N10/1	205	10	1	70 000	13	900
N10/370	205	10	370	26 250	93	1 949
N100/1	205	100	1	70 000	92	6 440
N100/370	205	100	370	20 000	310	6 200

[a] Obviously, the greater the number of neurons in the hidden and the output layer the less cycles can be computed per minute. The times were recorded on a Fujitsu UXP/M (S100) and are equivalent to up to 8.6 × 10^6 weight updates per second. The number of connections varies between 2060 (network N10/1) and 57 500 (network N100/370). [b] Number of cycles used to train each network. [c] Training time per 1000 cycles. [d] Total training time.

chemical structure as series of 0 and 1 values to the input layer of the neural network. One advantage of this encoding is its simple implementation. However, as a drawback, this way of encoding chemical structures is different for each class of compounds. If the molecules have no generic direction of presentation, *e.g.* no head or tail groups, the chemical structures have to be presented to the network in two orientations, *i.e.* from left to right and *vice versa*. The representation of the structures of smectic liquid crystalline compounds chosen here does not encompass all possible smectic liquid crystalline structures. However, more than 10^{11} different structures can be entered into the network using this encoding.

Theoretically, one can think of another possible way of encoding chemical structures by mapping each fragment to one neuron which can take a value between 0 (fragment is absent) up to the number of possible substructures for this fragment. In this encoding similarity of the code numbers would imply similarity of the chemical substructures. An attempt to train a neural network with this encoding did not lead to convergence.

We have developed two different ways to map the temperature range to the output neurons. The temperature range in our training set ranged from the minimum transition temperature, $T_{\min} = 46$ °C, to the maximum transition temperature, $T_{\max} = 396$ °C. In the first approach, the temperature range was mapped to the interval from 0 to 1, *i.e.* the output range of one neuron. The value x of the output neuron is obtained from the corresponding temperature T as shown in eqn. (1). In this case only one output neuron is needed

$$x = (T - T_{\min})/(T_{\max} - T_{\min}) \qquad (1)$$

and results in an analogue representation of the temperature. In the second approach, we used $(T_{\max} - T_{\min}) + 1$ neurons on the output layer such that the temperature is displayed like a digital thermometer. If the transition temperature is T, the first $(T - T_{\min})$ neurons are set to a value of '0.9' (the on-state) while the rest is set to a value of '0.1' (the off-state).

Finally, the number of neurons in the hidden layer had to be set. We tested two different numbers of hidden neurons of 10 and 100 each with the respective number of output neurons. This led to a total of four networks that were trained (see Table 2). We had a data set of 7879 different structural patterns generated from the chemical structures,[10,11] for which a smectic A phase exists, and their associated transition temperatures. This set was divided randomly in a ratio of 4 : 1 into two subsets, the training set containing 6304 patterns and their transition temperatures and the test set with 1575 patterns with their associated transition temperatures. The learning rate, which determines the speed and accuracy of learning, was initially set to 2.0 and subsequently decremented to 0.01 to reach a stable state of the network (*cf.* Fig. 3). The momentum, which specifies to what extent the previous search direction is retained in the current search direction, remained constant at 0.5.

After training was completed, each network was tested with the patterns of the test set unknown to the network. For networks N10/1 and N100/1 with just one neuron on the output layer the transition temperature is obtained from the value of the output neuron as shown in eqn. (2), where x is the value of

$$T = T_{\min} + x(T_{\max} - T_{\min}) \qquad (2)$$

the output neuron. If multiple output neurons are used as in networks N10/370 and N100/370 the neurons should ideally result in a step function (Fig. 4). Because of the large number of patterns in the test set an algorithm was devised to compute automatically the transition temperature from the values of the neurons on the output layer. The automatic interpretation of the output in networks with more than one output neuron was

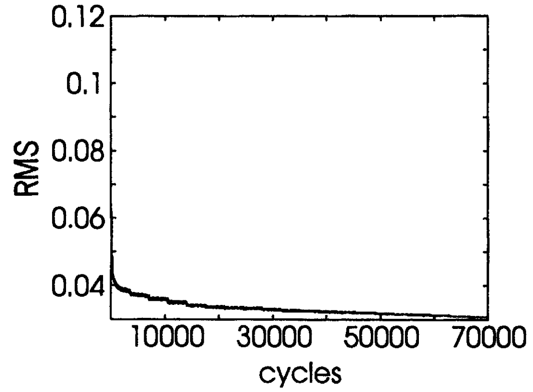

Fig. 3 Progress of the training of network N10/1 as measured by the RMS obtained from the patterns of the training set. The periodic steps in the curve originate from the decrease of the learning rate at these points. After each step the RMS remains nearly constant. When the learning rate gets too low, no further significant improvements were observed and the training was terminated.

done in two steps: first when the sequence of the output neurons from left to right is interpreted as a trajectory, all transitions over the values of 0.7, 0.6, 0.5, 0.4 and 0.3 were counted. If exactly one transition over each threshold exists, the transition temperature was obtained counting from the left those output neurons that have a value of greater than 0.5. For these patterns the standard deviation of the predicted from the actual transition temperature was calculated. Second, all patterns which had more than one transition over any of the values of 0.7, 0.6, 0.5, 0.4 and 0.3 were tested if the difference between x values of the first transition over 0.7 and the last over 0.3 was less than twice the standard deviation obtained in the first step. In this case the average of the two temperatures at the first crossing over 0.7 and the last crossing over 0.3 was taken as the transition temperature.

Results

We used two different ways to measure the quality of prediction of the networks. First, we grouped the differences between predicted and experimental transition temperatures in several intervals, *e.g.* of less than 5 °C, and counted the number of structures that were predicted within these boundaries (Table 3). Secondly, we computed the RMS (root mean square) of the difference between the predicted and measured transition temperatures. Fig. 5 shows an example of both quality tests that were obtained during the training procedure by testing the status of the networks with an independent test set after intervals of 200 training cycles each. In contrast, Fig. 3 shows the decline of the RMS of the training set itself during the training phase. By comparing Figs. 3 and 5 it is obvious that the RMS of the training set is not a good indicator of the quality of the network because the testing with data unknown to the network improves significantly after the RMS of the training set has converged. On the other hand, the RMS curve using the training set is a good indicator of the quality of the representation of the training set in the neural network. If there are only small changes of the RMS of the training patterns the learning rate can be reduced to improve the speed of the reduction of the RMS value, *i.e.* the steps in the curve in Fig. 3.

Fig. 6 shows the distribution of the errors of the test set for two networks at the end of the training. The networks predicted the transition temperatures of ≈40% of the unknown structures with absolute errors of ⩽5 °C and of 71% to 85% with absolute errors of ⩽20 °C (*cf.* Table 3). It is not possible to make an overall decision on which network has the best prediction capabilities. Network N100/370 provides the best results if an exact prediction of the transition temperatures is desired: the transition temperature of 42.4% of the pattern unknown to it were predicted with an error of less than ± 5 °C. If one takes as a reference for 100% not all structures presented to the network during testing but only those that were recognized, *i.e.* 79.4% of the total, the value of 42.4% improves to 53%. Thus, more than half of the predicted transition temperatures have an error of less than ± 5 °C. Furthermore, the same network has relatively few prediction errors of greater than 20 °C, *i.e.* 8.3%, while for other networks this value is found between 15% and 23%. If a network is desired that has a very high prediction rate network N10/1 should be used.

The reasons for deviations of the predicted from the actual temperatures may be twofold. First, there will most likely be

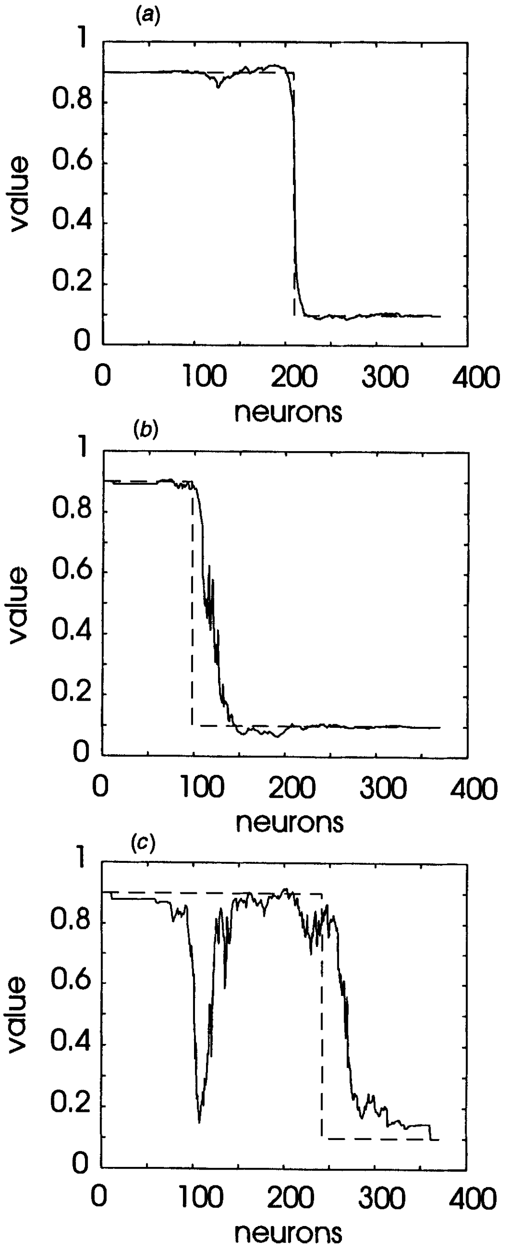

Fig. 4 The three panels show the response of the trained network N10/370 using patterns of the test data set. The dashed line in each panel indicates the expected curve (step function) and the solid line represents the curve obtained from the neural network. The panel at the top (*a*) shows a very good agreement between the calculated curve and the expected value with only one transition (step) in the network's response. This type of curve is obtained for 96.1% of the test patterns. The second panel (*b*) shows a curve with multiple transitions within a narrow range. The transitions are separated by less than twice the standard deviation obtained from network responses similar to that shown in the top panel. This type of curve is found for 2.9% of the network's responses. The curve in the last panel (*c*) shows multiple transitions that are far apart and is representative of 1% of the network's responses. This type of response is not interpreted even though one of the transitions is almost correct.

Table 3 Results of the networks with the test set of 1575 patterns encoding chemical structures that the neural network had not seen before. It is obvious that the best recognition rate with an error of ⩽ ± 100 °C is obtained by network N10/1. The best quality of prediction is obtained for network N100/370 as evidenced by the fraction of predictions with an error of more than 20 °C and the RMS value

Network	Rec.[a] (%)	RMS[b]/ °C	± 5 °C[c] (%)	± 10 °C[c] (%)	± 15 °C[c] (%)	± 20 °C[c] (%)	± 50 °C[c] (%)	± 100 °C[c] (%)	> ± 20 °C[d] (%)
N10/1	100	18.8	41.3	63.1	76.5	84.7	96.9	99.8	15.3
N10/370	99.0	19.0	38.2	59.0	71.4	79.5	95.8	99.0	19.5
N100/1	100	28.6	42.9	61.5	71.5	76.6	91.7	98.4	23.4
N100/370	79.4	14.3	42.4	58.4	65.7	71.1	78.5	79.3	8.3

[a] Overall recognition rate. [b] RMS value of all recognized patterns. [c] Fraction of the patterns of the test set that have an error of less than or equal to the value specified. [d] Fraction of the patterns whose transition temperature was predicted with an error of greater than 20 °C.

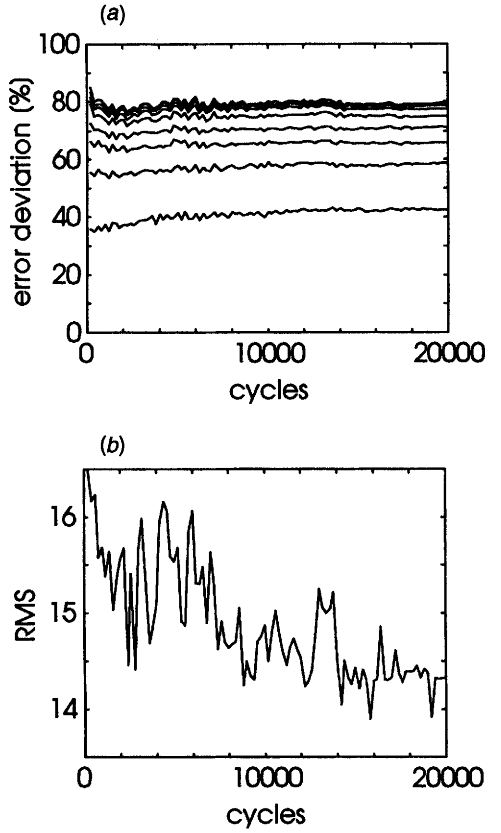

Fig. 5 Progress of the training of network N100/370 obtained by testing the network with the independent test set. The curves in panel (*a*) depict the fractions of the test patterns with errors of less than or equal to 5, 10, 15, 20, 30, 40, 50 and 100 °C (bottom to top), respectively, as a function of the training cycle. Panel (*b*) shows the development of the RMS of the test patterns during the training of the network. It shows that the RMS is mainly an indicator of the number of large errors in the prediction because it drops to lower values when the curves of panel (*a*) have already converged to seemingly constant values.

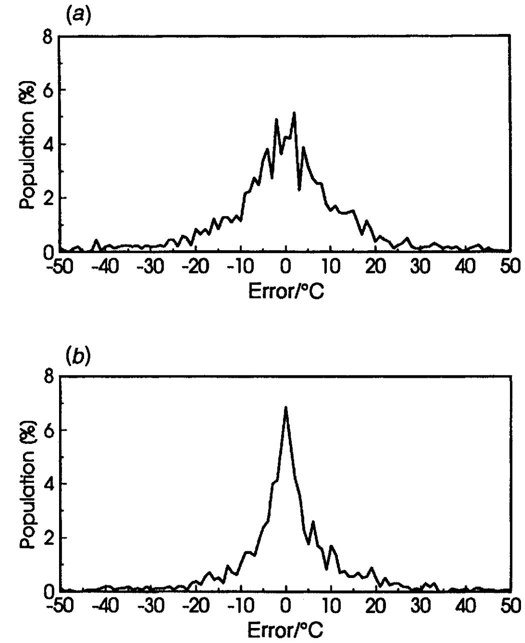

Fig. 6 Error distributions of the test set patterns for networks N10/1 and N100/370. A comparison of the two curves indicates that the distribution in panel (*b*) is more narrow at low values and has fewer large errors than that shown in panel (*a*). The references for the percent values are in both cases the whole test set. If only the recognized patterns were used as a reference the percent values in panel (*b*) would increase by 26% relative (*cf.* Table 3).

some errors in the database as well as in the underlying literature and, secondly, the neural net may not have had an adequate number of molecules in the training set to represent the breadth of liquid crystals that form a smectic A phase.

We have shown that it is possible to predict the clearing temperature of nematic liquid crystalline compounds using neural networks.[12] For this task classical methods can also be used.[13] However, these classical methods usually work only within a series of homologous compounds and cannot be used in general to predict the properties of a wide variety of chemical structures. In contrast to the prediction of clearing temperatures of nematic liquid crystalline compounds it is more difficult to predict the transition temperatures of compounds that form a smectic A phase because the formation of the latter phase requires that intermolecular forces are recognized by the algorithm, *i.e.* smectic phases occur only if the core groups and the tail groups have a significantly higher adhesion to each other than that between the core and the tail groups.

Further advantages of neural networks compared with classical methods lie in their ability to generalize from learned data, in the speed of predicting new information (about 50 000 per hour on a 486 PC) and in that they do not rely on the knowledge of explicit rules. Preliminary tests of classical regression analysis to predict transition temperatures show that a prediction of the quality shown here can be obtained when only one fragment is varied. That is, predictions within homologous series are possible, but when more than one group is changed at a time the prediction usually has much larger margins of error than the neural network based prediction. The development of a structural encoding that could potentially be used to encode more than a single class of molecules to one neural network is under way.

References

1 J. A. Burns and G. M. Whitesides, *Chem. Rev.*, 1993, **8**, 2583.

2 B. Meyer, T. Hansen, D. Nute, P. Albersheim, A. Darvill, W. York and J. Sellers, *Science*, 1991, **251**, 542.

3 J. P. Radomski, H. v. Halbeek and B. Meyer, *Nat. Struct. Biol.*, 1994, **1**, 217.

4 D. L. Clouser and P. C. Jurs, *Anal. Chim. Acta*, 1994, **3**, 221.

5 B. Curry and D. E. Rumelhart, *Tetrahedron Comput. Methodol.*, 1990, **3**, 231.

6 J. Gasteiger, X. Li, C. Rudolph, J. Sadowski and J. Zupan, *J. Am. Chem. Soc.*, 1994, **116**, 4608.

7 A. A. Gakh, E. G. Gakh, B. G. Sumpter and D. W. Noid, *J. Chem. Inf. Comput. Sci.*, 1994, **34**, 832.

8 L. M. Egolf, M. D. Wessel and P. C. Jurs, *J. Chem. Inf. Comput. Sci.*, 1994, **34**, 947.

9 D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing*, MIT Press, Cambridge, 1986, vol. 1.

10 V. Vill, *Liquid Crystals*, Landolt-Boernstein, New Series, Springer, Berlin, 1992–1995, vol. IV/7, subvol. 7a–7e.

11 V. Vill, *LiqCryst—Liquid Crystal Database*, LCI Publisher, Hamburg, 1995; Fujitsu Kyushu System (FQS) Ltd., Fukuoka, Japan, 1995.

12 H. Kränz, V. Vill and B. Meyer, submitted for publication in *J. Chem. Inf. Comput. Sci.*

13 V. Vill, *Liq. Cryst. Today*, 1995, **5**, 6.

Paper 6/00529B
Received 23*rd January* 1996
Accepted 1*st April* 1996