A Fast Mass Spectrum Screening Technique for Volatile Organic Compounds Based on Parallel Artificial Neural Networks

Thomas G. Thomas, Jr.*
704 48th Street South, Birmingham, AL 35222

Dennis G. Smith
University of Alabama at Birmingham, Department of Electrical and Computer Engineering, UAB Station, Birmingham, AL 35294

Abstract

A technique for screening mass spectra for the presence of volatile organic compounds (VOCs) is developed using probabilistic neural networks. A parallel neural network filter is designed to recognize benzene, toluene, ethyl benzene, and *o*-xylene in gas chromatography–mass spectrometry (GC–MS) chromatograms of VOC mixtures. The filter trained rapidly and was evaluated by analyzing a variety of VOC combinations. The performance of the network offers some significant advantages over the traditional GC–MS data processing techniques such as ion extraction and compound library searching. Advantages include speed, selectivity, and the ability to discriminate between overlapping compounds.

Introduction

A popular separation and detection technique for the analysis of airborne volatile organic compounds (VOCs) is gas chromatography–mass spectrometry (GC–MS). A large air sample containing VOCs is concentrated into a small volume by a preconcentration apparatus. The VOC mixture thus obtained is separated into its individual components by GC. An MS detector is then utilized to determine the mass spectrum of each peak in the chromatogram. Identification of the individual components is usually done off-line after the mass spectral data is taken.

A mass spectrum detector operates by ionizing a sample compound with high-energy electrons. The compound breaks into fragments that are then quantitated by an electrometer. In a typical GC–MS run, a range of masses are measured to obtain a spectral pattern. The mass numbers are expressed as the molecular weight of each fragment divided by the ion charge (or *m/z* values). The quantity of each ion is expressed as an abundance. Several scans are performed during each second of the chromatogram.

Because mass spectral patterns are highly characteristic of specific compounds, the automated identification of mass spectral patterns has been the subject of much research. The current most prevalent method for identifying compounds from mass spectra is automated database-searching, in which the spectrum of an unknown compound is matched with a reference spectrum. Reference spectrum libraries are constantly updated, and some libraries contain 100,000 spectra or more.

The majority of VOC analyses are conducted by operators who have a priori information about which VOC compounds are likely to be present. Knowledge of the compound structure can aid in the identification because certain characteristic ion fragments will be present. A technique known as ion extraction is often used in which a specific set of ion abundances is extracted from a full-range mass spectral scan. Usually six or eight ions are sufficient to make a positive identification. If only a few compounds are of interest, the detector is sometimes operated in single-ion mode, in which only the characteristic ions of a particular compound are sampled.

If a compound is unknown, the identification problem is harder. A routine library search can yield inconclusive results, especially if the MS used to analyze an unknown sample is different from the instrument used to obtain the library spectrum.

The availability of powerful and inexpensive computers in recent years has initiated a wide variety of GC–MS data processing techniques. The first such techniques, dating back to the 1970's, involved database searches. Numerical methods, such as *k*–nearest neighbor analysis and linear least squares, were tried with some success in the 1980's. In the early 1990's, artificial intelligence (AI) methods were introduced. Expert systems,

* Author to whom correspondence should be addressed.

fuzzy logic, and neural networks have all been applied to spectral classification and identification.

One of the most promising of the artificial intelligence techniques for MS data reduction is the utilization of neural networks. Some authors do not group neural networks with artificial intelligence because a neural network is mathematical in design, but like more traditional AI techniques, neural networks attempt to simulate the behavior of an expert in decision-making. A feed-forward network is trained with a set of data for which both the inputs and outputs are known. The network actually "learns" the test data and, with proper selection of the test data, is able to generalize to data it has never seen before.

The fundamental element of a neural network is a single neuron consisting of one or more inputs, a summing junction, and a nonlinear transfer function. The inputs are weighted and summed together at the neuron input. The neuron output is a nonlinear function of the sum of the weighted inputs. Neurons are usually clustered together into layers and fully connected (i.e., each neuron in a layer is connected to every neuron in a subsequent layer). If the signal flow is from the input to the output with no feedback loops, the network is known as a feed-forward network.

The basic structure of a typical feed-forward neural network consists of an input layer, a hidden layer, and an output layer. The inputs are processed by the input layer, sent to the hidden layer, and propagated to the output layer. Each of the connections between the layers is weighted, and the weights are adjusted to achieve the desired input/output transfer function.

The advantage of a neural network is that any arbitrary mathematical relationship between the input and output can theoretically be attained by the judicial selection of training data. The operator considers the network a "black box" that produces an output in response to an input. He does not usually know or care what the underlying individual weight values are as long as the right answer is obtained.

Adjustment of the weights is usually done by an iterative scheme in which the test data inputs are presented to the network together with their known outputs. The connection weights are adjusted by a scheme known as back-propagation until the inputs produce the desired outputs to within a user-specified amount of error (1). A complete pass through the data with back-propagation of error is known as a training epoch.

Although there are many types of neural networks, the most popular for MS data reduction is the feed-forward network with supervised back-propagation training as described above. Many applications have been described in the literature. For example, in 1989 Harrington et al. (2) used feed-forward neural networks for the recognition of pyrolysis mass spectra of bacteria. Long described the identification of jet fuel chromatographic data using back-propagating neural networks (3). A technique for the identification of the mass spectra of alditol acetates by neural networks was developed by Sellers et al. (4). In a series of papers published in 1993, Goodacre et al. applied neural networks to the recognition of pyrolysis mass spectra of amino acids, coliform bacteria, and seed oils (5–7). In all three papers, a three-layer neural network was employed with 150 input neurons, eight neurons in the hidden layer, and one output. The value of the output was a continuous variable representing the "amount of component" in each case.

The BP variety of feed-forward networks is virtually the only type of supervised network employed for pattern recognition in MS. Other types of supervised networks exist, however. An intriguing type of network is described by Specht, and is often as good as, if not better than, a BP network for pattern classification (8). The network, known as a probabilistic neural network (PNN), is similar in operation to a *k*–nearest neighbor classifier.

The topology of a PNN is the same as a BP-trained neural network, but there are some important differences. Most evident is the number of nodes in the hidden layer. A BP network has a number of nodes in the hidden layer that is usually less than the number of input nodes. In a PNN network, there is one node for each pattern in the training set. A BP network trains by iteration to achieve an arbitrary error for the training set, which can be slow. A PNN network trains in a single pass through the training set, which is very fast. Once trained, the BP network operates quickly because a relatively small number of weights are multiplied by input patterns to obtain an output. A PNN network must update a set of weights for each sample in the training set, which can be slow if the training set is large.

PNNs are based on probability density function estimation to make classifications of input patterns. A PNN is only useful as a classifier; it cannot predict the value of a continuous variable. A PNN is somewhat similar to a *k*–nearest neighbor classifier because a degree of pattern "averaging" from the training set is used to estimate the density of the categories.

The density function estimation task is usually accomplished by using a set of patterns with known classification. Specht described the use of the product of univariate kernels to estimate a multivariate probability density function. In particular, a Gaussian kernel can be used for a multivariate estimate of the probability function of the sample set, as shown in Equation 1.

$$f_A(X) = \frac{1}{(2\pi)^{p/2}\sigma^p}\,\frac{1}{m}\sum_{i=1}^{m}\exp\left[-(X-X_{Ai})^t\,(X-X_{Ai})/(2\sigma^2)\right] \qquad \text{Eq 1}$$

where *i* is the current pattern index, *m* is the total number of training patterns, X_{Ai} is the *i*th training pattern from category θ_A, σ is a smoothing parameter, and *p* is the dimensionality of the input space. It is assumed that the individual values of X_{Ai} are independent identically distributed random variables.

Equation 1 describes the sum of multivariate Gaussian distributions centered at each training sample. The smoothing parameter (σ) defines the amount of interpolation between the pattern locations. As σ approaches zero, the PNN approximates a nearest neighbor classifier. As the smoothing parameter is increased, the PNN acts as a *k*–nearest neighbor classifier.

The network structure consists of an input layer, a hidden layer with a summing/exponentiation node per training pattern, and an output layer for each category consisting of a summing node. Each input pattern vector *X* performs a series of *i* dot products with the weight vectors W_i, one for each training pattern, such that $Z_i = X \times W_i$. Provided W_i and *X* are normalized, each exponentiation of Equation 1 reduces to Equation 2.

$$\exp\left[(Z_i - 1)/(\sigma^2)\right] \qquad \text{Eq 2}$$

The nodes in the output layer simply sum the contributions from all of the training pattern nodes to arrive at a value. The output category node with the highest value "wins."

Training is done by setting the weight vector W_i in one of the pattern units equal to each pattern X in the training set. As a result, the PNN network trains very quickly. Unlike BP networks, it is easy to add or change training patterns in a PNN network. A particular pattern can be changed by simply replacing the corresponding weight vector. A pattern can be added by simply adding another hidden layer node without changing any of the other nodes. In a BP network, such changes would make it necessary to retrain the entire network.

The ultimate performance test of a classification method is how well it classifies unknowns. The simplest performance to measure is the percentage of correct answers; if a structure is present, the classifier correctly identifies it. If a structure is absent, the classifier should not indicate its presence. In practice, false positives are much more common than false negatives, so Curry and Rumelhart proposed two measures of performance called recall and reliability (9). Recall, given in Equation 3, is simply the number of correct identifications divided by the number of identifications attempted. Reliability, given in Equation 4, is the number of correct identifications divided by the sum of the correct identifications and the false positives.

$$\text{Recall} = I_c / N_c \qquad \text{Eq 3}$$

$$\text{Reliability} = I_c / (I_c + I_f) \qquad \text{Eq 4}$$

where I_c is the number of correctly asserted class members, N_c is the total number of class members, and I_f is the number of compounds falsely accused of being class members.

It is evident from the literature that many neural network applications for the interpretation of mass spectral data involve large data sets, many inputs, and large numbers of classes. Preprocessing of the data is usually necessary to generate input features for the neural networks. Some of the preprocessing techniques can get mathematically involved and can occupy considerable time and computational resources.

Many analytical problems encountered in VOC analysis are on a smaller scale. It is usually necessary to screen for a relatively small number of compounds in a GC–MS sample run. Evidently a need exists for small, simple networks that can be used for the identification of unknown compounds from low-resolution mass spectral data. The networks should be easily trained with user data, train rapidly, and be easily tailored to the needs of a variety of researchers. In addition, the development and use of the networks should be within the budget and computational resources of a small research laboratory. Such networks offer advantages over peak-matching techniques such as speed and selectivity. A PNN-based parallel adaptive filter that meets these criteria is proposed and described in this paper.

Experimental

The main challenge in designing a neural network for recognizing mass spectral patterns is to decide what input variables should be utilized in the classification process. For GC–MS data, the characteristic ion abundances for a specific compound, the compound retention time, and the compound peak height are logical choices. Some thought must be given to the selection of training data that is representative of the type of unknown data expected. In the case of mass spectrum identification, the neural network outputs are usually yes/no classifiers selecting one or more classes corresponding to the unknown compounds.

For the purposes of training and evaluating the neural network designs, eight VOC compounds were chosen. They were divided into two classes: target compounds and interferents. The target compounds were benzene, toluene, ethyl benzene, and *o*-xylene. The interferents were chloroform, 1,2-dichloroethane, 1,1,2-trichloroethane, and methyl isobutyl ketone. The interferents were chosen because they are all common industrial solvents that might be present in some concentration almost anywhere. The target compounds were chosen because they are regulated by the EPA at very low concentrations. Benzene, toluene, ethyl benzene, and *o*-xylene (sometimes referred to collectively as BTEX) are closely related in structure. Toluene, ethyl benzene, and *o*-xylene are substituted benzene compounds with similar mass spectra, and as such are good candidates for evaluating an automated compound identification technique.

Before large databases for mass spectra were available, analysts looked up compound mass spectra in tables. To save space, the tables were usually organized so that only the eight most intense ions were listed for a given compound. The ions were listed in order of abundance with the abundances normalized. The eight most abundant ions for the BTEX compounds are shown in Table I.

Table I. BTEX Ion Patterns

Benzene		Toluene	
Ion *m/z*	Abundance	Ion *m/z*	Abundance
78	1.000	91	1.000
77	0.200	92	0.715
52	0.195	65	0.135
51	0.175	51	0.110
79	0.060	63	0.105
76	0.050	89	0.050
74	0.045	93	0.050
63	0.025	62	0.050
Ethyl benzene		***o*-Xylene**	
Ion *m/z*	Abundance	Ion *m/z*	Abundance
91	1.000	91	1.000
106	0.310	106	0.400
51	0.140	105	0.175
65	0.080	51	0.175
77	0.080	77	0.150
78	0.070	65	0.100
103	0.040	79	0.085
79	0.040	92	0.075

The abundances are normalized so that the most abundant ion is assigned the value 1.00 and the other ions are assigned relative abundances with values between zero and one. The abundance values comprised eight inputs to the neural network filters. Two other inputs, the peak retention time and the peak height, were also used as inputs for a total of 10 inputs. The peak height was normalized by dividing by the largest peak in the total ion chromatogram; the retention time was normalized by dividing by the length of the sample run.

Hardware

A Hewlett-Packard (Palo Alto, CA) 5890A GC and a Hewlett-Packard 5971A mass spectral detector were used to collect the data in this study. The compound separations were performed with a 15-m, 0.53-mm megabore capillary column (DB-1 type) with a 1-µm film thickness. The temperature was programmed at 50°C for 1 min, then increased 50°C/min to a final temperature of 180°C. A typical analysis required 3 min with a 1-min solvent delay, which resulted in 2 min of data acquisition per chromatogram. The cycle time between analyses was 6 min. The carrier gas was helium at 15 mL/min, and the injection port temperature was 180°C. The MS interface temperature was also 180°C. A 5:1 sample splitter was used between the column outlet and the MS detector inlet to set the carrier flow to 3 mL/min within the MS detector. The MS detector was operated in electron ionization mode with an electron multiplier voltage of 1750 V. The mass scan range was 50–120 *m/z* at a scan speed of 5.9 scans/s. This resulted in approximately 700 mass spectral scans in a 120-s chromatogram.

Table II. Training and Test File Data Records*

Training file	
Output	either 0.1 or 0.9
Scan time	normalized by dividing by 180 s
Peak height	normalized by dividing by the largest peak in the chromatogram
Ion 1 abundance	most abundant ion
Ion 2 abundance	normalized by dividing by ion 1 abundance
Ion 3 abundance	normalized by dividing by ion 1 abundance
Ion 4 abundance	normalized by dividing by ion 1 abundance
Ion 5 abundance	normalized by dividing by ion 1 abundance
Ion 6 abundance	normalized by dividing by ion 1 abundance
Ion 7 abundance	normalized by dividing by ion 1 abundance
Ion 8 abundance	normalized by dividing by ion 1 abundance
Test file	
Record	number of scans in the chromatogram
Scan time	normalized by dividing by 180 s
Peak height	normalized by dividing by the largest peak in the chromatogram
Ion 1 abundance	most abundant ion
Ion 2 abundance	normalized by dividing by ion 1 abundance
Ion 3 abundance	normalized by dividing by ion 1 abundance
Ion 4 abundance	normalized by dividing by ion 1 abundance
Ion 5 abundance	normalized by dividing by ion 1 abundance
Ion 6 abundance	normalized by dividing by ion 1 abundance
Ion 7 abundance	normalized by dividing by ion 1 abundance
Ion 8 abundance	normalized by dividing by ion 1 abundance

* One per mass spectral scan.

Software

The neural networks were constructed, trained, and tested using the NeuroWindows data link library for Visual Basic. The NeuroWindows package is published by the Ward Systems Group (Frederick, MD). It is designed to utilize Visual Basic as a framework and is capable of building a variety of neural network types.

The data was collected using the Hewlett-Packard G 1034BMS Chemstation software package running on a Compaq Presario 866 PC. The neural network inputs were obtained by processing the spectral record sections of the Hewlett-Packard MS data files. A custom program was written to process the MS data files into a format that could be utilized to train and test the neural networks. The format of both files is shown in Table II. For each mass spectral scan in the training file, the program extracts the

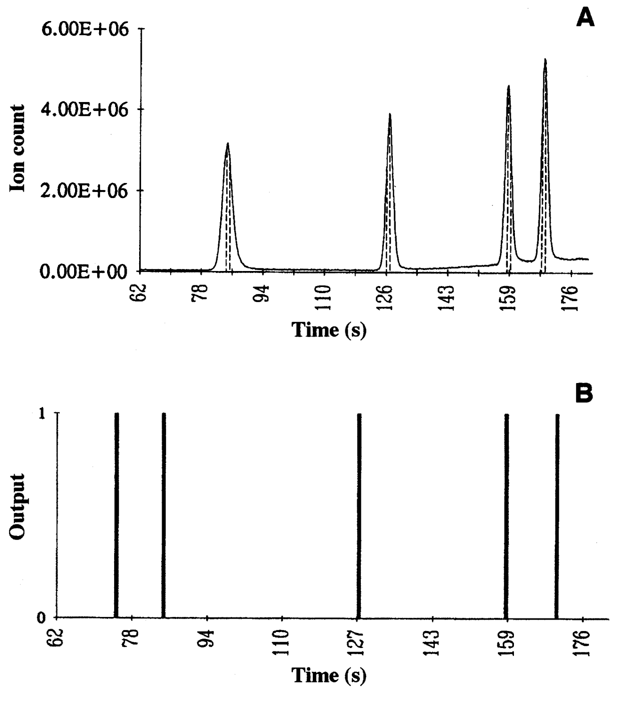

Figure 1. PNN training file input (A) and output (B).

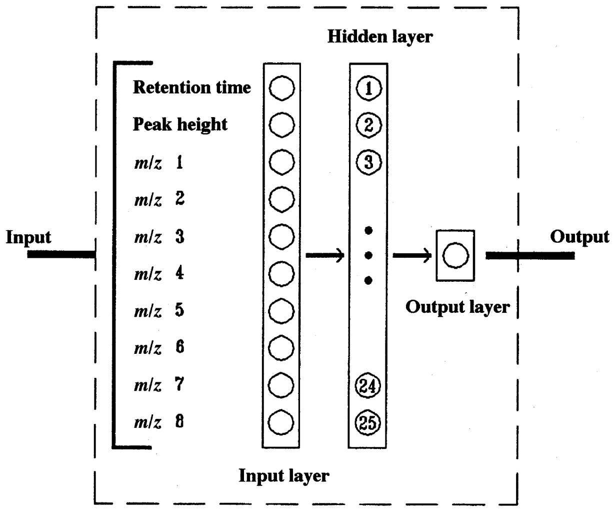

Figure 2. PNN filter block.

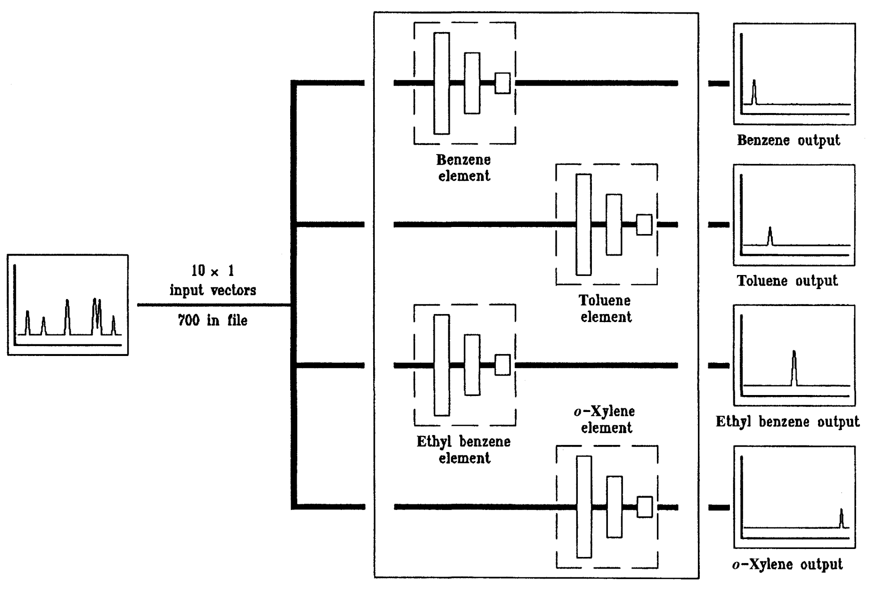

Figure 3. PNN filter block diagram.

Table III. Target Compound Calculation Spreadsheet

	PNN compound classification			
s1l1i1.put	btrn1.net	ttrn1.net	etrn1.net	xtrn1.net
Chloroform	0.00	0.00	0.00	0.00
1,2-Dichloroethane	0.00	0.00	0.00	0.00
Benzene	0.73	0.00	0.00	0.00
MIBK	0.00	0.00	0.00	0.00
1,1,2-Trichloroethane	0.00	0.00	0.00	0.00
Toluene	0.00	0.49	0.00	0.00
Ethyl benzene	0.00	0.00	0.75	0.01
Xylene	0.00	0.00	0.00	0.72

eight ions for each target compound, normalizes them, and writes them to an ASCII data file.

Figure 1 shows a typical training chromatogram and the corresponding output. The training file of Figure 1A is shown in a form known as a total ion chromatogram (TIC). This indicates that for each mass spectral scan, the sum of the abundances of all ions in the scan is displayed. A total of 25 scans were used as training samples for each of the four test compounds. Five scans centered at 75 s, five scans centered at 85 s, five scans centered at 127 s, five scans centered at 159 s, and five scans centered at 170 s. Note that the scans at 85, 127, 159, and 170 s were selected to fall at the maximum height of the compound peaks. The scans at 75 s were selected to represent a blank, which corresponds to a zero output for both filter outputs. The outputs of the training samples were set to 1.0 when a particular compound was present and to 0.0 when a particular compound was not present. This is shown in Figure 1B.

A PNN was designed for each target compound as follows. First, an input layer of 10 neurons was created, one for each input, as shown in Table II. The peak height, scan time, and eight ions for each mass spectral scan in the target compound chromatogram were utilized as inputs. Then a hidden layer of 25 neurons, one for each training sample, was connected to the input layer. An output layer with one neuron was connected to the hidden layer, and each network was trained in a single pass with the training data shown in Figure 1.

Figure 2 shows a single PNN filter block together with its inputs and outputs. Figure 3 shows four blocks configured as an adaptive filter. The input chromatogram was processed to obtain four outputs, each one of which indicates the presence or absence of one of the BTEX components. Unlike a BP network, the training of a PNN network is not an iterative process. Training consists of setting the connection weights between the input layer and a particular hidden layer node to the values of the inputs. As was stated before, there is one node for each input pattern, and training proceeds in a single pass. The individual components of the parallel network were trained separately.

The files used to test the networks were generated by the same custom program described above with the exception that the outputs were unknown. The output values in the test files were replaced by the data record numbers. Once trained, the networks operated as follows. An unknown chromatogram was processed to obtain a data file for each BTEX compound. The individual test files contained only the eight ion abundances specific to that compound. The target compound data files were submitted to the corresponding neural networks in parallel. For each mass spectrum scan, an output was calculated. This resulted in four graphic data files showing retention time versus output. If the target compound was present, a peak was observed at the retention time of the compound. The peak was integrated during a 2-s gate corresponding to the retention time of the specific compound. If the integrated peak value was above a user-selected threshold, the corresponding compound was assumed to be present.

The calculation of the presence or absence of a target compound was performed by an Excel spreadsheet. Table III shows the output of a typical spreadsheet for a mixture of the interferents and the BTEX compounds. As can be seen, the integrated values produced for each compound in each gate were tabulated. The four columns refer to each of the four neural networks. The rows correspond to the compound gates. The values at the intersections are the results of integrating the outputs of the respective networks during the individual compound gates.

Integration was performed by summing 10 network outputs bracketing the retention times for each compound. These 10 samples correspond to approximately 2 s of data. The sum was normalized by dividing by 10. If the integral value was larger than a user-selected threshold, a specific compound was assumed to be present. For example, the value of 0.73 for benzene calculated by the benzene network indicated the presence of benzene. All other compound values were zero, indicating that the benzene classifier showed good rejection of compounds other than benzene. Similar results were observed for the toluene network (0.49), the ethyl benzene network (0.75), and the *o*-xylene network (0.72).

Data collection

Training and test data for BTEX and the interfering compounds were obtained by making liquid injections of prepared standards into the GC–MS system. All standards were prepared from neat compounds in spectroscopic-grade methanol. Tables IV and V describe the preparation of the standard solutions.

For all of the standards except the blank, 10:1 and 100:1 dilutions were made. For each standard and the dilutions, five injections were made. The injections were 0.5, 1.0, 1.5, 2.0, and 2.5 μL. The data for each injection were then acquired and stored in a corresponding file. The standard levels were chosen to give data from easily detectable concentration levels to concentration levels that were at the threshold of detection for the MS detector.

Chromatographic conditions were selected for good separation between the BTEX compounds. The performance of each network was evaluated using the test data described in Tables IV and V. The blank data were used to evaluate the ability of the parallel network to reject noise in the absence of target compounds. The BTEX mixture was used to evaluate the ability of the network to differentiate between the target compounds. The mixture of interferents and BTEX was used to evaluate the ability of the network to differentiate between target compounds and other VOCs. The interferant mixture was used to evaluate the ability of the network to reject VOCs that are not target compounds.

Under the chromatographic conditions selected in this experiment, ethyl benzene and *m*-xylene have approximately the same retention times, and thus overlap. The ethyl benzene–*o*,*m*-xylene mixture and the *o*,*m*-xylene mixture were selected to evaluate the ability of the neural networks to discriminate between overlapping compounds.

Results and Discussion

Training

The networks were trained individually by using a single chromatogram: a 0.5-μL injection of the 1000 μg/mL BTEX mixture. The four individual training files contained 25 samples of the BTEX chromatogram mass spectrum. Each individual training file contained only the eight ions of interest for the corresponding compound. The chromatogram peak retention time for benzene was 85 s, that of toluene was 127 s, that of ethyl benzene was 159 s, and that of xylene was 169 s. Training was performed in a single pass through the data.

Evaluation

The performance of each network was evaluated by integrating each compound output in the six types of data and applying the recall and reliability measures of Curry et al. For each standard, five samples of each dilution level were analyzed. If a compound was incorrectly identified as present, the sample was recorded as a false positive. If a compound was present and not detected, it was recorded as an incorrect classification. If a compound was present and was detected or if a compound was absent and was not detected, it was recorded as a correct classification.

The data were most readily interpreted visually by superimposing the network outputs on the TIC of the sample. If a particular compound was present, a peak was clearly visible at the retention time in the corresponding network. If a compound was absent, there was little or no response at the compound retention time. Figure 4 shows a typical blank chromatogram in

Table IV. Standard Solutions of Test Compounds

Standard solutions	Benzene (μg/mL)	Toluene (μg/mL)	Ethyl benzene (μg/mL)	*o*-Xylene (μg/mL)	*m*-Xylene (μg/mL)
Blank	0	0	0	0	0
BTEX	879	867	867	880	0
10:1 dilution	87.9	86.7	86.7	88	0
100:1 dilution	8.79	8.67	8.67	8.8	0
BTEX and interferents	879	867	867	880	0
10:1 dilution	87.9	86.7	86.7	88	0
100:1 dilution	8.79	8.67	8.67	8.8	0
Interferents	0	0	0	0	0
10:1 dilution	0	0	0	0	0
100:1 dilution	0	0	0	0	0
Ethyl benzene and *m*,*o*-xylene	0	0	867	880	864
10:1 dilution	0	0	86.7	88	86.4
100:1 dilution	0	0	8.67	8.8	8.64
m-Xylene and *o*-xylene	0	0	0	880	864
10:1 dilution	0	0	0	88	86.4
100:1 dilution	0	0	0	8.8	8.64

Table V. Standard Solutions of Interferent Compounds

Standard solutions	Chloroform (μg/mL)	1,2-Dichloroethane (μg/mL)	MIBK (μg/mL)	1,1,2-Trichloroethane (μg/mL)
Blank	0	0	0	0
BTEX	0	0	0	0
10:1 dilution	0	0	0	0
100:1 dilution	0	0	0	0
BTEX and interferents	1492	1256	805	1338
10:1 dilution	149.2	125.6	80.5	133.8
100:1 dilution	14.92	12.56	8.05	13.38
Interferents	1492	1256	805	1338
10:1 dilution	149.2	125.6	80.5	133.8
100:1 dilution	14.92	12.56	8.05	13.38
Ethyl benzene and *m*,*o*-xylene	0	0	0	0
10:1 dilution	0	0	0	0
100:1 dilution	0	0	0	0
m-Xylene and *o*-xylene	0	0	0	0
10:1 dilution	0	0	0	0
100:1 dilution	0	0	0	0

which an injection of 0.5 µL of methanol was made. The target compound network outputs did not show the presence of any of the four target compounds. Figure 5 shows the chromatogram and network outputs for a 0.5-µL injection of the stock concentration of a mixture of all four target compounds. As can be seen, a clearly visible response was present in each network output at the retention time corresponding to the target compound retention time in the original chromatogram. Each network showed a

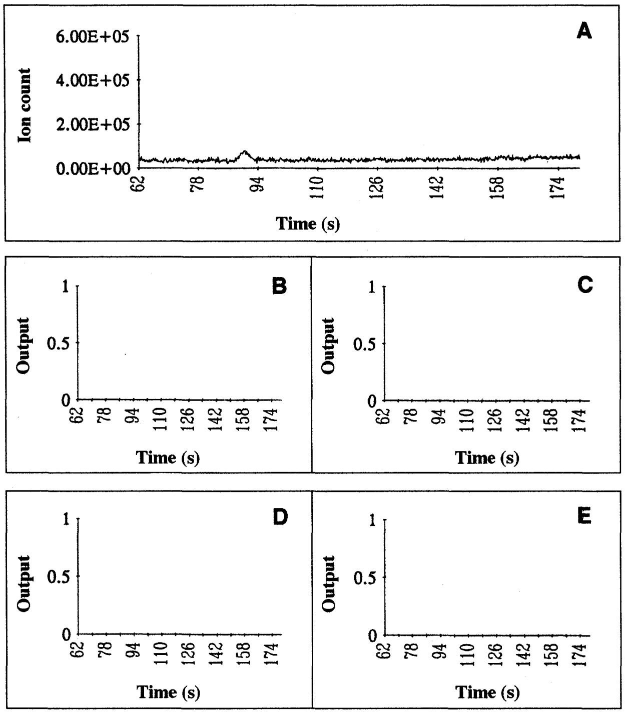

Figure 4. PNN filter output of blank data. (A) 0.5-µL injection of methanol, (B) benzene output, (C) toluene output, (D) ethyl benzene output, and (E) *o*-xylene output.

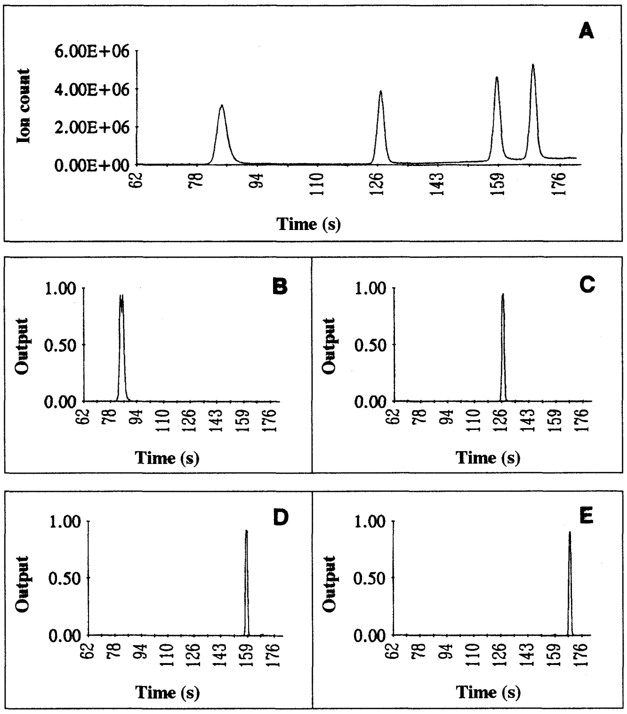

Figure 5. PNN filter output of BTEX mixture. (A) 0.5-µL BTEX injection (1 mg/mL standard), (B) benzene output, (C) toluene output, (D) ethyl benzene output, and (E) *o*-xylene output.

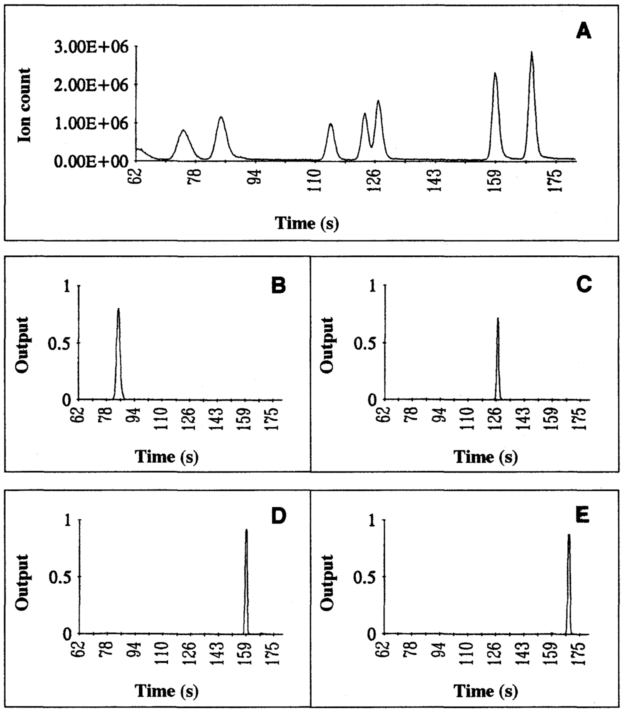

Figure 6. PNN filter output of BTEX with interferents. (A) 2.5-µL injection of BTEX with interferents (0.1 mg/mL standard), (B) benzene output, (C) toluene output, (D) ethyl benzene output, and (E) *o*-xylene output.

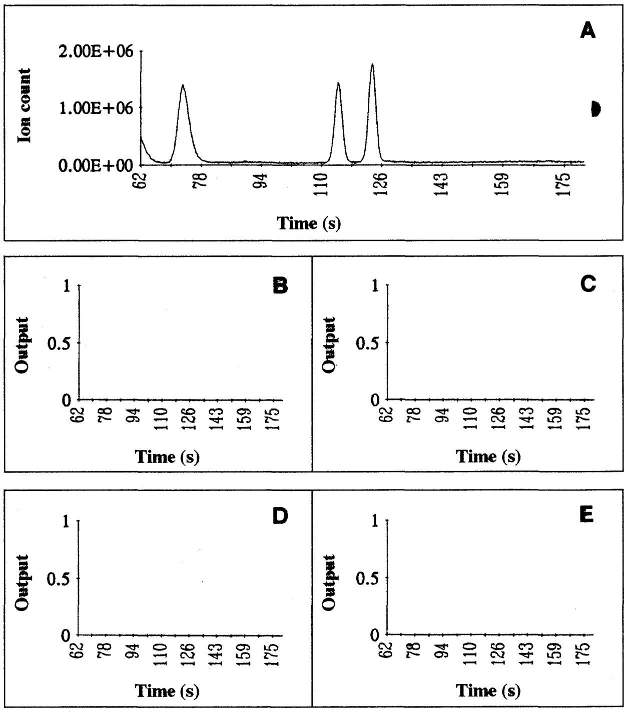

Figure 7. PNN filter output of interferents. (A) 2.0-µL injection of interferents mixture (0.1 mg/mL standard), (B) benzene output, (C) toluene output, (D) ethyl benzene output, and (E) *o*-xylene output.

response for its specific compound and rejected the other compounds very well. Figure 6 shows the chromatogram and network outputs for a 0.5-µL injection of the stock concentration of a mixture of all eight compounds. Again, the response for each target compound was clearly visible in its network output. Rejection of the other target compounds and the interferents was excellent. Figure 7 shows the chromatogram and network outputs for a 0.5-µL injection of the stock concentration of the interferents. The individual networks showed very low responses in the target compound gates and only a slight response in the 1,2-dichloroethane gate in the benzene classifier.

Figure 8 shows the chromatogram and network outputs for a 0.5-µL injection of the stock concentration of a mixture of ethyl benzene, *o*-xylene, and *m*-xylene. The benzene and toluene networks showed no response, as expected. The ethyl benzene network showed a clear response. The xylene network showed a clear response in the ethyl benzene gate that corresponds to *m*-xylene. A response was also visible for *o*-xylene. This indicated that the networks were capable of resolving overlapping peaks of ethyl benzene and *m*-xylene. Figure 9 shows the chromatogram and network outputs for a 0.5-µL injection of the stock concentration of a mixture of *o*-xylene and *m*-xylene. Clearly visible responses were present for the *m*-xylene and *o*-xylene peaks, but no response was noted in any of the other networks. This verified

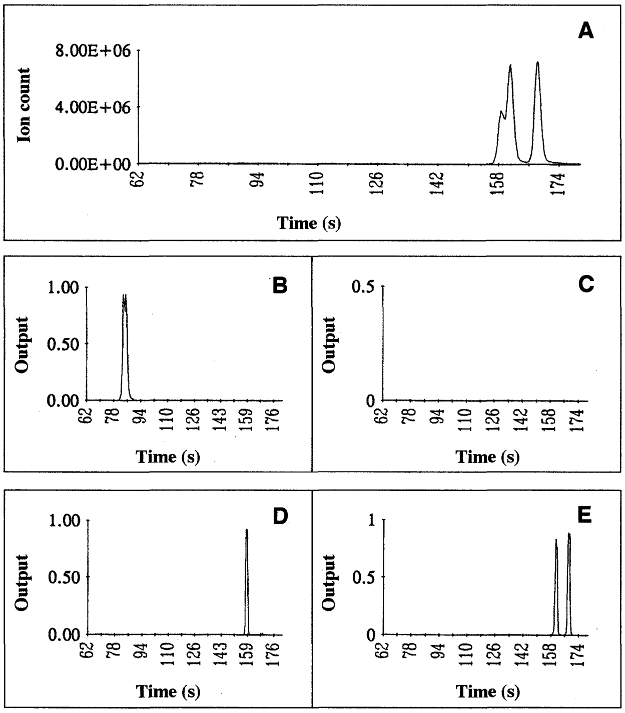

Figure 8. PNN filter output of ethyl benzene, *o*-xylene, and *m*-xylene. (A) 0.5-µL injection of ethyl benzene, *o*-xylene, and *m*-xylene (1 mg/mL standard), (B) benzene output, (C) toluene output, (D) ethyl benzene output, and (E) *o*-xylene output.

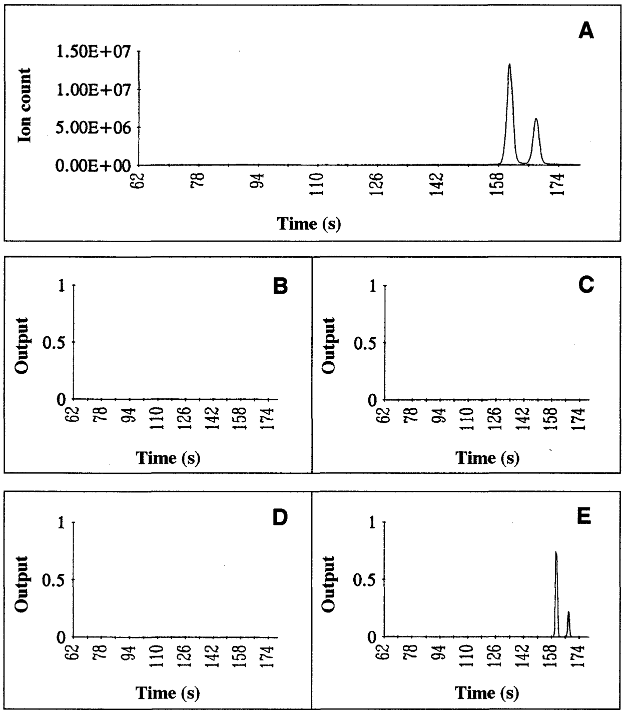

Figure 9. PNN filter output of o-xylene and m-xylene. (A) 0.5-µL injection of *o*-xylene and *m*-xylene (1 mg/mL standard), (B) benzene output, (C) toluene output, (D) ethyl benzene output, and (E) *o*-xylene output.

Table VI. Results of PNN Filter Classification of Compound Mixtures*

		Benzene output		Toluene output		Ethyl benzene output		*o*-Xylene output	
Standard solutions	Number of samples	I_c	I_f	I_c	I_f	I_c	I_f	I_c	I_f
Blank	5	5	0	5	0	5	0	5	0
BTEX	5	5	0	5	0	5	0	5	0
10:1 dilution	5	5	0	5	0	5	0	5	0
100:1 dilution	5	5	0	3	0	4	0	4	0
BTEX and interferents	5	5	0	5	0	5	0	5	0
10:1 dilution	5	5	0	5	0	5	2	5	0
100:1 dilution	5	0	0	0	0	0	0	0	0
Interferents	5	5	0	5	0	5	0	5	0
10:1 dilution	5	5	0	5	0	5	0	5	0
100:1 dilution	5	5	0	5	0	5	0	5	0
Ethyl benzene and *m*,*o*-xylene	5	5	0	5	0	5	0	5	0
10:1 dilution	5	5	0	5	0	5	0	5	0
100:1 dilution	5	5	0	5	0	5	0	3	0
m-Xylene and *o*-xylene	5	5	0	5	0	5	0	5	0
10:1 dilution	5	5	0	5	0	5	0	5	0
100:1 dilution	5	5	0	5	0	5	0	1	0
Total	80	75	0	73	0	69	2	68	0
Recall (%)	N/A	93.8		91.3		86.3		85.0	
Reliability (%)	N/A	100.0		100.0		97.2		100.0	

* I_c = number of correct classifications. I_f = number of false positives. Integration threshold is 0.02.

that ethyl benzene could be resolved from *m*-xylene, even though the peaks almost completely overlapped.

A total of 80 individual chromatograms were processed by the parallel network. Of these samples, 30 contained BTEX in some concentration, and 50 did not. Observation of the data files indicated that a choice of 0.02 as a threshold integral value gave good results for the recognition of target compounds. The number of false positives at this value was not zero, but it was acceptable. Table VI shows the results of the parallel neural network classification of the standard solutions. Calculation of the recall and reliability for each of the parallel networks was performed and tabulated at the bottom of Table VI.

The recall of both the benzene and toluene networks was above 90%, scoring 93.8 and 91.3%, respectively. The ethyl benzene network scored 86.3%, and the xylene network scored 85% in recall. The reliability for the standard concentrations investigated was 100% for benzene, toluene, and xylene. The reliability for the ethyl benzene was not as good, scoring 97.2%. This was due to false positive results at the integration threshold value of 0.02, which was selected for the data reduction. If the first two dilution levels were considered alone, the recall rate would be 100% for all four BTEX compounds.

All of the false positives for ethyl benzene were observed in the xylene gate of the ethyl benzene classifier. This is probably due to the fact that xylene and ethyl benzene share several ion fragments as inputs. Evidently, the networks have difficulty discriminating between the two compounds, especially at low concentrations. The integration threshold could have been selected to eliminate false positives entirely, but the recall rate would have been lower.

One of the advantages of parallel neural network identification of the target compounds over ion extraction is selectivity. In Figure 10, the HP Chemstation software was used to extract the six most abundant ions for each peak in the BTEX mixture. The ion abundances were plotted as overlapping graphs. The benzene peak is clearly visible in the ion-extracted chromatogram in Figure 10 corresponding to the benzene ions. However, the other three target compounds are difficult to tell apart based on the presence or absence of extracted ion peaks alone. A comparison with Figure 5 clearly shows the better selectivity of the parallel neural network.

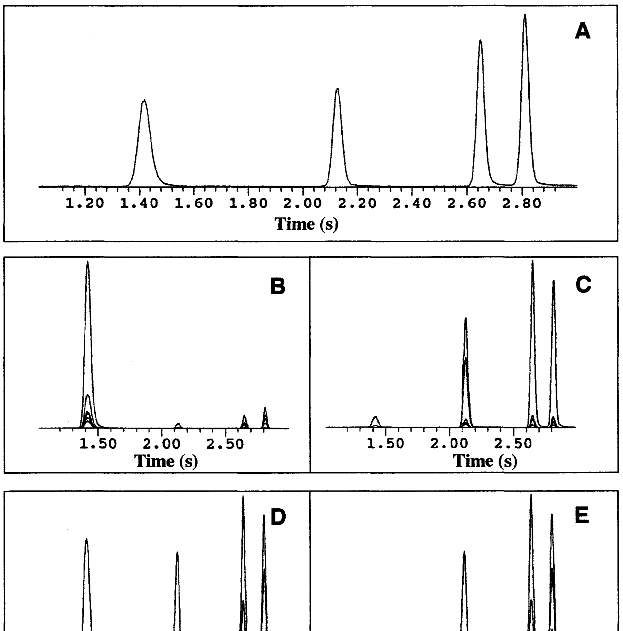

Figure 10. Extracted ion chromatogram of BTEX mixture. (A) 0.5-µL injection of BTEX (1 mg/mL standard), (B) benzene output, (C) toluene output, (D) ethyl benzene output, and (E) *o*-xylene output.

Conclusion

A detailed description of the design and implementation of a parallel neural network for VOC recognition has been given. The application of the network to the recognition of BTEX compounds in the presence of other VOCs has been successfully demonstrated. The network, consisting of four parallel neural networks, one for each compound, exhibits some unique properties when compared to ion extraction. The capability to reject noise and identify merged and overlapping compounds are all useful in a wide variety of GC–MS analyses.

These research results have demonstrated that small feedforward neural networks can be designed and trained with tools available to the average laboratory. Unlike a BP-trained network, the PNN networks can be trained very quickly. Individual networks could be trained on-line as a calibration injection is made. Once trained, the parallel network operates very rapidly and could be utilized to process GC–MS data in real time. Increasing the number of compounds in the parallel network is simply a matter of training individual networks based on eight-ion mass spectral patterns.

The capability to perform network training and unknown sample analysis with data obtained from the same instrument is an advantage. Because the operator controls the training conditions, the same compound identification networks could be trained to recognize mass spectral patterns under different analytical conditions. For example, a network could be trained to recognize the electron ionization spectrum of a compound in one analysis and could be retrained to recognize the chemical ionization spectrum in another analysis. In more traditional peak-matching techniques, the mass spectra in a library might be obtained from a wide variety of instruments and might be difficult to match. The fast screening of mass spectra has many potential applications. Because overlapping compounds can be resolved, GC conditions could theoretically be compromised for analysis speed. This can be important in such applications as explosives analysis or the detection of chemical warfare agents. The network could also be used as a preliminary "quick and dirty" test for the presence of target compounds. Once identified, more traditional analytical methods of quantitative analysis could be employed. If the compounds do not show up in the screen, the necessity for more expensive and time-consuming analysis could possibly be avoided.

Future efforts will be to refine the network architectures to improve selectivity and training speed. There are a variety of training algorithms in addition to back-propagation and PNN that may prove to have some advantages. The use of feedback from the network output to the input layer will also be investigated because the sequence of the training data is important in chromatography. Feedback will allow the network to "remember" previous data that may improve the performance. Also, the same architecture was utilized for all of the individual compound networks. This may not necessarily result in minimum training time or optimum performance.

The overall performance of the prototype parallel neural network for identifying mass spectral patterns was encouraging. In the near future, perhaps neural network signal processing of GC–MS data will be commonplace. When used in conjunction with more traditional means of data reduction, the neural network is a useful tool to the instrumental analytical chemist.

Acknowledgments

The author gratefully acknowledges the advice and comments of Dr. Dennis Smith of the University of Alabama at Birmingham's Department of Electrical and Computer Engineering. The use of the GC–MS system and the laboratory facilities at CMS Research Corporation in Birmingham, Alabama is also appreciated. The substantive results of this research were presented as an oral dissertation defense at the University of Alabama at Huntsville in partial fulfillment of the requirements for a doctorate in Electrical Engineering.

References

1. D.E. Rumelhart, G.E. Hinton, and R.J. Williams. *Parallel distributed processing: Explorations in the microstructure of cognition.* MIT Press, Cambridge, MA, 1986, pp. 322–28.
2. P. Harrington, T. Street, and K. Voorhees. Rule-building expert system for classification of mass spectra. *Anal. Chem.* **61:** 715–19 (1989).
3. J. Long, H. Mayfield, and M. Henley. Pattern recognition of jet fuel chromatographic data by artificial neural networks with back propagation error. *Anal. Chem.* **63:** 1256–61 (1991).
4. J. Sellers, W. York, P. Albersheim, A. Darvill, and B. Meyer. Identification of the mass spectra of partially methylated alditol acetates by artificial neural networks. Report, U.S. Department of Energy grant DE-FG09-85-ER13424, 1990.
5. R. Goodacre, A. Karim, M.A. Kaderbhai, and D.B. Kell. Rapid and quantitative analysis of recombinant protein expression using pyrolysis mass spectrometry and artificial neural networks: Application to mammalian cytochrome b5 in *E. coli. J. Biotech.* **34:** 185–93 (1993).
6. R. Goodacre and D.B. Kell. Rapid assessment of the adulteration of virgin olive oils by other seed oils using pyrolysis mass spectrometry and artificial neural networks. *J. Sci. Food Agri.* **63:** 297–307 (1993).
7. R. Goodacre, A. Edmonds, and D. Kell. Quantitative analysis of the pyrolysis mass spectra of complex mixtures using artificial neural networks: Application to amino acids in glycogen. *J. Anal. Appl. Pyrolysis* **26:** 93–114 (1993).
8. D. Specht. Probabilistic neural networks. *Neural Networks* **3:** 109–18 (1990).
9. B. Curry and D.E. Rumelhart. MSnet: A neural network which classifies mass spectra. *Tetrahedron Computer Methodology* **3:** 213–37 (1990).

Manuscript accepted December 12, 1997.