A Constraint Reasoning System for Automating Sequence-Specific Resonance Assignments from Multidimensional Protein NMR Spectra*

Diane E. Zimmerman†‡ and Casimir A. Kulikowski† and Gaetano T. Montelione‡
†Department of Computer Science and ‡Center for Advanced Biotechnology and Medicine
Rutgers University
Piscataway NJ 08854

Abstract

AUTOASSIGN is a prototype expert system designed to aid in the determination of protein structure from nuclear magnetic resonance (NMR) measurements. In this paper we focus on one of the key steps of this process, the assignment of the observed NMR signals to specific atomic nuclei in the protein; i.e. the determination of sequence-specific resonance assignments. Recently developed triple-resonance (^{1}H, ^{15}N, and ^{13}C) NMR experiments [Montelione *et al.*, 1992] have provided an important breakthrough in this field, as the resulting data are more amenable to automated analysis than data sets generated using conventional strategies [Wuethrich, 1986]. The "assignment problem" can be stated as a constraint satisfaction problem (CSP) with some added complexities. There is very little internal structure to the problem, making it difficult to apply subgoaling and problem decomposition. Moreover, the data used to generate the constraints are incomplete, non-unique, and noisy, and constraints emerge dynamically as analysis progresses. The traditional inference engine is replaced by a set of very tightly-coupled modules which enforce extensive constraint propagation, with state information distributed over the objects whose relationships are being constrained. AUTOASSIGN provides correct and nearly complete resonance assignments with both simulated and real 3D triple-resonance data for a 72 amino acid protein.

Background

The basic nuclear magnetic resonance experiment yields a one-dimensional spectrum of peaks reflecting the different frequencies at which various nuclei resonate in the presence of a magnetic field. Two-dimensional NMR is based on the analysis of *cross-peaks* which reflect the resonance frequencies of two nuclei interacting with one another. Similarly, three-dimensional NMR experiments yield crosspeaks reflecting the frequencies of three interacting nuclei. Crosspeaks are detected between atoms that interact either "through-bonds" (when nuclei are separated by 3 or fewer chemical bonds) or "through-space" (when interatomic distances are less than about 5 Å). These interactions are selectively detected by different types of NMR experiments [Wuethrich, 1986]. The newly developed NMR experimental techniques (Logan *et al*, 1992; Montelione *et al*, 1992; Lyons and Montelione, 1993a; Lyons *et al*, 1993b) which provide the three-dimensional data analyzed by AUTOASSIGN involve interactions between backbone ^{15}N atoms and groups of residue-specific protons in isotope-enriched protein samples. These are called CA-TOCSY [Lyons and Montelione, 1993a] and CO-TOCSY [Montelione *et al.*, 1992] experiments.

The process of protein structure determination by NMR involves four principal steps [Wuethrich, 1986]. In the first step, networks of protons which interact with one another through chemical bonds are identified. Each such network is called a *proton spin system*, and corresponds to a separate - but as yet unidentified - amino acid in the protein. Next, *sequence-specific assignments* for these spin systems are determined by establishing their respective positions in the polypeptide sequence. In the third step, conformational constraints are generated by correlating the *through-space* interactions detected in nuclear Overhauser effect (NOE) experiments with the resonance frequencies identified in the previous two steps. Finally, structure generation programs are used to compute three-dimensional models of the protein satisfying these conformational constraints. AI systems have been developed which perform this last step [Lichtarge *et al.*, 1987]; [Edwards *et al.*, 1992]. AUTOASSIGN is an object-oriented expert system which uses constraint reasoning to solve the second step, i.e. the sequential assignment problem.

*Support for this work was provided in part by grants from The National Institutes of Health (GM-47014) and The National Science Foundation (DIR-9019313). D.Z. was supported by a Biotechnology Training Grant from The National Institutes of Health (GM-08339)

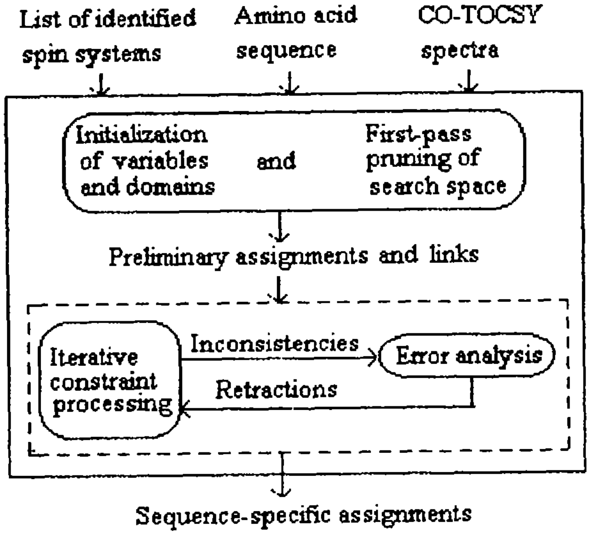
List of identified spin systems
Amino acid sequence
CO-TOCSY spectra
Initialization of variables and domains
and
First-pass pruning of search space
Preliminary assignments and links
Iterative constraint processing
Inconsistencies
Error analysis
Retractions
Sequence-specific assignments

Figure 1: Overview of AUTOASSIGN

Overview of AUTOASSIGN

The problem which AUTOASSIGN solves can be defined as follows:

Given: A (complete) list of spin systems, the amino acid sequence of the protein, and the inter-residue sequential connectivity information implied by a list of observed CO-TOCSY [Montelione *et al.*, 1992] crosspeaks;

Find: A one-to-one mapping of spin systems to sequence-specific amino acids which is most consistent with the spin system connectivity information implied by the CO-TOCSY spectra. Or equivalently, impose a complete order on the list of spin systems by establishing a complete set of adjacency relations among them.

Figure 1 gives a high-level overview of our approach. The expertise involved in performing this task interleaves simple geometric pattern matching with logical consistency reasoning using domain-specific knowledge. Each spin system specification submitted to AUTOASSIGN includes the *spin system type* (see next section), the nitrogen and amide proton resonance frequencies, and a list of aliphatic proton frequencies for that residue. A spin system can be represented as a set of points aligned parallel to the y axis in three-dimensional space. The coordinates in the z and x dimensions are defined by the magnetic resonance frequencies of the residue's backbone nitrogen and amide proton respectively. The aliphatic side-chain proton frequencies of a particular spin system appear as a "ladder" of crosspeaks occurring parallel to the y-axis at the same point in the xz-plane.

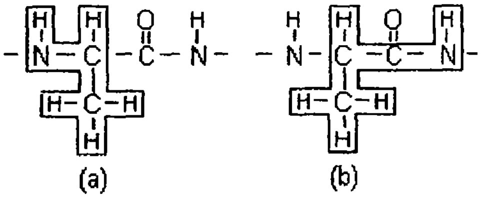

Figure 2: The CA- and CO-ladders of an Ala residue. (a) The *intra*-residue CA-ladder reflects the interactions of Ala's side-chain protons with its own backbone ^{15}N and H^N nuclei. (b) The *inter*-residue CO-ladder reflects the interactions of Ala's side-chain protons with the next residue's backbone ^{15}N and H^N nuclei.

Each crosspeak *within* a spin system reflects an *intra*-residue transfer of magnetization occurring between the three atoms which define its coordinates. These crosspeaks can be detected by CA-TOCSY experiments, and the resulting spin systems can be referred to [Lyons and Montelione, 1993a] as "CA-ladders". The connectivity information contained in the CO-TOCSY spectra reflects *inter*-residue interactions effected by a redirection of the transfer of magnetization. Specifically, the magnetization of residue i's side-chain protons is transferred to the backbone amide group atoms of residue $i+1$. The relation between "CA-ladders" and "CO-ladders" then, is analogous to a rigid translation of each CA-ladder to a new point in the xz-plane corresponding to the adjacent residue's backbone amide frequencies. This relationship is depicted schematically in Figure 2 and geometrically in Figure 3.

Complete analysis of the CO-TOCSY data involves two subtasks. Since the CO-TOCSY crosspeaks are presented as a simple list of three-dimensional coordinates, the first task is to cluster these into CO-ladders. Once this has been accomplished, CA-ladders can be matched to CO-ladders to infer adjacency relations between spin systems. But in order to describe how sequence and connectivity information can be combined to arrive at sequence-specific assignments, we first need to give a more detailed explanation of spin systems.

Amino Acid Spin Systems

Certain amino acids have spin systems which are uniquely characteristic of that residue type [Wuethrich, 1986]. Most residues bearing methyl groups fall into this category, i.e., Ala, Thr, Val, Ile, and Leu. Gly spin systems can also be uniquely identified as such, since they are the only residues bearing two α-protons (H^α) and no side-chain. The remaining 14 residue types do not have unique spin system patterns. Eight amino acids have what is called an

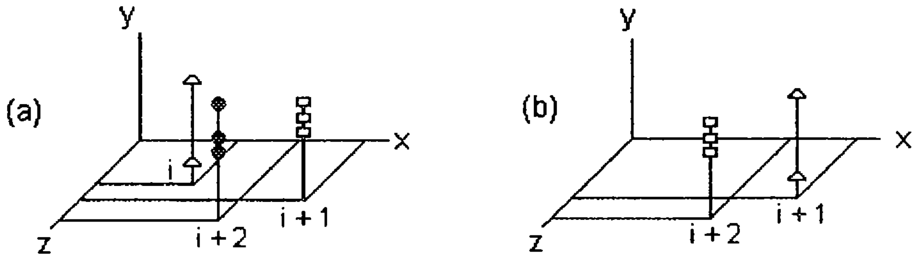

Figure 3: The CA- and CO-ladders of a sequence of 3 spin systems. (a) Each CA-ladder occurs in the xz-plane at a point defined by the spin system's own amide group. (b) The CO-ladders occur at points in the xz-plane defined by the sequence-adjacent amide group. The occurrence of only two CO-ladders reflects the fact that an N-terminal spin system has no CO-ladder associated with its amide group, while a C-terminal spin system has no "following" CO-ladder on which to project its side-chain resonance values

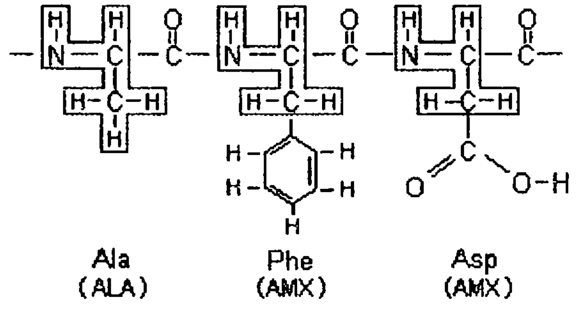

Figure 4: An ALA spin system followed by two AMXs.

AMX spin system, which is characterized by a single H^α and two H^β resonances. The AMX-type residues include Ser, Cys, Asn, Asp, Tyr, Phe, His, and Trp. A second type of "non-specific" spin system is called a LNG, and is characterized by having a single H^α, two H^βs, two H^γs, and possibly additional hydrogen resonances. LNG spin systems include Arg, Lys, Met, Pro, Glu, and Gln. Figure 4 illustrates the three spin systems associated with the tripeptide Ala-Phe-Asp, with each proton network schematized by enclosing boxes. Only those protons which are connected to other protons in the network by less than 4 bonds are included in the boxes. Neither the aromatic protons of Phe nor the hydroxyl proton of Asp are included in the spin systems associated with those residues.

In AUTOASSIGN, whenever adjacencies between spin systems can be established, the system checks the sequence to see if a unique position is defined by that segment of linked spin system types. One measure of the complexity of the problem is the "spin system degeneracy" (i.e. non-uniqueness) of the sequence. For example, in a hexamer composed of Gly-Ala-Val-Thr-Ile-Leu, each amino acid generates a unique spin system type which has only one possible position. In contrast, a hexamer composed of six LNG-type amino acids would be maximally degenerate.

The sequential assignment problem resembles in many ways the classical problem in logic known as the "Five Houses Puzzle" [Van Hentenryck, 1989]. In that problem, five individuals living on the same block each have different professions, nationalities, pets, hobbies, and so forth. We are given only partial information, such as "the Italian drinks tea" and "the Englishman lives in a red house", and from these clues, must deduce who drinks water and owns a zebra. In our case, the "individuals" are spin systems, and the order in which different-colored houses appear is specified by the amino acid sequence. The "clues" we are given are the spin system adjacencies which can be inferred by matching CA-ladders to CO-ladders. These clues however, may be unreliable and incomplete.

The spin systems themselves may have extensive overlap of their nitrogen and amide proton resonances and/or non-distinctive patterns of side-chain resonances. Further complications arise from noise and digitization errors in the spectra and incompleteness in the observed set of crosspeaks. In most cases, each pair of xz-values is uniquely defined by that residue's backbone resonance frequencies, but not always. When both the x and z values overlap, the "rungs" of the associated ladders become interleaved, and it is difficult to determine which aliphatic protons are interacting with which pair of backbone nuclei.

In summary, the sequential assignment problem is similar to many constraint satisfaction problems but with some additional challenges. In particular, many of the constraints which must be applied to arrive at a solution are not known *a priori*, but must instead be extracted as analysis progresses. The following section describes in more detail how we define sequential assignment as an instance of constraint satisfaction.

Sequential Assignment as a Constraint Satisfaction Problem (CSP)

In the CSP paradigm [Kumar, 1992], one is given a set of variables, a set of values each variable can assume (domains), and a list of constraints which further restrict how these variables can be assigned. The goal is to find a complete assignment of the variables which violates none of the constraints. AUTOASSIGN uses the following link and assignment variables. Associated with each spin system is an *N-link* variable, a *C-link* variable, and an (amino acid) *assignment* variable. The domains of the two link variables associated with a spin system initially include all of the other spin systems. Amino acids have only a (spin system) *assignment* variable; the N- and C-links are simply the two surrounding amino acids in the sequence. The domains of the assignment variables of spin systems and amino acids are complementary. For example, each Ala residue in the sequence is initially included in every ALA spin system's domain of possible assignments, and vice versa.

The goal of AUTOASSIGN is to reduce each of these domains to a single, unique value, or equivalently, to assign a unique value to each of the assignment and link variables. Traditionally, constraint satisfaction problems have been solved by applying node-, arc-, and path-consistency algorithms [Mackworth, 1977] to prune the domains before any variable assignments are made. Node- and arc-consistency algorithms can be performed in polynomial time, and in some cases have been shown to reduce exponential search problems to linear execution times [Kumar, 1992]. These algorithms assume however, that all of the constraints are reliable and known *a priori*. But in our case, constraints can only be inferred incrementally as analysis progresses. We do not have space here to enumerate all the ways in which constraints can be defined and propagated, but a few examples are given to indicate how these emerge.

The domains of a spin system's link variables can be constrained as follows. A first condition is that in order for one spin system to be followed by another, there must be a "reasonable" match between the first spin system's side-chain resonance frequencies and the CO-ladder associated with the amide resonances of the second spin system. When this condition is not satisfied, the two spin systems can be removed from each others' C- and N-link domains respectively. A second condition requires that all possible remaining link values be consistent with the order in which residue-types occur in the sequence, as well as with the currently established sequence-specific assignments. For example, if the CO-TOCSY data suggests that some ALA-type spin system is followed by some LEU-type spin system, but there is no instance of Ala-Leu in the sequence, then the respective link domains should be pruned of this inconsistency. Alternatively, if there is an instance of Ala-Leu in the sequence but both of these residues have already been assigned to other spin systems, then the link domains of all unassigned ALA and LEU spin systems can be pruned. Similar pruning is possible when a spin system has been assigned to a particular amino acid but one or both of its link variables remains unassigned.

The domains of the assignment variables can also be constrained according to the links which have been established. For example, once two or more spin systems have become linked to form a sequential segment, the domains of their amino acid assignment variables should be pruned for mutual consistency. The spin system assignment domains of amino acids can also be pruned as possible links between spin systems are eliminated. Specifically, if a spin system currently included in an amino acid's domain of possible assignments no longer has any way of establishing either an N-link or a C-link which can support this hypothesis, then AUTOASSIGN removes that spin system from the residue's domain.

All of these methods for domain-pruning correspond in principle to arc- and path-consistency arguments. In effect, each of these is a mechanism for "ruling-out" inconsistent assignments. But very little information of this type is available initially, and it is only by alternating between ruling-out and ruling-in mechanisms that the system can progress to a complete assignment. In particular, this incremental approach allows the system to base its earliest decisions on only the strongest empirical evidence, and to subsequently use these most reliable inferences to filter out many of the errors that might otherwise occur. Some of the mechanisms by which assignments are ruled-in are discussed in the following section which describes how these embedded variables are represented.

Representation and Implementation

Table 1 summarizes the data structures used to represent the relationships between spin systems, sequence-specific amino acids, and CO-ladders. Each spin system is represented by an object whose attributes specify the spin system's internal resonance frequencies, the three embedded variables described in the preceding section, plus three additional slots used to track the domains of these variables. Tracking the domains of the link variables is a bit more complicated than implied by our previous discussion. Each spin system for which a CO-ladder could be identified maintains a pointer to that ladder (under the attribute CO-ladder). The domain of that spin system's N-link variable is then defined as the other spin systems whose side-chain resonances matched those in the CO-ladder. These matches are stored with the CO-ladder however, so are accessed indirectly. Similarly, the domain of the

C-link variable is defined to be those other spin systems whose CO-ladders matched the side-chain values of the spin system under consideration. These CO-ladders are listed in the Matches slot. Access to the actual domain values is again indirect, as the spin systems associated with these ladders are stored with the ladders themselves.

Each amino acid in the sequence is also represented by an object whose attributes include the N- and C-links, the spin system assignment variable, and its associated domain. The attributes of a CO-ladder specify the spin system associated with that ladder in the XZ-plane, the CO-TOCSY peaks included in the ladder, a list of other spin systems whose side-chain resonances match the ladder, and a heuristic score reflecting how reliable the ladder is. This score is a linear function of the number of peaks included which occur on other ladders, the number of overlapping ladders, and the scatter of x and z values in the included peaks.

Defining what constitutes a "good" match between a spin system's y-values and the peaks (rungs) included in a ladder is complicated by noise, degeneracy, and incompleteness. We have found it useful to apply a "measure of goodness" to matches as well as to CO-ladders. The link-score between two spin systems is then taken as the product of these two scores. Match scores are computed as:

$$py \times pr \times \left(\#\text{matches} - \frac{err}{tol}\right)$$

py and *pr* represent the percentage of matched y-values for the spin sytem and the percentage of matched rungs for the ladder. The third term represents the total number of matches occurring between the two objects, less the sum of errors (*err*) which occurred in matching, normalized by the match tolerance (*tol*) used.

Ruling in Variable Assignments

In this section we describe several mechanisms for reliably ruling in certain variable assignments. In order to set a "high-link", a spin system's best match to a CO-ladder must also be that ladder's best match to any spin system, and the link-score between the two spin systems must surpass all other link-scores by a significant threshold. A second way of setting links roughly corresponds to what has been referred to as k-consistency algorithms [Cooper, 1989]. Using a branching factor B, all possible paths from the N- and C-termini of all previously assigned segments are generated. When two such paths moving in opposite directions cross each other and the only way to reach certain unlinked spin systems is via these paths, the implied "mutually-exclusive" links are committed to for these spin systems. Because some of our implied constraints may be unreliable, we cannot use this principle in general to prune the domains. But when a domain is forced by this mechanism to converge to a single value, we have found that it is always the correct one.

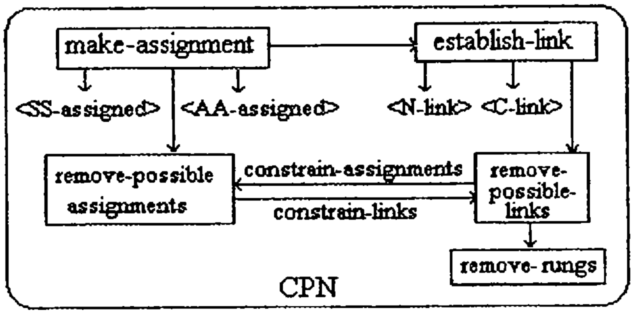

Figure 5: The Constraint Propagation Network.

Similar to the manner in which possible paths can be used to impose "sequence-consistency" on the unassigned links of spin systems, it is possible to impose "match-consistency" on unassigned amino acids. Working from the sequence, unique triples whose central residues have not yet been assigned can be identified. Then, if a single spin system has the requisite links or matches consistent with this position, and very high link-scores in both directions, the assignment to the central amino acid is made. A second way of using unique triples is to allow the domains of the surrounding amino acids to constrain the spin system of assignment of the central residue. Once all such assignments have been evaluated, additional assignments can be made by a process of elimination.

The Constraint Propagation Network (CPN)

With each assignment of a variable, the entailed constraints are immediately propagated to all other variables to ensure that global consistency is maintained and that maximal pruning occurs with each decision. There are two modules which are used to rule in variable assignments: make-assignment and establish-link. Both of these are an integral part of the constraint propagation network (Figure 5). Each of the "rule-in modules" in turn triggers the "rule- out" modules which remove possible links (matches between spin systems and CO-ladders) and possible assignments as described. In addition, the pruning of matches offers the opportunity to re-examine peaks which may have been inappropriately included on overlapped CO-ladders. If a CO-ladder currently includes some peaks which no longer have matches to any spin system's side-chain values, these peaks (rungs) can now be eliminated. This has the effect of promoting both match scores and ladder scores, thus allowing new constraints to emerge.

Control Flow

Figure 6 gives an abstract representation of the overall control flow and interaction of AUTOASSIGN's four

Object-Type	Attributes	Description
Spin-system	X	The H^N frequency of this spin system
	Y-values	The side-chain proton frequencies of this spin system
	Z	The backbone ^{15}N frequency of this spin system
	N-link	A spin system assigned to be this spin system's predecessor in the sequence
	C-link	A spin system assigned to be this spin system's successor in the sequence
	AA-assigned	A sequence-specific amino acid assigned to this spin system
	AA-domain	A list of possible amino acid assignments for this spin system
	CO-ladder	A CO-ladder uniquely associated with this spin system's x and z values
	Matches	A list of CO-ladders whose rungs matched this spin system's y-values
Amino Acid	N-link	The preceding amino acid in the sequence
	C-link	The following amino acid in the sequence
	SS-domain	A list of possible spin system assignments for this amino acid
	SS-assigned	A spin system assigned to this amino acid
CO-ladder	Spin-system	A spin system uniquely associated with this ladder's x,z-values
	CO-peaks	The CO-peaks which define this ladder's rungs
	Matches	A list of spin systems whose y-values matched this ladder's peaks
	Lscore	A measure of intra-ladder scatter (noise) and inter-ladder separation (overlap)

Table 1: Objects Used in the Representation

main modules with the CPN. The initialization routines create the objects described in Table 1 and initialize the domains of their embedded variables. The first step in STARTUP, which is invoked immediately after initialization, is to filter the domains of the assignment variables based on spin system types and the observed resonances of the spin systems. This step narrows the possible assignments of certain AMX- and LNG-type spin systems according to characteristic patterns that sometimes occur, and corresponds to node-consistency. The next goal is to assign the link attributes of as many spin systems as possible based on the highest link-scores. The net effect of processing inside STARTUP is that the most reliable links and assignments are established, while the number of remaining possible assignments and links is dramatically reduced (see results in Table 2).

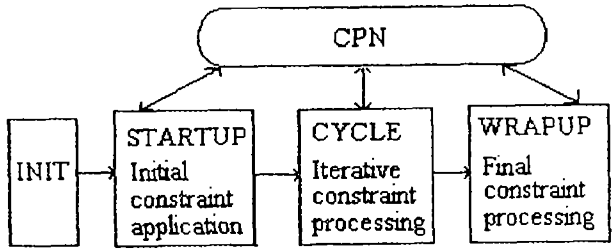
CPN
INIT
STARTUP
Initial constraint application
CYCLE
Iterative constraint processing
WRAPUP
Final constraint processing

Figure 6: The Overall Control Flow of Execution

CYCLE alternates between establishing definite links by discovering convergent paths and making definite assignments by analyzing unique triples. Paths are generated with a fixed branching factor until no further links can be established. At that point, unique triples are analyzed, and if any assignments are made, the algorithm returns to trying to establish links. When no further progress can be made the module WRAPUP is executed. At this point, all but the most degenerate ladders have been pulled apart by gradually removing the unmatched rungs as possible links (matches) are eliminated. The most problematic cases occur when a spin system motif is repeated two or more times in the sequence, or when spin systems have severely overlapped backbone resonance frequencies. In order to pull these ladders apart, the peaks included in each ladder are now redefined using a much smaller radius centered about the spin system's x and z values. We also tested the use of these tighter match criteria in the x and z dimensions during the initialization of CO-ladders, but found that many of the true cross-peaks were then excluded from their appropriate ladders. WRAPUP also uses the current assignments of amino acids to constrain those that remain unassigned.

Backtracking

Situations arise where a request is sent to the CPN to establish an assignment which conflicts with the current matches and/or established links. For example, if links have been established between two spin systems and one of these is subsequently given a sequence-specific assignment, then the next round of constraint processing will try to assign the other spin system to the adjacent position in the sequence. But before making requested assignments, the CPN checks to see if any contradictions will result - e.g., a newly assigned spin system becomes adjacent to some previously assigned spin system with which it has no way of establishing a link. When a contradiction is discovered, the assignment is not made. Instead, the system evaluates the competing assignments and links and attempts to determine where the error has occurred. In most cases, the correct decision can be made based on the simple scoring mechanisms used to evaluate matches, links, and ladders. Once the error has been identified, the offending link or assignment is retracted along with any other actions which may have propagated from it, and AUTOASSIGN resumes execution.

Although not shown in Table 1, each of the objects also keeps track of a few local "history" variables to facilitate backtracking. Amino acids and spin systems maintain a list of possible assignments which have been eliminated. Similarly, CO-ladders and spin systems maintain a list of matches which have been ruled-out. When it becomes necessary to restore a previous state, these lists are scanned, and any values which do not conflict with currently assigned variables are restored. This actually works fairly well, but provides no way of ensuring that *only all* of the constraints which may have been propagated erroneously are retracted. An important issue is whether or not the additional storage and complexity of more detailed bookkeeping would be worth the trade-off in performance. The current mechanisms form a preliminary foundation for future development of efficient backtracking schemes.

Performance Results

Table 2 shows the results obtaind for a 72 amino acid domain derived from the Staphylococcal Protein A using the actual data obtained from a 3D CO-TOCSY triple resonance experiment [Lyons *et al.*, 1993b] and a simulated data set for the same protein molecule. The four columns show respectively the number of remaining possible assignments, the number of confirmed assignments, the number of remaining possible links, and the number of these which have been established. Each row correpsonds to one of the stages of execution, i.e., INIT, STARTUP, CYCLE and WRAPUP. For the simulated data (results shown in parentheses), all of the expected crosspeaks were computed from the spin system list and then analyzed with match tolerances of ±0.02 ppm, ±0.08 ppm, and ±0.35 ppm in the x, y, and z dimensions respectively. This domain of Protein A is composed of three helices and a 15 residue leader sequence which has a random coil conformation, and many of its backbone amide frequencies are very similar. Accordingly, the degeneracy of x and z values is fairly pronounced, and the initial CO-ladders are overlapped with one another. This problem is overcome by the system as well-separated regions are assigned first, and the subsequent elimination of inconsistent matches permits substantial pruning of the degenerate ladders. For these simulated data, the system makes over 83% of the assignments in STARTUP before resorting to iterative constraint processing. One pass through CYCLE completes the assignments without any errors (Table 2).

AA Domain	Assigned	Link Domain	Linked
1378 (1378)	0 (0)	656 (668)	0 (0)
536 (28)	22 (60)	294 (85)	18 (51)
105 (0)	55 (72)	103 (71)	46 (71)
2 (0)	70 (72)	66 (71)	66 (71)

Table 2: Results on Real and (Simulated) Data for Protein A

Having demonstrated the reliability of the system using simulated data, we next carried out automated analysis of the real 3D CO-TOCSY data. Compared to the simulated data, the real data contains only 65% of the expected crosspeaks. The spin system list is also incomplete as only 71 spin systems were identified for 72 residues in the sequence. The initial x, y, and z match tolerances estimated from clustering of the crosspeak frequencies were the same as those used in the analysis of simulated data. With these real data, early constraint processing (STARTUP) yields a dramatic reduction in the number of possible assignments and matches, but only about 30% of the definite assignments. Iterative constraint processing inside CYCLE results in assignments for all but 16 of the 71 spin systems. Two instances of backtracking occur, and in both cases consistent assignments are found once the inconsistent assignments have been retracted. The final stage of processing (WRAPUP) yields an additional 15 assignments. The remaining two residues in the sequence, Met(-14) and Gln(-5), are both LNGs, but neither is assigned to the single remaining LNG spin system, as there is nothing in the CO-TOCSY data to support a decision either way.

In evaluating the system's performance, we observed that no matches were incorrectly eliminated, but in one case, His(-4), the correct assignment has been deleted from the appropriate spin system's domain of possible assignments. This is due to the fact that the expected crosspeaks to both the preceding and succeeding spin systems in the sequence were not detected in the NMR experiment. However, in the final stages of WRAPUP, the correct assignment is made by a process of elimination, as there is only one remaining AMX-type spin system and His(-4) is the only remaining unassigned AMX-type residue.

Related Work

Automated Sequential Assignment

Because through-space interactions often occur between the protons on adjacent residues, nuclear Overhauser effect (NOE) data used to measure through-space distances has also been used routinely in manual analysis to infer connectivity information [Wuethrich, 1986]. Attempts to automate sequential assignment using 2-D NOE data have had limited success, largely because the solution is greatly underconstrained by the connectivity information. The problem with inferring spin system adjacencies from

NOE data is twofold: (1) through-space interactions also occur between *non*-adjacent residues, and (2) the occurrence of "through-space adjacent" interactions is conformation dependent.

In one case [Billeter *et al.*, 1988], the system was tested on both real and simulated data sets and could typically make only 30-50% of the complete assignments. Similar results were achieved in a semi-automated implementation reported by Eads and Kuntz (1989). The system first establishes the most reliable links on the basis of strong supporting evidence in the data, and then writes a list of potential "next" and "previous" spin system relations to a separate file. This information is then used by the expert to make manual assignments. Using bovine pancreatic trypsin inhibitor (BPTI) as a test case, manual analysis of the established links led to 21 unambiguous assignments (41% of the sequence), and logical consistency arguments similar to those used by AUTOASSIGN were then applied to make an additional 23 assignments by a process of elimination. It is interesting to note that the percentage of assignments which can be most reliably established by AUTOASSIGN for Protein A fall within the range reported by Billeter et al (i.e. 30% in STARTUP), and that the percentage of additional assignments obtained by extensive constraint propagation qualitatively agrees with the results of [Eads and Kuntz, 1989]. Although several authors (Montelione and Wagner, 1990; Ikura *et al*, 1990; Logan *et al*, 1992; Montelione *et al*, 1992; Lyons and Montelione, 1993a; Lyons *et al*, 1993b) have noted that the new data sets being generated by various multiple-resonance multi-dimensional NMR experiments (such as the CA-TOCSY and CO-TOCSY experiments) are more amenable to automated analysis, no fully automated systems have yet been reported. A recently published semi-automated implementation called ALFA [Bernstein *et al.*, 1993] uses energy minimization techniques to make sequential assignments. The connectivity information is taken from three-dimensional NOESY data. In ALFA, all spin systems are initially given unique but arbitrary assignments to amino acids in the sequence. The system then proceeds to examine arbitrarily selected pairs of segments of random lengths varying from two to seven residues. The current assignments of these residues are exchanged wherever such a modification will lead to a reduction in the total "energy". Terms in the energy equation include a measure of spin system-residue type compatibility and detected sequential crosspeaks to surrounding spin systems. This process is repeated until no further minimization is possible. For the single test case reported, the system made 83% correct assignments using the NOE data.

Dynamic Constraint Satisfaction

In most constraint satisfaction problems, the variables, domains, and constraints are included in the problem specification, and the solution space can be dramatically reduced before search is initiated by applying various consistency algorithms. In AUTOASSIGN, only the variables, their domains, and *unary* constraints are available initially. In our "constraint graph", we have all of the nodes (variables) but none of the edges (binary or n-ary constraints). Instead, these emerge as dependency relations among amino acids, spin systems, and CO-ladders as the links and assignments are established. A second advantage in knowing all the constraints *a priori* is that the order in which decisions are made can be guided by "least-commitment" or "most-constrained-first" strategies. One possibility would be to do a preliminary dependency analysis which might be able to anticipate the more important choice points, e.g., which links or assignments will trigger the maximum number of additional assignments.

Synthesis tasks (e.g. configuration, design, etc.) have also been characterized as dynamic constraint satisfaction problems [Mittal and Falkenhainer, 1990]. But in these tasks, not even the variables themselves are predetermined. Thus the focus is often on establishing a means of coupling the creation of variables to the propagation of the constraints they entail. In our own experience with the ACONS configuration system [Hagerty *et al.*, 1991], we addressed this issue by distributing the constraint information over the objects being constrained rather than storing it in a central location such as a goal stack. Although the creation of variables is not an issue for AUTOASSIGN, we have found the distributed representation of object-specific state information to be an effective means of reducing search. A second principle applied in both ACONS and AUTOASSIGN is the thorough propagation of constraints as each commitment is made.

Conclusions

AUTOASSIGN demonstrates that generic constraint-satisfaction methods can be successfully adapted to a complex real-world problem. The use of these methods evolved naturally in the course of trying to model the type of reasoning the expert brings to bear on the sequential assignment problem. It is difficult to compare the performance results of systems which perform the same task using different inputs, particularly when each system has only one real data set to work with. Further testing and development on additional proteins is needed to better assess the system's strengths and weaknesses and to enhance its robustness and flexibility. Qualitatively however, AUTOASSIGN appears to outperform other systems designed to perform the same task which have been reported to date.

The sequential assignment problem is a special type of dynamic constraint satisfaction problem, where the variables and domains are given but the constraints must be discovered in the process of analyzing the data. This description fits many data interpretation problems, and it would be interesting to explore how well the approach we have taken maps to other domains.

References

Bernstein, R.; Cieslar, C.; Ross, A.; Oschkinat, H.; Freund, J.; and Holak, T. A. 1993. Computer-assisted assignment of multidimensional NMR spectra of proteins: Application to 3D NOESY-HMQC and TOCSY-HMQC spectra. *Journal of Biomolecular NMR* 3:245–251.

Billeter, M.; Basus, V. J.; and Kuntz, I. D. 1988. A program for semi-automatic sequential resonance assignments in protein ^{1}H nuclear magnetic resonance spectra. *Journal of Magnetic Resonance* 76:400–415.

Cooper, M. C. 1989. An optimal k-consistency algorithm. *Artificial Intelligence* 41:89–95.

Eads, C. D. and Kuntz, I. D. 1989. Programs for computer-assisted sequential assignment of proteins. *Journal of Magnetic Resonance* 82:467–482.

Edwards *et al.*, 1992. In Hunter, L., editor 1992, *AI and Molecular Biology.* AAAI/MIT Press.

Hagerty, C. G.; Zimmerman, D. E.; and Kulikowski, C. A. 1991. Distributing constraint satisfaction for configuration tasks. In *Proceedings IMACS '91: 13th World Congress on Computation and Applied Mathematics.* Dublin. 987–988.

Ikura, M.; Kay, L. E.; and Bax, A. 1990. A novel approach for sequential assignment of ^{1}H, ^{13}C, and ^{15}N spectra of proteins: Heteronuclear triple-resonance three-dimensional nmr spectroscopy. Application to calmodulin. *Biochemistry* 29(19):4659–4667.

Kumar, V. 1992. Algorithms for constraint-satisfaction: A survey. *AI Magazine.*

Lichtarge, O.; Cornelius, C.W.; Buchanan, B. G.; and Jardetzky, O. 1987. Validation of the first step of the heuristic refinement method for the derivation of solution structure of proteins from nmr data. *PROTEINS: Structure, Function, and Genetics* 2:340–358.

Logan, T. M.; Olejniczak, E. T.; Xu, R. X.; and Fesik, S. W. 1992. Side chain and backbone assignments in isotopically labeled proteins from two heteronuclear triple resonance experiments. *FEBS* 314(3):413–418.

Lyons, B. A. and Montelione, G. T. 1993a. An HC-CNH triple resonance experiment using carbon-13 isotropic mixing for correlating backbone amide and sidechain aliphatic resonances in isotopically-enriched proteins. *Journal of Magnetic Resonance* 101B:206–209.

Lyons, B. A.; Tashiro, M.; Cedergren, L.; Nilsson, B.; and Montelione, G. T. 1993b. A novel strategy for determining sequence-specific nuclear magnetic resonance assignments in isotopically-enriched proteins. *Biochemistry.* in press.

Mackworth, A. K. 1977. Consistency in networks of relations. *Artificial Intelligence* 8(1):99–118.

Mittal, S. and Falkenhainer, B. 1990. Dynamic constraint satisfaction problems. In *Proceedings of the Eighth National Conference on Artificial Intelligence.* Morgan Kaufman.

Montelione, G. T. and Wagner, G. 1990. Conformation-independent sequential NMR connections in isotope-enriched polypeptides by H-C-N triple-resonance experiments. *Journal of Magnetic Resonance* 87:183–188.

Montelione, G. T.; Lyons, B. A.; Emerson, S. D.; and Tashiro, M. 1992. An efficient triple resonance experiment using carbon-13 isotropic mixing for determining sequence-specific resonance assignments of isotopically-enriched proteins. *Journal of the American Chemical Society* 114:10974–10975.

Van Hentenryck, P. 1989. *Constraint Satisfaction in Logic Programming.* MIT Press, Cambridge.

Wuethrich, K. 1986. *NMR of Proteins and Nucleic Acids.* Wiley, New York.