January 1987

Report No. STAN-CS-87-1142
Also numbered KSL-87-05

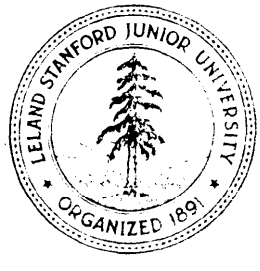
A Heuristic 'Refinement for Spacial Constraint Satisfaction Problems

by

James E. Brinkley, Bruce G. Buchanan, Russ B. Altman,
Bruce S. Duncan, Craig W. Cornelius

Department of Computer Science

Stanford University
Stanford, C A 94305



Knowledge Systems Laboratory
Report No. 87-05

January 1987

A Heuristic Refinement Method for Spatial Constraint Satisfaction Problems

by

James F. Brinkley, Bruce G. Buchanan, Russ B. Altman,
Bruce S. Duncan, Craig W. Cornelius

KNOWLEDGE SYSTEMS LABORATORY
Computer Science Department
Stanford University
Stanford, California 94305

· A Heuristic Refinement Method for Spatial Constraint Satisfaction Problems

James F. Brinkley, Bruce G. Buchanan, Russ B. Altman,
Bruce S. Duncan, Craig W. Cornelius

1 January 1987

PROTEAN Project'
Knowledge Systems Laboratory
Department of Computer Science
701 Welch Road
Palo Alto, California 94303
415-723-8379, BRINKLEY@SUMEX-AIM.STANFORD.EDU

Full paper, Track: Science, Topic: Reasoning, Search

Additional Keywords: spatial reasoning, space planning, arrangement assembly,
multi-dimensional constraint satisfaction

Abstract

The problem of arranging a set of physical objects according to a set of constraints is formulated as a geometric constraint satisfaction problem (GCSP), in which the variables are the objects, the possible locations of the objects are the possible values for the variables, and the constraints are geometric constraints between the objects. A GCSP is a type of multi-dimensional constraint satisfaction problem in which the number of objects and/or the number of possible locations per object is too large to permit direct solution by backtrack search. A method is described for reducing these numbers by refinement along two dimensions. The number of objects is reduced by refinement of the structure, representing a group of objects as a single abstract object before considering each object individually. The abstraction used depends on domain specific knowledge. The number of locations per object is reduced by applying node and arc consistency algorithms to refine the accessible volume of each object. Heuristics are employed to control the order of operations (and hence to affect the efficiency of search) but not to change the correctness in the sense that no solutions that would be found by backtrack search are eliminated. Application of the method to the problem of protein structure determination is described.

---

[1]The PROTEAN project consists of members of the Stanford Magnetic Resonance Laboratory and the Stanford Knowledge Systems Laboratory (in alphabetical order): Russ Altman, James Brinkley, John Brugge Bruce Buchanan, Craig Cornelius, Bruce Duncan, Alan Garvey, Barbara Hayes-Roth, Oleg Jardetzky, Olivier Lichtarge, Marcella Madrid. This work was funded in part by NIH grants RR-00785, RR-0071 1, and GM07365, NSF grant DMB84-2038, NASA Grant NCC 2-274, Boeing Grant W266875, DARPA contract N00039-83-C-0136.

1 Introduction

Current generation expert systems are able to perform reasonably well in heuristic classification problems such as medical diagnosis, fault diagnosis, etc. [Clancey 85]. A different type of problem is constraint satisfaction. In general a constraint satisfaction problem (CSP) involves the assignment of values to a set of variables subject to constraints [Dechter 85]. Example problems that have been formulated in this way include N-queens and line labelling in computer vision [Waltz 75].

The computational complexity of these problems depends both on the number of variables and the number of possible values per variable. If these numbers are small enough then direct solution by backtrack search, possibly involving learning [Dechter 86], is feasible. (Backtrack search is essentially a method for enumerating all possible solutions). If the numbers are too large then filtering algorithms such as node, arc and path consistency may be applied prior to backtrack search in order to reduce the number of possible values per object [Mackworth 77).

In general these methods have been applied to problems in which the values for the variables are one-dimensional scalars, such as the row number for a queen in the N-queens problem, or the label type for the line labelling problem. However, there are many problems which may be formulated as *multi-dimensional* constraint satisfaction problems, in which the possible values for the variables are vectors. For example, each variable may be a frame or object-class, and the range of possible values for the variable may be mapped to the possible attribute value pairs defining the set of allowed object-instances within the object-class. The number of dimensions for the value vector is the number of attributes for the object. A single solution to the multi-dimensional constraint satisfaction problem is then a list of one object-instance per object-class, such that the constraints between object-classes are satisfied, while the complete solution is the list of all such individual solutions. Since an attribute of an object may itself be an object, the possible values of the attribute may themselves be vectors representing object-instances of the object-class defining the attribute. The number of possible values for the top level objects is therefore the cross product of all possible values for the attributes, each of which may in turn be the cross product of its own sub-attributes. Thus, the total number of possible values for the top level objects can become too large for direct solution by backtrack search, even after the application of network consistency algorithms.

An important type of multi-dimensional constraint satisfaction problem is the class of spatial, or geometric constraint satisfaction problems (GCSP), which may also be described as arrangement assembly problems [Hayes-RothB 86a]. A GCSP may be formulated as a CSP in which the variables are geometric objects, the values are the possible locations of those objects in space (in general six-dimensional vectors with three position components and three orientation components), and the constraints are binary or n-ary geometric constraints between the objects. In object-attribute-value terminology the geometric objects are the object-classes, the six-dimensional locations are six attributes of the objects, and the possible object-instances are the possible locations of the geometric objects in space. A solution to a GCSP is a list of one location per object such that all the constraints are simultaneously satisfiable. The complete solution is all such lists of locations. Since space is infinitely decomposable the solution must be a representative sample, at some reasonable sampling resolution compatible with the computational resources.

Examples of GCSP's include construction site planning, furniture arrangement, bathroom design [Willey 81], landscape design, robot assembly, and protein structure determination [Buchanan 85]. Temporal reasoning can also be thought of as a geometric constraint satisfaction problem, in which the objects are events and the locations are one-dimensional time intervals [Allen 83].

GCSP's may be design problems (in which the constraints are design considerations) or data interpretation problems (in which the constraints are measured data). In both cases it is often desirable to retain all solutions compatible with the data in order to study the effect of sequentially introducing or retracting constraints. For example, in designing a furniture arrangement there may be certain necessary constraints such as the requirement that the television be visible from the couch. If all solutions compatible with this constraint are found first then it may become clearer how to introduce additional, aesthetic constraints in order to

narrow the possible arrangements further. For data analysis the data may not be complete enough to determine a unique solution, but it may be desirable to determine all solutions compatible with the data before introducing more theoretical constraints to narrow the range of possibilities.

In this paper we describe a method, called heuristic refinement, for geometric constraint satisfaction problems in which a representative set of all possible solutions is desired. The method should also be extendable to other types of multi-dimensional constraint satisfaction problems. Although the search space is too large for direct backtrack search, our working hypothesis is that the number of representative and useful solutions is small enough to be enumerated. If this hypothesis is true then the goal is to find efficient means for searching the solution space which do not eliminate any arrangements that would be found in a backtrack search. The method is described in the context of a system, called PROTEAN, which is being developed for determination of three-dimensional protein structure from experimental constraints [Buchanan 85, Altman 86, Duncan 87, Hayes-RothB 86b, Brinkley 86, Lichtarge *86).*

2 Elements of the Heuristic Refinement Method

The goal of the heuristic refinement method (applied to geometric constraint satisfaction problems and embodied in the PROTEAN program) is to enumerate (at a reasonable sampling resolution) all conformations of a set of objects compatible with a set of constraints. Since the complexity of backtrack search depends primarily on the number of objects and the number of locations per object, PROTEAN first attempts to reduce these numbers in three main ways before performing backtrack search:

1. representing objects and constraints at several levels of abstraction.
2. using- constraint satisfaction algorithms to limit the number of possible conformations that must be enumerated.
3. employing heuristics to decide the order of operations to perform.

The effect of these techniques is that the objects are refined along two main dimensions: that of structure and that of accessible volume. The *structure* is first imprecisely determined at an abstract level before being refined to a more precise but computationally expensive level. The rationale is that there is no reason to immediately consider all the individual objects in a subgroup of locally highly constrained objects since the relative placement of those objects will not change radically in the final structure. If these objects are represented by a geometric solid defining the minimum enclosing volume of the vast majority of structures of this type, then the volume of space occupied by that solid determines bounds on the positions of the objects within that solid. These bounds can drastically limit the amount of search required when the solid is refined to its constituent parts. The representation of a group of objects by a single abstract object corresponds to the imposition of a structural model, which embodies certain additional theoretical constraints. The degree to which the additional constraints eliminate solutions that would be found without the imposition of the model is the degree of bias inherent in the model. The PROTEAN framework aims to allow the human user (or an expert system) to explicitly choose the degree of bias, thereby trading off efficiency for completeness.

The other main refinement dimension is *accessible volume.* All solid objects are initially considered to be located within some infinite region of space with respect to one of the solids, which is chosen to be a fixed *anchor.* The region of space occupied by each solid is called its accessible volume. The accessible volume of each solid is first reduced from all of space to the volume compatible with the fixed anchor, given the constraints between the solid *anchoree* and the anchor. The accessible volumes of each anchoree are then further reduced by incrementally applying constraints with other anchorees. Reduction of the accessible volumes in this way corresponds to the application of node and arc consistency algorithms to a geometric constraint network [Mackworth 77].

Different subparts of the object may be assembled separately in this way, each with respect to

a fixed anchor. The subassemblies may then be corn bi ned into a larger whole by incorporating constraints between the subassemblies.

Each of these operations results in reduction of the accessible volumes of the individual subparts of the object, leading to accessible volumes that are small enough so that 1) the solid level conformations can be effectively enumerated by backtrack search, or 2) the solids can be refined to their subparts. These individual operations are possible because the local constraints between individual objects, when acting together, are able to adequately constrain the overall conformation. Each local operation causes a correct reduction of the accessible volume of the relevant solids in the sense that no locations are eliminated that would be found in an exhaustive enumeration (except for locations eliminated by bias in the structural model), yet the search is much more efficient since inaccessible volumes are eliminated before the exhaustive enumeration is done.

Although each possible reduction in accessible volume is correct, there are always many possible such actions that may be taken. For example, decisions must be made as to which subparts of the object to assemble separately, which to make the global anchor, and which anchorees to consider together first. Poor decisions as to which actions to take can lead to such inefficiency that the program will not arrive at an answer within a reasonable time. Thus, the order of refinement operations requires intelligence, supplied either by the human user or by a set of *heuristics* encoded in an expert system.

3 The Heuristic Refinement Method Applied to Protein Structure Determination

In the following sections we discuss an application of the heuristic refinement method to the problem of protein structure determination. The current representations and actions of PROTEAN are described, followed by an example of the use of these actions to assemble an abstract representation of a protein. The choice of these actions is made by an experienced user of the system. Our attempts to encode heuristics in an automatic system for assembling -geometric objects in the most efficient manner is described elsewhere [Hayes-RothB 86a].

3.1 The Protein Structure Deterruination Problem

Proteins perform a wide range of functions in living organisms, ranging from chemical catalysis to signal transduction. The three-dimensional structure of a protein determines its function [Fersht 77]. Accurate determination of protein structure would greatly increase our understanding of basic biological mechanisms, and should allow custom design of proteins for new types of materials, drugs and other industrial products [Tucker 85].

Amino acids are the building blocks of proteins. The twenty different amino acids are linked in sequences to form proteins (much like different colored beads on a string). The sequence of amino acids of a particular protein is its *primary structure*, which may contain from five to more than a thousand amino acids. Subsequences of amino acids within a primary structure often arrange themselves in recognizable repeating patterns to form *secondary structures.* For example, part of the chain of amino acids may form a helix with a characteristic pitch and rise: the *alpha-helix.* The final three-dimensional configuration of all the atoms in the -protein is its *tertiary structure.* Methods to identify the primary and secondary structures have been developed, but the problem of determining how they associate to form the tertiary structure is largely unsolved, and is the problem addressed in this research.

The tertiary structures of the relatively few proteins whose structures are known have been obtained from X-ray crystallography [Blundell 76], which gives accurate three-dimensional coordinates of each constituent atom. However, many proteins do not form crystals, in which case other, less precise structural information must be used, such as the volume and shape of the molecule, indications of which atoms are on the surface, and proximity information for pairs of atoms.

Proximity information can be obtained from the Nuclear Overhauser Effect (NOE) observed in nuclear magnetic resonance experiments [Wutrich 76]. This effect provides evidence that

two atoms are within a certain distance of each other (usually 2-5 angstroms). Structure determination methods that utilize NOE information generally employ optimization procedures to produce a single structure consisting of atomic coordinates [Kuntz 79]. However, the NMR data is usually not complete enough to determine a unique solution, and in fact more than one solution may exist since the protein may have internal motion [Welch 82]. It is therefore undesirable to prematurely rule out alternate structures.

The goal of retaining all tertiary structures compatible with the data, as well as the fact that the NOE constraints are binary, allows us to formulate the protein tertiary structure determination problem as a binary geometric constraint satisfaction problem in which the number of objects, the number of possible locations per object, and the dimensionality of the search space are too large to permit direct solution by backtrack search. The heuristic refinement method is therefore used to refine the protein along the structural and accessible volume dimensions in order to allow backtrack search to be employed.

3.2 Refinement Along the Structural Dimension

The secondary structure of a protein consists of groups of atoms that form reasonably regular geometric objects. Thus, these groups may be represented at an abstract level since the entire group moves more or less as one unit when the gross topology of the protein is being determined. PROTEAN therefore first represents groups of locally constrained atoms as a small number of geometric solids defining the minimum enclosing volume of the individual atoms. The locations of these solids determine bounds for more refined solutions at the atomic level.

Figure 1 shows solid level representations of three alpha helices as cylinders defined within local coordinate systems. The positions of backbone atoms (such as the carbon atoms Cl and C2 in the figure) may be defined with respect to this local coordinate system using known parameters of a helix. These analytic equations are additional theoretical constraints reflecting the-structural model associated with the assumption of an ideal helix. Intervening unstructured sequences of amino acids are called *random* coils and are represented as spheres centered on the middle amino acid of the sequence with radius equal to half the length of the coil. The only theoretical constraint introduced in this case is the assumption that the amino acids comprising the coil are covalently linked together with a certain maximum distance between the linked amino acids. Although this assumption is more likely to be true than the assumption of ideal helices, it does not produce as great an increase in efficiency since the locations of the component backbone atoms are only known within the volume of the sphere.

If one solid is chosen to be a fixed *anchor,* then the location of any other solid (an *anchoree)* with respect to the anchor coordinate system is defined by the location of the local coordinate system of the anchoree. A *location* is a six-dimensional vector (three position components and three orientation components). Given the location of the solid the anchor coordinates of the fixed po'ints may be calculated by a suitable transformation, thereby allowing points on separate anchorees to be related to each other by means of the anchor coordinate system.

The main source of binary constraints for PROTEAN are NOE measurements, which state that two atoms are within 2 to 4 angstroms of each other. At the solid level these constraints are abstracted to state that the backbone carbon atoms involved in the NOE must be within a fixed distance range (D in figure 1). If the NOES involve random coils then the distances are referred to the center of the coil coordinate system by adding an appropriate amount to the al lowed distance range.

A pair of locations of two solid objects are tested for compatibility as shown in figure 1. The two objects are placed in their respective locations, the backbone atom points are transformed to anchor coordinates, and the distance between the points is calculated. If this distance is not within the allowed range then the locations are not compatible and one or both may possibly be eliminated. If the distance range is satisfied then any other binary constraints are checked in the same way, accepting the locations only if all the constraints are simultaneously satisfied.

3.3 Refinement along the Accessible Volume Dimension

Figure 2 shows a geometric constraint network corresponding to the solid level representation of objects and constraints. Each solid object is a node in the constraint graph and each conjunctive set of binary constraints is an arc. The numbers on the arcs are the number of binary constraints inferred from the data.

If one of the objects (say H1) is chosen to be a fixed anchor, then a single solution to this geometric constraint satisfaction problem consists of one location per object, relative to the fixed anchor, such that all the constraints are simultaneously satisfied. We call such a solution a *coherent instance.*

A straightforward approach to generating coherent instances would be to enumerate all possible locations of the anchorees with respect to the anchor and check each one for simultaneous compatibility with the constraints. However, even though the number of solid objects is small compared to the number of atoms there are still too many possible locations per object to allow solution by enumeration or backtrack search. Therefore, PROTEAN employs network consistency algorithms [Mackworth 77] to reduce the number of locations per object, thereby performing refinement along the *accessible volume* dimension.

At the beginning of problem solving each anchoree is initially assumed to be located within some large region of the six-dimensional location space with respect to the anchor, typically a 64 angstrom cube. This region is called the accessible volume of the anchoree, and is represented (implicitly at first) as a list of discrete locations sampled at some resolution (typically 2 angstroms for the position component and 30 degrees for the orientation component).

The first network consistency algorithm applied is *node consistency,* which is achieved by a procedure we call *anchoring,* shown schematically in figure 3. Conceptually, each possible location in the original "infinite" accessible volume of an anchoree is generated, then tested for compatibility with the anchor, using any constraints with the anchor, Only those locations satisfying the constraints are retained in an explicit *location table.* All rejected locations could never be part of a coherent instance since the constraints with the anchor would not be satisfied. The resulting list of locations may be displayed graphically as a *halo* of points showing the accessible volume of the anchoree with respect to the anchor. In practice the constraints are used to limit the search, using the fact that the distance constraints determine maximum possible values for the position components of the location.

The second network consistency algorithm employed is *arc consistency,* which is achieved by multiple applications of a procedure we call *yoking,* shown schematically in figure 4. A pair of accessible volumes is arc consistent if for each location in the accessible volume of the first object there is at least one location in the accessible volume of the second object such that the constraints between the two objects are satisfied. The yoke procedure examines all pairs of locations within the two accessible volumes, retaining only those that satisfy the arc consistency condition. The procedure is called multiple times until the number of locations in all accessible volumes does not decrease further.

3.4 Heuristic Control

At any time in the assembly process there are many operations that may be performed, although not all of the operations outlined above are always possible. For example, at the beginning of the problem solving it is necessary to decide which groups of atoms to consider as solid objects and which to leave as atoms, which sets of constraints to check first, and which groups of solids to consider as a partial arrangement. Within a partial arrangement it is necessary to decide which object to make the anchor, and the order of anchoring, yoking, and coherent instance generation. If a partial arrangement turns out to be unconstrained it may be necessary to consider al ternate partial arrangements before com bining partial arrangements.

Each one of the potential actions leads to a partial solution of the problem that is correct in the sense that it does not eliminate any conformations that would be present in the final atomic solution. However, it is our hypothesis that different choices of actions will result in

different degrees of efficiency in finding the solutions, and that unintelligent choice of actions may lead to unacceptably long execution times. As an extreme example, the option should always be available to refine the protein to the atomic level and to perform backtrack search to find the solution conformations. However, for any reasonable sized protein this procedure could take on the order of years and hence should not be performed. In PROTEAN the choice of which action to perform next is determined either manually or with the aid of a blackboard system called BB1 [Hayes-Roth 85]. At any time in the problem solving process the feasible geometric actions are represented as executable knowledge sources. Control knowledge sources employ *heuristics* to determine the best action to perform next. More details on the operation of the heuristic control portion of PROTEAN and on an implementation of an abstract language for con trolling arrangement assembly problems are described elsewhere [HayesRothB 86a].

4 An Example: Assembly of a Small Protein

Figure 2 shows a constraint network representing a single partial arrangement for a small protein at the solid level. This protein has 51 amino acids, three alpha helices and 4 intervening random coils. Analysis of NMR data from this protein produced 14 NOE constraints and 6 covalent constraints (points that are near each other because they are close to each other in the primary sequence) [Jardetzky 84]. These constraints were abstracted by PROTEAN to solid level binary constraints distributed according to the numbers shown on the arcs of the graph. In general a larger number of constraints between two objects results in fewer locations of one object with respect to the other.

OBJECT	LOCATIONS POSSIBLE	LOCATIONS SEARCHED	LOCATIONS FOUND	CPU TIME (SECONDS)
H2	1185408	39432	943	9
H3	1185408	170324	121	24
RC2	1428	4	3	< 1
RC1	1428	249	186	< 1
RC3	1428	263	48	< 1
RC4	1428	329	41	< 1

Table 1: Results of anchoring six objects to helix 1 (H1)

For the anchoring operation H1 was heuristically chosen to be the fixed anchor because it was the largest helix and had the greatest number of constraints to other objects. The system then chose to anchor helices H2 and H3, followed by each of the random coils in order of increasing size.

Table 1 shows the results (including timings) of anchoring each of the six anchorees to H1. In the case of the helices the three position components of the six-dimensional location space were sampled within a 64 angstrom cube centered about the origin of the anchor coordinate system. The three orientation components were sampled at 30 degree increments. For the coils only a three-dimensional search for the coil center was required since the coils are rotationally symmetric. The number of compatible locations for each anchored object ranged from 3 to 943, or 0.01% to 13.0% of the possible locations. Optimization of the search program allowed only a small percentage (from 0.2% to 23%) of the possible locations to actually be searched.

Once the six solid objects had been anchored the following yoke operations were possible, reflecting the binary constraints between anchorees shown in figure 2: RCI-H2, H2-H3, H2-RC4, RC3-H3, and H3-RC4. Table 2 shows the results, including timings, of successive iterations of the yoke procedure for the six anchorees. In this demonstration all 5 of the possible yokings were performed on the next iteration when any location table changed for an

	H2	H3	RC1	RC2	RC3	RC4	TIME
Anchoring	943	121	186	3	48	41	35
Iteration 1	114	11	114	3	6	22	19
Iteration 2	21	11	113	3	6	17	3
Iteration 3	21	11	113	3	6	17	3

Table 2: Results of three iterations of yoking, numbers of locations, time is CPU time in seconds

anchoree. After 3 such iterations no location tables changed, so arc consistency was achieved. The most dramatic drop in location table size was seen in H2, initially anchored with 943 locations, but retaining only 21 locations after being yoked 3 times to H3 and RC4. Other location tables were also pruned significantly.

Figure 5 is a 3D display of part of the final arc consistent network, showing halos for the three helices H1, H2 and H3, as well as a single coherent instance generated by backtrack search from the final network. These locations are constrained enough to provide meaningful biochemical information about the topology of the molecule. Detailed analysis of the accuracy and precision of these and other results are described elsewhere [Lichtarge 86 3.

5 Discussion

This paper describes a method for approaching geometric constraint satisfaction problems, and presents an example of a partial implementation of the method for protein structure determination. Other publications expand on this method and show its application to more complex proteins as well as other arrangement problems [Buchanan 85, Altman 86, Duncan 87, Hayes-Roth8 86b, Brinkley 86, Lichtarge 86].

The results show that refinement along the structural and accessible volume dimensions reduces the num ber of objects and the number of possible locations per object to the point that solid level solution by backtrack search is feasible. The application of additional structural knowledge allows this refinement to be accomplished without losing too much information. The solid level accessible volumes, together with the solid representations, can also be used to obtain the initial accessible volumes of the individual atoms of the protein. These accessible volumes will be much smaller than the initial accessible volumes obtainable without the solid level solutions. The atoms, together with their accessible volumes and the initial constraints, constitute the original atomic level constraint satisfaction problem, in which the variables are the atoms and the values are the accessible volumes of the atoms. The initial step of solving the constraint satisfaction problem at the solid level results in a smaller number of possible values for each atom, leading to a small enough problem to be solved by consistency algorithms and backtrack search. If the size of the problem becomes small enough then memory-intensive algorithms such as path consistency [Mackworth 77) or backtrack with 'learning [Dechter 86] may be employed to further increase the efficiency of the geometric constraint satisfier.

An important goal of the heuristic refinement method is to ensure that each refinement operation produces *correct* results in the sense that it does not eliminate any solutions that would be found by backtrack search. Heuristics are mainly employed to improve the *efficiency* of the search while retaining correctness. However, in general this complete separation of heuristic efficiency from correctness cannot be maintained since the imposition of a structural model implies the introduction of theoretical constraints which represent an idealization of reality. In fact no helix is actually in the shape of a cylinder although most helices come very close to that ideal. Nevertheless, for complex problems the introduction of simplifying assumptions is necessary in order to allow the problem to be solved in a reasonable amount of

time. These assumptions are the result of applying domain specific knowledge and are almost certainly an important component of human intelligence as well.

Although the heuristic refinement method has been developed in the context of protein structure determination, it should have direct application to other types of geometric constraint satisfaction problems. In fact many problems are simpler than the protein structure problem in one or more ways: the number of objects or the number of possible locations per object may be smaller, the dimensionality of the solution may be less, or the need to find all solutions may be reduced. However, in all cases the basic operations of heuristic refinement along the structural and accessible volume dimensions may be employed by suitable alterations to the parameters of the program.

For example, the location of each object in a furniture arrangement consists of two position components and one rotation component, making this a three-dimensional spatial constraint satisfaction problem. Further, the number of possible locations per object may be reasonably small since the possible orientations may be limited. Given two possible locations for two pieces of furniture any binary constraint that can be formulated as a geometric predicate may be included in the constraint tester. For example, predicates may require that two pieces of furniture be touching, or that there is a line of sight between the couch and the television. Groups of highly constrained pieces may be considered as single objects, for example an end table and a couch that are always kept together.

The notion of heuristic refinement should also be generalizable to other types of multi-dimensional constraint satisfaction problems since the grouping of objects is a ubiquitous activity. In complex real-world situations it is necessary both for humans and computers to group objects in order to reduce the number of entities that must be considered. In order to be useful the grouping should impose an abstract structure or model on the individual objects, in the form of theoretical constraints, such that the large number of objects may be replaced by a more concise description. The adequacy of the model for problem solving depends on its ability to increase efficiency while still satisfying the original constraints. Decisions as to which objects to group together and which model to impose are made with the aid of domain-specific knowledge learned from experience. The adequacy of these decisions seems to be an important factor in determining the amount of intelligence exhibited by a human or by a computer.

[Allen 83] Allen, J.
Maintaining knowledge about temporal intervals.
CACM 26:832-843, 1983.

[Altman 86] Altman, R. and Jardetzky, 0.
New strategies for the determination of macromolecular structure in solution.
J. Biochemistry 100(6):1403-1423, 1986.

[Blundell 76] Blundell, T.L. and Johnson, L.N.
Protein Crystallography.
Academic Press, 1976.

[Brinkley 86] Brinkley, J., Cornelius, C., Altman, R., Hayes-Roth, B., Lichtarge, O., Duncan, B., Buchanan, B., and Jardetrky, 0.
Application of constraint satisfaction techniques to the determination of protein tertiary structure.
Technical Report KSL-86-28, Stanford University, March, 1986.

[Buchanan 85] Buchanan, B., Hayes-Roth, B., Lichtarge, O., Altman, R., Brinkley, J., Hewett, M., Cornelius, C., Duncan, B., Jardetzky, 0.
The heuristic refinement method for deriving solution structures of proteins.
Technical Report KSL-85-41, Stanford University, Knowledge Systems Laboratory, 1985.

[Clancey 85] Clancey, W.J.
Heuristic Classification.
Artificial Intelligence 27(3):289-350, 1985.

[Dechter 85] Dechter, R. and Pearl, J.
The anatomy of easy problems: a constraint satisfaction formulation.
In *IJCAI85*, pages 1066-1075. Morgan Kaufmann, 1985.

[Dechter 86] Dechter, R.
Learning while searching in constraint-satisfaction problems.
In *Proceedings of the Fifth National Conference on Artificial intelligence,* pages 178-183. American Association for Artificial Intelligence, Philadelphia, August 11-15, 1986.

[Duncan 87) Duncan, B., Buchanan, B., Lichtarge, O., Altman, R., Brinkley, J., Hewett, M., Cornelius, C., and Jardetzky, 0.
PROTEAN: A new method for deriving solution structures of proteins.
Bull. Mug. Res. , 1987.
In press.

[Fersht 77] Fersht, A.
Enzyme Structure and Mechanism.
W.H. Freeman & Co., 1977.

[Hayes-Roth 85] Hayes-Roth, B.
A blackboard architecture for control.
Artificial Intelligence 26:251-321, 1985.

[Hayes-RothB 86a]
Hayes-Roth, B., Johnson, M.V., Garvey, A., Hewett, M.
A layered environment for reasoning about action.
J. Artificial Intelligence in Engineering: Special issue on Blackboard systems, October, 1986.

[Hayes-RothB 86b]
Hayes-Roth, B., Buchanan, B., Lichtarge, O., Hewett, M., Altman, R., Brinkley, J., Cornelius, C., Duncan, B., Jardetzky, 0.
PROTEAN: Deriving protein structure from constraints.
In *Proceedings of the Fifth National Conference on Artificial Intelligence,* pages 904-909. Americal Association for Artificial Intelligence, Philadelphia, Penn., August 11-15, 1986.
Also published as Stanford University Technical Report KSL 86-38.

[Jardetzky 84] Jardetzky, 0.
A method for the definition of the solution structure of proteins from NMR and other physical measurements: the lac-repressor headpiece.
In Proceedings of the International Conference on the Frontiers of Biochemistry and Molecular Biology. Elsevier, Alma Ata, June, 1984.

[Kuntz 79] Kuntz, I.D., Crippen, G.M. and Kollman, P.A.
Application of distance geometry to protein tertiary structure calculations.
Biopolymers 18:939-957, 1979.

[Lichtarge 86) Lichtarge, O., Cornelius, C.W., Buchanan, B.G., Jardettky, 0.
Validation of the First Step of the Heuristic Refinement Method for the Derivation of Solution Structures of Proteins from NMR Data.
1986.
Submitted to *Proteins.*

[Mackworth 77] Mackworth, A.K.
Consistency in Networks of Relations.
Artificial Intelligence 8:99-118, 1977.

[Tucker 85) Tucker, J.B.
Proteins to Order.
High Technology :26-34, Dec, 1985.

[Waltz 75] Waltz, D.
Understanding line drawings of scenes with shadows.
In Winston, P.H. (editor), *The Psychology of Computer Vision,* . McGraw-Hill, New York, 1975.

[Welch 82] Welch, G.R., Somogyi, B., Damjanovich, S.
The Role of Protein Fluctuations in Enzyme Action: A Review.
Prog. Biophys. Molec. Biol. 39:109-146, 1982.

[Willey 81] Willey, D.S. and Toller, D.R.
SPA: Automating bathroom design.
Computer-Aided Design 13(3):137-144, 1981.

[Wutrich 76] Wutrich, K.
NMR in Biological Research: Peptides and Proteins.
North Holland, Amsterdam, 1976.

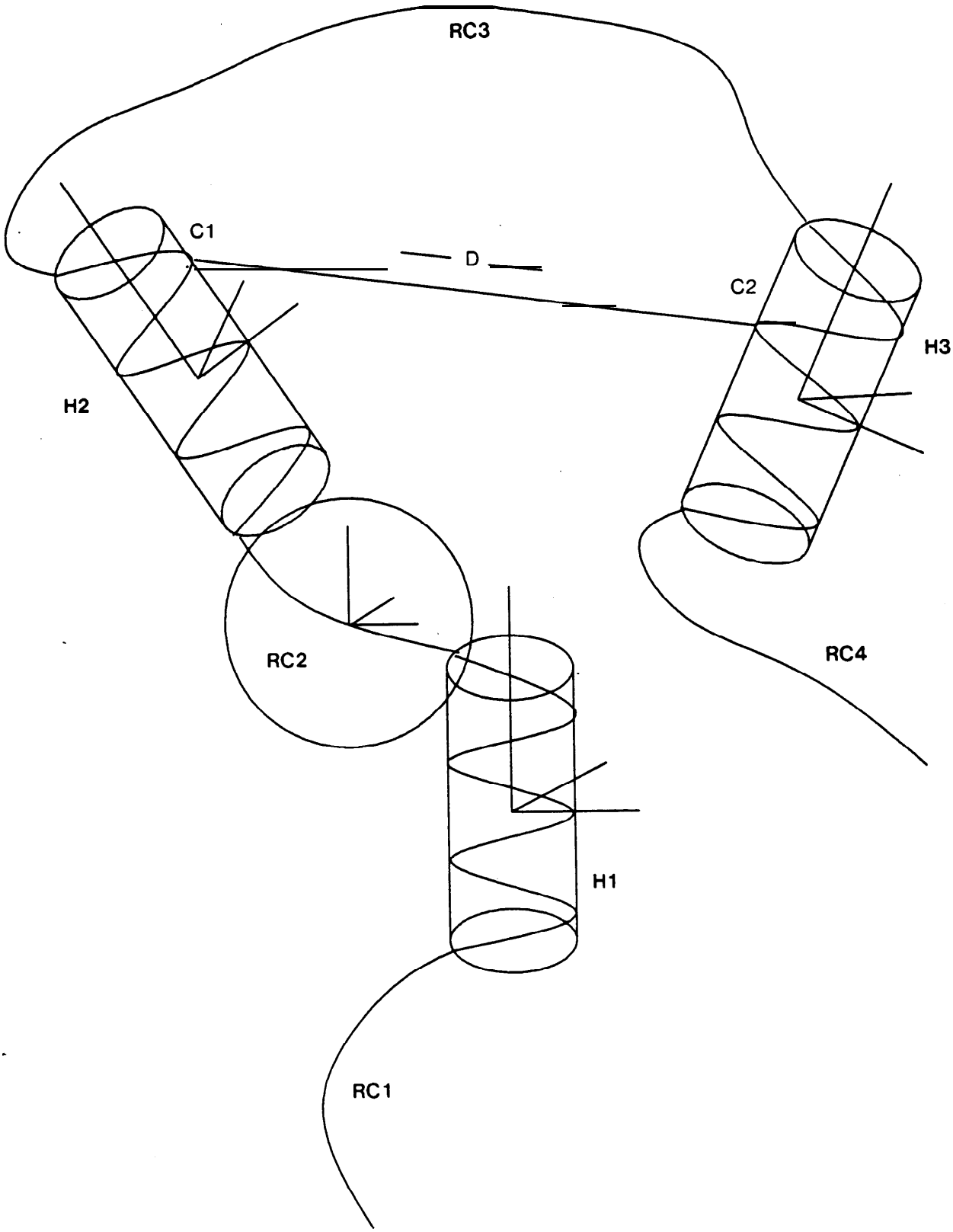


Figure I: Secondary structures represented as solids within local coordinate systems. Distance constraints are abstracted to distances D between fixed points Cl and C2 on solids. RC1-RC4 are random coils, HI-H3 are helices.

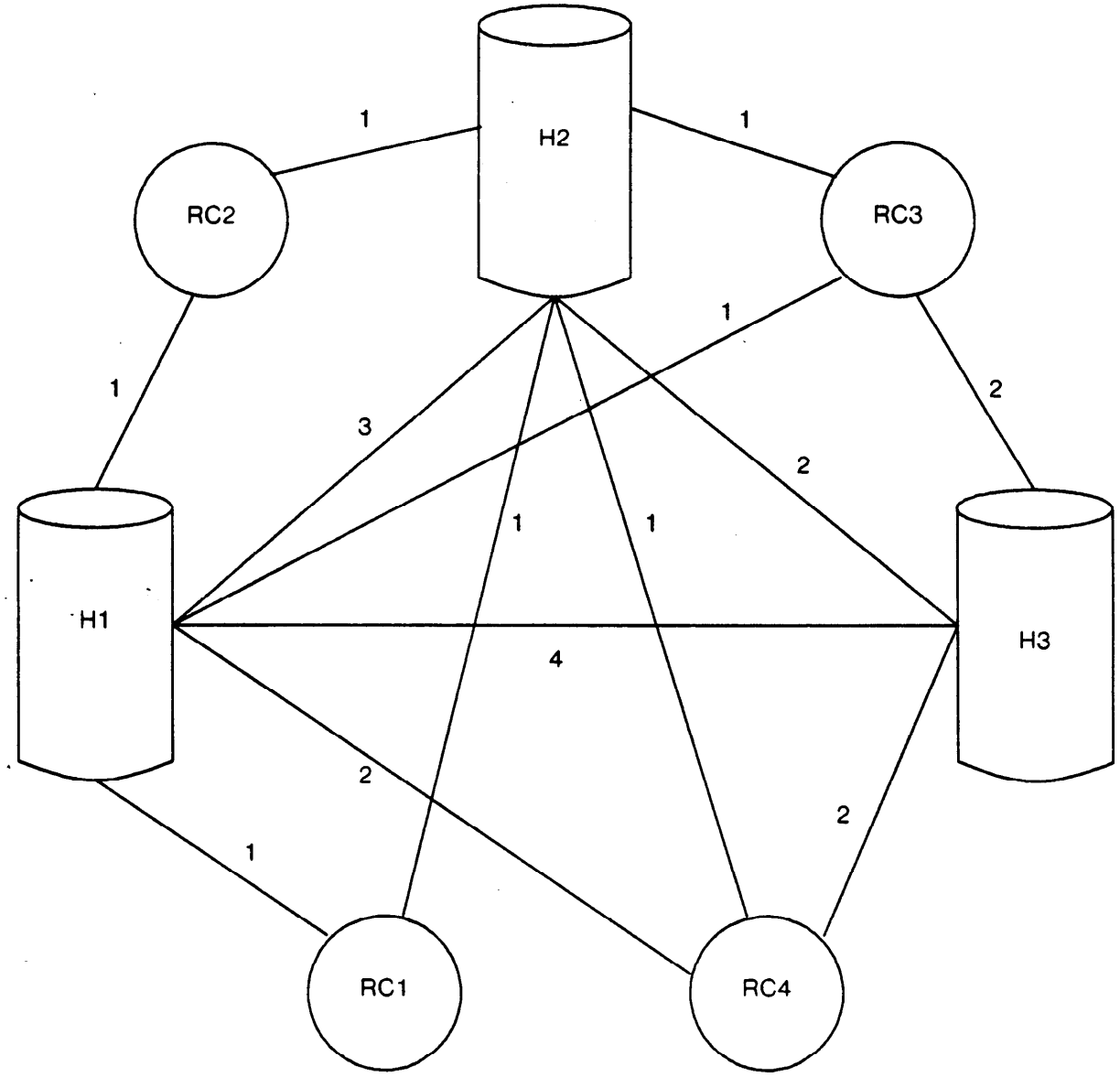


Figure 2: **Solid level geometric constraint network. The nodes are geometric solids, the arcs are constraints. Numbers on the arcs are the number of conjunctive constraints.**

Figure 3: Anchoring of Helix2 and Helix3 to Helix1 results in accessible volumes defined by only those locations compatible with the anchor Helix1

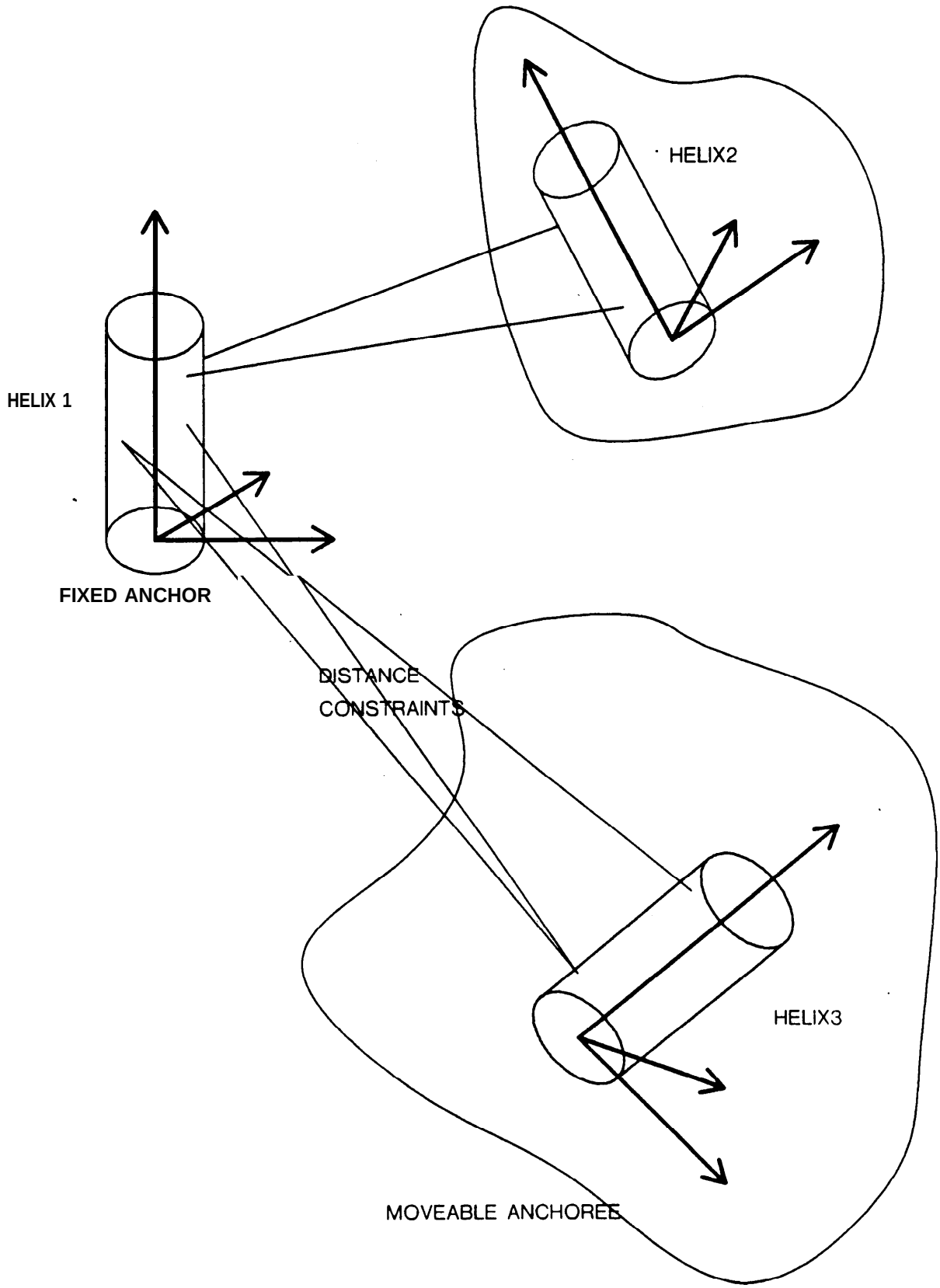
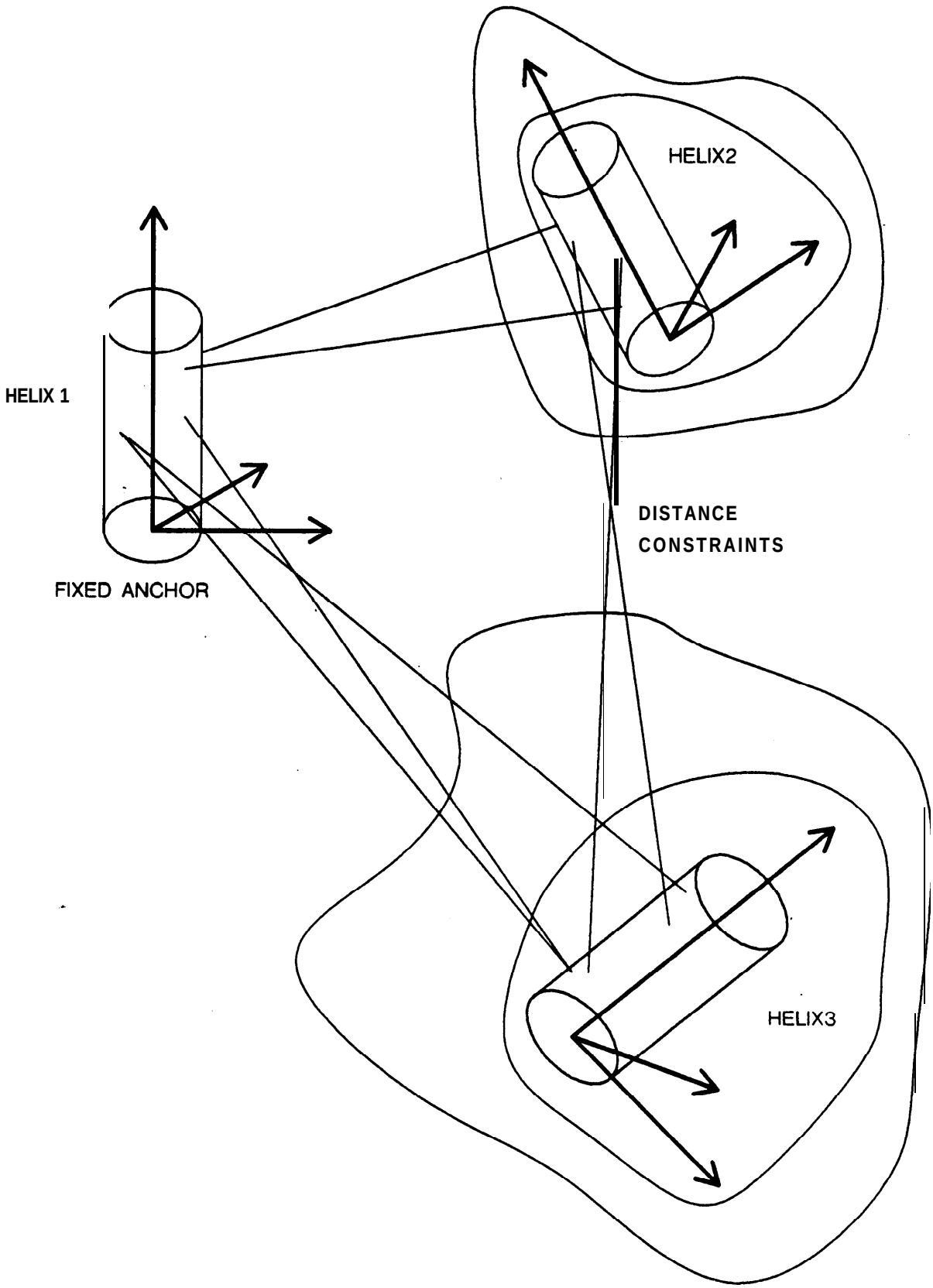


Figure 4: Yoking of Helix2 and Helix3 results in accessible volumes in which each location for Helix2 is compatible with at least one for Helix3, and vice versa



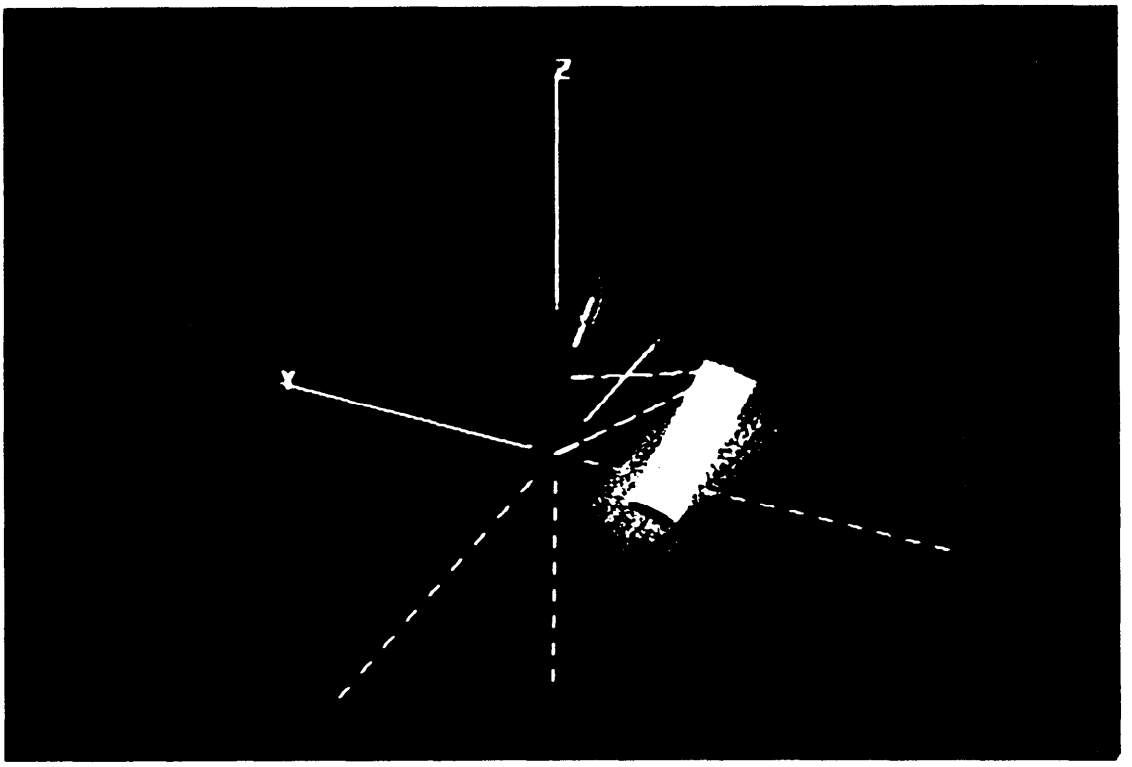


Figure 5: 3D display of arc consistent solid level helix halos, and a single coherent instance.