Journal of Biomolecular NMR, **23:** 263–270, 2002.
KLUWER/ESCOM
© 2002 *Kluwer Academic Publishers. Printed in the Netherlands.*

Protein sequential resonance assignments by combinatorial enumeration using ^{13}Cα chemical shifts and their (*i, i*−1) sequential connectivities

Michael Andrec* & Ronald M. Levy
Department of Chemistry and Chemical Biology, Rutgers, The State University of New Jersey, 610 Taylor Road, Piscataway, NJ 08854-8087, U.S.A.

Received 5 April 2002; Accepted 27 June 2002

Key words: computational complexity, high throughput, structural genomics, tree search, triple-resonance

Abstract

The need for the structural characterization of proteins on a genomic scale has brought with it demands for new technology to speed the structure determination process. In NMR, one bottleneck is the sequential assignment of backbone resonances. In this paper, we explore the computational complexity of the sequential assignment problem using only ^{13}Cα chemical shift data and Cα $(i, i-1)$ sequential connectivity information, all of which can potentially be obtained from a single three-dimensional NMR spectrum. Although it is generally believed that there is too much ambiguity in such data to provide sufficient information for sequential assignment, we show that a straightforward combinatorial search algorithm can be used to find correct and unambiguous sequential assignments in a reasonable amount of CPU time for small proteins (approximately 80 residues or smaller) when there is little missing data. The deleterious effect of missing or spurious peaks and the dependence on match tolerances is also explored. This simple algorithm could be used as part of a semi-automated, interactive assignment procedure, e.g., to test partial manually determined solutions fo uniqueness and to extend these solutions.

Introduction

The explosion of genomic data in recent years and initiatives in structural genomics have lead to the demand for new technologies to speed the protein structure determination process. Protein structure determination by NMR as it has typically been practiced is relatively slow, insofar as it requires the collection of many multidimensional spectra, their processing and peak-picking, the sequential assignment of backbone and side chain resonances, the translation of NOE cross-peaks into specific internuclear constraints, and the generation of protein conformations which satisfy all of the available constraints. Considerable efforts have been made to automate the latter steps of this process (Moseley and Montelione, 1999), and new types of data such as residual dipolar couplings promise to substantially reduce data collection and analysis time by avoiding the especially time consuming steps of side chain resonance and NOE crosspeak assignment (Delaglio et al., 2000; Andrec et al., 2001). Another means by which increased efficiency could be achieved is by reducing the amount of time needed to acquire and process the NMR data itself, since most current automated assignment strategies require input peak lists from a relatively large number of spectra (e.g., Zimmerman et al., 1997).

There exist triple-resonance NMR experiments such as the HNCA (Kay et al., 1990) which provide interresidue connectivity information by correlating amide ^{1}H and ^{15}N shifts with intraresidue and preceding ^{13}Cα nuclei. In favorable cases, these experiments can be run on proteins with natural abundance ^{13}C (Tian et al., 2001), further increasing their usefulness in a structural genomics context. However, it has been generally accepted that there is too much chemical shift degeneracy among the Cα nuclei to allow for unambiguous sequential assignment using only the data provided by such an experiment. In order to reduce

*To whom correspondence should be addressed.
E-mail: andrec@lutece.rutgers.edu

this degeneracy, various laboratories have recently proposed assignment strategies using data from triple-resonance experiments which have been modified to allow the measurement of scalar couplings, residual dipolar couplings, and/or differential line broadenings (Zweckstetter and Bax, 2001; Tian et al., 2001). This additional data can then be used to provide additional match criteria to supplement the Cα $(i, i-1)$ connectivities (Zweckstetter and Bax, 2001) or to provide a structural filter to reduce the number of possible sequential linkages (Tian et al., 2001). In this paper, we investigate the nature of the combinatorial problem of generating a sequential resonance assignment using only ^{13}Cα chemical shifts and their $(i, i-1)$ connectivities. We will assume throughout that the appropriate three-dimensional NMR spectrum has been properly acquired, processed, and peak-picked, that the resulting peaks have been correctly collated into a list of ordered pairs of ^{13}C chemical shifts corresponding to intra- and interresidue Cα nuclei (e.g., Table 1), and that the amino acid sequence of the protein is known. We show that such data can in fact lead to correct and unambiguous sequential assignments for proteins under approximately 80 residues in size, provided that the input peak lists are sufficiently clean. We also investigate the deleterious effect of missing or spurious peaks and the dependence on match tolerances. Such a minimalist strategy is a workable approach to the sequential assignment problem for those cases where the data are accurate and complete. This study provides a baseline result for protein sequential resonance assignment using minimal connectivity information. This approach may be further developed by incorporating additional information from other NMR experiments. With modifications, the combinatorial enumeration described here may also be used simply as an aid to confirm and extend manual assignments.

Theory and methods

The algorithm used in this paper is a recursive depth-first tree search similar to approaches used by earlier combinatorially oriented automated assignment strategies (e.g., Xu et al., 1994). We begin by listing all possible spin pairs (e.g., lines in Table 1) that could be assigned to the second residue (the N-terminal residue not being visible in an amide proton detected experiment). We then take the first element of that list and generate a second list containing all spin pairs that could possibly follow that element. We then take the first element of that second list and continue in a similar manner until we either have assigned all of the residues in the protein or we reach a dead end (i.e., the list of possible successors is empty). In the either case, we then back up to the nearest unused list element and repeat the process (Figure 1). Proline residues (which do not give rise to peaks in amide-detected NMR experiments) are treated as placeholders in our algorithm. In other words, no spin pair is assigned to a proline position, and that position provides no connectivity information with regards to the feasibility of a succeeding spin pair. Our algorithm can be expressed very concisely as a single recursive subroutine, and is easily implemented in languages such as *perl* that support list manipulation and recursion.

The size of the resulting search tree is limited by constraints on the interresidue chemical shift connectivities and amino acid type, which are given by match tolerances T_{conn} (typically 0.1 ppm) and T_{aa} (ranging from 3.6 to 7.6 ppm). For example, suppose that the $(i, i-1)$ spin pair (44.97 ppm, 54.25 ppm) has been assigned to residue number 2 and that residue 3 is a glutamic acid. Candidate spin pairs for residue 3 must have an $i-1$ chemical shift in the range of $44.97 \pm T_{conn}$ ppm, and an i chemical shift in the range of $56.6 \pm T_{aa}$ ppm, where 56.6 ppm is the random coil shift for glutamic acid (Wishart and Case, 2001). Since most triple-resonance experiments are detected through the amide proton, there will be no $(i, i-1)$ spin pairs corresponding to proline residues or the N-terminal residue. However, the Cα chemical shifts corresponding to those residues do appear as the $i-1$ shift of the spin pair for the succeeding residue, and we insist that that shift lie within $\pm T_{aa}$ ppm of the random coil shift for prolines or the N-terminal residue. It should be noted that our search procedure is deterministic and exhaustive, and therefore is guaranteed to find all possible sequential assignments consistent with the T_{aa} and T_{conn} match tolerances.

The algorithm was tested on a total of nineteen peak lists ranging in size from 33 to 85 residues. Eighteen of these were ‘synthetic’ lists generated using chemical shift data from the BioMagResBank (BMRB) (Seavey et al., 1991, http://www.bmrb.wisc.edu) re-referenced by the Wishart laboratory (http://redpoll.pharmacy.ualberta.ca/RefDB). The peak lists were generated by creating $(i, i-1)$ spin pairs based on the sequential assignment in the database after adding an error ε to each $i-1$ shift. Each ε was independently and identically distributed according to $0.6\ U\ (-0.09, 0.09) +$

Table 1. An example of the input chemical shift data used in this study. Shown is the data for the rubredoxin hydrophobic core mutant (W3Y, I23V, L32I) obtained by Tian et al. (2001) (data courtesy of J.H. Prestegard). Index numbers have been assigned arbitrarily, and asterisks indicate missing data due to spectral overlap or missing peaks

Index	Cα(i) (ppm)	Cα(i − 1) (ppm)	Index	Cα(i) (ppm)	Cα(i − 1) (ppm)
2	59.08	59.82	30	55.71	54.97
3	59.79	58.28	31	57.05	65.67
4	54.93	58.00	32	52.70	57.03
5	55.71	63.96	33	53.86	56.97
6	51.50	57.57	34	56.97	55.39
7	51.37	45.93	35	60.49	62.17
8	60.44	45.88	36	56.91	63.56
9	57.57	59.26	37	54.76	51.75
10	56.35	55.71	38	63.96	45.47
12	59.30	60.45	39	53.20	44.67
13	61.98	55.71	40	55.56	58.76
14	58.01	53.86	41	56.95	59.64
15	58.86	51.48	42	58.77	61.95
16	58.17	61.19	43	52.61	57.02
17	59.65	53.15	44	53.15	56.91
18	60.78	55.57	45	45.48	64.70
19	61.20	46.28	46	57.02	58.82
20	58.28	56.62	47	45.92	59.10
21	61.10	63.71	48	45.88	59.04
22	62.43	59.07	49	60.23	*
23	56.63	54.74	50	46.26	52.58
24	59.10	62.42	51	44.66	52.78
26	57.12	*			
28	55.41	60.28			
29	59.04	61.09			

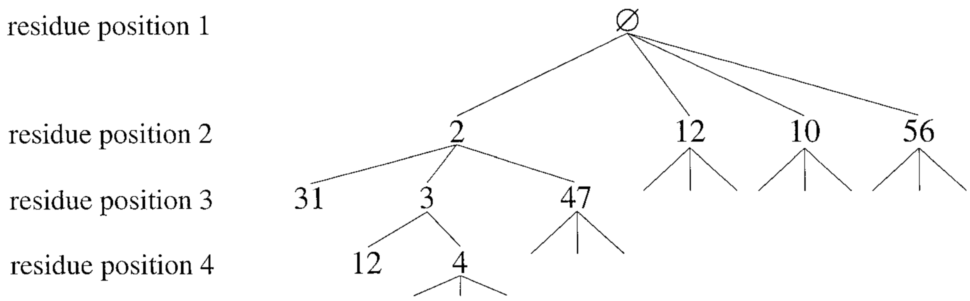

Figure 1. A fragment of a hypothetical search tree for the sequential assignment process. Since the N-terminal residue cannot be seen in amide proton detected NMR experiments, no spin pairs can be assigned to residue position 1. There are four spin pairs (2, 12, 10, and 56) that could potentially correspond to residue position 2. We examine the first of these (spin pair 2), and find that three spin pairs (31, 3, and 47) could potentially follow it. Taking the first of these (spin pair 31), we find that none of the remaining spin pairs satisfy the requirements for following spin pair 31. This constitutes a dead end, and we move on to the next nearest unexamined spin pair (spin pair 3 at residue position 3). It does have potential successors (spin pairs 12 and 4). Spin pair 12 at residue position 4 is another dead end, therefore we continue on to spin pair 4, and proceed in a similar manner until we have completely searched the entire tree. Note that the same spin pair can appear in more than one place in the tree (e.g., spin pair 12 at residue positions 2 and 4), provided that no repeats occur on a given descending path. Proline residues are treated as placeholders to which no spin pairs are assigned (see text for details).

0.4 $N(0, 0.005)$, where $U(a, b)$ is the uniform probability density between a and b, and $N(m, s)$ is the normal probability density with mean m and standard deviation s (in units of ppm). Only those regions of the proteins with no missing data (as shown in Table 2) were considered, therefore the data in these peak lists are complete. The nineteenth was an experimental peak list for rubredoxin obtained from the Prestegard laboratory (Tian et al., 2001) (Table 1). In contrast to the 18 'synthetic' peak lists, this data set contains two spin pairs with missing $i-1$ data due to spectral overlap. All calculations were done using a *perl* script on an SGI server with a 180 MHz MIPS R10000 processor. The software used in this study is available from the authors upon request.

Results and discussion

Combinatorial searches were first performed on the nineteen lists of $(i, i-1)$ spin pairs using a connectivity tolerance T_{conn} of 0.1 ppm and an amino acid tolerance T_{aa} of 6.5 or 7.0 ppm (Table 2). While a connectivity tolerance of 0.1 ppm is fairly tight by some standards, it is not unreasonable, especially since the digital resolution in the ^{13}C dimension can be made larger than usual as only one NMR spectrum is needed for the assignment. Of course, this will be limited by increased T_2 relaxation for larger proteins, however direct assignment methods such as those described here will likely be of limited utility for such proteins even for small T_{conn} values. A T_{aa} value of approximately 7.0 ppm is consistent with previous estimates of the variability of ^{13}Cα chemical shifts (Grzesiek and Bax, 1993). For most of the proteins examined, these choices of T_{conn} and T_{aa} gave a very small number of possible assignments, typically 1 or 2, and required less than 30 minutes of CPU time for proteins under 70 amino acids long (Table 2). The two notable exceptions to this were the *de novo* designed 3-helix bundle protein (BMRB #4126), which gave 32 possible assignments at $T_{aa} = 7.0$ ppm, and the cardiotoxin from *Naja Atra* (BMRB #4966), for which no assignment could be found. The distribution of secondary shifts (equal to the observed shift minus the random coil shift) for the BMRB #4966 data set appears to be somewhat upfield shifted (perhaps due to residual misreferencing), resulting in one of the residues having a -7.57 ppm secondary shift. Since this is larger than our cutoff of ± 7.0 ppm, no assignment was found. The reasons for the relatively large number of assignments for BMRB #4126 may be due to the fact that this data set has more chemical shift degeneracy than typical among the other data sets studied. We can measure the degree of degeneracy by a mean connection number, which we define to be the average number of $(i, i-1)$ spin pairs which can possibly precede or succeed a given spin pair at a given T_{conn} tolerance irrespective of amino acid type. The BMRB #4126 peak list has a mean connection number of 4.8 at $T_{conn} = 0.1$ ppm. This is higher than the connection numbers for the other proteins studied, which typically are in the range of 3 to 4, and is consistent with the hypothesis that the large number of assignments for BMRB #4126 is due to unusually large chemical shift degeneracy.

We next investigated the degree to which the T_{aa} tolerance could be made tighter and still give correct solutions. Since in our case the correct assignments are known, the minimal T_{aa} which will still give the correct assignment for a given protein is simply the secondary shift with the largest absolute magnitude. The results of the combinatorial search using those T_{aa} tolerances are shown in Table 2 as the second entry for each protein. As expected, the number of possible assignments using the minimal T_{aa} are generally smaller (going from 32 down to 2 for BMRB #4126), and the CPU time required decreases substantially (by nearly a factor of 80 in the case of BMRB #1642). Of the nineteen cases tested, only five (BMRB #4223, #4126, #4160, #4162, and rubredoxin) did not give a unique solution at the minimal T_{aa}. Of these, four resulted in only two solutions, and for three of those (BMRB #4126, #4160, and rubredoxin) the two solutions were identical except for the interchange of two residues (13 and 37, 9 and 20, and 9 and 42, respectively).

Since the minimal T_{aa} is *a priori* unknown, it is important to know whether it could be reliably found in practice. In those cases where the minimal T_{aa} gives only one solution, that solution is the correct one and obviously any further reduction in T_{aa} would give no solutions. Therefore, in these cases the minimal T_{aa} can be found to arbitrary precision by simple bracketing and bisection. Specifically, if one has found a T_{aa} value $T_{aa}^{(l)}$ which gives no solutions, and a larger T_{aa} value $T_{aa}^{(u)}$ which gives one solution, then one can rerun the combinatorial search algorithm using a T_{aa} value of $T_{aa}^{(m)} = 1/2(T_{aa}^{(l)} + T_{aa}^{(u)})$. If the $T_{aa}^{(m)}$ tolerance gives no solutions, then we set $T_{aa}^{(l)}$ equal to $T_{aa}^{(m)}$, otherwise we set $T_{aa}^{(u)}$ equal to $T_{aa}^{(m)}$. This procedure can

Table 2. A summary of results for the combinatorial assignment procedure (in order of increasing number of residues). All results shown here were obtained using $T_{conn} = 0.1$ ppm. Each protein has two entries, the second of which corresponds to the minimal T_{aa} value (see text)

Protein (residue range)	BMRB accession number	Number of residues in peak list	T_{aa} (ppm)	Number of solutions	CPU time (min:sec)[a]
p53 dimer(1-33)	4934	33	7.00	2	0 : 01
			4.37	1	0 : 01
Ovomucoid 3rd domain (17-49)	4864	33	7.00	1	0 : 01
			4.98	1	0 : 01
Cardiotoxin from *Naja Atra* (21 − 60)	4966	40	7.00	0	0 : 15
			7.57	1	3 : 16
Lac repressor DNA complex (4-46)	4813	43	7.00	1	0 : 02
			6.19	1	0 : 01
TATA binding protein (4-49)	4223	46	7.00	2	0 : 01
			5.35	2	0 : 01
Chicken cartilage matrix protein (1 − 47)	4055	47	7.00	4	0 : 07
			4.85	1	0 : 02
RNase H1 (16-63)	4424	48	7.00	1	0 : 01
			3.64	1	0 : 01
M13 major coat protein (1-50)	4209	50	7.00	3	0 : 05
			5.78	1	0 : 05
RNA1 modulator protein (7-57)	4072	51	7.00	1	3 : 11
			5.92	1	0 : 30
Rubredoxin (1-53)	–[b]	53	7.00	6	1 : 45
			5.25	2	0 : 17
HHCC domain of HIV 1 integrase (1 − 55)	4619	55	7.00	1	5 : 58
			6.09	1	2 : 14
Apokedarcidin (1-56)	4036	56	7.00	1	0 : 05
			4.13	1	0 : 01
de novo designed 3-helix bundle protein (3-58)	4126	56	7.00	32	10 : 56
			4.80	2	0 : 30
Tn916 integrase DNA-binding domain (3-63)	4160	61	7.00	2	1 : 11
			4.78	2	0 : 10
Neural cell adhesion molecule module-1 (1-63)	4162	63	7.00	6	29 : 49
			4.72	3	0 : 30
Adenylate kinase complex w/ inhibitor AP5A (1-67)	4193	67	7.00	1	0 : 08
			5.20	1	0 : 02
EH1 domain of mouse Eps15 (2-69)	4140	68	7.00	2	6 : 11
			5.45	1	0 : 13
Tendamistat (1-74)	1642	74	7.00	1	160 : 02
			5.05	1	2 : 03
Phosphocarrier protein (1-85)	2371	85	6.50	1	≈ 78 h
			5.90	1	≈ 10 h

[a] CPU times of less than one second have been rounded up to one second.

[b] see Table 1.

then be repeated until the minimal T_{aa} has been found to the desired precision.

In cases where the minimal T_{aa} gives more than one solution, it is conceivable that reducing T_{aa} further might eliminate the correct solution while retaining the others, thereby leading to an erroneous assignments. To test this, we ran the combinatorial assignment algorithm using a T_{aa} tolerance equal to the minimal tolerance minus 0.01 ppm for the five proteins that gave more than one solution at the minimal T_{aa}. In all five cases, no solutions were found. If this result turns out to be general, then the correct solution will not he missed by setting T_{aa} too low, and therefore the minimal T_{aa} can always be found by bracketing and bisection. It is not clear how general this result may be in practice, especially when the data may be corrupted by inaccurate peakpicking, though it is interesting to note that no such problem arose in the case of the experimental rubredoxin peaklist. Nonetheless, a user of this algorithm should exercise caution when using small T_{aa} values.

As with any exhaustive combinatorial search algorithm, we expect the running times to increase exponentially with the size of the protein and with the size of the match tolerances. In order to explore this, we repeated the combinatorial assignment for the TATA binding protein peak list (BMRB #4223) using different values of T_{aa}. The results are shown in Figure 2. As expected, the running time increases exponentially as T_{aa} increases from 5.4 to 18 ppm, then flattens to a plateau at a CPU time of approximately 50 min for very large T_{aa} (where the amino acid type information is effectively being ignored in the assignment process). The location of the 'knee' at approximately 18 ppm coincides with the total range of ^{13}Cα random coil shifts (from glycine at 45.1 ppm to trans-proline at 63.3 ppm) (Wishart and Case, 2001). The number of assignments found also increases exponentially, reaching a plateau value of nearly 1000. This result confirms the intuition of most protein NMR spectroscopists that Cα $(i, i-1)$ connectivities alone contain too much degeneracy to allow for unambiguous sequential assignment. However, the incorporation of even vague amino acid type information dramatically simplifies the combinatorics of the problem: A T_{aa} cutoff of 15 ppm reduces the number of solutions by more than an order of magnitude, while a cutoff of 10 ppm gives only five solutions. While we expect the exponential dependence of CPU time on T_{aa} to be a general feature, the details can vary considerably from protein to protein. For example, the analogous results for RNase H1 (BMRB #4424) (data not shown) also shows an exponential increase in CPU time as a function of T_{aa} which plateaus at approximately $T_{aa} = 18$ ppm, however the plateau CPU time is less than 9 s, and no more than two solutions are ever found. This dramatic difference can be explained in part by the fact that the BMRB #4424 peak list has much less degeneracy than BMRB #4223 as measured by the mean connection number (2.7 vs 3.7). Similar results are also seen for the running time as a function of T_{conn}. For example, doubling T_{conn} from 0.1 to 0.2 ppm for tendamistat (BMRB #1642) increases the running time from 2 min to 49 min but still results in only 5 solutions. It is unlikely that such a large T_{conn} value will be necessary, given that the ^{13}C digital resolution could be made appropriately large. In addition, the use of more sophisticated peak fitting procedures can produce precision in peak location estimates far in excess of the digital resolution (Prestegard et al., 1999).

The reason for the surprisingly small number of solutions is in part due to the restrictions on amino acid type imposed by T_{aa}, but is also inherent in the combinatorics of the problem itself. This can be seen, for example, if we attempt to assign only the first 30 residues of BMRB #4424 using the complete peak list and a T_{aa} of 25.0 ppm (in order to remove all amino acid type information). In this case the algorithm still returns rather quickly (less than 9 s CPU time), however it now gives 89 solutions, compared to the two solutions obtained using the full sequence. Eighty seven of these 89 solutions turn out not to be feasible for the full protein, however, since all attempts to extend them to the full 48 residues lead to dead ends. In other words, the requirement to find assignments for an entire sequence of known length itself introduces constraints on the combinatorics, whereby otherwise feasible solutions are rendered infeasible when the assignment cannot be extended to the required length using the spin pairs left over. Interestingly, reducing T_{aa} to the minimal value of 3.64 ppm yields only the correct assignment for the 30 residue truncated assignment problem. In addition to providing insight into the nature of the combinatoric problem of sequential assignment, these results also suggest that the presence of spurious peaks in the peak list due to spectral artifacts, sample heterogeneity, or slow chemical exchange may not lead to serious consequences. At worst, they may increase the number of feasible solutions, but they are unlikely to significantly increase the running time of the algorithm.

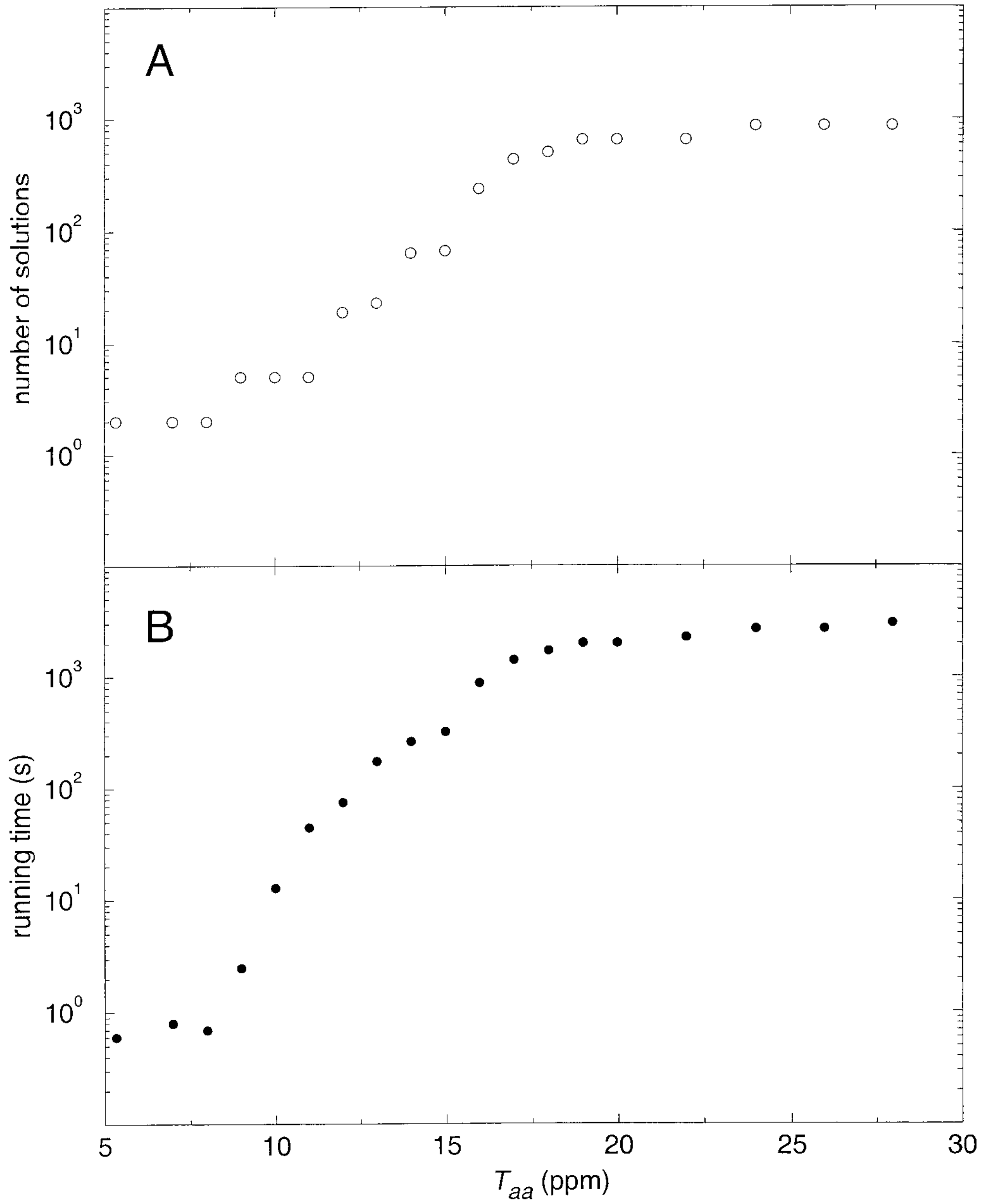

Figure 2. Dependence of the number of solutions (A) and running time (B) on the amino acid type tolerance T_{aa} for the 'TATA binding protein' (BMRB #4223) peak list.

The presence of missing data, however, can be a much more serious problem. The case where one member of a small number of spin pairs is missing is not a severe problem, as can be seen in the results for rubredoxin presented above. However, if several spin pairs are completely missing (as can happen at N- and/or C-termini or in flexible loops), this can greatly complicate the combinatorics of the problem. This is not surprising, since in our scheme we are forced to include a 'wildcard' spin pair in the peak list corresponding to each missing residue. Each of these wildcard spin pairs can potentially follow any other spin pair in the peak list, and can potentially appear at any position in the amino acid sequence. In the case of very small proteins and few missing residues, the problem is still manageable. For example, replacement of the spin pairs corresponding to the residues 2 and 3 of the p53 dimer data (BMRB #4934) with wildcards results in 18 solutions using the minimal T_{aa}, only 9 of which are distinct (due to the equivalence of the two wildcards), and requires less than 6 minutes of CPU time. As in the complete data case, reducing T_{aa} below its minimal value results in no solutions. For more missing residues or larger proteins, the problem quickly becomes unmanageable.

Conclusions

We have shown that in favorable cases one can in fact go reasonably far in backbone sequential assignments of small proteins using a simple combinatorial search on ^{13}Cα chemical shifts and their $(i, i-1)$ connectivities. It appears that such a strategy will be sufficient for proteins approximately 70 amino acids in length or smaller with little or no missing data, provided that the peak list input is sufficiently clean. Of course, this represents a tiny fraction of all proteins which NMR spectroscopists may wish to study, especially since real-world peaklists are often far from clean due to spectral artifacts, overlap, or inaccurate peak picking. However, this work provides a better understanding of the real information content of the simplest inter-residue connectivity data. Furthermore, it provides a useful baseline for what can be accomplished using exhaustive enumeration. More extensive data providing more matching criteria are easily obtainable using HNCA-type experiments via the measurement of scalar and residual dipolar couplings (Zweckstetter and Bax, 2001; Tian et al., 2001). The addition of such data will expand the feasibility of direct combinatorial methods for the sequential assignment of backbone resonances in proteins. It should be made clear, however, that the algorithm described here can be easily applied to data from multiple triple-resonance NMR experiments provided that the data have been collated into appropriate 'spin systems' and that appropriate amino acid type and connectivity tolerances have been defined.

The approach described here differs considerably from that used by human beings in the course of manually assigning NMR resonance data. In our approach, an exhaustive enumeration scheme is used, and otherwise completely feasible assignments which lead to dead ends are eliminated. In other words, our algorithm operates globally by trying to find the complete assignment of the entire protein. Human beings, by contrast, tend to generate 'islands' of assignments having varying degrees of confidence, which are then arranged and connected with the remaining resonances. It may be possible to combine the 'machine' and 'human' approaches in an interactive, semi-automated manner for systems which are not amenable to fully automatic sequential assignment. For example, one can use the software tool described here to assess the uniqueness of partial assignments generated manually, and to generate alternative assignments which satisfy the given tolerances if they exist. Alternatively, further development of the algorithm could be made to incorporate some of the processes used by a human being in solving the assignment problem, such as the construction and independent evaluation of partial solutions.

Acknowledgements

We thank the laboratory of James H. Prestegard for providing the rubredoxin peak list shown in Table 1 and for their encouragement and helpful discussions. We also thank Jean Baum and Yinan Wei for their useful input. This work was supported by a National Institutes of Health grant GM-30580.

References

Andrec, M., Du, P. and Levy, R.M. (2001) *J. Biomol. NMR*, **21**, 335–347.

Delaglio, F., Kontaxis, G. and Bax, A. (2000) *J. Am. Chem. Soc.*, **122**, 2142–2143.

Grzesiek, S. and Bax, A. (1993) *J. Biomol. NMR*, **3**, 185–204.

Kay, L.E., Ikura, M., Tschudin, R. and Bax, A. (1990) *J. Magn. Reson.*, **89**, 496–514.

Moseley, H.N.B. and Montelione, G.T. (1999) *Curr. Opinion Struct. Biol.*, **9**, 635–642.

Prestegard, J.H., Tolman, J.R., Al-Hashimi, H.M. and Andrec, M. (1999) In *Structure Computation and Dynamics in Protein NMR*, Krishna, N.R. and Berliner, L.J., Plenum Publishers, New York, pp. 311–355.

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

Tian, F., Valafar, H. and Prestegard, J.H. (2001) *J. Am. Chem. Soc.*, **123**, 1179–11796.

Wishart, D.S. and Case, D.A. (2001) *Meth. Enzymol.*, **338**, 3–34.

Xu, J., Straus, S.K., Sanctuary, B.C. and Trimble, L. (1994) *J. Magn. Reson. B*, **103**, 53–58.

Zimmerman, D.E., Kulikowski, C.A., Huang, Y., Feng, W., Tashiro, M., Shimotakahara, S., Chien, C.-Y., Powers, R. and Montelione, G.T. (1997) *J. Mol. Biol.*, **269**, 592–610.

Zweckstetter, M. and Bax, A. (2001) *J. Am. Chem. Soc.*, **123**, 9490–9491.