

# A Random Graph Approach to NMR Sequential Assignment

CHRIS BAILEY-KELLOGG,<sup>1</sup> SHEETAL CHAINRAJ,<sup>2</sup> and GOPAL PANDURANGAN<sup>2</sup>

## ABSTRACT

Nuclear magnetic resonance (NMR) spectroscopy allows scientists to study protein structure, dynamics and interactions in solution. A necessary first step for such applications is determining the resonance assignment, mapping spectral data to atoms and residues in the primary sequence. Automated resonance assignment algorithms rely on information regarding connectivity (e.g., through-bond atomic interactions) and amino acid type, typically using the former to determine strings of connected residues and the latter to map those strings to positions in the primary sequence. Significant ambiguity exists in both connectivity and amino acid type information. This paper focuses on the information content available in connectivity alone and develops a novel random-graph theoretic framework and algorithm for connectivity-driven NMR sequential assignment. Our random graph model captures the structure of chemical shift degeneracy, a key source of connectivity ambiguity. We then give a simple and natural randomized algorithm for finding optimal assignments as sets of connected fragments in NMR graphs. The algorithm naturally and efficiently reuses substrings while exploring connectivity choices; it overcomes local ambiguity by enforcing global consistency of all choices. By analyzing our algorithm under our random graph model, we show that it can provably tolerate relatively large ambiguity while still giving expected optimal performance in polynomial time. We present results from practical applications of the algorithm to experimental datasets from a variety of proteins and experimental set-ups. We demonstrate that our approach is able to overcome significant noise and local ambiguity in identifying significant fragments of sequential assignments.

**Key words:** nuclear magnetic resonance (NMR) spectroscopy, automated sequential resonance assignment, random graph model, randomized algorithm, Hamiltonian path.

## 1. INTRODUCTION

**N**UCLEAR MAGNETIC RESONANCE (NMR) spectroscopy plays a vital role in post-genomic tasks including determination of protein structure, dynamics, and interactions. Full automation is required in order for NMR to fulfill its potential in supporting such applications at a genomic scale (Montelione *et al.*, 2000; Stevens *et al.*, 2001). In particular, much work (Moseley and Montelione, 1999) has focused on

---

<sup>1</sup>Department of Computer Science, Dartmouth College, Hanover, NH 03755.

<sup>2</sup>Department of Computer Science, Purdue University, West Lafayette, IN 47907.

automated *sequential resonance assignment*, an essential step for these applications. NMR spectra operate in terms of atomic “IDs” called *chemical shifts*; the problem of resonance assignment is to determine the correspondence between these chemical shifts and the atoms in the known primary sequence. Significant progress has been made in assignment algorithms, employing approaches including best-first search (Li and Sanctuary, 1997; Zimmerman *et al.*, 1997), exhaustive and heuristic search (Atreya *et al.*, 2000; Bailey-Kellogg *et al.*, 2000; Güntert *et al.*, 2000; Vitek *et al.*, 2004), branch-and-bound (Lin *et al.*, 2002), genetic algorithms (Bartels *et al.*, 1997), simulated annealing (Buchler *et al.*, 1997), and Monte Carlo optimization (Lukin *et al.*, 1997; Leutner *et al.*, 1998; Hitchens *et al.*, 2003), as well as alternative protocols using additional or different sources of information (Bailey-Kellogg *et al.*, 2000; Erdmann and Rule, 2002; Langmead and Donald, 2004).

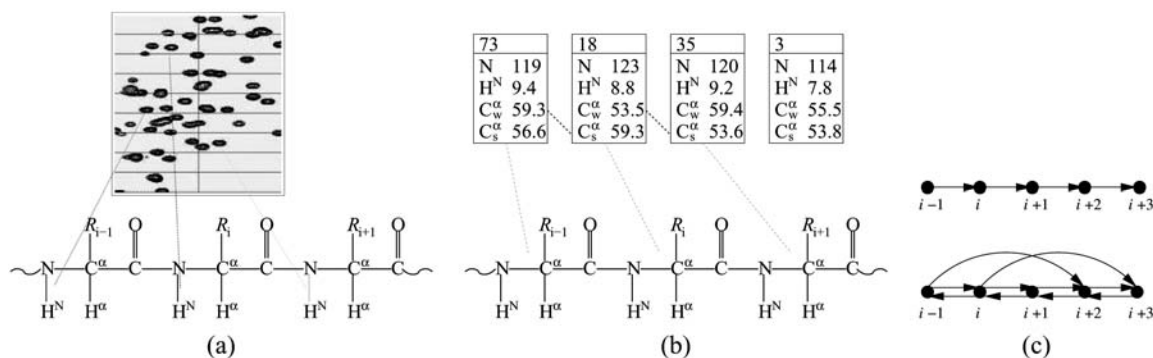
The primary information content in NMR data utilized by typical sequential resonance assignment algorithms can be categorized into *connectivity* information, indicating possible atomic interactions (through a small number of bonds, or through space), and *amino acid type* information, roughly characterizing the type of residue containing the atoms. Both types of information are often highly ambiguous; for example, an NMR peak could be explained by several or even tens of possible mutually exclusive connectivities, and amino acid type information might be more or less consistent with any or all residues. Assignment algorithms typically use connectivity information to piece together short strings of connected atoms (spanning multiple residues) and use amino acid type information to identify a consistent substring in the primary sequence. These two aspects can be employed in separate “growing” and “aligning” phases (e.g., Bailey-Kellogg *et al.*, [2000], Güntert *et al.*, [2000], Lin *et al.*, [2002]) or in a simultaneous approach of growing aligned strings (e.g., Zimmerman *et al.* [1997], Vitek *et al.* [2004], and many others). A goal essential to the wide-spread acceptance and utilization of automated assignment algorithms is the analysis of assignment algorithms in terms of the (ambiguous) connectivity and sequence information in data provided by the experimentalist.

This paper explores the role of connectivity information alone in sequential resonance assignment. Both the two-phase and simultaneous assignment approaches described above either implicitly or explicitly rely on some “good” connectivity information to help overcome the significant ambiguity in amino acid type information. For example, the two-phase MAPPER program (Güntert *et al.*, 2000) starts with unambiguously connected strings and explores possible matches against the primary sequence. In such an approach, the runtime and output quality depend on the length of the connected strings, and thus the number of permutations of them that must be explored. In fact, it might be worthwhile to allow some ambiguity in the strings grown (e.g., by independently trying multiple possibilities at connectivity choice points) in order to increase their length and thereby decrease the number of consistent alignments against the primary sequence. Intuitively, each additional match provides another “bit” helping distinguish possible locations in the primary sequence (Bailey-Kellogg *et al.*, 2000).

While the role and algorithmic impact of amino acid type information have been studied, the corresponding information content and algorithmic impact of connectivity alone have not. We demonstrate here that enforcing *global consistency* of *local connectivity choices* overcomes a great deal of ambiguity in growing, so that only a small number of the possible strings need to be explicitly explored for alignment. We develop a novel growing algorithm that *effectively reuses* connected strings, and we analyze the algorithm in terms of a novel random graph formalism that captures the key source of ambiguity in connectivity. Our approach is generic to different approaches to assignment and, in complementing other existing work on aligning, promises to lead to a solid algorithmic basis for resonance assignment.

## 2. GRAPH REPRESENTATION OF NMR DATA

NMR spectroscopy captures magnetic interactions between atoms as peaks in multidimensional spectra (Cavanagh *et al.*, 1996). For example, the HSQC (heteronuclear single quantum coherence spectroscopy) experiment of Fig. 1(a) correlates bonded  $H^N-N$  atom pairs and yields a two-dimensional spectrum. Peaks in a spectrum indicate pairs of atoms that exhibit a particular interaction (amide bond in HSQC); axes are resonance frequencies, or *chemical shifts*, of the interacting atoms. Three-dimensional experiments correlate triplets of atoms connected by a few bonds; e.g., HN(CO)CA correlates the pair  $H^N-N$  of one residue with  $C^\alpha$  of the preceding residue, thereby capturing *sequential* interactions, while HNCA correlates  $H^N-N$  with



**FIG. 1.** NMR assignment. (a) HSQC spectrum. The axes indicate  $H^N$  and N chemical shifts, so that a peak corresponds to a bonded  $H^N$ -N atom pair with those shifts. However, the correspondence (assignment) between chemical shifts and atoms is unknown. (b) Assignment of spin systems. Each spin system here has within-residue N,  $H^N$ , and  $C^\alpha$ , as well as sequential  $C^\alpha$ . Matching sequential- $C^\alpha$  of one to within- $C^\alpha$  of another helps identify sequential connectivities, and aligning the connected spin systems to positions of consistent amino acid type then defines an assignment. (c) NMR interaction graphs: nodes represent residues, and edges ideal connectivities for (top) sequential backbone (e.g., based on within vs. sequential  $C^\alpha$  match as above) and (bottom) NOESY  $\alpha$ -helix.

both preceding and within-residue  $C^\alpha$ , capturing both sequential and *within-residue* interactions. Similar experiments involve interactions with  $C^\beta$ ,  $H^\alpha$ , and CO.

Individual NMR spectra provide no information on which residue locations in the protein sequence originated each peak. In order for spectral information to be useful, a scientist must determine the mapping from chemical shifts to individual atoms, the *resonance assignment*. Typically, data from a set of through-bond experiments is maintained in a data structure called a *spin system*, which references peaks by shared  $H^N$ -N shifts and contains sequential and within-residue shifts of the remaining atom types. Then assignment proceeds (Fig. 1(b)) by matching sequential and within-residue chemical shifts to determine sequential connectivity and aligning the connected spin systems to substrings of the primary sequence (Wüthrich, 1986). Evidence for the alignment is given by consistency of the observed chemical shifts with those expected for the corresponding residue type, based on the well-characterized effect of primary sequence on resonance (Seavey *et al.*, 1991).

An alternative approach to assignment (Stefano and Wand, 1987; Bailey-Kellogg *et al.*, 2000) uses through-space (NOESY) interactions rather than the through-bond interactions described above. For example, in an  $\alpha$ -helix, sequential residues have  $H^N$ - $H^N$  NOE interactions, while residues  $i$  and  $i+3$  have  $H^\alpha$ - $H^N$  NOEs. It is worth noting that in these  $^{15}\text{N}$ -only approaches, the  $H^N$ - $H^N$  interactions are symmetric (i.e., one from  $i$  to  $i+1$  and one from  $i+1$  to  $i$ ), but the  $H^\alpha$ - $H^N$  interactions are only from  $i$  to  $i+3$ . More complex interaction patterns arise in  $\beta$ -sheets, involving networks of cross-sheet connectivities, and sparse connectivity information (primarily sequential  $H^\alpha$ - $H^N$ ) is also available in random coil regions. The basic approach to assignment based on NOESY still follows the generic pattern of determining connectivities and aligning connected portions against the primary sequence.

This paper focuses on the analysis of connectivity information and is generic to the source of connectivity—NOESY or any of many possible through-bond experiments used in traditional assignment algorithms. Thus we abstract the NMR data into a graph which will be the input for our algorithm.

**Definition 1 (NMR interaction graph).** An NMR interaction graph  $G = (V, E)$  is a labeled, weighted, directed graph with vertices  $V$  corresponding (via an unknown mapping) to residues (or to noise, for extras), and edges  $E \subset V \times V$  such that  $e = (v_1, v_2) \in E$  iff there are observed NMR peaks that could be explained by interactions between atoms of  $v_1$  and atoms of  $v_2$ . Edges are labeled with an interaction type indicating which atoms interact in spectral peaks, and a match score indicating the quality of the chemical shift match.

For example (see Fig. 1(c)), a correct edge for an interresidue peak from an HNCA experiment would connect the vertex for one residue (via  $H^N$  and N) with the vertex for the preceding residue (via  $C^\alpha$ ).



**FIG. 2.** Chemical shift degeneracy. (a) A correct  $A \rightarrow B$  edge will also introduce an incorrect  $C \rightarrow B$  edge, since  $C$ 's chemical shift for the given atom type (on the  $y$ -axis) is very similar to  $A$ 's. (b) Random graph model introduces noise edges (dotted lines) to vertex (b) from vertices (c,d) in a window of width  $w$  around the correct vertex (a). The order of vertices represents sorted order by chemical shift, so that the correlation structure among edges models that in NMR data.

Similarly, correct edges for interresidue peaks from an  $^{15}\text{N}$ -NOESY in an  $\alpha$ -helix region would connect the vertex for one residue (via  $\text{H}^{\text{N}}$  and  $\text{N}$ ) with the vertex for the preceding residue (via  $\text{H}^{\text{N}}$ ), with the vertex for the subsequent residue (via  $\text{H}^{\text{N}}$ ), and with the vertex for the third residue back (via  $\text{H}^{\alpha}$ ). The match score in each of these cases would simply compare the chemical shift observed in the interresidue peak with the shift expected for the partner residue (from intrasidue peaks). When multiple peaks are combined in an edge (e.g., with  $\text{HNCACB}$ , using both  $\text{C}^{\alpha}$  and  $\text{C}^{\beta}$ ), scores must be appropriately combined, accounting for missing atom types (Vitek *et al.*, 2004).

For the remainder, we assume that an interaction graph has been derived from the spectral data, by standard compilation of spin systems (vertices) and identification of “reasonable” matches between inter- and intrasidue chemical shifts (edges) (Zimmerman *et al.*, 1997). Let  $G_* = (V_*, E_*)$  denote the (unknown) interaction graph containing only physically correct vertices and edges. The graph  $G$  derived from NMR data differs substantially from  $G_*$ , due to noise and missing data; one of the goals of assignment is to identify the correct subgraph. Basic addition and deletion of vertices and edges can be treated as essentially random (e.g., a peak has some chance of falling below the noise floor, thereby deleting the corresponding edge). However, there is an interesting structure in NMR data that gives rise to a large amount of ambiguity in defining the edges from the data and thus shows up as *correlated* noise edges. Recall that an edge is generated by matching the observed interresidue chemical shift for a peak associated with one vertex, with the expected intrasidue chemical shift associated with the other vertex. The key source of ambiguity, called *chemical shift degeneracy* (Fig. 2(a)), is the fact that, after accounting for peak width and potential measurement error, multiple possible vertices have intrasidue chemical shifts “close enough” to the interresidue observation. Thus each correct edge is obscured by several or many incorrect edges. The incorrect edges are not randomly distributed, but rather correlated, since when two vertices have atoms that are similar in chemical shift, they will tend to share edges—each edge for the one will also appear for the other.

Traditional  $G_{n,p}$  random graph models (Bollobas, 2001) essentially add noise edges randomly and independently and do not capture the correlation, due to chemical shift degeneracy, of noise edges among themselves and with correct edges. In order to develop and analyze an assignment algorithm that focuses on connectivity, we now define a simple model that does account for this correlation. The following focuses on edges derived by matching an interresidue chemical shift of a particular atom type  $a$  (e.g.,  $\text{C}^{\alpha}$ ). For edges comprised of matches with  $d$  atom types, e.g., using both  $\text{C}^{\alpha}$  and  $\text{C}^{\beta}$ , treat each atom type independently and apply the following to add correlated noise edges:

1. Choose a permutation of  $V$ ; denote by  $\pi(v)$  the index of  $v$  in the permutation. This corresponds to sorting the vertices by the chemical shift of atom type  $a$ . There is no systematic, global correlation between positions of atoms in the primary sequence or in space and positions in the sorted list of chemical shifts. Thus the order of atoms with respect to chemical shift is well-modeled by a random permutation. Prior information on permutations (e.g., from chemical shift predictions [Neal *et al.*, 2003]) can be incorporated when available.
2. Let  $w(v)$  be the window width around vertex  $v$ , indicating with how many other vertices  $v$  are degenerate (with respect to the single atom type  $a$  being considered). This captures peak width and local “clustering”

- in chemical shift space. We note that, if  $w(v)$  is to be sampled from a distribution, care must be taken to ensure that degeneracy is symmetric (except for small perturbations capturing different peak width).
- For each  $u \in V$ , let  $C(u)$  be the set of vertices  $\{v \in V \mid |\pi(v) - \pi(u)| \leq w(u)\}$ . (If there is more than one atom type, then  $C(u)$  will include a vertex  $v$  if and only if  $|\pi^a(v) - \pi^a(u)| \leq w(u)$ , for “enough” atom types  $a$ .) These vertices are degenerate with  $u$  with respect to atom type  $a$ .
  - For each directed edge  $(u, v) \in E_*$ , add directed edges  $\{(u', v) \mid u' \in C(u)\}$  to  $E$ .
  - Set edge weights as a function of distance within the permutation plus noise. This captures typical scoring rules (e.g., Zimmerman *et al.* [1997], Güntert *et al.* [2000], Vitek *et al.* [2004]) that compare absolute or squared difference in chemical shift. Note that, except for noise (reasonably modeled as Gaussian), the correct edge should match exactly and thus have the best score.

Figure 2(b) illustrates a chemical shift degeneracy window and resulting noise edges. In typical 3D NMR experiments, the ambiguity is one-sided, since the peak has two chemical shifts ( $H^N$ ,  $N$ ) anchoring it in the residue at which the peak was observed, but only one interresidue chemical shift on which the match is based. It is straightforward to extend this model to allow nonzero ambiguity on the “to” side of the edge, or to allow two matches on the “from” side (e.g., for 4D spectra [Erdmann and Rule, 2002]).

### 3. PATH-FINDING FOR ASSIGNMENT

As discussed in the introduction, the “growing” aspect of sequential resonance assignment uses connectivity information (edges in an NMR interaction graph) to identify chains of possibly connected atoms. We develop here an algorithm that pushes that aspect to its logical extreme and seeks to find not just short unambiguous chains, but rather groups of chains that connect the entire set of vertices.

**Definition 2 (sequential fragment).** For an NMR graph  $G = (V, E)$ , a sequential fragment  $F = (V', E')$ , where  $V' = \langle v_1, v_2, \dots, v_{|F|} \rangle \subset V$  is a sequence of vertices and  $E' \subset \{(u, v) \in E \mid u, v \in V'\}$  is a set of edges supporting that sequential order.

This is not necessarily a path; for example, in an  $\alpha$ -helix, there might be only an  $H^N$ – $H^N$  edge from  $v_{i+1}$  to  $v_i$  (in the opposite direction from sequentiality), and there might be an  $H^\alpha$ – $H^N$  edge from  $v_i$  to  $v_{i+3}$ . However, since sequential connectivity is the motivation, we will loosely refer to it as a path. We extend the notion of match score to fragments by summing over all appropriate edges.

**Definition 3 (sequential cover).** For an NMR graph  $G = (V, E)$ , a sequential cover  $C = \{F_1, F_2, \dots, F_{|C|}\}$  is a set of fragments such that each vertex  $v \in V$  appears in exactly one fragment.

Some of the vertices (e.g., extras) can appear in singleton fragments. We extend the notion of match score to covers by summing over all fragments. Note that the physically correct graph  $G_*$ , with vertices ordered by residue number, corresponds to a sequential cover. The goal of NMR assignment is to find the underlying  $G_*$  given a noisy  $G$ , but the best we can hope to do is to find “good” subgraphs  $G' \subset G$  and evidence that  $G_*$  is among them. In particular, we focus here on finding sequential covers of  $G$  with near-optimal match score. Our optimization problem is then to find the sequential cover of maximum score for a given NMR graph. We note that, due to variability in the data and in order to not be too sensitive to the scoring model, it is also desirable to find and compare near-optimal covers as well. We discuss in Section 4 a practical implementation that does that.

The NP-hardness of this problem is immediate by reduction from Hamiltonian path. We focus here on a randomized algorithm that takes advantage of the problem structure to perform very well in practice (see Section 4). We also use our model to analyze the algorithm in various versions. The key insight is that, in searching for good covers, the best ones tend to share a lot of substructure—fragments that differ in some chunks but are largely the same. It is hard for traditional branching-based searches to take advantage of this shared structure because it can appear on many different branches. For example, multiple different paths can take a jump to a common vertex, after which they are all the same; each of

these would be on a different branch. However, algorithms as developed for finding Hamiltonian paths for traditional  $G_{n,p}$  random graphs (Angluin and Valiant, 1979) perform only small rearrangements to a path in order to continue extending it. Inspired by the success of the rotation and extension algorithms for finding Hamiltonian paths, we develop a simple-to-implement randomized algorithm that exploits shared substructure in a natural way and also handles breaks and scoring.

The basic randomized algorithm is as follows (Fig. 3). Let  $C$  be a sequential cover. Denote by  $\text{succ}(u, C)$  and  $\text{pred}(u, C)$  the successor and predecessor of  $u$  in  $C$  (these can be null). Initially,  $C = V$ ; i.e., each vertex is a fragment by itself. We will view the algorithm in two phases. In Phase 1, we extend all *unambiguous* vertices—those with out-degree 1 to vertices with in-degree 1. At the end of Phase 1,  $C$  will be a collection of potentially nontrivial fragments. In Phase 2, we extend *ambiguous* vertices (with out-degree more than 1, or with an edge to a vertex with in-degree more than 1), choosing edges with probability proportional to their scores, and potentially swapping a chosen edge for a previously chosen edge. Note that cycles can be formed during the execution of the algorithm, when an edge connects “back” to a vertex in a path. Thus, at all times,  $C$  will consist of disjoint sets of (simple) paths or cycles. The algorithm is simple to implement and has two desirable properties: it naturally reuses already-found paths (i.e., makes only local changes in each step) and also gives preference to edges with higher scores. Assuming that scores capture the probability of the edge being the “correct edge” (refer again to edge weights in our random graph model for a justification), the algorithm nicely balances both local and global criteria.

We will present an analysis of the algorithm on a simplified NMR interaction graph model: the random graph model assuming a random permutation and a fixed window size  $w$  generating an NMR interaction

```

Given  $G = (V, E)$ 
Let initial cover  $C = V$ 
Let vertices with successors  $W = \emptyset$ 
Choose vertex  $u$  from  $V$ 

Phase 1:
Let visited vertices  $U = \emptyset$ 
While  $U \neq V$  do
  Add  $u$  to  $U$ 
  If  $u$  has single out-edge  $e = (u, v)$  and  $v$  has a single in-edge then
    Add  $e$  to  $C$ 
    Add  $u$  to  $W$ 
    Set  $u$  to  $v$ 
  Else Choose  $u$  from  $V - U$ 
endwhile

Phase 2:
While  $C$  is not a Hamiltonian path or cycle do
  Choose  $u$  from  $V - W$ 
  Choose an edge  $(u, v)$  with probability proportional to its weight
  If  $\text{pred}(v, C)$  is null then
    Join the two fragments in  $C$ 
    Add  $u$  to  $W$ 
  Else
    Create two fragments in  $C$ :  $\langle \dots u, v \dots \rangle$  and  $\langle \dots, \text{pred}(v, C) \rangle$ 
    Add  $u$  to  $W$ 
    Remove  $\text{pred}(v, C)$  from  $W$ 
endwhile

```

FIG. 3. The basic randomized algorithm for NMR interaction graph sequential cover.

graph  $G = (V, E)$  from the correct sequential connectivities  $G_* = (V_*, E_*)$ . For now, we assume no edge weights, no breaks, and no other sources of noise (additions and deletions); the result can be generalized.

Our analysis relates the performance of the algorithm to  $w$ . We show that if  $w$  is not very large, then the algorithm will find a long path (of length at least  $\Omega(n/\log n)$ ) in polynomial time (cf. Theorem 1).<sup>1</sup> Actually, our analysis shows more. By giving a simple and natural *random walk* interpretation of our algorithm, we show that the algorithm will terminate with a “good” long path, i.e., one whose expected value is close to the optimum (cf. Corollary 1).

We first need the following lemmas whose analysis relates the performance of the algorithm to a random walk. For simplicity of analysis, henceforth we will assume that the input NMR graph has a Hamiltonian cycle (instead of just a Hamiltonian path); i.e., the two endpoints of the path are connected by directed edge; it is easy to modify the analysis to work without this assumption.

The following lemma analyzes the performance of the randomized algorithm when the input graph is Hamiltonian and each vertex has at most two outgoing edges. We note that it is still NP-hard to find a Hamiltonian cycle in such a graph (Plesnik, 1979). A *2-factor* of  $G$  is a collection of vertex-disjoint cycles which cover all the vertices of  $G$ ; i.e., every vertex belongs to a nontrivial cycle.

**Lemma 1.** *Let  $G$  be a directed unweighted graph on  $n$  vertices such that each vertex has at most two outgoing edges. Assuming that  $G$  has a (directed) Hamiltonian cycle, the randomized algorithm will either terminate with a Hamiltonian cycle or with a 2-factor of  $G$  in expected  $O(n^2)$  steps.*

**Proof.** We first analyze the basic randomized algorithm given in Fig. 3. Let  $H$  be a Hamiltonian cycle in  $G$ . We focus on the algorithm finding  $H$ . At the end of Phase 1, let  $C \neq H$ ; otherwise we are done. The algorithm will terminate if each vertex has the “correct” outgoing edge, i.e., the edge belongs to  $H$ . Phase 1 takes  $O(n)$  steps.

We view the algorithm as a random walk on a line numbered  $0, \dots, n$  as follows. In some step (one execution of the while loop), say  $t$ , the algorithm is stationed at number  $k$  where  $k$  is the number of “correct” edges, i.e., those belonging to  $H$ . In the beginning,  $k = 0$ . We claim that in step  $t + 1$ , the algorithm can move to  $k + 1$  with probability at least  $1/2$ . The reasoning is as follows. Let  $u$  be the vertex, without a successor, considered at this step. Then the probability that the correct edge, say  $(u, v)$ , is chosen is at least  $1/2$ . Now, how does this affect the probability of some other edge? The only other edge which is affected is  $(\text{pred}(v, C), v)$ . There are two cases: (a)  $\text{pred}(v, C)$  is null, in which case we have  $k + 1$  correct edges; (b) otherwise, we lose the edge  $(\text{pred}(v, C), v)$ ; but the probability that that edge is correct *conditioned* on the fact that  $(u, v)$  is correct is 0. Thus, with probability at least  $1/2$ , we add one correct edge. With probability at most  $1/2$ , the algorithm can move to  $k - 1$  or stay at  $k$ . We are interested in the expected number of steps needed to reach  $n - 1$ . By setting up a difference equation (for example, see Grimmett and Stirzaker [1992, pp. 73–74]), we can show that the expected number of steps needed to reach  $n$  is at most  $n^2$ .

The above analysis assumes that in every step there is a vertex that does not have a successor. If not, this implies that  $C$  will consist of a set of disjoint (nontrivial) cycles covering every vertex, i.e., a 2-factor. ■

The following theorem shows that we can get a polynomial time performance: our randomized algorithm will find a Hamiltonian cycle or a path of length at least  $\Omega(n/\log n)$  in expected polynomial (in  $n$ ) steps, even for relatively large  $w$  (i.e., substantial chemical shift degeneracy). *This gives us a theoretical explanation as to why our algorithm performs pretty well in practice.* In the following, the number of atom types  $d$  is a constant, established by the experimental set-up. While the following arguments still hold (somewhat more weakly) when only one atom type is used, the most interesting results are obtained when there are at least two atom types matched per edge, so that information is combined in order to overcome noise. This is true in all typical backbone NMR protocols, as well as in the NOESY  $\alpha$ -helix case described above. For simplicity of formulas, we apply a uniform window width  $w$  that is the same for each atom type.

<sup>1</sup>It is interesting to mention that an algorithm of Frieze *et al.* (Broder *et al.*, 1994) finds a Hamiltonian path in a graph containing a Hamiltonian path and a *large* number (at least  $cn$  for some large constant  $c$ ) of random edges.

We first show that finding a 2-factor is enough to find a cycle (or path) of length at least  $\Omega(n/\log n)$  w.h.p. (i.e., with probability at least  $1 - 1/n$ ) in our random graph model.

**Lemma 2.** *Let  $G$  be an NMR interaction graph generated by the simplified model using  $d > 1$  matched atom types for each edge and such that each vertex has at most two outgoing edges. Then w.h.p. any 2-factor will contain a cycle of length at least  $\Omega(n/\log n)$ .*

**Proof.** The NMR random graph model with at most two outgoing edges essentially behaves like a directed 2-regular random Hamiltonian graph, i.e., a Hamiltonian cycle with a random perfect matching added. We can show that such a random graph can be partitioned into at most  $\Theta(\log n)$  vertex-disjoint cycles w.h.p. (Pandurangan, 2005). Hence, any 2-factor will consist of a cycle of length  $\Omega(n/\log n)$  w.h.p. We show the proof for the NMR random graph model as follows.

Let  $G$  be an NMR random graph minus the Hamiltonian edges. (We refer to the edges on the Hamiltonian cycle as *Hamiltonian edges* and refer to the other edges as *random edges*.) We can show that the number of (disjoint) cycles in any 2-factor is  $\Theta(\log n)$  as follows. Consider any 2-factor which induces a permutation  $\sigma$  on the vertex set. Consider the *cycle permutation* of  $\sigma$ , i.e., the cycles of the permutation  $\sigma$  arranged in increasing order of the smallest element of the cycles. Let  $X_i$  be the indicator r.v. for the  $i$ th element of the cycle permutation to be the last element of a cycle. Then  $\Pr(X_i = 1) = 1/(n - i + 1)$  because there are  $n - i + 1$  equally likely possibilities for  $\sigma(i)$  (one of the remaining elements, or closing the cycle). The number of cycles is  $\sum_{i=1}^n X_i$  and the expectation is  $\sum_{i=1}^n 1/(n - i + 1) = \Theta(\log n)$ . Also, the random variables  $X_i$  are almost independent. The dependence is due to the fact that if  $(k, l)$  is a random edge then so is  $(l - 1, k + 1)$ , and vice versa, because to generate random edge  $(k, l)$  from Hamiltonian edge  $(l - 1, l)$ ,  $k$  must be in the window of  $l - 1$ , and then since uniform windows are symmetric, Hamiltonian edge  $(k, k + 1)$  also generates random edge  $(l - 1, k + 1)$ . Thus each  $X_i$  can depend on at most one other  $X_j$ . By partitioning the sum  $X$  into two disjoint sets of independent r.v.s and applying the Chernoff bound to each, we can show that the number of cycles is  $\Theta(\log n)$  w.h.p. This is still true after adding the Hamiltonian edges, since the number of vertex disjoint cycles with  $b$  random edges ( $b \geq 1$ ) and  $n - b$  Hamiltonian edges is stochastically dominated by the number of disjoint cycles in an NMR random graph with  $b$  nodes (collapsing the paths of Hamiltonian edges) with only random edges.

Now, there can be at most  $O(n^2)$  (i.e., polynomial in  $n$ ) different 2-factors in  $G$  w.h.p. This is because the expected number of different 2-factors is at most  $\sum_{b=1}^n \binom{n}{b} b! O(1/n^{bd/2}) = O(n)$ , since  $b$  random edges can be inserted in place of  $b$  Hamiltonian edges in  $\binom{n}{b}$  ways and there are at most  $b!$  different 2-factors each with probability  $O(1/n^{bd/2})$  (the occurrence of at least half of the edges is independent and an edge will occur if it falls into the window in all  $d$  atom types). Markov's inequality gives the probability bound. The proof is finished by using the union bound since w.h.p. each 2-factor has only  $\Theta(\log n)$  disjoint cycles and there are only at most  $O(n^2)$  2-factors w.h.p. ■

**Theorem 1.** *Let  $G = (V, E)$  be an NMR interaction graph of degeneracy window width  $w$  on  $n$  nodes generated by the simplified model, using  $d > 1$  matched atom types for each edge ( $d$  is an integer constant independent of  $n$ ). Then, if  $w = o(n^{1-(3/2d)})$ , a path of length  $\Omega(n/\log n)$  can be found asymptotically almost surely in expected  $O(n^2)$  steps.*

**Proof.** The probability that a *noisy* directed edge  $(i, j)$  exists, based on a match for one atom type, is  $O(w/n)$ , since such an edge will exist if  $i$  falls within the window of width  $w$  surrounding the true predecessor of  $j$ . Thus the probability of a *fooling edge* between two vertices, with matches for all  $d$  atom types, is  $O(w^d/n^d)$  (since we assume that matches with different atom types are independent). Thus the expected number of fooling edges is  $O(w^d/n^{d-2})$ .

The probability that a vertex is bad (i.e., has more than two fooling edges) is  $O(w^{2d}/n^{2d-2})$ . Thus the expected number of bad vertices is  $O(w^{2d}/n^{2d-3})$ .

If  $w = o(n^{1-(3/2d)})$ , then by the first moment method (Alon and Spencer, 2000),  $G$  will have no bad vertex asymptotically almost surely (a.a.s.).<sup>2</sup> Then, a.a.s., Lemma 1 can be applied, and it will terminate with either a Hamiltonian path or a 2-factor in expected  $O(n^2)$  steps. ■

<sup>2</sup>Probability  $\rightarrow 1$  as  $n \rightarrow \infty$ .



For example, with two atom types ( $C^\alpha + C^\beta$  in backbone protocols, or symmetric  $H^N - H^N$  interactions in NOESY  $\alpha$ -helix), if  $w = o(n^{1/4})$ , then a.s. we get the results in expected  $O(n^2)$  steps.

The above theorem can be generalized for a weighted NMR graph as follows. Consider an NMR graph  $G = (V, E)$  with parameters as stated in Theorem 1 and with edge weights. For each vertex  $v$ , assume that the weights of the outgoing edges sum to 1 and interpret the weight of an incident outgoing edge to be the probability that it is present in the correct Hamiltonian path; this is reasonable under the scoring model. The goal is to find a path with large edge weight. We have a simple, weighted version of Theorem 1:

**Corollary 1.** *Let the edges of  $G_*$  (i.e., the Hamiltonian edges) have weight at least  $p$ , for some constant  $p > 0$ . Then the randomized algorithm will find a path of weight  $\Omega(n/\log n)$  (in other words, the path found will be within a factor of at least  $\Omega(1/\log n)$  of the optimal weight path) asymptotically almost surely in expected  $O(n^2)$  steps.*

**Proof.** The proof for the expected time and path length is similar to that of Theorem 1. We note that when an outgoing edge is chosen for a vertex, the randomized algorithm chooses with probability proportional to its weight. Linearity of expectation gives the expected weight of the output path. ■

To summarize our results, Theorem 1 essentially says what parameters will give us long paths in polynomial time; Corollary 1 says that paths are also statistically significant in the context of weights. In practice, we expect correct edges to have higher scores, and this helps the randomized algorithm in making better choices, and hence finding longer paths, even in the presence of substantial noise (i.e., large  $w$  and higher number of outgoing vertices per vertex). More importantly, the paths are made up of predominantly correct (i.e., Hamiltonian) edges. Our experimental results agree with this quite well. Finally, we note that in practice we can apply this algorithm even when there are “breaks” in the input graph, i.e., if there is not a (complete) Hamiltonian path in the input graph  $G$ . We assume that  $G$  is weakly connected; otherwise, we run the algorithm on the individual weakly connected components. Our results in the next section demonstrate that the algorithm performs well in this situation also.

## 4. RESULTS

We tested an implementation of our random graph algorithm on a total of 10 experimental datasets: 7 complete backbone experiments provided with AUTOASSIGN (Zimmerman *et al.*, 1997); and the  $\alpha$ -helical subset of the  $^{15}\text{N}$ -edited NOESY data for the three proteins from the Bushweller lab previously tested with JIGSAW (Bailey-Kellogg *et al.*, 2000). In both cases, we constructed an NMR interaction graph from the spin systems previously compiled by the respective programs. The graph edges from the backbone data were formed by comparing the inter- (sequential) and intra- (within) residue chemical shifts of candidate spin systems; edge scores were computed from the difference in chemical shifts, combined over the number of atom types matched, with a penalty for missing atom types. Chemical shift differences are the basis for scoring in all NMR assignment; the precise scoring model is not our focus here and is detailed as part of a complete Bayesian model and approach in Vitek *et al.* (2004). In the  $\alpha$ -helix tests, symmetric pairs of edges were constructed for the potential  $(i, i + 1)$   $H^N - H^N$  edges, to account for lack of inherent direction; scores and penalties were calculated based on a log-likelihood function of the frequency of occurrence of these edges. Thus during path-finding, an  $\alpha$ -helix graph is treated the same as a backbone graph. Directionality in a resulting cover is established by consensus of the directed  $(i, i + 3)$   $H^\alpha - H^N$  edges.

The various experimental datasets display a range of difficulty for assignment; Table 1 roughly characterizes contributing factors. ZDOM and NS1 are small and highly  $\alpha$ -helical, but show significant chemical shift degeneracy for the  $C^\alpha$  and  $C^\beta$  atom types and also high overlap for  $H^N$ . Apart from this, the significant problem with NS1 dataset is the five entirely missing spin systems. This was also the case with BPTI dataset (eight missing spin systems). CSP, a  $\beta$ -sheet protein, had incomplete spectra and more extra spin systems. FGF, RNASEWT, and RNASEC6 were the toughest backbone datasets because of the significantly high noise, evident from the number of noise edges in the input NMR interaction graph. Though these two datasets had no entirely missing spin systems, the difficulty was increased by the large

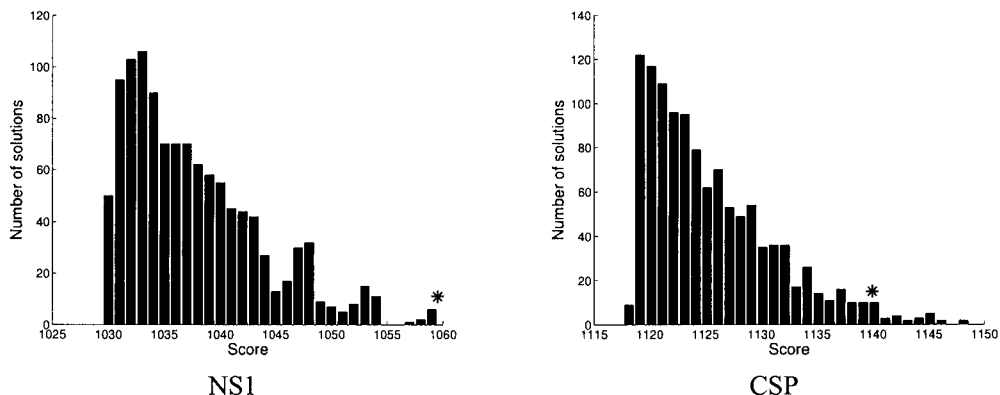
TABLE 1. SUMMARY OF DATASETS, INCLUDING FACTORS MAKING ASSIGNMENT DIFFICULT<sup>a</sup>

Dataset	Atom types	Residues	Pro	Missing	Extra	Out degree	Wrong edges	Breaks
BPTI	C <sup>α</sup> ,C <sup>β</sup> ,H <sup>α</sup>	58	4	8	5	0.90	4	4
NS1	C <sup>α</sup> ,C <sup>β</sup> ,H <sup>α</sup>	73	1	5	1	1.30	22	0
ZDOM	C <sup>α</sup> ,C <sup>β</sup> ,H <sup>α</sup>	69	3	1	1	1.50	38	0
CSP	C <sup>α</sup> ,C <sup>β</sup> ,H <sup>α</sup>	68	2	0	4	1.72	50	0
RNASEC6	C <sup>α</sup> ,C <sup>β</sup> ,CO,H <sup>α</sup>	124	4	0	30	2.07	193	1
RNASEWT	C <sup>α</sup> ,C <sup>β</sup> ,CO,H <sup>α</sup>	124	4	0	30	2.19	209	12
FGF	C <sup>α</sup> ,C <sup>β</sup> ,CO,H <sup>α</sup>	154	9	2	3	2.90	338	0
CBF $\alpha$ -helix	H <sup>N</sup> ,H <sup>α</sup>	36	NA	0	0	4.16	114	1
HUGRX $\alpha$ -helix	H <sup>N</sup> ,H <sup>α</sup>	38	NA	0	0	5.00	160	8
VACGRX $\alpha$ -helix	H <sup>N</sup> ,H <sup>α</sup>	42	NA	0	0	3.67	112	4

<sup>a</sup>The top seven use traditional backbone experiments, while the bottom three use <sup>15</sup>N-NOESY (only residues in  $\alpha$ -helices are considered in our tests). Node characteristics are summarized by the numbers of prolines and entirely missing spin systems, which cause breaks in a cover, and the number of extra spin systems (i.e., not corresponding to any residue), which can mislead the assignment by filling in for correct or missing spin systems. Edge characteristics are summarized by the mean out degree over the entire graph, including both correct and noise edges; the total number of wrong edges; and the number of missing edges (breaks), not counting those due to missing spin systems and prolines. All datasets also have some extra and missing peaks within individual spin systems; these aren't detailed here, as they only locally affect the scoring of an edge.

number (30) of extra spin systems. More details are available in the AUTOASSIGN paper (Zimmerman *et al.*, 1997). HUGRX and VACGRX are fairly small, closely related proteins with a relatively large  $\alpha$ -helical content. The spectra for these two proteins have quite different noise characteristics, in terms of particular missing edges (i.e., breaks), a key source of difficulty in obtaining a complete assignment. We note that JIGSAW provided only partial assignments of the  $\alpha$ -helical regions, due to internal breaks that fragment the helices and allow much more combinatorial swapping of fragments. CBF is a larger protein, but with a smaller total amount of  $\alpha$ -helical content. It is much more readily assigned due to having only one intrahelical break.

For each dataset, the cover-finding algorithm was run for 20,000 iterations, and all unique solutions having scores within a constant fraction (20%) of the highest score were collected. We characterize the quality of our algorithm on this ensemble of results. Figure 4 illustrates the scores in the ensemble for two example datasets. Note that the reference (published) solution need not have the highest score according to our metric, so the ensemble allows us to avoid committing to a single assignment subject to parameters in a particular scoring model. The first three columns of Table 2 summarize the growing results.



**FIG. 4.** Example score distributions over ensembles. Reference solution scores (1059.53 for NS1 and 1139.94 for CSP) are marked.

TABLE 2. GROWING AND MAPPING RESULTS<sup>a</sup>

<i>Dataset</i>	<i>Ensemble size</i>	<i>Wrong grown edges</i>	<i>Wrong mapped edges</i>	<i>Assigned residues</i>
BPTI	6	1.00	0.5	34.66
NS1	816	4.60	1.29	28.3
ZDOM	126	2.90	0.7	43
CSP	1169	4.39	1.5	41.3
RNASEC6	8852	7.28	0.95	93.88
RNASEWT	3327	11.39	1.5	50.08
FGF	81	18.27	2.02	44.95
CBF $\alpha$ -helix	111	5.53	NA	NA
HUGRX $\alpha$ -helix	19	9.40	NA	NA
VACGRX $\alpha$ -helix	785	16.60	NA	NA

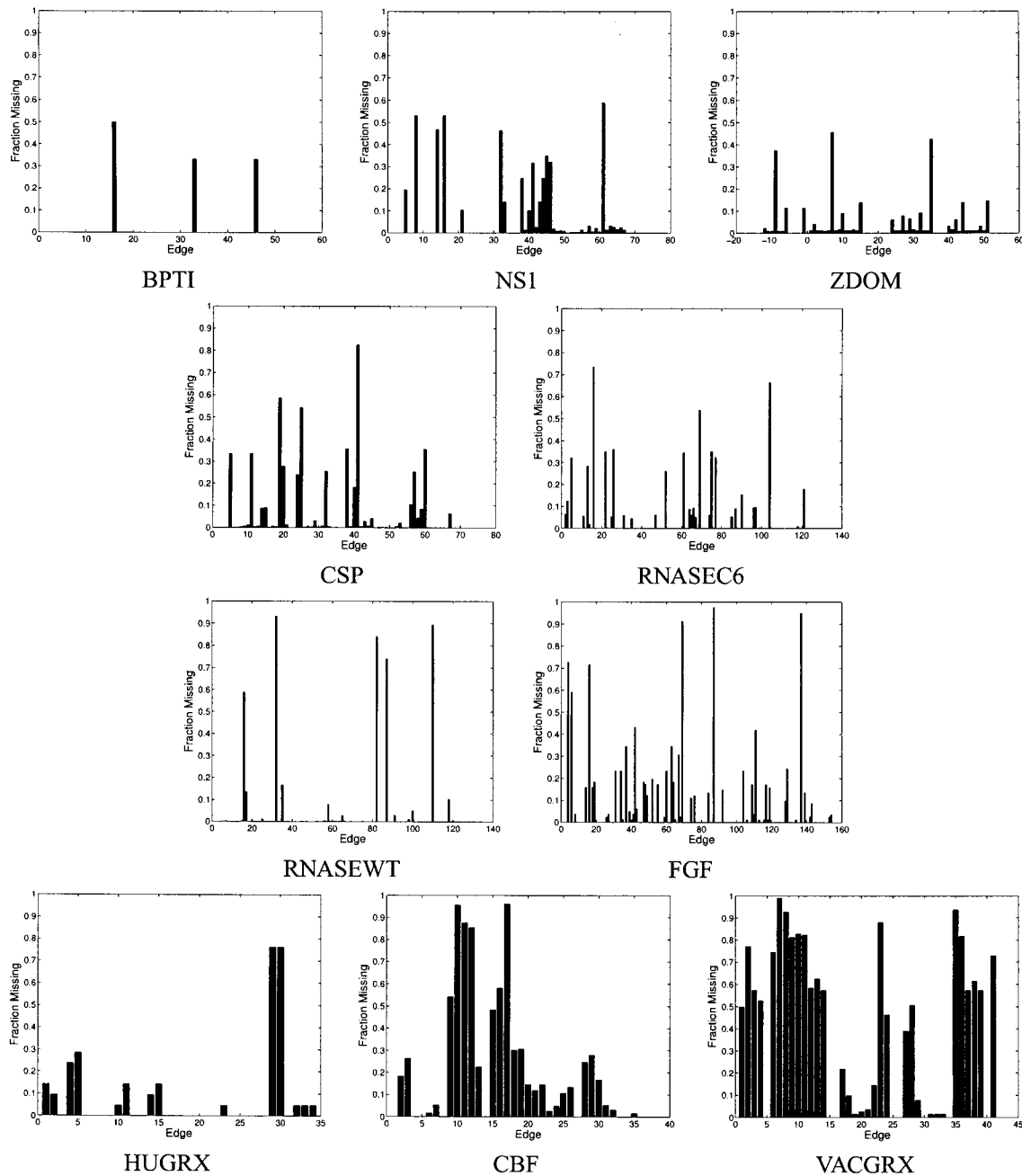
<sup>a</sup>Our algorithm generates a number of solutions (20,000) and keeps an ensemble of acceptable solutions with scores near (within 20% of) the best found. Results are averaged over these ensembles. Error rate is summarized by the mean number of wrong edges per cover. Growing alone yields a surprisingly good error rate, due to the requirement of global consistency of a cover. A simple application of MAPPER then filters the set of fragments, keeping only long enough ones that unambiguously map to the primary sequence. The error rate drops, and a large number of the residues are unambiguously assigned.

The set of nonredundant, high-enough-scoring solutions tends to be quite small. Further, the average number of wrong edges selected per cover is also quite small, indicating that most members of the ensemble are typically quite good.

The ensemble further allows us to elucidate the shared structure of high-quality assignments. We expect the same edges, mostly correct, to show up repeatedly over the solution ensemble. Figure 5 illustrates that this indeed holds true, even for extremely noisy datasets. Each plot shows the frequency of *missing* the correct sequential edges in the set of covers in the ensemble. That is, a bar of height  $h$  at position  $i$  in the chart indicates that the  $(i, i + 1)$  edge was correctly found in a fraction of  $1.0 - h$  members of the ensemble. These plots show that many correct edges are conserved in the ensembles. For clarity, the bars for prolines and entirely missing spin systems are not shown (they necessarily have a height of 1.0). More detailed analysis of these plots follows.

We begin with the analysis of the toughest datasets, in order to get insight into difficulties. RNASEC6 provides significant problems (correct edge missed in over half the covers) at residues 16, 69, and 104. To a large extent, these ambiguities are due to extra spin systems—the residues in question had good edges to extras, or succeeding residues had good edges from extras. Since these extras in turn were weakly connected to other residues, once the algorithm chose a wrong edge, there was less chance to correct it. Additional ambiguities were typically due to particularly high out-degree. But in many cases the algorithm was able to overcome these ambiguities. For example, the region from 65 to 69 had an average out-degree of about 3 and an average of two edges to extras, and yet the search for a high-scoring *globally-consistent* cover forced the correct choice in most of the members of the ensemble. Similar reasons for ambiguity exist for RNASEC6. Here we see that in spite of the noisiness of the data, most of the correct edges appear consistently. For example, only three edges (from 14, 69, and 104) were missing half the time, and this was due to high-scoring out edges from those nodes to extras. Similar arguments hold for CSP, where the ambiguities at 19, 25, and 41 are due to high out-degree (degree of 6 at 19) or good edges to extras (two edges each at 25 and 41). The FGF results, on the other hand, demonstrate the ability of the algorithm to overcome this type of ambiguity. Spin system 110 has 9 out-edges and 117 has 17, each with two edges to extras, but the correct edges appeared in 97% and 83% of the solutions. This is also the case at residue 20 in CSP, which has out-degree of 14 but yields a correct choice in 73% of the solutions. The remaining datasets (NS1, BPTI, and ZDOM) were relatively cleaner, and the algorithm was able to largely overcome most ambiguity. One interesting case was the interleaving of two correct fragments in NS1. Because of a good scoring edge (8, 17), the two correct fragments  $\langle 2 \dots 8, 9 \rangle$  and  $\langle 11 \dots 17, 18, 19 \rangle$  appeared as  $\langle 2 \dots 8, 17, 18, 19 \rangle$  and  $\langle 11 \dots 16 \rangle$  and  $\langle 9 \rangle$ .

For the  $\alpha$ -helix dataset HUGRX, the highly problematic edges occur at spin systems 94 and 95. This was seen because of the fact that the spin systems 95 and 96 had very good edges from the end-point



**FIG. 5.** Frequency of *missing* the correct edge (indexed by “from” residue number), over the ensemble of solutions.

of another helix (ending at 9). Also, the region 54–58 was highly noisy, with the symmetric  $H^N$  edges between the consecutive spin systems missing. The plot for CBF shows that the ambiguity is localized at the two regions corresponding to the end of one  $\alpha$ -helix and beginning of another. For the spin systems corresponding to these regions, the data was especially noisy, and the high number of missing  $H^\alpha(i, i + 3)$  correlations added to the ambiguity. VACGRX was the noisiest of the  $\alpha$ -helix datasets with 50% of the  $H^\alpha$  correlations in the  $\alpha$ -helices missing. Furthermore, there were significantly many good edges between spin systems of different  $\alpha$ -helices, and a number of breaks. A further indication of the noisiness of the data is the fraction of possible edges provided. For example, of the expected 27  $H^\alpha$  edges possible in VACGRX, only 13 were present, along with 103 noise ones. The situation was similar with the  $H^N$  edges.

The global consistency requirement prevents any particular solution from having too many incorrect edges. Table 2 characterizes the average number of incorrect edges over the different test cases. There is a strong correlation between the percentage of wrong edges out and measures of the dataset difficulty. For example, when average out-degree is high, then, as shown by our theoretical analysis, the algorithm must work harder to find the optimal solution. As a practical consequence, it finds more wrong solutions during the allotted time. Substantial numbers of extras and missings had the expected impact on the quality of solutions. For example, when the four poorly connected vertices for residues 54–58 in HUGRX  $\alpha$ -helix were removed (in fact, JIGSAW did not identify them), the performance improved to a mean of 2.8 wrong edges. In general, the impact of extras and missings is largely localized, as Fig. 5 demonstrates.

While this paper concentrates on connectivity information, as proof-of-concept, we explored the ability of MAPPER (Güntert *et al.*, 2000) to complete the assignment, by using sequence information to align the identified fragments to the primary sequence (refer again to the discussion of growing and aligning in the introduction). MAPPER takes as input short fragments of sequentially connected residues, along with  $C^\alpha$  and  $C^\beta$  chemical shifts (indicative of amino acid type) or other data about amino acid type. MAPPER first considers each fragment individually and lists all acceptable mappings and their scores (how well each spin system matches the expectation for the mapped residue). MAPPER can, for each fragment, give no acceptable mappings, a unique mapping, or multiple scored mappings. Given the ambiguity in amino acid type information, MAPPER typically gives numerous alignments for short fragments, but the increasing number of amino acid type constraints that must be satisfied by longer fragments drastically reduce the number of consistent mappings. Thus we expect *longer and correct* fragments to have unique mappings and incorrect fragments to have none or poor-scoring mappings. Based on this reasoning, along with our claim that we expect the sequential covers in the ensemble to contain long chunks of correct fragments, we apply MAPPER to each sufficiently long fragment in the ensemble, and we keep those fragments that have a unique mapping. We expect the union of the long, mapped fragments from a solution to provide a reasonable global assignment.

We applied this basic heuristic to the output ensembles from each backbone dataset (we did not attempt to apply MAPPER to the particularly weak amino acid type information provided by JIGSAW from  $^{15}\text{N}$ -TOCSY data). Table 2 shows that the combined approach—growing ambiguous fragments with our algorithm and mapping/filtering them with MAPPER—effectively assigns a large number of residues with a low error rate. There were no false negatives (i.e., MAPPER correctly aligned correct fragments). The false positive rate is reduced by at least half compared to growing alone. It is still possible to map incorrect edges to the primary sequence, particularly when an extra spin system with several missing peaks serves as a “wild card” replacing the correct spin system or a missing spin system. The average number of residues correctly mapped was reasonably large, more than 50% of the alignable length (actual length minus number of missing, including Pro) except for NS1 (42%) and FGF (32%). Results in Table 2 are obtained with a minimum fragment length of 6; when that threshold is raised to 8, yielding even more constraint, the error rate drops to 0. Naturally, the average number of assigned residues drops, too, but the experimentalist might consider this a trade-off worth making. This approach would lend itself naturally to further heuristics, including iterative growing/aligning combining information across the ensemble, considering global alignments or ambiguous alignments, etc. However, our focus here is on the study of the information content available from sequential connectivities, and we believe this simple heuristic provides sufficient evidence to justify the analysis of connectivity information separate from mapping information.

## 5. CONCLUSION AND FUTURE WORK

In this paper, we developed a novel random-graph theoretic framework for algorithmic analysis of NMR sequential assignment. We then gave a simple and natural randomized algorithm for finding an optimum sequential cover and presented a probabilistic analysis of the algorithm under our random graph framework. Our analysis shows that our algorithm can tolerate a relatively large ambiguity while still giving polynomial time expected performance. To study the algorithm’s performance in practice, we tested it on experimental data sets from a variety of proteins and experimental set-ups. The algorithm was able to overcome significant noise and local ambiguity and consistently identify significant sequential fragments.

These preliminary results are encouraging, and our next goal, for which we already conducted preliminary tests, is to more tightly integrate the connectivity phase (solved by our algorithm) and the aligning phase (solved by MAPPER, for example). In this process, we also plan to incorporate rigorous statistical scoring techniques, in order to support inference about the results. Finally, we plan to extend the search to higher-order connectivities, as in the  $\beta$ -sheet patterns utilized by JIGSAW.

## ACKNOWLEDGMENTS

This work benefited greatly from discussions with Olga Vitek, Department of Statistics, Purdue University, on uncertainty in NMR data, as well as her assistance in compiling and properly scoring the input NMR interaction graphs. We would like to thank Drs. Gaetano Montelione and Hunter Moseley of the Center for Advanced Biotechnology and Medicine, Rutgers University, for providing access to peak list files and the AutoPeak and AutoAssign programs, and Dr. John H. Bushweller, University of Virginia, for providing the NOESY datasets. This work was supported in part by a grant to CBK from the National Science Foundation (IIS-0237654).

## REFERENCES

- Alon, N., and Spencer, J. 2000. *The Probabilistic Method*, John Wiley, New York.
- Angluin, D., and Valiant, L. 1979. Fast probabilistic algorithms for Hamiltonian circuits and matchings. *J. Comput. System Sci.* 18, 155–193.
- Atreya, H.S., Sahu, S.C., Chary, K.V.R., and Govil, G. 2000. A tracked approach for automated NMR assignments in proteins (TATAPRO). *J. Biomol. NMR* 17, 125–136.
- Bailey-Kellogg, C., Widge, A., III, J.J.K., Berardi, M.J., Bushweller, J.H., and Donald, B.R. 2000. The NOESY jigsaw: Automated protein secondary structure and main-chain assignment from sparse, unassigned NMR data. *J. Comp. Biol.*, 7, 537–558. Conference version: *Proc. RECOMB 2000*, 33–44.
- Bartels, C., Güntert, P., Billeter, M., and Wüthrich, K. 1997. GARANT—A general algorithm for resonance assignment of multidimensional nuclear magnetic resonance spectra. *J. Comp. Chem.* 18(1), 139–149.
- Bollobas, B. 2001. *Random Graphs*, Cambridge University Press, UK.
- Broder, A., Frieze, A., and Shamir, E. 1994. Finding hidden Hamilton cycles. *Random Structures and Algorithms* 5, 395–410.
- Buchler, N.E.G., Zuiderweg, E.P.R., Wang, H., and Goldstein, R.A. 1997. Protein heteronuclear NMR assignments using mean-field simulated annealing. *J. Mol. Resonance* 125, 34–42.
- Cavanagh, J., Fairbrother, W.J., Palmer III, A.G., and Skelton, N.J. 1996. *Protein NMR Spectroscopy*, Academic Press, New York.
- Erdmann, M., and Rule, G. 2002. Rapid protein structure detection and assignment using residual dipolar couplings. Technical Report CMU-CS-02-195, Computer Science Department, Carnegie Mellon University, Pittsburgh, PA.
- Grimmett, G., and Stirzaker, D. 1992. *Probability and Random Processes*, 2nd ed., Oxford University Press, UK.
- Güntert, P., Saltzmann, M., Braun, D., and Wüthrich, K. 2000. Sequence-specific NMR assignment of proteins by global fragment mapping with program Mapper. *J. Biomol. NMR* 17, 129–137.
- Hitchens, T.K., Lukin, J.A., Zhan, Y., McCallum, S.A., and Rule, G.S. 2003. MONTE: An automated Monte Carlo based approach to nuclear magnetic resonance assignment of proteins. *J. Biomol. NMR* 25, 1–9.
- Langmead, C., and Donald, B. 2004. An expectation/maximization nuclear vector replacement algorithm for automated NMR resonance assignments. *J. Biomol. NMR* 29(2), 111–138. Conference version: *Proc. RECOMB 2003*, 176–187.
- Leutner, M., Gschwind, R.M., Liermann, J., Schwarz, C., Gemmecker, G., and Kessler, H. 1998. Automated backbone assignment of labeled proteins using the threshold accepting algorithm. *J. Biomol. NMR* 11, 31–43.
- Li, K.-B., and Sanctuary, B.C. 1997. Automated resonance assignment of proteins using heteronuclear 3D NMR. 1. Backbone spin systems extraction and creation of polypeptides. *J. Chem. Info. Comp. Sci.* 37, 359–366.
- Lin, G., Xu, D., Chen, Z.-Z., Jiang, T., and Xu, Y. 2002. A branch-and-bound algorithm for assignment of protein backbone NMR peaks. *First IEEE Bioinformatics Conference*, 165–174.
- Lukin, J.A., Gove, A.P., Talukdar, S.N., and Ho, C. 1997. Automated probabilistic method for assigning backbone resonances of ( $^{13}\text{C}$ ,  $^{15}\text{N}$ )-labeled proteins. *J. Biomol. NMR* 9, 151–166.
- Montelione, G.T., Zheng, D., Huang, Y.J., Gunsalus, K., and Szyperski, T. 2000. Protein NMR spectroscopy in structural genomics. *Nature America*, Suppl.

- Moseley, H.N.B., and Montelione, G.T. 1999. Automated analysis of NMR assignments and structures for proteins. *Curr. Opin. Struct. Biol.* 9, 635–642.
- Neal, S., Nip, A., Zhang, H., and Wishart, D. 2003. Rapid and accurate calculation of protein <sup>1</sup>h, <sup>13</sup>c and <sup>15</sup>n chemical shifts. *J. Biomol. NMR* 26, 215–240.
- Pandurangan, G. 2005. On a simple randomized algorithm for finding a 2-factor in sparse graphs. *Information Processing Letters* 95, 321–327.
- Plesnik, J. 1979. The np-completeness of the Hamiltonian cycle problem in planar digraphs with degree bound two. *Information Processing Letters* 8(4), 199–201.
- Seavey, B.R., Farr, E.A., Westler, W.M., and Markley, J. 1991. A relational database for sequence-specific protein NMR data. *J. Biomol. NMR* 1, 217–236, [www.bmrb.wisc.edu](http://www.bmrb.wisc.edu).
- Stefano, D.D., and Wand, A. 1987. Two-dimensional <sup>1</sup>H NMR study of human ubiquitin: A main-chain directed assignment and structure analysis. *Biochemistry* 26, 7272–7281.
- Stevens, R.C., Yokoyama, S., and Wilson, I.A. 2001. Global efforts in structural genomics. *Science* 294(5540), 89–92.
- Vitek, O., Vitek, J., Craig, B., and Bailey-Kellogg, C. 2004. Model-based assignment and inference of protein backbone nuclear magnetic resonances. *Statistical Applications in Genetics and Molecular Biology* 3(1), article 6, 1–33, [www.bepress.com/sagmb/vol3/iss1/art6/](http://www.bepress.com/sagmb/vol3/iss1/art6/).
- Wüthrich, K. 1986. *NMR of Proteins and Nucleic Acids*, John Wiley, New York.
- Zimmerman, D., Kulikowski, C., Huang, Y., Feng, W., Tashiro, M.S., Shimotakahara, S., Chien, C., Powers, R., and Montelione, G.T. 1997. Automated analysis of protein NMR assignments using methods from artificial intelligence. *J. Mol. Biol.* 269, 592–610.

Address correspondence to:  
Chris Bailey-Kellogg  
Department of Computer Science  
Dartmouth College  
6211 Sudikoff Laboratory  
Hanover, NH 03755

E-mail: [cbk@cs.dartmouth.edu](mailto:cbk@cs.dartmouth.edu)