Journal of Biomolecular NMR, **15:** 15–26, 1999.
KLUWER/ESCOM
© 1999 *Kluwer Academic Publishers. Printed in the Netherlands.*

RESCUE: An artificial neural network tool for the NMR spectral assignment of proteins

J.L. Pons & M.A. Delsuc*
Centre de Biochimie Structurale UMR 9955, U 414, UM1, 15 avenue Charles Flahault, F-34000 Montpellier, France

Received 19 January 1999; Accepted 30 June 1999

Key words: artificial neural network, automated assignment, BioMagResBank, chemical shift, fuzzy logic, NMR of protein

Abstract

The assignment of the ^{1}H spectrum of a protein or a polypeptide is the prerequisite for advanced NMR studies. We present here an assignment tool based on the artificial neural network technology, which determines the type of the amino acid from the chemical shift values observed in the ^{1}H spectrum. Two artificial neural networks have been trained and extensively tested against a non-redundant subset of the BMRB chemical shift data bank [Seavey, B.R. et al. (1991) *J. Biomol. NMR*, **1**, 217–236]. The most promising of the two accomplishes the analysis in two steps, grouping related amino acids together. It presents a mean rate of success above 80% on the test set. The second network tested separates down to the single amino acid; it presents a mean rate of success of 63%. This tool has been used to assist the manual assignment of peptides and proteins and can also be used as a block in an automated approach to assignment. The program has been called RESCUE and is made publicly available at the following URL: http://www.infobiosud.univ-montp1.fr/rescue.

Introduction

High quality NMR spectra are relatively easy to obtain from a suitable protein or peptide sample. The spectra can be directly used to investigate for possible ligand binding, to measure pKa titrations, to investigate the oligomerization state or other physical studies.

However, for a deeper study, and notably if a structural study is to be undertaken, assignment of the ^{1}H spectrum cannot be avoided. Carrying out this assignment is usually a long and tedious process, and the difficulty and the length of this operation is certainly what limits the rate at which NMR studies are completed. Thus, any technique which could help in speeding up this procedure is certainly much needed.

Assignment is usually performed from a set of 2D and 3D J-correlated spectra, by considering the logical constraints introduced in the spin systems by the observation of cross-correlations, and by matching these constraints with the skeleton of the different side-chains of the polypeptide.

The information obtained from the J-correlated spectra is not sufficient, and NOE-correlated spectra must also be considered for the determination of the sequential linking of the amino acids. This sequential approach, first proposed by Wüthrich and co-workers (Wüthrich, 1986) is still used with great success. Since then, the technique has been extended with the use of ^{15}N labelling in the case of larger systems. The determination of the spin-system types is central in this assignment technique and depends to a large extent on the skill of the assignor to recognize spin-systems in the J-correlated spectra. This procedure is impeded by three main difficulties: (i) chemical shift values, strongly influenced by the environment can rarely be used; (ii) spin-system aliasing, which arises from the fact that different amino acids may have exactly the same spin-system topology (the AMX and the AMPTX families), or very closely related

*To whom correspondence should be addressed. E-mail: Marc-Andre.Delsuc@cbs.univ-montp1.fr.

spin-systems (I and L); (iii) the disappearance of some signals, too weak or hidden by others.

^{13}C and ^{15}N labelling is currently used as a standard procedure to circumvent these difficulties, as this technique permits a rapid and unambiguous assignment of the polypeptide backbone. However, this technique cannot be considered in the case of extracted proteins and the associated price tag rules it out for small proteins and peptides.

In all these procedures, the actual value of the ^{1}H chemical shift of each observed spin is rarely used, as it is usually considered that the environment of each residue has such a strong and unpredictable impact on the line positions that no reliable information can be extracted from it. Rather, chemical shifts are used in latter stages of the study, for predicting the secondary structure (Wishart et al., 1991; Wishart and Sykes, 1994) or in the process of structure refinement (Kuszewski et al., 1995).

The purpose of this work is to investigate whether the chemical shift information can be used directly to assist the amino acid determination, and to implement a program realizing this operation.

The artificial neural network software technology appears perfectly appropriate in our case. The basic principle is to build a function, taking the raw experimental data as input (in our case chemical shifts), and yielding the assignment as its output (here the type of the amino acid). This function is designed as a generic parameterized non-linear function. The parameters of this function are optimized against a large body of examples for which the answers are already known. This optimization is a lengthy process, performed by iteratively minimizing the differences between the actual output of the function being optimized and the ideal output profile. In the literature, this optimization step is usually called the training process. In a second step, the trained artificial neural network can be exploited, first on independent test data in order to assess the quality of the program, and finally on real cases.

The development of a good artificial neural network thus requires the definition of an optimum function design and the availability of a large database of representative examples, which will be split into a training and a test set.

The use of artificial neural networks based on chemical shift information, as an aid for protein NMR studies, has already been demonstrated, either for the study of homologous proteins (Hare and Prestegard, 1994), or by combining assignment and secondary structure (Choy et al., 1997; Huang et al., 1997). The present work differs from these pioneer studies in several aspects. First, we chose to use the large chemical shift database collected in the BioMagResBank database (Seavey et al., 1991). We selected from this database a representative subset of proteins which was used to perform the optimization of the artificial neural network. This approach permits to build a general purpose artificial neural network while the previous approaches, built on more specialized training sets, had less generality. Secondly, the usual artificial neural network design has been modified by adding a fuzzy logic layer on the input, thus obtaining a higher rate of success in the analysis process. Finally, because of its general use, this tool has been made publicly available under the name RESCUE on our web server at http://www.infobiosud.univ-montp1.fr/rescue.

Materials and methods

Database

The artificial neural network used in this work was trained on a set of chemical shifts extracted from the BioMagResBank (BMRB) database (Seavey et al., 1991). The BMRB database contains NMR chemical shifts derived from proteins and peptides, reference data, and amino acid information, along with data describing the source of the protein and the conditions used to study the protein. In constructing the database, proteins and larger peptides have been given priority.

The entire database as of July 1996 was downloaded as a flat file and used in this state for the present study. At this date, the BMRB contained over 100 000 unique ^{1}H chemical shifts, measured for over 1169 peptides or proteins. ^{1}H chemical shifts were extracted from the BMRB and used directly. In this database ^{1}H chemical shifts are referenced to TSP or DSS, and no corrections for reference, pH, or temperature bias were applied (Wishart et al., 1995b).

For each amino acid, only the ^{1}H spins which make a clear TOCSY correlation with the Hα proton were considered for the study. This excludes the aromatic protons of aromatic residues, the methyl group of methionine, the HN terminal protons of basic residues as well as all the labile protons of alcohols and acids.

A first selection was made out of the BMRB by removing all protein entries with paramagnetic centre, and by considering only amino acids with a minimum number of assigned resonances. This minimum number was set to 2 for G and A, 3 for T and AMX residues, 4 for E, Q, M and V, 6 for the long chain residues (L, I, P) and 7 for K and R.

A final selection was then made by removing proteins not fully assigned (i.e. with less than 4 assignments per residue in average). Then all redundant protein entries were removed, in order to create a set of proteins which spans in an unbiased manner all the proteins present in the database and which is, at the same time, representative of the various chemical shifts found. This final set contains 142 different proteins and was used as a training set for building the different artificial neural networks, and will be referred to as the training set in this paper. The names of the selected files are available as supplementary material at the following URL: http://www.infobiosud.univ-montp1.fr/rescue/file_training.html.

The first selection was used to carry out the test procedures of the different artificial neural networks studied, with the additional condition that entries used for training were not used in the test phase. Tests were made on a total of 8033 assigned amino acid entries spanning 786 different proteins.

Neural network

In this study two different approaches have been tried. A first artificial neural network setup (called NN1 in the following) was designed to discriminate the 20 different amino acids. With this approach some systematic errors were observed for closely related amino acids such as I and L; M, E, and Q; or D and N. Another network was designed, where related amino acids are grouped, and predicted as a group in a first stage. A series of specialized networks were then trained to separate individual amino acids among the predicted groups of the first stage. The groups are presented in Table 1. This set of networks (first stage and second stage) will be called NN2 in the following.

All the artificial neural networks used in this work consist of a classical perceptron design (Rosenblatt, 1957, 1958; Rumelhart et al., 1986) in which the input data (chemical shifts) are presented to the input layer, and results (the amino acid types) are obtained from the output layer. The topology retained for this work is a classical 3-layer network with one hidden layer and simple forward connections with no data retroconnection. The schematic of the computation is shown in Figure 1. Each neuron *n* performs a weighted sum of its inputs I_i^n, and computes its output as a function of this sum.

$$x_n = \sum_i W_i^n I_i^n \tag{1}$$

Table 1. Amino acid grouping used for the artificial neural network NN2

First stage	Second stage
IL	I
	L
A	A
G	G
P	P
T	T
V	V
KR	K
	R
AMX *(FYWHDNC)*	FYWHC
	DN
AMPTX *(EQM)*	EQ
	M
S	S

The application functions used for the connections in the network are the following:

$$f(x_n) = \frac{2}{1 + \exp(-2(x_n + B_n))} - 1 \tag{2}$$

and

$$\begin{aligned} g(x_n) &= \max(0, \frac{1}{2}(x_n + 1)) && \text{iff} \quad x_n < 0 \\ &= \frac{1}{1 + \exp(-(x_n + B_n))} && \text{iff} \quad x_n \geq 0 \end{aligned} \tag{3}$$

The function $f(x)$ is used for the internal connections, and $g(x)$ is used for the output layer. These functions were chosen among other functions by trial and error as giving the best results. The parameters W_i^n (weight) and B_n (bias) are adapted for each connection by the training process (see below).

The number of chemical shifts to be entered is different for each amino acid, and may even vary for a given amino acid, because of the possibility of missing assignments. This cannot be handled easily by a perceptron design, so we used an additional fuzzy logic layer (Zadeh, 1988) in order to code on a constant number of inputs the set of chemical shifts to analyse. This layer consists of a grid on the chemical shift scale on which the position of each spectral line is coded. In order to obtain a resolution along the chemical shift finer than the grid spacing, a technique close to graphic antialiasing was chosen, in which the intensity of a grid entry is proportional to the distance of the spectral line to the middle of the division (see Figure 1).

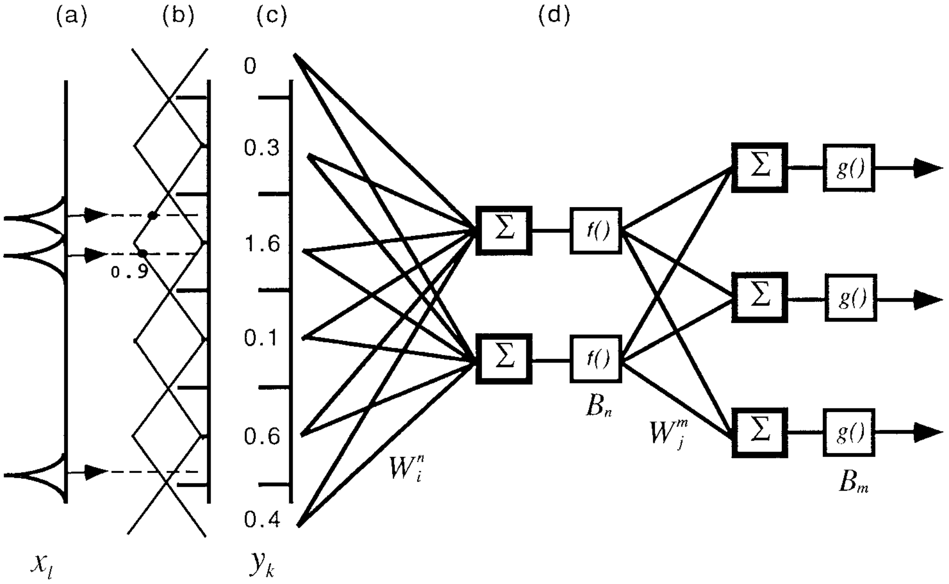

Figure 1. Outline of the structure of the artificial neural network used in this work presented here with 6 input neurons, 2 hidden neurons and 3 output neurons. (a) represents the NMR spectrum to be analysed, consisting of 3 different lines; (b) describes the fuzzy logic layer coding the chemical shift into the input layer; (c) the actual values used to feed the input layer; (d) the arithmetic operations performed by the neural network, where $\sum$ represents the summation operation, *f()* and *g()* are the non-linear transfer functions, and *W* and *B* are the weights and bias optimized during the training step.

We thus define a fuzzy logic grid as consisting of n entries, located at the positions δ_k regularly sampling the chemical shift axis with a spacing of $\Delta\delta$. An output value y_k is computed for each grid position. The following equation is used to compute the y_k values from the chemical shift values x_l present in the spectrum to be coded:

$$y_k = \sum_l \left(\max \left(1 - \frac{\|\delta_k - x_l\|}{\Delta\delta}, 0 \right) \right) \tag{4}$$

The n y_k values are then used as input of the input layer of the artificial neural network. The intensities of the lines are ignored. The grid used in this study collapses the spectral data into 32 input neurons, spanning the −2 ppm to +14 ppm range in intervals of 0.5 ppm; chemical shifts outside this range are discarded. With this coding, the positions of a few lines can be coded exactly on the grid, and the coding becomes blurrier in the case of a more densely packed spectrum. No provisions are made for overlapping or degenerate resonances.

The output layer consists of one neuron per possible output value. The ideal network should produce a 1.0 on the output corresponding to the true answer, and 0.0 on all the other outputs.

Training (optimization) of all the parameters of the network (weights and bias for each neuron input) was performed by minimizing the Euclidean distance (χ^2) between the actual output of the trained network and the ideal output. A gradient back propagation algorithm with momentum and adaptive learning rate was used (Rumelhart and McClelland, 1986).

Training of NN1 was realized on the training set as described above; 20 representatives of each amino acid type were randomly chosen from all the proteins in this set, thus realizing a training on 400 different residues. For the training of the artificial neural networks used in NN2, the number of representatives for each group was increased to 80. In any case, training was performed for a minimum of 12 500 iterations.

The artificial neural network being optimized was also evaluated against the complete test set presented above. For each spin system entry in the test set, the predicted amino acid was compared to the real one, and the ratio of correct answers was used as an overall measure of the rate of success. This rate of success was monitored during the whole training phase, and the state corresponding to the best rate of success was always selected, even if it did not correspond to the smallest χ^2.

The whole process was first implemented in version 5 of the Matlab program (Matlab, 1998) using the artificial neural network tool box. All data management was performed in the perl language. A simple version of the final optimized artificial neural networks has also been written in the perl language in order to produce a small and stand-alone version of the program. It is the perl versions of NN1 and NN2 which are used in the Internet-based version of the program. All computations have been carried out on a HP K250 computer.

In normal use, the optimized artificial neural network is presented with a series of chemical shifts extracted from a given spin-system. The artificial neural network output consists of a vector O_i with a value for each targeted amino acid. The largest value in the output vector ($O_{\max}$) is considered to be the predicted amino acid.

The difference between the actual output vector and the ideal vector for this target I_i^t is used to evaluate the reliability of the answer. The program computes the quantity $p^t(O)$:

$$p^t(O) = \exp(-\sum_i \frac{(O_i - I_i^t)^2}{\sigma_i^t}) \times R^t \qquad (5)$$

where σ_i^t is the variance of the i^{th} element of the output vector, evaluated during the test phase from the neural network output for all the amino acids of type t; and R^t is the rate of success observed for this amino acid, as given in Tables 2 and 3. A final coherence test is also added at the end of the processing, verifying that the number of spins of the predicted amino acid is compatible with the number of signals on the input. The output vector issued to the user, as well as the quantity *p(O),* is expressed in percents.

Results

database analysis

Chemical shift statistics for each resonance of the amino acids were first computed from the database selection. We verified that the mean chemical shift values are very close to the random coil values previously published (Wüthrich, 1986; Merutka et al., 1995; Wishart et al., 1995a). On the other hand, the observed standard deviations around the mean values are much larger than the variation of this mean value from one amino acid to another, and are of the order of 0.4 ppm. This amply confirms the well known fact that the value of a single chemical shift cannot assess the amino acid type, except in very special cases. It could also be observed that aromatic residues present slightly larger deviations than non-aromatic ones.

Table 2. Results of the optimized artificial neural network NN1 for the different types of residues

Residues	Rate of success *(%)*
All residues	63.5
G	91.2
A	92.2
V	94.5
L	64.9
I	82.7
P	77
T	90.8
K	92.7
R	90.3
E	39.8
Q	51.8
M	50.7
S	89
C	23.7
D	48.3
N	10.4
F	8
Y	0
W	60
H	1.9

Table 3. Result of the optimized artificial neural network NN2 for the different types of residues

First stage		Second stage	
Group name	Rate of success *(%)*	Group name	Rate of success *(%)*
All residues	91.9	All residues	79.9
I L	93.4	I	74.4
		L	70.8
A	94.5		
G	94.4		
P	96.5		
T	90.8		
V	93.9		
K R	91.7	K	91.1
		R	80.5
AMX	89.1	F Y W H C	67.9
		D N	59.2
AMPTX	93.5	EQ	71.6
		M	67.8
S	88.1		

First design

The artificial neural network with one output for each of the 20 different amino acids (NN1) is first considered. To evaluate the possibilities of such a design and to optimize its efficiency, we varied the size of the hidden layer and monitored the optimum rate of success obtained for each network. Figure 2 shows the variation of different parameters of the fully trained NN1 obtained upon varying the hidden layer size.

It can be seen that χ^2, the quantity being minimized during the training, decreases with increasing size of the hidden layer. Larger artificial neural networks contain more degrees of freedom, and are thus more capable of adapting the arbitrary function which is fitted during the training. However, it can be observed that a better χ^2 does not imply a larger rate of success, and that the rate of success estimated in the test phase drops for more than 4 hidden neurons.

This can be explained by the fact that additional degrees of freedom available in the larger artificial neural networks are used to fit features in the training set which are not relevant to the predicting process, and that the larger artificial neural networks somehow overfit the available data. This interpretation is strengthened by the fact that the χ^2 computed on the test set increases for the largest networks.

From this study, we chose to use the geometry with 4 neurons in the hidden layer. The rate of success of the optimized NN1, as obtained on the test set, is 63.5%. This means that for 63.5% of the residues in the test set, the correct amino acid is inferred solely from the set of chemical shifts. The results are detailed in Table 2 and Figure 3. It can be seen that there is some departure from the mean rate of success depending on the amino acid tested. Some amino acids, such as G, A, V, S, T, R or K are found in 90% or more of the tested cases. On the other hand, residues such as C, N, F, Y and H are detected with a rate of success which is smaller or only marginally larger than the random value of 5%. From Figure 3 it can be seen that I and L appear as a pair and are often mistaken for each other, the group of the AMX or AMPTX spin-systems are correctly detected as a whole, but not correctly separated.

Hare and Prestegard (1994), in an analogous study, proposed to remove the chemical shifts of the HN and Hα spins from the analysis. Indeed, such spins seem to mostly bear information on the secondary structure (Wishart et al., 1991; Merutka et al., 1995). However, when we tried to build an artificial neural network using only the side chain spins, the system was found to be much less efficient.

Second design

The second approach (NN2) with amino acids grouped in similar classes (Table 1) was then applied. This grouping is largely designed from the results of the NN1 study. With this second approach the analysis is performed in two steps: a first artificial neural network determines in which group a given spin-system falls, then if this group consists of more than one amino acid, a second independent network, specialized on this group, determines more precisely the amino acid type. No attempt was made to separate down to a unique amino acid in each class. The NN2 approach consists thus of a set of five related artificial neural networks; one for the first separation step, and one for each sub-group separation: I-L, K-R, AMX and AMPTX. The reliability (Equation 5) is computed for each predictive step, and the user may stop at the first stage, or decide to use the final prediction. The reliability coefficient returned to the user at the issue of the second step is the product of the reliability coefficients of each steps, normalized to the overall rate of success.

In a procedure similar to NN1, the optimum number of neurons in the hidden layer was determined for each artificial neural network of the NN2 set. The network used in the first separation step has 6 hidden neurons, and the specialized ones have 2 or 3 neurons in their hidden layer.

The results obtained with this second approach are given in Table 3. The mean rate of success of the first stage on the test set is 91.9%, and the cumulated rate of success of the two-stage analysis is 79.9%.

It can be seen that the global figures obtained with NN2 are much better than those obtained with NN1. All the amino acids and groups are predicted with rates of success higher than 88%, whereas in the previous approach, two thirds of the residues were below this threshold. Except for V and S, all the residues which are handled individually by this second approach are predicted with an equal or better efficiency than in NN1.

By grouping together the amino acids that are intrinsically difficult to separate, we have eased considerably the determination process of the artificial neural network. This fact probably fully accounts for the observed gain in rate of success. However, it should be noted that the inherent complexity of the problem has not really been solved, but is mostly hidden in the groups of the final stage.

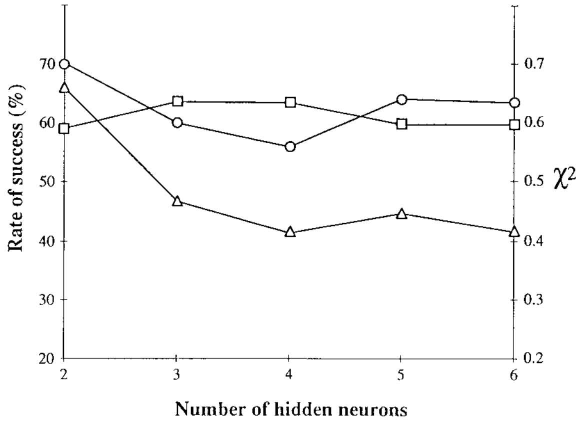

Figure 2. The value of the χ^2 after optimization (triangles), the χ^2 obtained on the test set (circles), and the global rate of success (squares, in percent) for different sizes of the hidden layer.

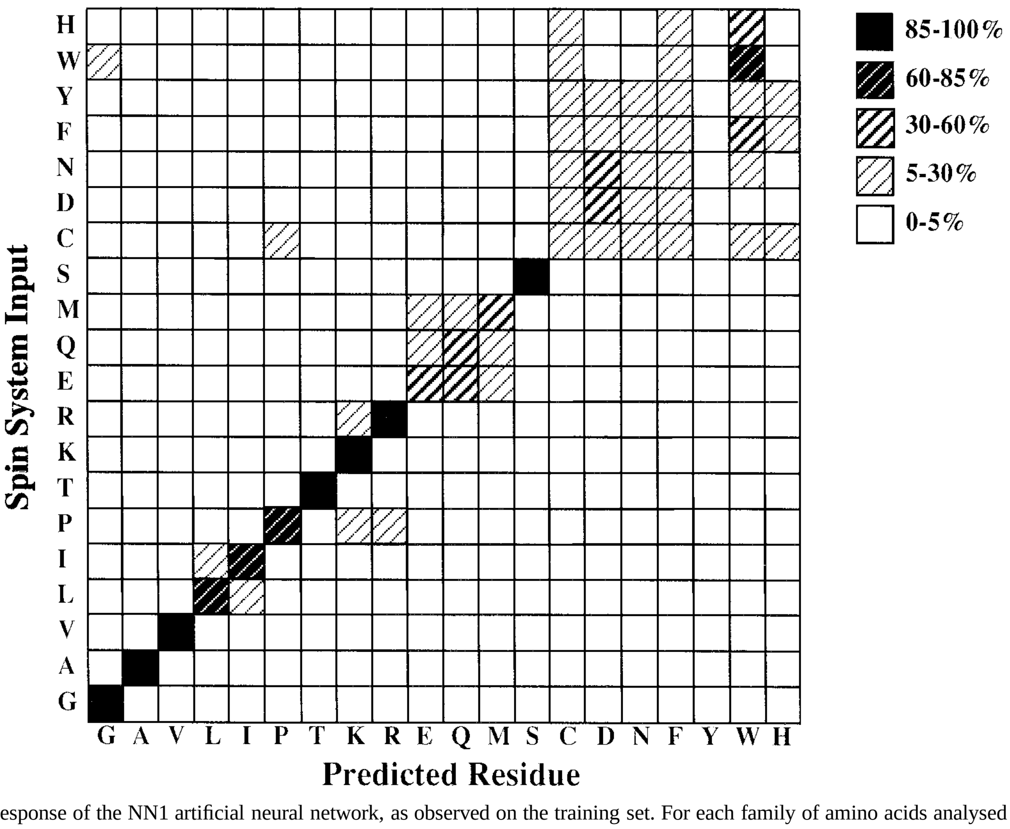

Figure 3. Response of the NN1 artificial neural network, as observed on the training set. For each family of amino acids analysed (located on the left) the number of answers is given graphically: white: less than 5%; light crosshatching: 5 to 30%; medium crosshatching: 30 to 60%; crossed dark crosshatching: 60-85%; black: more than 85%.

Finally, it should be noted that the NN2 approach gives the user a choice between a good separation and a good confidence with a less precise separation. This latter analysis can be well suited for automatic assignment analyses.

Reliability coefficients

One important feature of the program presented here is that it evaluates the confidence of its answers, using a reliability coefficient (Equation 5). To check how this reliability relates with the quality of the answer, we plot in Figure 4 the rate of correct answers of the NN2 neural network versus the reliability coefficient. It can be seen that, as the rejection threshold on the reliability coefficient is raised from 0 to 90%, the number of analysed entries is reduced (given here as the percentage of rejected entries ranging from 0 to 65%) and at the same time the rate of success on the remaining entries increases substantially. For instance, one half of the erroneous answers are rejected by selecting only the answers with reliability coefficients over 60%, while 75% of the data can still be analysed.

Optimization

Other artificial neural network geometries were also tested, such as a one-step network predicting on the final 14 groups presented in Table 1, or a two-stage network predicting down to the single amino acid. However, it is the approach presented here which yielded the best balance between global efficiency and usefulness.

Proteins present in the BMRB are usually not fully assigned, and many assigned amino acids are present with missing chemical shifts. The complete assignment statistics of the training set are given in Table 4. It appears that the assignment of long-chain amino acids is slightly less complete than short-chain ones. On the other hand, no simple correlation appears between the mean assignment level and the rate of success as presented in Table 3. Indeed, the mean assignment level of the BMRB is faithfully reported in the training and test sets, and the artificial neural networks presented here are optimized for such a level of missing entries.

However, because a published assignment, as given in the BMRB, is always more comprehensive than the level of analysis of an assignment still in progress, we thought that the level of assignment, as found in the BMRB, could be too high for every day practice. We thus trained another set of artificial neural networks, lowering the gap level needed to accept a given amino acid in the training set. This new set of artificial neural networks presents slightly lower rates of success than the previous set. However, it presents mixed results on the two real cases presented below, and its usefulness has yet to be further investigated.

The fuzzy logic layer used as input to the artificial neural network is certainly important for the quality of the results presented here, as it permits the formatting of the input data independently of the number of entered resonances. With this coding, no chemical shift information is lost as long as all the resonances of the spin system being considered are separated by more than the grid spacing. When two resonances closer than the grid spacing fall in different grid cells or if resonances become even more clustered, precise information on the spectral positions is lost, but the information on the number of observed resonances always remains.

The grid spacing chosen for this study is 0.5 ppm. We carried out tests with a grid spacing of 0.25 ppm, but did not observe significant improvements of the rate of success; moreover, the impact on the computation time was very large.

Real cases

Two proteins currently studied in our laboratory were used to evaluate the behaviour of the artificial neural network under circumstances close to a real assignment process.

The proteins used for the tests are two small oncoproteins, involved in rare forms of human leukaemia, the solution structures of which were recently determined in our laboratory (Barthe et al., 1997; Yang et al., 1998).

Both of these proteins present a new folding pattern. The first protein, called P13, contains 116 residues and is mostly in β-structure, with an original eight-stranded β-barrel fold; the second, called P8, contains 78 residues and is mostly α-helical, structured as three anti-parallel helices, stabilized by three disulfide bridges. It should be noted that no protein homologous to any of these two proteins is present in the BMRB.

The ^{1}H NMR spectra of these two proteins are assigned at 90.8% and 85.4% of the possible chemical shifts for P13 and P8, respectively. The chemical shifts of the two proteins, as available from the assignment listings, were directly input into the program; the results are given in Table 5 for the first and second stages of the NN2 program; with various reliability levels. It was observed that most of the assignment errors were

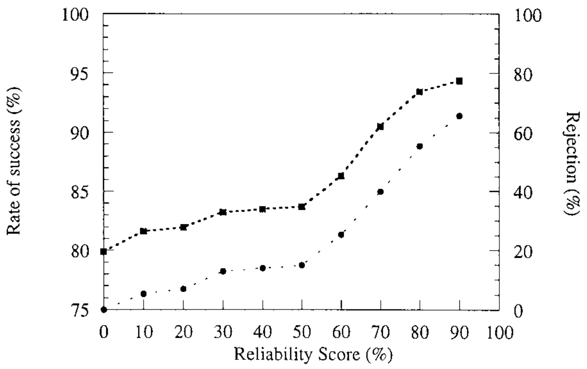

Figure 4. Rate of success (correct answers ratio) of the NN2 neural network (closed squares–left axis) for all the answers over a given reliability coefficient (abscissa). The percentage of entries below the given reliability is also given (open circles–right axis).

Table 4. Assignment completeness in percent, observed on the training set. The mean value is given in the case of stereospecific assignments

	HN	Hα	Hβ	Hγ	Hδ	Hϵ	Mean
G	95.65	99.19					98.01
A	95.37	99.75	99.51				98.21
V	92.83	100	100	99.37			98.31
L	93.50	99.00	95.25	91.00	94.50		94.71
I	95.04	100	99.17	91.32[a]	92.56		95.10
				96.28			
P		97.25	91.76	87.05	93.72		91.76
T	96.02	99.33	99.33	100			98.67
K	93.22	99.36	92.37	71.82	65.67	64.19	78.07
R	95.39	98.46	94.78	81.59	82.51		88.95
E	92.98	99.69	97.25	85.67			93.09
Q	91.17	99.50	96.07	90.19			93.87
M	84.33	100	93.97	77.70			87.95
S	93.42	98.63	98.63				97.33
C	96.64	97.68	98.19				97.68
D	94.33	99.68	99.68				98.34
N	94.63	97.70	99.42				97.79
F	93.22	99.43	99.43				97.88
Y	93.00	99.00	100				98.00
W	95.31	98.43	96.87				96.87
H	90.64	99.28	99.28				97.12

[a] Hγ methyl.

Table 5. Results of NN2 on the P8 and P13 proteins

Reliability	P8				P13			
(%)	First stage (%)		Second stage (%)		First stage (%)		Second stage (%)	
	Success	Reject	Success	Reject	Success	Reject	Success	Reject
0	88.5	0	82.1	0	81.9	0	63.8	0
10	89.4	2.5	82.4	5.1	88.6	24.1	75.6	25.9
70	87.8	15.4	84.4	42.3	90.6	26.7	83.9	51.7

found for the residues for which many chemical shifts were lacking.

Discussion

Two artificial neural network programs are presented in this manuscript. The first, called NN1, is set up so as to directly extract the type of the amino acid under study from the values of the observed ^{1}H chemical shifts. This program appears not to be usable as it presents very high error rates for certain types of amino acids. The second, called NN2, is very similar to NN1 except that some grouping has been made in the possible answers of the program. This different design permits to present a program which yields the correct answer in 80% of the tested cases.

Such a rate of success is certainly higher than the values previously reported in comparable previous studies (Hare and Prestegard, 1994; Huang et al., 1997), and should be high enough for this tool to be of real help during the assignment process.

When looking at Table 5, it can be seen that P8 is predicted by NN2 with a higher rate of success than P13 (82% versus 64%). This difference could be accounted for by the fact that α-helix structures are more often present in the BMRB database, and consequently in the training set used here, and that β-sheets are less often present. The mean ratio of correct answers for these two tests is slightly lower than the mean value obtained on the test case (80% for NN2), but this is certainly due to the fact that both P8 (three-helix bundle) and P13 (eight-stranded β-barrel fold) represent folding patterns absent in the BMRB.

The reliability coefficient (Equation 5) can be used to modulate the confidence of the output given by the program. Figure 4 shows that the number of correct answers can be increased from 80% to values over 90%, at the cost of rejecting 40% of the analysed spin systems. The number of wrong answers is strongly reduced by the operation. This feature can be of great help when one searches for a small number of secure answers, for instance in the case of partial assignment, or to help an automatic assignment program.

One distinctive characteristic of artificial neural network studies is that it is possible to design and optimize an artificial neural network for a given task, but it is usually difficult to tell how the produced artificial neural network actually works, and on which features it constructs its discrimination.

In our case, much can be said on how this prediction process takes place. It appears that the artificial neural networks presented here do not work by extracting some striking and characteristic features of a given amino acid (special chemical shift, number of resonances, etc.), but rather, use subtle global correlations between the different chemical shifts entered. For instance, long-chain amino acids are regularly recognized given only the HN, Hα and Hβ chemical shift values; in certain cases wrong glycine predictions are given for entries with more than 3 chemical shifts, thus underlining the necessity for the final coherence test mentioned in the Materials and methods section.

Global correlations between the different chemical shifts do exist, as it is well known that the secondary structure of a given amino acid has some known influence on the mean values of the HN and Hα chemical shifts. The CSI analysis (Wishart et al., 1991) is indeed based on such a phenomenon.

This influence of the environment on the chemical shifts might well be more general, not being restricted to the secondary structure but also including for instance the type of the preceding and following amino acids (Wishart et al., 1995a), the global tertiary folding, the presence of shielding or deshielding aromatic groups, etc. It might also be more subtle, as the effect may be a complex and correlated move of all the spectral lines of the studied amino acid. It is possible that the programs presented here base the discrimination on the extraction of these subtle and correlated moves away from the random coil shifts.

The tool presented here is certainly accurate enough to be used on a regular basis, and it is worth envisioning now what are the reasons of the errors which are still observed, and how these errors could be eventually detected or circumvented. Systematic errors, such as an amino acid being regularly taken for another, predominant in the NN1 approach, are nearly absent in NN2. This is indeed due to the fact that the inherently difficult cases have been grouped together, somehow hiding the difficulties in the design itself. However, prediction rates below 80% are still observed.

There are many examples in the literature of chemical shifts which are observed to be very far from the random coil values. These exotic chemical shifts are usually rationalized by the presence of unusual contacts with ring current-inducing structures of irregular 3D patterns, or of some exotic pK_a or chemical activation. These effects are quite complex and probably not easy to analyse during the course of the assignment itself.

Such exotic chemical shifts are also present in the training set as well as in the test set, as no special attention was paid to the value of the chemical shifts during the selection process (except for the rejection of the paramagnetic proteins altogether). It is thus expected that the neural network presented here should be able to handle outliers without too many errors. To check more precisely the behaviour of the NN2 neural network, we investigated for each amino acid type how the NN2 rate of success relates with respect to the spread in chemical shifts (data not shown).

In all cases it is observed that the rate of success drops for the extreme outliers (representing a small percentage of the total database); however, this drop is more or less pronounced depending on the amino acid type. Looking to the remaining 90 percent of the database, two distinct effects can be separated: for some amino acid types, the rate of success does not depend significantly on the spread of the chemical shifts from their mean values, this corresponds roughly to the amino acids with mean rates of success over 90% (see Table 3) plus the aromatic subset (A, G, T, V, F, Y, W, and to a lesser extent I, C, P, and H); for the other amino acids, the quality of the determination is directly related to the distance of the chemical shifts from their mean values: the closer to this mean value, the better the prediction.

It should finally be noted that the level of errors reported for this work is perturbed by the possible errors in the BMRB itself. There are several reports in the literature of errors in the NMR analysis due either to assignment errors (Massefski et al., 1990; Bontems et al., 1991) or even to primary structure errors (Foray et al., 1993; Nishio et al., 1998), in addition to the possible typing errors in the data entry of the BMRB entries.

The tool presented here can certainly be used successfully in several typical situations. First, it can serve as an aid during manual inspection and assignment of protein and peptide spectra. As such, it has been inserted as a tool in our assignment program (Malliavin et al., 1998) available as a module of the Gifa program (Pons et al., 1996). While using this program in our laboratory, we have found this tool to be very useful in helping people who are new to the assignment art. We also have found that the robustness of the artificial neural network approach permits the use of this tool in situations far from the learning conditions.

Secondly, this tool can certainly be used in an automatic assignment project by coupling its output to some primary sequence analysis. However, this approach would necessitate analysing, in some manner, the NOESY spectra, in order to extract some information on the preceding and following residues.

The approach presented here has been nicknamed RESCUE standing for RESidue prediCtion with neUral nEtworks. It consists of a set of programs written in the perl and Matlab languages. A CGI program implementing all the functions presented here can be used from our web site at: http://www.infobiosud.univ-montp1.fr/rescue. A stand-alone version of RESCUE, written in perl, can also be obtained from the web site or from the authors.

Conclusions

We have demonstrated here that for a protein or a structured polypeptide, the values of the ^{1}H chemical shifts contain some information on the type of the amino acid. We have shown that under certain conditions, this information can reliably be used to characterize the amino acids under study. We have presented a set of programs called RESCUE, based on the artificial neural network technology, that achieve this determination with errors below 10% in certain cases.

Previous attempts to use artificial neural networks to analyse chemical shift have been reported in the literature for proteins (Hare and Prestegard, 1994; Choy et al., 1997; Huang et al., 1997) or oligosaccharides (Radomski et al., 1994). However, this study presents a much higher prediction success. This can be accounted for by the following specific features: (i) the artificial neural network design has been augmented by a fuzzy logic input layer which permits to format adequately the values to be input into the artificial neural network; (ii) training and test have been realized on a large database of assigned chemical shifts, issued from the BMRB; (iii) grouping of amino acids inherently difficult to separate greatly improves the efficiency of the approach while reducing only marginally the use of this tool.

In this study, only ^{1}H chemical shifts were used as we concentrated on data as would be detected from a 2D TOCSY. It is probable that adding to the input data the ^{15}N or ^{13}C chemical shifts would give some room for prediction improvement. However, this improvement can only be warranted if the training and test databases are of sufficient size. The BMRB does not appear to meet this criterion fully at present. However, this could change rapidly considering the increasing rate of protein structure determination with the labelled protein technique; provided the authors deposit their assignment information to the BMRB database.

Acknowledgements

The authors want to acknowledge Julien Delsuc for initiating this study, and André Aumelas for fruitful discussions. This work was funded by CNRS, INSERM and UM1, and was realized as part of the Infobiosud project.

References

Barthe, P., Yang, Y.S., Chiche, L., Hoh, F., Strub, M.P., Guignard, L., Soulier, J., Stern, M.H., van Tilbeurgh, H., Lhoste, J.M. and Roumestand, C. (1997) *J. Mol. Biol.*, **274**, 801–815.

Bontems, F., Roumestand, C., Gilquin, B., Ménez, A. and Flavio, T. (1991) *Science*, **254**, 1521–1523.

Choy, W.Y., Sanctuary, B.C. and Zhu, G. (1997) *J. Chem. Inf. Comput. Sci.*, **37**, 1086–1094.

Foray, M.F., Lancelin, J.M., Hollecker, M. and Marion, D. (1993) *Eur. J. Biochem.*, **211**, 813–820.

Hare, B.J. and Prestegard, J.H. (1994) *J. Biomol. NMR*, **4**, 35–46.

Huang, K., Andrec, M., Heald, S., Blake, P. and Prestegard, J.H. (1997) *J. Biomol. NMR*, **10**, 45–52.

Kuszewski, J., Gronenborn, A.M. and Clore, G.M. (1995) *J. Magn. Reson.*, **B107**, 293–297.

Malliavin, T.E., Pons, J.L. and Delsuc, M.A. (1998) *Bioinformatics*, **14**, 624–631.

Massefski, W., Redfield, A.G., Hare, D. and Miller, C. (1990) *Science*, **249**, 521–524.

Matlab (1998) The MathWorks, Inc., Natick, MA.

Merutka, G., Dyson, H.J. and Wright, P.E. (1995) *J. Biomol. NMR*, **5**, 14–24.

Nishio, H., Nishiuchi, Y., De Medeiros, C.L., Rowan, E.G., Harvey, A.L., Katoh, E., Yamazaki, T., Kimura, T. and Sakakibara, S. (1998) *J. Pept. Res.*, **55**, 355–364.

Pons, J.L., Malliavin, T.E. and Delsuc, M.A. (1996) *J. Biomol. NMR*, **8**, 445–452.

Radomski, J.P., Van Halbeek, H. and Meyer, B. (1994) *Nat. Struct. Biol.*, **1**, 217–218.

Rosenblatt, F. (1957) Report, Cornell Aeronautical Laboratory, Ithaca, NY.

Rosenblatt, F. (1958) *Phys. Rev.*, **65**, 386–408.

Rumelhart, D.E., Hinton, G.E. and Williams, R.J. (1986) In *Parallel Distributed Processing* (Eds. Rumelhart, D.E. and McClelland, J.J.), MIT Press, Cambridge, MA, pp. 318–362.

Rumelhart, D.E. and McClelland, J.J. (1986) *Parallel Distributed Processing*, MIT Press, Cambridge, MA.

Seavey, B.R., Farr, E.A., Westler, W.M. and Markley, L. (1991) *J. Biomol. NMR*, **1**, 217–236.

Wishart, D.S., Bigam, C.G., Holm, A., Hodges, R.S. and Sykes, B.D. (1995a) *J. Biomol. NMR*, **5**, 67–81.

Wishart, D.S., Bigam, C.G., Yao, J., Abildgaard, F., Dyson, H.J., Oldfield, E., Markley, J.L. and Sykes, B.D. (1995b) *J. Biomol. NMR*, **6**, 135–140.

Wishart, D.S. and Sykes, B.D. (1994) In *Nuclear Magnetic Resonance, Pt C*, Vol. 239 (Eds. Oppenheimer, T.L.and James, N.J.) Academic Press, San Diego, CA, pp. 363–392.

Wishart, D.S., Sykes, B.D. and Richards, F.M. (1991) *J. Mol. Biol.*, **222**, 311–333.

Wüthrich, K. (1986) *NMR of Proteins and Nucleic Acids*, Wiley, New York, NY.

Yang, Y.S., Guignard, L., Padilla, A., Hoh, F., Strub, M.P., Stern, M.H., Lhoste, J.M. and Roumestand, C. (1998) *J. Biomol. NMR*, **11**, 337–354.

Zadeh, L. (1988) *Computer*, **21**, 83–93.