Journal of Biomolecular NMR (2005) 32: 219–233
DOI 10.1007/s10858-005-7944-6

Article

Probabilistic identification of spin systems and their assignments including coil–helix inference as output (PISTACHIO)

Hamid R. Eghbalnia[a,b,d,*], Arash Bahrami[a,b,c], Liya Wang[a,b,c], Amir Assadi[d] & John L. Markley[a,b,c]

[a]*Biochemistry Department, National Magnetic Resonance Facility at Madison, 433, Babcock Drive, Madison, WI, 53706, USA;* [b]*Biochemistry Department, Center for Eukaryotic Structural Genomics;* [c]*Graduate Program in Biophysics, University of Wisconsin-Madison, Madison, WI, 53706, USA;* [d]*Mathematics Department, University of Wisconsin-Madison, 811, Van Vleck Hall, 480 Lincoln Drive, Madison, WI, 53706, USA*

Received 9 March 2005; Accepted 12 May 2005

Key words: automation, backbone assignments, particle interaction model, sidechain assignments

Abstract

We present a novel automated strategy (PISTACHIO) for the probabilistic assignment of backbone and sidechain chemical shifts in proteins. The algorithm uses peak lists derived from various NMR experiments as input and provides as output ranked lists of assignments for all signals recognized in the input data as constituting spin systems. PISTACHIO was evaluate00000000d by comparing its performance with raw peak-picked data from 15 proteins ranging from 54 to 300 residues; the results were compared with those achieved by experts analyzing the same datasets by hand. As scored against the best available independent assignments for these proteins, the first-ranked PISTACHIO assignments were 80–100% correct for backbone signals and 75–95% correct for sidechain signals. The independent assignments benefited, in a number of cases, from structural data (e.g. from NOESY spectra) that were unavailable to PISTACHIO. Any number of datasets in any combination can serve as input. Thus PISTACHIO can be used as datasets are collected to ascertain the current extent of secure assignments, to identify residues with low assignment probability, and to suggest the types of additional data needed to remove ambiguities. The current implementation of PISTACHIO, which is available from a server on the Internet, supports input data from 15 standard double- and triple-resonance experiments. The software can readily accommodate additional types of experiments, including data from selectively labeled samples. The assignment probabilities can be carried forward and refined in subsequent steps leading to a structure. The performance of PISTACHIO showed no direct dependence on protein size, but correlated instead with data quality (completeness and signal-to-noise). PISTACHIO represents one component of a comprehensive probabilistic approach we are developing for the collection and analysis of protein NMR data.

Introduction

NMR resonance assignment, the key step in data analysis in which chemical shifts are assigned to individual nuclei in a covalent structure, has persisted as one of the main challenges in solving structures of proteins from NMR data. Automation of this step is desirable from the standpoints of speeding up the process of structure determination and of providing a more objective analysis of the input data. Most assignment strategies analyze data from multidimensional, multinuclear datasets by a stepwise approach that derives from one

* To whom correspondence should be addressed. E-mail: eghbalin@nmrfam.wisc.edu

described by Kurt Wüthrich and coworkers for two-dimensional proton spectra (Billeter et al., 1982; Wider et al., 1982). First, signals from different experiments are *aligned* in comparable dimensions. Individual resonances are grouped into *spin systems* that are used in the *typing* stage to score the amino acid identity of spin systems. In the *local assembly step*, sequentially related spin systems are identified. These are then mapped onto the primary sequence in the *global assembly (or mapping) stage*. Different automation programs implement each step with varying degrees of success.

A number of computerized approaches to the backbone assignment problem have been described (Gronwald and Kalbitzer, 2004). Some software packages achieve partial automation of sidechain assignments, for example, GARANT (Bartels et al., 1997) or the combination of GARANT and AUTOPSY (Koradi et al., 1998). However, to our understanding, a satisfactory, flexible, and truly automated approach to the overall assignment problem has yet to be presented. Many laboratories continue to invest considerable effort in manual assignments to ensure quality.

The need for a mechanism for modifying the allowed input data to include results from new NMR experiments and combinations of experiments can be understood by considering the problem of experiment selection. In larger proteins, differences in resonance transfer efficiencies make it difficult to obtain signals from all sidechain carbons in a single spectrum. Complete sidechain assignments may require the collection of data from specialized experiments (Celda and Montelione, 1993; Lin and Wagner, 1999), whose choice will depend on the size of the protein, the amino acid sequence, the labeling, pattern, and the instrumentation available for data collection. For these reasons, a workable approach for automated assignment should be capable of utilizing any combination of experiments deemed necessary by the experimenter (Bax et al., 1990a, b; Fesik et al., 1990; Gronwald and Kalbitzer, 2004). Therefore, the algorithm used for automated assignment should allow for easy extension of experiment lists and should not impose any restrictions on the combination of experiments used.

Noise is a common factor in most real-world optimization problems. Sources of noise include limitations in instrument sensitivity and precision, incomplete sampling, and inconsistent human–computer interactions. Major sources of noise in the NMR assignment problem are variability in data quality (peak overlaps and peak widths), variability in peak picking (extra peaks or missing peaks), and variability in spin-system scoring. The formal theoretical underpinning we seek requires approaches that deal with noisy data and quantify the effects of noise on the output. Our approach is to build methods within the well-established structure of probability theory, which offers the formal theoretical "glue" that can connect individual pieces of the structure analysis process. Probabilistic methods have the advantage of being able to deal with noise and uncertainty within the same rigorous framework.

Resonance assignment programs can be categorized by the methods they use in the mapping steps. These methods include stochastic approaches, such as simulated annealing/Monte Carlo algorithms (Buchler et al., 1997; Lukin et al., 1997; Leutner et al., 1998), genetic algorithms (Bartels et al., 1997), exhaustive search algorithms (Atreya et al., 2000; Andrec and Levy, 2002; Coggins and Zhou, 2003; Jung and Zweckstetter, 2004), heuristic comparison to predicted chemical shifts derived from homologous proteins (Gronwald et al., 1998), and heuristic best-first algorithms (Zimmerman et al., 1994; Li and Sanctuary, 1997; Hyberts and Wagner, 2003). For example, AutoAssign (Moseley et al., 2001) is a constraint-based expert system that uses a heuristic best-first mapping algorithm. Among these algorithms, those that rely on stochastic approaches have the best potential for satisfactorily addressing issues arising from noise.

As we discuss in the next section, mathematical arguments indicate that a general solution to the assignment problem requires a new approach separate from those realized to date. The straightforward combinatorial approach to the NMR assignment problem is to consider the set of all possible configurations for the assignment and to find the one with the lowest "cost". This approach has two notable drawbacks. From a practical computing point of view, exploring the set of all configurations is computationally intractable – particularly for large proteins. The more formidable difficulty lies in the fact that all acceptable cost functions involve noisy, local experimental data.

The approach we use in PISTACHIO recasts the cost function as a measure of "system energy"

and restates the optimization problem in the physical terms of finding a "ground state". This restatement enables us to ask questions that turn out to have tractable solutions with practical applications. For example, we can ask for a "typical low energy configuration" and find a computationally feasible (polynomial time) solution that gives a great deal of information about the "ground state". This is pertinent to the NMR assignment problem, since the available data may not support a unique set of assignments. An additional advantage to this approach is that it yields a set of assignment configurations with their associated probabilities that can be explored and refined further as additional data become available.

Another difficulty arises from the local nature of the data provided by the relevant NMR experiments. Even for an individual amino acid residue, the database of available chemical shifts is insufficient to construct the accurate multidimensional distributions necessary for building an optimal score function. Our solution to this challenge has been to build highly accurate analytic expressions for one-dimensional chemical shift distributions, to use correction factors to extend these to multidimensional score functions, and to take advantage of the sharpening of probabilities afforded by the multiple dimensions. This last step is achieved by parsing the sequence of the protein into overlapping tripeptides and by computing scores and overlap functions for these. The novelty of our approach comes from the analysis of overlapping tripeptides and the algorithms we have devised for scoring each tripeptide and for assembling them into a single scored sequence.

Probabilistic Identification of Spin Systems and their Assignments including coil–helix Inference as Output (PISTACHIO) is part of a larger effort on the automated analysis of protein NMR data. A single software platform has been developed that uses as input the sequence of the protein and peak lists derived from various experimental multidimensional, multinuclear magnetic resonance datasets and that provides as output chemical shift assignments and secondary structure analysis. The output is conveniently reported in NMR-STAR format that can be readily deposited in BMRB. PISTACHIO uses the package PECAN (Eghbalnia et al., 2005) to identify secondary structure elements from the protein sequence and associated assigned chemical shifts. Both packages are available for general use from a server at http://bija.nmrfam.wisc.edu

Materials and methods

Analysis of the NMR assignment problem

NMR experiments used for backbone assignments yield two types of chemical shift correlations: inter-residue and intra-residue. These correlations are used to construct intra- and inter-residue spin systems, consisting of connected nuclei and their chemical shifts. After intra-residue spin systems have been constructed, the process of *spin system typing* assigns a positive score to each spin system. The score signifies a measure of correspondence between spin system j and residue i.

If we consider the cost of assignment, $C(j,i)$, as the negative of the score, then the following assignment strategy can be formulated. Number the residues and spin systems with integers from 1 to n as shown in the first two rows of Table 1. Assume that as many spin systems have been identified as there are residues – we will relax this assumption shortly. Next, rearrange the residue numbers to get the perfect assignment by minimizing a total cost function. This is shown, for example, by the correspondence between the first and the third row. Note that the third row is obtained by a permutation of the second row in which the numbers shown in bold are not permuted.

To formalize this problem, let $\sigma(i)=j$ denote a permutation of i to j. Then we can describe the optimal assignment as a search in the space of permutation matrices to obtain the minimum cost as follows:

Table 1. Conventional method of assigning protein NMR signals

Spin systems	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	...
Residues	1	**2**	3	4	**5**	6	7	8	9	10	11	**12**	13	14	**15**	...
Assigned residues	6	**2**	9	3	**5**	4	1	7	8	11	10	**12**	14	13	**15**	...

$$\arg\min_{\sigma} \sum_{i} C(\sigma(i), i)$$

In the classical mathematical setting, this problem is also known as the "assignment problem" where the term "assignment" describes the problem concerned, typically finding the best way to assign n persons to n jobs. The permutation matrix, which minimizes the cost expression and determines the maximum matching by use of an efficient algorithm of $O(n^3)$, was first demonstrated by Kuhn in 1955. This was a rediscovery of work by two Hungarian mathematicians, D. Konig and E. Egervary, that predated the birth of linear programming by more than 15 years. The approach, therefore, is known as the "Hungarian method". Starting with the work of Edmonds (1965) and building on earlier work of Tutte (1947), this approach has been used to address satisfactorily a number of optimization problems, in particular the "optimal weighted graph matching problem".

In NMR, the inter-residue connectivity information constrains the minimization problem outlined above by restricting the set of permutations to those that do not violate the observed data in the experiments. To take this into account, we could reformulate the problem by associating a "connectivity cost" to each permutation as given by

$$\arg\min_{\sigma \in S_n} \sum_{i=1}^{n} \sum_{j=1}^{n} l_{ij} B(\sigma(i), \sigma(j)) + \sum_{i=1}^{n} C(\sigma(i), i)$$

The second term is the same as before, whereas the new term B represents the cost of assigning the spin system pair (i,j) to residues $(\sigma(i),\sigma(j))$. This cost is weighted by l_{ij}, which is a "small number" if the connectivity between pair (i,j) is observed and a "large number" otherwise. Although the "best matching problem" illustrated in Table 1 has an $O(n^3)$ solution, the more complicated optimization problem formulated above, commonly known as the "quadratic assignment problem" (QAP) or the "weighted bipartite graph matching problem" (a restricted case of QAP), is known to be *NP*-hard (Gonzalez, 1996).

The constrained weighted bipartite graph-matching formulation approach to automated NMR assignments (Ikura et al., 1990; Xu et al., 1993; Geerestein-Ujah et al., 1996; Moseley and Montelione, 1999; Bailey-Kellogg et al., 2000; Permi and Annila, 2001; Andrec and Levy, 2002; Coggins and Zhou, 2003; Malmodin et al., 2003), can be considered as a specialized version of the QAP approach in which the values for l_{ij} and B are specified in advance. The remarkable number of practical applications formulated as QAP has spurred research in a wide community and has led to a number of specialized solutions. Several useful protein NMR software tools attempt to solve the QAP problem by the use of various approximations (Eccles et al., 1991; Nelson et al., 1991; Leopold et al., 1994; Bartels et al., 1997; Leutner et al., 1998; Atreya et al., 2000; Chen et al., 2003; Hitchens et al., 2003), possibly inspired by existing specialized solutions of the QAP. The best-known approximations for finding solutions to typical QAP problems are variations of the "branch and bound" algorithms. Other approximations have been proposed, including genetic algorithms, neural networks, and simulated annealing. These approximations address the practical issue of finding solutions; however, the important question of determining the quality of solutions has remained unanswered. Theoretical work (Burkard and Fincke, 1985) has shown that the landscape of score functions for QAP problems of practical size is essentially flat. An inescapable consequence of this theory is that any deterministic algorithm, such as branch and bound or any of its heuristic modifications, can yield a solution that is trapped in a local minimum without providing any information about the global minimum. These considerations place serious practical limits on the applicability of branch and bound-type methods to the NMR assignment problem.

The nature of noise in determining the NMR "cost function" makes the application of deterministic combinatorial methods even less attractive. The difficulty arises at the spin system typing and assembly stages, where one must assign a cost or probability that a spin system (or group of spin systems) belongs to a particular amino acid (or sequence of amino acids) in the protein sequence. After the time-domain data have been processed, the peak-picking step identifies the chemical shifts associated with each peak in the spectrum. The accuracy of this step depends on the resolution and sensitivity of the data collected, as well as choices made by the expert. The intermediate construction of spin systems and their scores depend on the information content, noise, and ambiguity of the

data, and these vary from dataset to dataset. Noise and ambiguity are inherent features of derived NMR data; they are ever present and vary only in degree. In addition, the noise in the scoring depends on the model distributions of chemical shifts. These problems are complicated further upon lifting the earlier assumption regarding the equality of resides in the sequence and intra-residue spin systems identified in the experimental data. In reality, the number of intra-residue spin systems identified can exceed or fall short of the number of residues. All the above sources of noise will make it difficult to identify the single global minimum, particularly within the context of deterministic algorithms.

Evolutionary algorithms (EA) are general, nature-inspired heuristics for numerical search and optimization that are frequently observed to be particularly robust with regard to the effects of noise. This class of techniques, initiated by the work of John Holland (1975), includes search algorithms inspired by the process of natural selection (biological evolution). The approach been used for NMR assignments (Bartels et al., 1997). The operation of EA depends on a multitude of parameters in combination with fitness environments; together they form stochastic dynamical systems that are not easily analyzed or understood. The class of evolutionary algorithms for optimization, which encompass the methods of genetic algorithms, evolutionary programming, and evolutionary strategies (Davis, 1987, 1991; Goldberg, 1989; Holland, 1992; Koza, 1996), have been reported to perform well under noisy conditions, as supported by numerical results in test cases (Rana et al., 1996; Nissen and Propach, 1998; Stroud, 2001). However, the successful implementations of EA (Baum et al., 2001) in the presence of noise have been limited to problems in which the noise has a very special structure; and, as has been pointed out (Michalewicz and Fogel, 2000), ''there really are no effective heuristics to guide the choices to be made that will work in general''.

The NFL (no free lunch) theory (Wolpert and Macready, 1997) presents a formal analysis of search algorithms for optimization, including the EA, random search, and simulated annealing approaches. The essence of NFL analysis is that in the absence of prior knowledge or models, the average performance of a given optimization algorithm across all problems is as good as any other algorithm. This sobering statement reaffirms that no general algorithm can replace careful modeling of the specific problem.

Our earlier attempt to address these issues and to develop a statistical approach to automated assignments was embodied in the CONTRAST software package (Olson and Markley, 1994; Olson, 1995), which provides a ranked list of assignment probabilities. CONTRAST attempted to model noise, but is subject to the general limitations discussed above. PISTACHIO, the new approach we lay out here, addresses these challenges by transforming the deterministic, combinatorial optimization problem into a search for the ground state configuration of a statistical system. The local state of the statistical system is constructed in three steps. First, we parse peak lists associated with particular experiments into the set of all possible tripeptide spin systems specified by the peptide sequence. Second, we score the list of possible tripeptide assignments on the basis of our prior analysis of the BMRB database of chemical shifts, which has allowed us to create models for the chemical shifts for each nucleus and each pair of nuclei within amino acids taken singly and in pairs. This makes use of a formula we have derived that accurately computes probability scores for matching chemical shift data to tripeptides. Third, we assemble the overlapping tripeptides to match the sequence and to achieve the optimal probabilities for correct assignments. Each of the three steps serves to restrict the size of the combinatorial search space and to minimize the impact of noise on the analysis. The impact of noise is further ameliorated because the global ground state of the model corresponds to a state that is ''statistically closest'' to the reported observations of data (see the supporting information for an outlined proof and analysis of noise impact).

Mathematical model

Our starting point is to define a ''local cost model'' for our system that has ''suitable emergent global properties''. By a local cost model, we mean that the overall cost of assignment can be obtained as the sum of costs for local assignment. By suitable emergent global properties, we mean that our solution for the global cost will satisfy all local constraints and cost functions in the most optimal way. We start by defining a set of local cost

functions J_N as a function of the configurations Y_N that represent the spin system for a given tripeptide. We define the global cost as:

$$J(Y) = \sum_{Y_N \subseteq Y} J_N(Y_N)$$

To minimize the overall cost J,we can equivalently maximize the following expression:

$$\begin{aligned} e^{-J(Y)} &= \exp\Big(\sum_{Y_N \subseteq Y} -J_N(Y_N)\Big) \\ &= \prod_{Y_N \subseteq Y} \exp(-J_N(Y_N)) \end{aligned}$$

We can treat this as a probability by dividing by a normalization factor:

$$P(Y) = \frac{1}{Z} \prod_{Y_N \subseteq Y} \exp(-J_N(Y_N))$$

We base the cost J on a QAP cost formula that we restate as:

$$\begin{aligned} J &= \sum_{i=1}^{n} \sum_{j=1}^{n} l_{ij} B(\sigma(i), \sigma(j)) \\ &\quad + \sum_{i=1}^{n} C(\sigma(i), i) = \sum \Psi + \sum \Phi \qquad (1) \\ e^{-J} &= e^{-\sum \Psi} e^{-\sum \Phi} \end{aligned}$$

Upon dividing by the normalization factor to obtain probabilities, and by denoting the set of all permutations as Σ, we obtain:

$$\begin{aligned} P(\Sigma) = \frac{1}{Z} &\prod_{\substack{i,j \\ i \neq j}} \exp(-\Psi(\sigma(i), \sigma(j)) \\ &\times \prod_{i} \exp(-\Phi(\sigma(i), i)) \qquad (2) \end{aligned}$$

In the expression above, the second part measures the merit of assigning a tripeptide spin system Φ to a given amino acid triple, whereas the first part Ψ provides a measure of the compatibility of adjacent and overlapped spin systems. We describe below how the value of Φ for each tripeptide is obtained. We define the value of Ψ by

$$\Psi(x,y) = \begin{cases} u = (-k\langle C^n x, C_n y\rangle) & \text{if } u < -\varepsilon \; x \neq y \\ -\varepsilon & \text{if } u \geq -\varepsilon \; x \neq y \\ B & \text{if } x = y \end{cases}$$

In the above formula, k is a fixed and empirically obtained positive constant that sets the scale of local costs. The parameter ε is used as a control parameter to test the local "flatness" of the solution surface, and B is set to a value much larger than k to inhibit duplicate selection. The expressions $C^n x$ and $C_n y$ represent projection of the chemical shift coordinates onto the last n and first n coordinates, respectively. Intuitively, when n represents the spin-system size for a single residue overlap and the neighboring tripeptides are AKC and CDE, with the overlapping residue being C, we assign a score to the compatibility of the chemical shift values of C in each triplet pair. The number of projected coordinates (or overlapping residues) is changed in a controlled manner to obtain information regarding the consistency of local scores. The above prescription for the value of Ψ encodes our objective to make assignments that violate connectivity information "energetically unfavorable". One major practical advantage of this formulation is its generality. It does not rely on a specific experiment or set of experiments and thus enables easy and efficient incorporation of new experiments into the framework. The resulting model can be represented in the form of a graph $G = (V, W)$, with vertices V and edges W (Figure 1). In this graphical representation of our model, two overlapping residues in neighboring tripeptides are connected by an edge in order to enforce the neighborhood constraint. Tripeptides and the triplet spin systems (also represented by vertices) are connected by an edge that represents the possibility that the spin system may correspond to the given

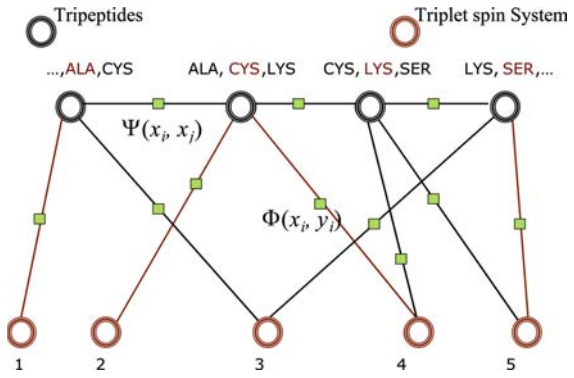

Figure 1. Graphical representation of the energetic model for the case of two overlapping residues in neighboring tripeptides. The tripeptides are the upper vertices, and the triplet spin systems are the lower vertices. An edge between a tripeptide and a triplet spin system represents the possibility that the spin system belongs to that residue.

residue. The weights along the edges or those for individual vertices represent local information, and the task of the algorithm (see below) is to find the most consistent global assignment described by the probability distribution.

To achieve our goal of deriving probabilities for the pairing of pair spin systems with residues (the Φ term), we must calculate the marginal probability instead of the joint probability given by Equation 2. Whereas the joint probability provides the frequencies of pairing of spin systems and residues occurring simultaneously across the amino acid sequence, the marginal probability indicates the fidelity achieved in pairing a specific spin system with a particular residue.

Brute force computational approaches to deriving marginal probabilities grow exponentially in computational time with the number of assignments to be made, and, therefore, are impractical even for small proteins. Our approach is to derive the marginal probability by an iterative approximation algorithm. The algorithm exploits parallels between the formula above and particle interaction problems in statistical physics. To achieve global consistency, the algorithm works on clusters of residues that share common residues and evaluates the cumulative measure of merit (joint and marginal probabilities) for each cluster. Values in every overlap cluster are adjusted in a way that moves the global cost in the direction of the minimum. In each iterative step, the error gradients in local clusters are transmitted to larger clusters, and relative weight adjustments are performed. This process of adjusting the values in the overlap clusters continues until values computed from two starting points differ by only a small ε value.

In recent years, a number of techniques based in probability, geometry, and functional analysis have yielded great insight into probability distributions in multi-dimensional settings (Talagrand, 1995). In intuitive language, the probability distribution of joint events tends to become more concentrated around its mean (or median) as we consider more and more events. In the assignment problem, one can take advantage of this property by considering tripeptide resonance systems whenever an accurate estimate of their probabilities can be obtained. The tripeptide spin systems are assembled from spectral data that provide intra-residue and inter-residue connectivities. One of the advantages of this approach is that it accommodates conditional probabilities: neglecting end effects, each residue forms a part of three overlapping tripeptides.

To obtain accurate joint probability values, we proceed as follows. Let A, B, and C be subsets representing events with probabilities of occurrence $|A|, |B|$, and $|C|$, respectively. The following basic relationship and its generalization to subsets denoted by $|A_i|$ follow a standard counting argument and the joint probability is obtained by considering the value of the complement:

$$|A \cup B \cup C| = |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C| \tag{3}$$

$$|A| = |A_1 \cup A_2 \cup \ldots \cup A_n| = \sum_{i=1}^{n} |A_i| - \sum_{\text{pairs}(ij)} |A_i \cap A_j| + \sum_{\text{triples}(ijk)} |A_i \cap A_j \cap A_k| - \ldots + (-1)^{n+1} |A_i \cap A_j \cap A_k \cap \ldots \cap A_n| \tag{4}$$

This formula, however, does not lead to sufficiently accurate probabilities for the tripeptide spin systems or its complement $1 - |A|$. It is clear from Equation 4 that an accurate estimate of individual probabilities $|A_i|$ in the first term of the expansion is required and that the first- and higher-order correlations (dependent events) also need to be evaluated. However, we have sufficient database information to build only the first two terms in the expansion. For this reason, we have rewritten the expression in a way that yields a good estimate from first two terms alone.

$$|A| \approx 1 - \prod_i N_i \prod_{j>i} N_{ij}$$

$$N_i = 1 - |A_i|, \quad N_{ij} = N_i^{-1} + N_j^{-1} - N_i^{-1} N_j^{-1} (1 - |A_i \cap A_j|)$$

Computational aspects

The input data consist of peak lists derived from a set of multidimensional NMR experiments. This set can include experiments such as HSQC, HNCO, CBCA(CO)NH, HNCACB, HN(CO)CACB,

HNCA, HN(CO)CA, HN(CA)CO, HBHA(-CO)NH, C(CO)NH, HN(CO)(CA)CB, HN(CA)CB, H(CCO)NH, N15-TOCSY, and HCCH-TOCSY. Any combination of these 15 experiments can be used to generate candidate spin systems. The most sensitive experiment (typically HSQC or HNCO) is used as the reference starting point for spin system generation. For missing chemical shifts, an empty value is assigned to the nucleus. In cases where assignments are ambiguous, all chemical shift possibilities for a given nucleus are carried forward throughout the computation. PISTACHIO uses a grid search for the best tolerance for generating spin systems (see below). The values typically searched are in the range of 0.02–0.03 ppm for ^{1}H, 0.2–0.275 ppm for ^{15}N, and 0.2–0.3 ppm for ^{13}C dimensions. User can specify a search range different from the default. PISTACHIO treats peak lists from different experiments in an unbiased way and uses all information entered to create the set of candidate spin systems. By constructing profiles (experiment description matrixes), one can customize PISTACHIO so as to: (1) add additional experiments, (2) assign different weightings to experiments, (3) handle data from selectively labeled proteins, or (4) incorporate prior assignments made by other means. Only (1) and (3) are currently available from the PISTACHIO web server using special instructions; (2) and (4) are achievable but must be coded in by hand.

PISTACHIO performs preprocessing steps prior to spin system generation. Of these, the most important are *alignment* and *best density and probability estimates*. These are discussed briefly below.

Alignment

An alignment algorithm is used to compensate for possible referencing errors or other source of global shifting errors. Note that the matching of peaks from different experiments is an ill-posed problem, because the peaks expected from nuclei common to more than one experiment cannot be assumed to have been observed in each case. In addition, non-linear peak displacements in different spectra have been observed in real data. Our algorithm attempts to obtain a global alignment of common chemical shift axes in all datasets in an automated fashion without trying to match individual peaks. A linear shift is tried first, and if this fails, then local adjustments are made. The intuition behind the approach is to replace each peak location with a probability distribution and to attempt to match the distributions.

Best density and probability estimates

The chemical shift distribution model used by PISTACHIO for an individual atom, amino acid residue, or tripeptide is not Gaussian. Our approach to deriving an accurate estimate for the probability distribution is to obtain densities from the largest class of distributions that (1) satisfy the law of large numbers in probability, (2) have the most parsimonious form, and (3) permit robust estimates of their parameters from Monte Carlo simulations. We have used data available from BMRB to parameterize a set of distributions for each nucleus in each amino acid. The details of the methods involved will be presented in a separate publication. The resulting distributions properly capture the current statistics for the available data and provide more accurate local probably estimates than those derived from histograms (Figure 2).

Description of the PISTACHIO algorithm

The free energy minimization approach used by PISTACHIO employs an iterative strategy that is guaranteed to converge. The algorithm uses a sequence of non-increasing energy states, called strata, to control the combinatorial search in the exponentially large number of states. For each stratum, a sequence of sub-strata is used to control the descent in energy by iteratively ''squeezing'' the energy between a lower and upper bound. The final ''minimal energy state'' specifies the configuration corresponding to ''highest probability states'' for assignments ''most consistent'' with the observed data in the presence of noise. This correspondence utilizes the well-known equivalence of minimum free energy state and the minimal statistical distance. The state that the algorithm converges to is exact if the model described by Equation 1 is correct. The consequences of noise or ambiguities in the data are naturally reflected in the assignment probabilities.

In order to describe the PISTACHIO algorithm in a simple and accessible form, a number of notational conventions are necessary. Equation 2 is rewritten as $P(X) = Z^{-1} \prod_{u \in U} \psi_u(X_u)$, where u can be a single or a multi-index and U is a set of multi-

indices. We refer to u as a cluster and to U as a cluster set. In the context of the graph notation ($G=(V, W)$) introduced above, a cluster is a subset of vertices of V and a cluster set is a set of these subsets. For example, when $u=\{i\}, \psi_u(X_u)$ represents $\phi(i,\sigma(i))$, and when $u=\{i,j\}, \psi_u(X_u)$ represents $\Psi(\sigma(i),\sigma(j))$. The multi-index notation is implied whenever an index is used with a capitalized variable such as X or a function such as ψ. The notation X_u is the shorthand for the random vector $(x_{i_1},x_{i_2},\ldots, x_{i_k})$, where the indices i_j form the multi-index or cluster u. The free energy for our model (Figure 2), $F \approx \sum \langle \log P_u(X_u) - \log \psi_u(X_u) \rangle_{p_u(X_u)} + \sum a_v \langle \log P_v(x_v) \rangle_{P_v(X_v)}^{u}$ represents a refined approximation to the "full" free energy term (Landau and Lifshitz, 1980). A lower case x is meant to denote our interest in the values of the function for a cluster of size exactly one. In our formulation some clusters are not disjoint, and we need to correctly reflect the number of times each cluster and the vertices in the intersection of clusters are counted. The constants a_v are used to count the number of times a vertex v appears in the intersection of clusters in a cluster set U and the constants b_v are used to count the number of clusters in the set U. PISTACHIO also selects a multi-index set U in each iteration, which is defined in the display describing the PISTACHIO algorithm below. The outer loop of the algorithm repeats the iteration until no further changes in the priority of assignments is observed. A change in the priority of assignments is observed if a previously unassigned spin system is assigned or a previously assigned spin system changes its assignment. Details are shown in Figure 3.

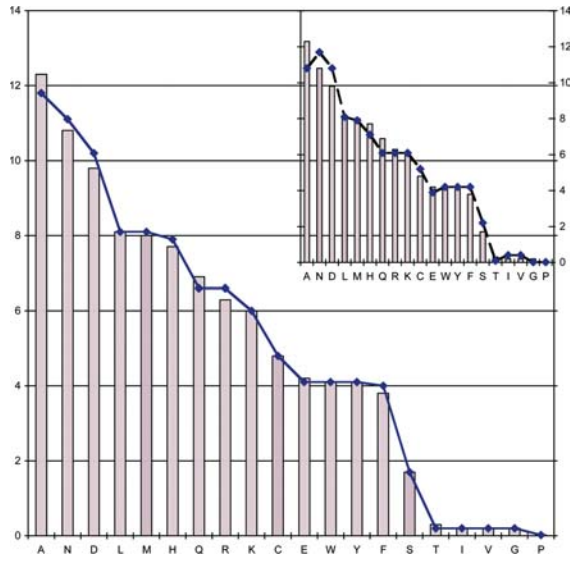

Figure 2. Results from the classification of a random sample of 1000 $^{13}C^{\alpha}$ chemical shifts for alanine to one of the 20 amino acids by three different distribution models: the analytical distribution used by PISTACHIO, a normal distribution, and an optimal histogram. The bars indicate the percentage of classification for each residue (vertical axis) based on our analytical distribution. The blue line in the main figure indicates the percentage of classification for each residue based on optimal histograms. The approach used by PISTACHIO follows the trend of results by an optimal histogram and leads to a small but noticeable improvement in the classification along this single chemical shift dimension ($^{13}C^{\alpha}$). When spread over multiple nuclei, enhancements of this type lead to evident cumulative improvements in the assignments. The inset compares the classification of our analytical model with one obtained by fitting a normal distribution to the histogram data. When fit with a normal distribution, the distribution of choices is more entropic (less selective) and alanine is no longer recognized as the most likely residue.

Results

In the early testing stage of PISTACHIO, we used as input simulated datasets obtained by deconstructing BMRB entries into peak lists. Simulated peak lists for one small protein (8.6 kDa), two medium sized proteins (12–18 kDa), and one large protein (40.6 kDa) led to average assignments accuracies of 98% (results not shown). The BMRB deposition for one of these proteins (BMRB 5106) contained the raw peak lists, which enabled us to compare the use of real vs. simulated data. Whereas the simulated data yielded 69 spin systems with 99% of the spin systems assigned, a subset of the raw data (N15-HSQC, HNCO, HNCACB, CBCA(CO)NH) yielded 66 spin systems with 90% correctly assigned. This result confirmed our expectation that true quality measurements can only be obtained from tests on raw data.

Raw data in the form of peak lists were obtained from collaborators at the National Magnetic Resonance Facility at Madison (NMRFAM) and the Center for Eukaryotic Structural Genomics (CESG). To get the best cross section of results for testing our system, we adopted the single requirement that input data be reported in either the XEASY (Bartels et al., 1995) or NMR-STAR (defined on the BMRB website http://www.bmrb.wisc.edu) format. Users were free to choose their experimental dataset combinations,

Set $P_u(x_u) = 1$ While ***Changing*** increment iteration count l	Set $P_u(x_u)$ initial value to scaled uniform distribution. If any assignments change, repeat the steps. ***Changing*** means change in assignment probabilities that change assignment priorities.
Set ***Defaults***	***Defaults*** (below) sets starting values
While *not* ***converged***	***Converged*** means Delta $F < 0.000001$.
$\psi_u(X_u) \leftarrow M(\psi_u(X_u))$	$M(\psi_u(X_u)) = \psi_u(X_u) e^{-\sum_{v \subset u} a_u / b_u \log P_u(x_v)}$
Estimate	***Estimate*** algorithm take ($\psi_u(X_u)$ as input and returns new $P_u(x_u)$ & $P_v(x_v)$
End of While	
Return $P_u(X_u)$ and $P_v(x_v)$	***Converged*** values for probabilities
End While	
Estimate Algorithm	The algorithm obtains probabilities based on the current upper bound.
Set $P_u(x_u) = \psi_u(X_u)$ and $P_v(x_v) = 1$	
While *not* ***converged***	
For all subsets indexed by v $P_v(x_v) \leftarrow L(P_v(x_v))$	$L(P_v(x_v)) \propto P_v(x_v)^{\frac{a_v}{a_v+b_v}} \left[\prod_{u \supset v} P_u(x_v) \right]^{\frac{1}{a_v+b_v}}$
For all neighbors u $P_u(x_u) \leftarrow T(P_u(x_u))$	$T(P_u(X_u)) \propto P_u(X_u) P_u^{-1}(x_v) L(P_v(x_v))$
End For	
End For	
End While	
Return all $P_u(x_u)$ and $P_v(x_v)$	
Defaults Algorithm	The algorithm sets default values for PISTACHIO.
For each {C, N, H} search grid {0.2,0.2,0.02} to {0.3, 0.275, 0.03} in increments {0.01, 0.01, 0.002}	A sensitivity range is searched in order to obtain the best-predicted assignment quality. This step is typically necessary only once but is included in the iteration in cases where re-tuning after "fixing" of some spin systems.
Q = quality(C,N,H)	Quality is measured as defined by equation 5
End For	
Prune high probability assignments	If this is not the first stratum, spin systems with high probability of assignment (>99.5%) are fixed to their current assignment.
For each triplet **Y** in the model Find *maximum number of* edges with lowest probabilities such that the sum of probabilities is bounded by $10^{-(l+3)}$ Reconnect the spin system end of vertex all removed edges to a *ghost* vertex. Add all vertices for this triplet to the set U *Renormalize* all probabilities	Select cluster set U for current stratum. A cluster set forms the multi-index set for the variable u used above. l counts the number of iterations. The set U is constructed from the set of vertices of the graph that are incident to the edges for triplet **Y**. A *ghost* vertex is used to represent the average state of removed spin systems. *Renormalization* is necessary to correct for removals that impact multiple triplets.

Figure 3. PISTACHIO algorithm (left column) and a corresponding description of the steps (right column).

the method they selected for peak picking (manual vs. automatic), and the number of peaks reported. The PISTACHIO assignments were scored against the best set of assignments achieved by other means. In some cases, the manual assignment used for comparison benefited from corrections occurring farther down the structure determination pipeline. Nonetheless, our criterion for correctness was agreement of the PISTACHIO results with the best known assignments for the protein at hand.

In 10 of the 15 datasets tested (Table 2), PISTACHIO achieved better than 90% correct assignments for backbone resonances. Various peak picking software packages (NMR_VIEW, SPARKY, NMRPipe) were used in generating the raw peak lists. The datasets contained different fractions of noise peaks. Depending on the experiment and method of peak picking, the peak counts varied from approximately 0.7 times to 7 times the expected number of peaks.

The quality of raw peak list data was the key factor in determining the extent and correctness of the assignments as well as the run times. In order to assess the quality of our data, we use two notions of quality. First, we use a post-analysis quality number that represents the percentage of peaks assigned from a single typical experiment, generally the HNCACB experiment, which produces a moderately large number of ''information rich'' peaks. We also use a pre-analysis quality measure that reports a statistical expectation for the number of assignments (see supporting information for further discussion). To define this pre-analysis quality, let S_A be the combinatorial set of vectors enumerating chemical shift possibilities for spin system '*A*' (for example, two different CO choices for one NH). S_A can be viewed as a matrix $S_A(k,l)$ where the columns are chemical shift vectors and rows are chemical shift values for a given nucleus. In the same way, we define S_{ABC} for the triplet spin

Table 2. Proteins assigned with PISTACHIO for which assignments made by other means were available for comparison

Protein designator	Backbone				Sidechain		Experiments represented in the input peak lists[a]
	Residues correctly assigned[b]/assignable residues (number of Pro residues)	CPU time (h)	$P>0.95$	% correct[b]	CPU time (h)	% correct[b]	*1 2 3 4 5 6 7 8*
At2g24940	106/106 (3)	0.2	100%	100%	1	95%	
At1g77540	96/97 (6)	0.2	100%	99%	0.1	95%	
At2g23090	82/84 (2)	0.1	98%	97%	c	c	
Mm202773	92/94 (4)	0.1	97%	97%	0.1	85%	
At5g22580	99/108 (3)	4	95%	92%	1	86%	
CE5073	108/118 (2)	4	95%	92%	c	c	
At3g17210	96/107 (5)	5	95%	90%	1	90%	
At3g51030	108/120 (4)	4	95%	90%	1	85%	
At5g01610	95/115(5)	6	90%	83%	c	c	
At3g16450[d]	232/291 (8)	5	85%[b]	80%	2	75%	
At1g23750	122/152 (5)	6	85%	80%	c	c	
Acyl carrier	70/72 (0)	0.7	99%	97%	c	c	
BMRB 5106	61/68 (2)	1	95%	90%	c	c	
YggX	71/89 (2)	4	80%	80%	1	75%	
P-gamma	62/77 (10)	4	85%	80%	c	c	

[a]Each dataset included an HSQC or HNCO experiment; other experiments are indicated by numbers: *1* – CBCA(CO)NH or HN(CO)CACB; *2* – HNCACB; *3* – HNCA; *4* – HN(CO)CA or CA(CO)NH; *5* – HN(CA)CO; *6* – H(CCO)NH or N15 TOCSY; *7* – C(CO)NH; *8* – HBHA(CO)NH. [b]Correct assignments were achieved independently of PISTACHIO and based on the best available assignment model; in most cases these assignments made use of additional information not available to PISTACHIO, such as structural constraints. [c]Sidechain data had not been collected yet in these cases. [d]Stereo array isotope labeled (SAIL) protein; isotope shifts due to labeling were not accounted for.

system in a tripeptide *ABC*. The quality factors *b* (spin system quality factor), *c* (connectivity quality factor), and overall quality factor *q* are defined as:

$$b = \langle I_k(S_A)\rangle_{A,k}, \quad c = \langle (I_m(S_{ABC})\rangle_{ABC,m},$$
$$q = \sqrt[4]{bc} \tag{5}$$

$I_k(S_A)$(defined below) represents a count of unique chemical shifts for spin system A.$\langle\Box\rangle_{A,K}$ denotes the expectation taken over all k in the spin systems A. For residue B, the index m runs over all chemical shift entries that belong only to the residues A and C. I is defined by the following function:

$$I_k(S_A) = \begin{cases} 1 & S_A(k,l) = S_A(k,l')\forall l,l' \\ 0 & \text{Otherwise} \end{cases}$$

$$I_m(S_{ABC}) = \begin{cases} 1 & S_{ABC}(m,l) = S_{ABC}(m,l')\ \forall l,l' \\ 0 & \text{Otherwise} \end{cases}$$

All proteins with datasets of average or good quality yielded assignments $\geq 90\%$ correct. Proteins with PISTACHIO assignment scores $< 90\%$ were ones with spectra characterized as being of low quality or with many overlapped and missing peaks. Also in this category was a stereo array isotope labeled (SAIL) protein, discussed below, for which no corrections were made for isotope effects on chemical shifts. Input datasets containing high noise (many more peaks than expected in particular experiments) were not necessarily more complex to assign than those with low noise. The most difficult cases were ones in which "real" and "noise" peaks were interspersed within the tolerances for separate peak recognition. For example, spectra from protein At1g23750 contained many overlapping peaks in a disordered region of the protein. The overall assignment result for At1g23750 was 80% correct, with assignment difficulties confined mostly to the unstructured region (Table 3).

We found that the workstation time required was inversely related the quality of the input data and did not depend necessarily on the size of the protein. According to the protein and data available, the computation required minutes to hours on a single workstation (Table 2). The computationally demanding part of the assignment is the iterative algorithm used to determine global consistency described above.

Table 3. Completeness and quality of data used for test of automated assignments by PISTACHIO

Name	PEAKS			Quality
	% Reported	% "True"	% Noise	
At2g24940	78	71	7	0.85
At1g77540	93	63	30	0.78
At2g23090	102	70	31	0.71
Mm202773	109	76	32	0.73
At5g22580	93	79	13	0.88
CE5073	102	82	20	0.76
At3g17210	105	91	13	0.75
At3g51030	105	62	42	0.69
At5g01610	166	76	89	0.65
At3g16450	99	72	27	0.72
At1g23750	95	52	42	0.60
Acyl carrier	153	61	92	0.60
BMRB 5106	113	90	23	0.78
YggX	216	47	168	0.63
P-gamma	100	80	20	0.78

Percentages of peaks under the heading PEAKS are for HNCACB experiment. % Reported peaks is the ratio of total reported peaks to the maximum number of theoretically possible peaks. % "True" peaks is the ratio of total assigned peaks to the maximum number of theoretically possible peaks. This number is indicative of data completeness. % Noise peaks is the difference between % reported peaks and % "true" peaks. This number is indicative of one source of noise in the input data and may be larger than 100%. Quality is an indicator computed according to formulas (Eq. 5) and is a statistical measure of quality.

The initial steps of alignment and preliminary quality check of the data prior are performed rapidly, and these are used on an interactive basis to decide whether the assignment should proceed. PISTACHIO rapidly computes the number of spin systems represented in the data and quality scores for backbone and sidechain assignments. These can be used to immediately predict, in advance of the longer computation, the minimum number of spin systems that likely will be assigned (see supporting information). These quality factors can indicate possible sources of error, such as alignment problems, an unacceptable number of noise peaks, or data provided in unacceptable format. If no difficulties are detected or if they can be corrected automatically, PISTACHIO proceeds to the iteration step. Otherwise, the user must intervene to provide improved input data.

The workstation time needed for sidechain assignment is substantially less than that for backbone assignments. This is because of the cost function Equation 1 is substantially simplified and because an exact calculation of Equation 2 can be performed. The time usually is linearly proportional to the number of residues in protein. In practice, our observations show a strong correlation between the quality of sidechain and backbone assignments.

One of the advantages of PISTACHIO is that it provides a ranked list of possible assignments for ambiguous cases. In the case of the P-gamma protein (Table 2), the presence of multiple peaks resulted in PISTACHIO assignments to three residues that later were corrected manually from additional information. The corrected assignments were those ranked second in the PISTACHIO results. The reporting of alternatives allows experimenters or structure determination software to rapidly find alternative assignments that satisfy additional information as it is added.

Protein At3g16450 contained SAIL amino acids. In this labeling pattern, each methyl group is $-^{13}C^{1}H(^{2}H)_2$ and each methylene group is $-^{13}C^{1}H^{2}H-$. As a result, different nuclei are affected by isotope shifts that can be conformationally dependent. For the purposes of testing robustness, the resulting shifts can be viewed as random noise present in the chemical shift data. The range of isotope shifts (now viewed as random noise) had varied effects (shifts of 0.4–0.8 ppm) on the $^{13}C^{\alpha}$ and $^{13}C^{\beta}$ chemical shifts. Shifts beyond 0.5 ppm generally are considered to be significant for the purpose of assignment and structure determination. Even though no corrections were introduced for these isotope shifts, PISTACHIO achieved 80% correct assignments for this 299-residue protein. These results demonstrate that PISTACHIO behaves robustly with respect to noise. With the introduction of appropriate corrections for expected isotope shifts, we expect that the performance of PISTACHIO with proteins containing SAIL amino acids will improve measurably.

Discussion

Automation afforded by PISTACHIO has benefits beyond improvements in the efficiency of the assignment step. The introduction of formal theoretical protocols at each computational processing step should lead to outcomes that are less subjective and more reproducible. This will enable objective comparisons that exclude variability arising from the interpretive evaluation of experts.

PISTACHIO provides an extensible and robust system for assignment. Experiments are described by means of an experiment description matrix, and models for assignment are constructed on the basis of this matrix. Therefore, the addition of new experiments does not require rewriting of the software. PISTACHIO maintains internal records of hard to assign regions and their corresponding experimental data. This database, when sufficiently populated, will be used as a statistical knowledge base to identify difficult regions in new proteins submitted for assignment.

PISTACHIO is part of a larger effort aimed at introducing automated probabilistic analysis at each step in a protein NMR structural study. Automation procedures at each step should lead to consistency and verifiability for the overall process. The tools should also be flexible enough to allow improvements in each step separately and to allow the introduction of new steps that may amend or replace existing steps. Since the overall robustness and accuracy of the process is gated by the weakest step, establishment of formal theoretical protocols at each step is paramount to the overall success of the effort. In its current implementation, PISTACHIO uses peak lists supplied by the user for experiments that report through-

bond connectivities. Our goal is to derive these peak lists in a probabilistic manner, as achieved by the HIFI-NMR approach (H. R. Eghbalnia et al., submitted). Once an initial set of assignments is available, these can be evaluated according to the linear analysis of chemical shifts (LACS) algorithm (Wang et al., 2005) to identify and correct referencing problems and to identify outliers that may indicate misassignments or unusual secondary structure. In addition, assignments can be refined by making use of probabilistic secondary structure determinations from the set of assigned chemical shifts and the peptide sequence as provided by the PECAN approach (Eghbalnia et al., 2005) to refine the chemical shift models used in scoring the tripeptide assignments. One can envision a further assignment iteration that makes use of NOESY data and the consistency of assignments with structural models. These refinements are included in our future goals for this project.

Supporting information available

Intuitive discussion of the assignment algorithm, examples of additional datasets analyzed by the software, predictive value of the initially derived quality factors (1 table and 3 figures) at http://dx.doi.org/10.1007/s10858-005-7944-6.

Acknowledgements

This research was carried out at the National Magnetic Facility at Madison (NMRFAM), which is supported by NIH Grant P41 RR02301 from the Biomedical Research Technology Program, National Center for Research Resources. A.B. and L.W. had partial support from Grant 1 P50 GM64598 from the National Institute of General Medical Science's Protein Structure Initiative, which supports the Center for Eukaryotic Structural Genomics (CESG). During part of this work H.E. was supported as a postdoctoral trainee by the National Library of Medicine under Grant 5T15LM005359. We thank Eldon L. Ulrich and William M. Westler for advice and encouragement and the various persons at NMRFAM and CESG who provided the peak lists and assignments used in this study. Data on the SAIL protein were obtained in collaboration with Masatsune Kainosho and members of his group at Tokyo Metropolitan University. This work made extensive use of the BioMagResBank and the Protein Data Bank.

References

Andrec, M. and Levy, R.M. (2002) *J. Biomol. NMR* **23**, 263–270.
Atreya, H.S., Sahu, S.C., Chary, K.V. and Govil, G. (2000) *J. Biomol. NMR* **17**, 125–136.
Bailey-Kellogg, C., Widge, A., Kelley, J.J., Berardi, M.J., Bushweller, J.H. and Donald, B.R. (2000) *J. Comput. Biol.* **7**, 537–558.
Bartels, C., Güntert, P., Billeter, M. and Wüthrich, K. (1997) *J. Comput. Chem.* **18**, 139–149.
Bartels, C., Xia, T.-H., Billeter, M., Güntert, P. and Wüthrich, K. (1995) *J. Biomol. NMR* **5**, 1–10.
Baum, E.B., Boneh, D. and Garrett, C. (2001) *Evol. Comput.* **9**, 93–124.
Bax, A., Clore, G.M. and Gronenborn, A.M. (1990a) *J. Magn. Reson.* **88**, 425–431.
Bax, A., Clore, G.M., Driscoll, P.C., Gronenborn, A.M., Ikura, M. and Kay, L.E. (1990b) *J. Magn. Reson.* **87**, 620–627.
Billeter, M., Braun, W. and Wüthrich, K. (1982) *J. Mol. Biol.* **155**, 21–346.
Buchler, N.E.G., Zuiderweg, E.R.P., Wang, H. and Goldstein, R.A. (1997) *Biophys. J.*, **72**, WP447.
Burkard, R.E. and Fincke, U. (1985) *Discrete Appl. Math.* **12**, 21–29.
Celda, B. and Montelione, G.T. (1993) *J. Magn. Reson. Series B* **101**, 189–193.
Chen, Z.Z., Jiang, T., Lin, G., Wen, J., Xu, D., Xu, J. and Xu, Y. (2003) *Theor. Comput. Sci.* **299**, 1–3.
Coggins, B.E. and Zhou, P. (2003) *J. Biomol. NMR* **26**, 93–111.
Davis, L. (1987) *Genetic Algorithms and Simulated Annealing* Pitman, London.
Davis, L. (1991) *Handbook of Genetic Algorithms* Van Nostrand Reinhold, New York.
Eccles, C., Güntert, P., Billeter, M. and Wüthrich, K. (1991) *J. Biomol. NMR* **1**, 111–130.
Edmonds, J. (1965) *J. Res. Nat. Bur. Standards Sec. B* **69**, 125–130.
Eghbalnia, H., Wang, L., Bahrami, A., Assadi, A. and Markley, J.L. (2005) *J. Biomol. NMR* (in press).
Fesik, S.W., Eaton, H.L., Olejniczak, E.T. and Gampe, R.T. (1990) *J. Am. Chem. Soc.* **112**, 5370–5371.
Geerestein-Ujah, E.C., Mariani, M., Vis, H., Boelens, R. and Kaptein, R. (1996) *Biopolymers* **39**, 691–707.
Goldberg, D. (1989) *Genetic Algorithms in Optimization Search and Machine Learning*, Addison Wesley, New York.
Gonzalez, T.F. (1996) Multi-message multicasting: complexity and approximation. Proceeding of 30th Hawaii International Conference on System Sciences HICSS-30.
Gronwald, W. and Kalbitzer, H.R. (2004) *Prog. Nuc. Magn. Reson. Spectr.* **44**, 33–96.
Gronwald, W., Willard, L., Jellard, T., Boyko, R.E., Rajarathnam, K., Wishart, D.S., Sonnichsen, F.D. and Sykes, B.D. (1998) *J. Biomol. NMR* **12**, 395–405.
Hitchens, T.K., Lukin, J.A., Zhan, Y., McCallum, S.A. and Rule, G.S. (2003) *J. Biomol. NMR* **25**, 1–9.

Holland, J.H. (1975) *Adaptation in Natural and Artificial Systems an Introductory Analysis with Applications to Biology Control, and Artificial Intelligence*, University of Michigan Press, Ann Arbor.

Holland, J.H. (1992) *Adaptation in Natural and Artificial Systems an Introductory Analysis with Applications to Biology Control, and Artificial Intelligence*, MIT Press, Cambridge, Mass.

Hyberts, S.G. and Wagner, G. (2003) *J. Biomol. NMR* **26**, 335–344.

Ikura, M., Kay, L.E. and Bax, A. (1990) *Biochemistry* **29**, 4659–4667.

Jung, Y.S. and Zweckstetter, M. (2004) *J. Biomol. NMR* **30**, 11–23.

Koradi, R., Billeter, M., Engeli, M., Güntert, P. and Wüthrich, K. (1998) *J. Magn. Reson.* **135**, 288–297.

Koza, J.R. (1996) *Genetic Programming: Proceedings of the First Annual Conference 1996*, MIT Press, Cambridge Mass.

Leopold, M.F., Urbauer, J.L. and Wand, A.J. (1994) *Mol. Biotech.* **2**, 61–93.

Landau, L.D. and Lifshitz., E.M. (1980) *Statistical Physics (Part 1)* (3rd edn.). Pergamon Press Oxford New York.

Leutner, M., Gschwind, R.M., Liermann, J., Schwarz, C., Gemmecker, G. and Kessler, H. (1998) *J. Biomol. NMR* **11**, 31–43.

Li, K.B. and Sanctuary, B.C. (1997) *J. Chem. Inform. Computer Sci.* **37**, 467–477.

Lin, Y. and Wagner, G. (1999) *J. Biomol. NMR* **15**, 227–239.

Lukin, J.A., Gove, A.P., Talukdar, S.N. and Ho, C. (1997) *J. Biomol. NMR* **9**, 51–166.

Malmodin, D., Papavoine, C.H. and Billeter, M. (2003) *J. Biomol. NMR* **27**, 69–79.

Michalewicz, Z. and Fogel, D.B. (2000) *How to Solve it: Modern Heuristics* Springer, Berlin.

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

Moseley, H.N., Monleon, D. and Montelione, G.T. (2001) *NMR Biol. Macromol. Pt B*, **339**, 91–108.

Nelson, S.J., Schneider, D.M. and Wand, A.J. (1991) *Biophys. J.* **59**, 1113–1122.

Nissen, V. and Propach, J. (1998) Parallel Problem Solving from Nature, Vol. 5, pp. 159–168.

Olson, J.B., Jr. (1995) Ph.D. thesis, University of Wisconsin-Madison.

Olson, J.B. Jr. and Markley, J.L. (1994) *J. Biomol. NMR* **4**, 385–410.

Permi, P. and Annila, A. (2001) *J. Biomol. NMR* **20**, 127–133.

Rana, S., Whitley, L.D. and Cogswell, R. (1996) *Lecture Notes in Computer Science (LNCS)* **1141**, 196–207.

Stroud, P.D. (2001) *IEEE Trans. Evol. Comput.* **5**, 66–77.

Talagrand, M. (1995) *Publications Mathématiques de l'I. H. E. S.* **81**, 73–205.

Tutte, W.T. (1947) *J. London Math. Soc.* **22**, 107–11.

Wang, L., Eghbalnia, H., Bahrami, A. and Markley, J.L. (2005) *J. Biomol. NMR* (in press).

Wider, G., Lee, K.H. and Wüthrich, K. (1982) *J. Mol. Biol.* **155**, 367–388.

Wolpert, D.H. and Macready, W.G. (1997) *IEEE Trans. Evol. Comput.* **1**, 67–82.

Xu, J., Straus, S.K., Sanctuary, B.C. and Trimble, L. (1993) *J. Chem. Inf. Comput. Sci.* **33**, 668–682.

Zimmerman, D., Kulikowski, C., Wang, L.Z., Lyons, B. and Montelione, G.T. (1994) *J. Biomol. NMR* **4**, 241–256.