Rachel Kolodny

Department of Structural Biology and
Computer Science Department
Stanford University
Stanford, CA 94305, USA
trachel@cs.stanford.edu

Leonidas Guibas

Computer Science Department
Stanford University
Stanford, CA 94305, USA

Michael Levitt
Patrice Koehl

Department of Structural Biology
Stanford University
Stanford, CA 94305, USA

Inverse Kinematics in Biology: The Protein Loop Closure Problem

Abstract

Assembling fragments from known protein structures is a widely used approach to construct structural models for new proteins. We describe an application of this idea to an important inverse kinematics problem in structural biology: the loop closure problem. We have developed an algorithm for generating the conformations of candidate loops that fit in a gap of given length in a protein structure framework. Our method proceeds by concatenating small fragments of protein chosen from small libraries of representative fragments. Our approach has the advantages of ab initio *methods since we are able to enumerate all candidate loops in the discrete approximation of the conformational space accessible to the loop, as well as the advantages of database search approach since the use of fragments of known protein structures guarantees that the backbone conformations are physically reasonable. We test our approach on a set of 427 loops, varying in length from four residues to 14 residues. The quality of the candidate loops is evaluated in terms of global coordinate root mean square (cRMS). The top predictions vary between 0.3 and 4.2 Å for four-residue loops and between 1.5 and 3.1 Å for 14-residue loops, respectively.*

KEY WORDS—inverse kinematic problem, loop closure, protein structure, protein fragment libraries

1. Introduction

Inverse kinematics is classically defined as the process of characterizing the geometry of an open kinematic chain composed of rigid links, given the position of its end points. This process arises often in robotics, where it is reformulated as computing the geometry of a closed chain system corresponding to a given end-effector configuration. Examples of closed systems include reconfigurable robots (Kotay et al. 1998), as well as closed chains formed by multiple robots grasping an object (Khatib et al. 1996). Many problem-specific solutions to the inverse kinematics problems on closed chain systems have been developed (see, for example, Zhao and Badler 1994; Deo and Walker 1995; Nielsen and Roth 1999; Tolani, Goswami, and Badler 2000; Park, Chang, and Yang 2003; Wampler 2004; Wang and Chirikjian 2004; and references therein). Interestingly, computational chemists and structural biologists implicitly solve the same problem in their attempts to model the structure of a cyclic molecule, or to define the conformation of a molecular segment whose end-points are geometrically constrained (Manocha and Zhu 1994; Manocha, Zhu, and Wright 1995). In this paper we consider the latter problem for protein segments, often referred to as the loop closure problem.

Detailed knowledge of the tertiary structure of a protein is required for an understanding of its biological function. Experimental data at atomic resolution can usually be obtained by X-ray crystallography and/or nuclear magnetic resonance techniques. It is not feasible however to determine experimentally the structure of the millions of proteins whose corresponding genes have been sequenced as part of the multiple

The International Journal of Robotics Research
Vol. 24, No. 2–3, February–March 2005, pp. 151-163,
DOI: 10.1177/0278364905050352
©2005 Sage Publications

genome projects. This has resulted in the appearance of a large number of protein structure modeling techniques. Although considerable theoretical effort has led to progress in recent years (Bonneau and Baker 2001; Moult et al. 2003), successful *ab initio* protein structure prediction from the protein sequence alone does not appear imminent. As an alternative to these theoretical approaches, there is hope that biologically useful models can be derived by inference from the databases of known protein structures. This hope is based on the common knowledge that proteins with homologous sequences share similar structures. In such cases, models for the unknown structure of a new protein (the target) can be derived from the structure of a homologous protein (the template) using comparative modeling techniques (Browne et al. 1969). Figure 1 provides a simple overview of comparative modeling. It starts from the alignment of the sequences of the target and template proteins. Parts of the template structure corresponding to conserved regions in this alignment are copied, defining the three-dimensional framework for the structure of the target protein. The variable regions in the alignment correspond to the gaps in the framework. These are usually the results of substitutions, insertions and deletions of residues between members of the same structural family, and frequently correspond to exposed loop regions that connect elements of secondary structure in the protein fold. Thus, the problem of filling the gaps in the framework is often referred to as the loop closure problem. It is worth noticing that as the framework is not exact, we are only looking for approximate solutions to this problem. In testing the fitness of a candidate loop in a gap of the framework, a tolerance of 1 Å in the positions of its end points is usually considered acceptable. This 1 Å should be compared to the length of each link of the loop, i.e., 3.8 Å.

Many loop building procedures have been described. Analogously to the prediction of whole protein structures, these methods can be divided into *ab initio* methods, and database search techniques. The *ab initio* loop building is based on a conformational search, often guided by a scoring or energy function. The search can be deterministic. Analytical loop building dates back to the pioneering work of Go and Scheraga (1970). It is possible to predict the conformation of short constrained segments by solving a set of algebraic equations that describe the geometry of the system (Go and Scheraga 1970; Bruccoleri and Karplus 1985; Palmer and Scheraga 1991; Manocha and Canny 1994; Manocha, Zhu, and Wright 1995; Wedemeyer and Scheraga, 1999). Unfortunately, this statement does not extend to loops with more than six degrees of freedom (Go and Scheraga 1970; Palmer and Scheraga 1991). Even under the assumption of rigid geometry, i.e., keeping idealized bond lengths and bonds angles, this sets the limit of analytical methods to loops of two residues or less. Building longer loops relies therefore on heuristics, and sampling becomes a critical issue. *Ab initio* methods for building long loops should provide many different closures which sample a large conformational space (C-space) such as to maximize the probability that the correct (or very close) structure is included. Complete sampling of long loops requires that the C-space be discrete. Usually, a restricted set of (Φ, Ψ) torsion angles is used to approximate all possible conformations. The search is then performed either using uniform sampling, or biased sampling based on the known population of the (Φ, Ψ) maps (Bruccoleri and Karplus 1985; Moult and James 1986; Dudek and Scheraga 1990; Deane and Blundell 2000). Various methods have been developed to optimize the initial loops obtained through these sampling procedures. These include molecular dynamics simulations (Bruccoleri and Karplus 1990), Monte Carlo searches with simulated annealing (Collura, Higo, and Garnier 1993; Carlacci and Englander 1993), dynamic programming (Vajda and DeLisi 1990; Finkelstein and Reva 1992), genetic algorithms (McGarrah and Judson 1993; Ring and Cohen 1994), bond scaling with relaxation (Zheng et al. 1993a, 1993b; Rosenbach and Rosenfeld 1995; Zheng and Kyle 1996), and multicopy searches (Zheng et al. 1994). Other optimization methods have been directly derived from robotics (see, for example, Lavalle et al. 2000). Shenkin and colleagues described a "random tweak" algorithm based on the Jacobian matrix of first derivatives of distances between atoms of the terminal residues of the loop, with respect to its degrees of freedom (Fine et al. 1996; Shenkin et al. 1987). In their method, all the torsion angles of the loop are modified together in each step of the iteration process, until the distance constraints between the end residues are satisfied. Canutescu and Dunbrack (2003) have recently proposed an improved version of that method, in which the degrees of freedom are varied one at a time, using the cycle coordinate descent (CCD) algorithm (Wang and Chen 1991).

The wealth of information available on protein structures makes the protein loop closure problem different from standard inverse kinematics problem. As of today the Protein Databank (PDB; Bernstein et al. 1977; Berman et al. 2000) contains more than 20,500 experimentally determined protein structures. This has led to an alternative approach to *ab initio* loop building that searches this database for loop candidates, based on geometric fitness criteria. This method was originally introduced by Jones and Thirup (1986) to facilitate model building for crystallographic refinement by selecting protein fragments to fit in the electron density map. Methods of this type have the advantage of guaranteeing rapid results that have physically reasonable conformations. However, Fidelis et al. (1994) concluded that the use of segments from the PDB was useful only for loops up to a length of four residues, as the completeness of the database degrades rapidly with increasing length. Recent studies have shown that this limit can be extended to longer loops (nine residues based on the van-Vlijmen and Karplus 1997 study, and 15 residues according to Du, Andrec, and Levy 2003), due to the enormous increase in the PDB in the last years.

In the present study, we develop a method for protein loop building that combines the two basic approaches described

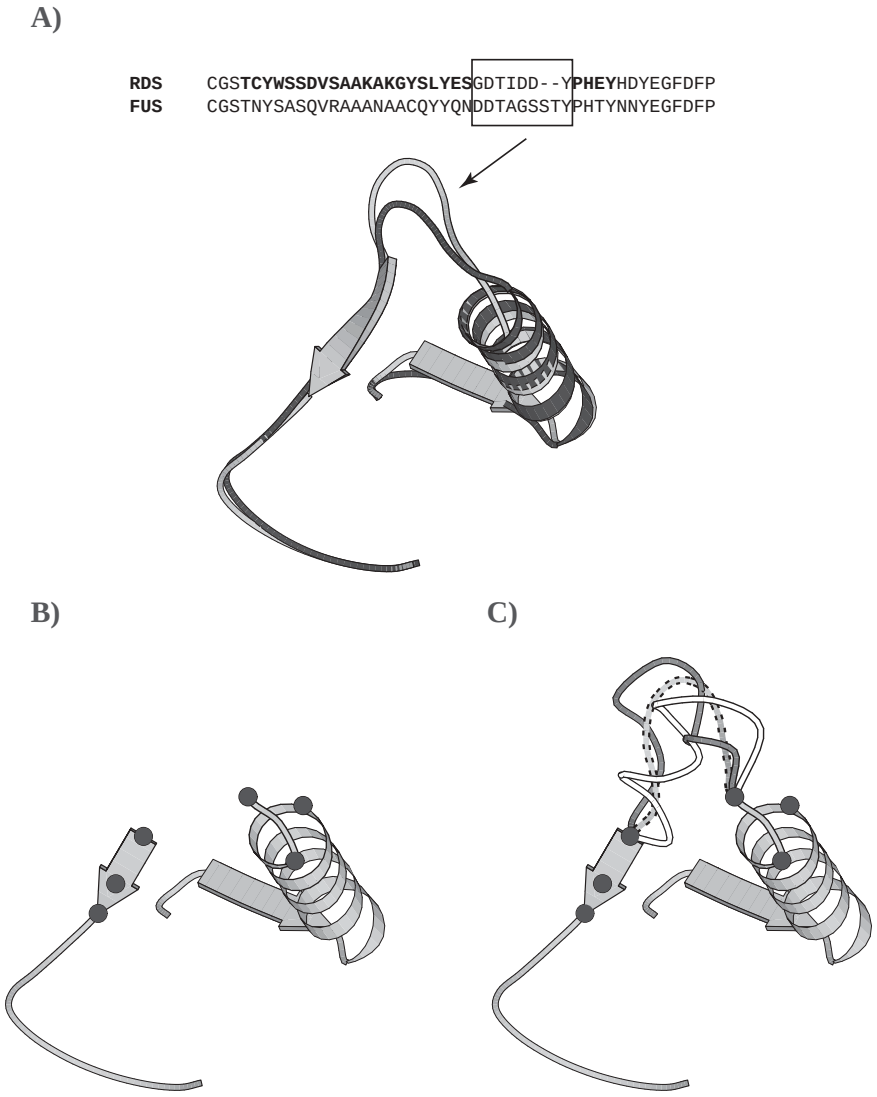

Fig. 1. Homology modeling of the Ribonuclease F1 from fusarium monoliforme (FUS). RF1 is a small protein of 106 residues whose structure is known (PDB code 1FUS). A toy experiment designed to predict the structure of RF1 proceeds via a number of steps. (A) The identification of an homologue of the known structure (the "template"). The sequence of FUS is found to be similar to the sequence of Ribonuclease MS from molsin (RDS) (Fasta E-value: 1e-22; 57,8% sequence identity in 102 amino acid overlap). The sequence alignment between FUS and RDS is compared to the structural alignment between their corresponding structures, 1FUS and 1RDS (shown as gray and black, respectively). The structural alignment was computed using STRUCTAL (Subbiah, Laurents, and Levitt 1993). For the sake of clarity, only the region between residues 20 and 50 of FUS is shown. Conserved regions in the sequence alignment correspond to conserved regions in the structural alignment (coordinate root mean square (cRMS) = 0.8 Å over 102 residues). The two structures differ in the loop region between two secondary structures (highlighted in bold in the sequence alignment). The loop in FUS is longer by two residues. (B) Building the framework. A framework for FUS is built by using the backbone regions of RDS corresponding to the conserved secondary structures. The framework has a gap of nine residues. The position of the three residues preceding and following the gaps are marked with balls centered at their C_α. These residues define the "stems" of the loop to be modeled. (C) The loop building process. Fragments of nine residues with proper end-to-end geometry are either selected from a database of protein segments, or built using small libraries of protein building blocks. The dashed loop is the correct, native loop, while the gray and white loops are the best loops derived from the database (5.6 Å) and through construction (2.7 Å), respectively. Once a protein segment is selected to fill the gap, the side chains are added to yield a full atom model for the target protein FUS. This figure was generated using MOLSCRIPT (Kraulis 1991).

above. Following *ab initio* procedures, we include an exhaustive search of a discretized version of the C-space. We represent the candidate loops as a sequence of rigid building blocks that are concatenated without any degrees of freedom. Following the database approach, the building blocks are chosen from libraries of short protein backbone fragments, which represent protein chains accurately, and are economical, i.e., of low complexity (Kolodny et al. 2002). To mitigate combinatorial explosion, the procedure simultaneously builds two parts of the loop by starting from both ends of the gaps in the framework, and tests for closure where the two parts should join. The coarseness of the sampling is defined by the size of the library of fragments used to build the loop. In principle, this method combines the advantage of exhaustive enumeration, with the ability of the database approach to generate segments that are locally physically reasonable.

The paper is organized as follows. In Section 2, the technique is described in detail, as well as the criteria used for testing. Section 3 provides a comparison of the results of simple database searches for candidate loops, with the exhaustive set of loops generated with our fragment-based approach, based on the modeling of 38–40 loops of known structure at each length from four to 14 residues. A concluding discussion ends the paper.

2. Methods

In this section, we describe two approaches for generating the conformation of candidate loops that satisfy given end-to-end constraints. The first approach searches through a database of protein fragments of appropriate size for those that satisfy the constraints, while the second approach builds systematically an ensemble of loops obtained by concatenating short protein fragments. Note that we are interested in the generation of the loops, and not in the subsequent step of selecting hopefully native-like conformations. We only check for geometric matching (i.e., satisfaction of the end-to-end criteria) and ignore any possible collision of the loop with the rest of the protein, or with itself.

2.1. Database Approach to the Protein Loop Closure Problem

We follow a procedure similar to that of Jones and Thirup (1986) and Summers and Karplus (1990). A conformational filter is designed to screen a protein crystal structure database for possible segments. The filter consists of the set of C_α–C_α interatomic distances generated from the three C_α atom positions immediately preceding and following an n-residue gap in the protein of interest (note that this filter is sequence-independent). It includes 15 target distances (*TDn*). The protein database consists of high-quality, non-homologous, X-ray structures ($R < 2$ Å). This representative subset is a selection of ungapped protein domains with a BLAST (Altschul et al. 1990) 0.0001 level of sequence similarity; that is, the similarity between any two sequences in this subset has an expectation value (*E* value) greater than 0.0001. It includes 3307 protein domains from SCOP 1.63 (Murzin et al. 1995), and can be retrieved from the ASTRAL compendium (Brenner, Koehl, and Levitt 2000; Chandonia et al. 2002).

Possible segments that fit in the gap under study are obtained by computing the root-mean-square deviation, $RMS(n, P, S)$, between the set of target distances *TDn* and the equivalent set of distances $RNn(P, S)$ for every $n + 6$ residue segment S in each protein P in the database (Summers and Karplus 1990):

$$RMS(n, P, S) = \sqrt{\frac{\sum_{i=1}^{15} (TDn(i) - RNn(P, S)(i))^2}{15}}. \quad (1)$$

This subset is filtered by removing all segments with RMS values exceeding 1 Å. The selected segments must also match specific conformational restrictions related to the sequence of the gap. Fragments in which the conformation of a non-Gly residue in the gap is defined by ($\Phi > 0$, $\Psi < 0$) are eliminated. Similarly, Pro residue conformations are restricted to $\Phi_{Pro} \in [-90°, -30°]$ and $\Psi_{Pro} \in [-85°, 0°]$ or $\Psi_{Pro} \in [115°, 175°]$, and residues X preceding a Pro must be characterized by $\Phi_X \in [-210°, -30°]$ or $\Phi_X \in [30°, 90°]$, and $\Psi_X \in [60°, 180°]$ (Summers and Karplus 1990). If none of the segments verifies all these criteria, the next segment (in terms of root mean score) are considered until at least one possible segment is found.

2.2. Generating Loops Using Libraries of Small Protein Fragments

2.2.1. Protein Structure Building Blocks

The approximations we use for loop structures are based on libraries of commonly observed, yet geometrically diverse, fragments of protein backbone. The fragments we consider are five amino acids long, and are represented by the three-dimensional coordinates of their C_α atoms. We employ libraries of $L = 20, 40, 60, 80$ and 100 fragments. The libraries are compiled by clustering (non-overlapping) fragments from 200 high-resolution structures of protein domains, based on their geometric similarity, and then selecting one representative per cluster. We use the cRMS as a geometric similarity measure of two fragments. The cRMS is computed after optimal superposition of the two fragments in three dimensions (Kabsch 1976). This work has been described in detail (Kolodny et al. 2002). Note that the fragment libraries do not contain any information about the sequence of the proteins from which they were built. These libraries were originally designed to represent entire protein structures.

2.2.2. Loop Generation

Protein loops of any length can be constructed by concatenating library fragments. Depending on loop length, we use two different strategies, as follows.

Strategy A: unidirectional construction (for short loops). Candidate loops are built by concatenating fragments from the library. The position of each new fragment is determined by best superimposing its first three C_α atoms onto the last three C_α atoms of the chain already constructed. Even if the two C_α triplets do not match perfectly, this defines the relative orientation of the fragments uniquely, provided that the atoms of the triplets do not lie along a line. Figure 2 illustrates this procedure on a two-dimensional example. We emphasize that the library fragments are used as mere templates; any fragment can be used repeatedly along the constructed chain. Most of the loops constructed using this scheme will not allow closure. However, since the number of possible fragments is finite, we can enumerate all loops of the appropriate length, and filter the ones that reach loop closure. Notice that, due to the noisy nature of experimentally determined structures, loops need only to be approximately closed; we allow a 1 Å tolerance in the atomic position. There are several (essentially equivalent) ways for enumerating all loops of length l. We can construct systematically all chains of $l+4$ atoms and select the ones that fit in the gap of the framework. Alternatively, we can construct a chain of $l+5$ atoms, starting from the anchor points on the N-terminal side of the gap in the framework, and test the position of the last two atoms of the chain, with respect to the position of the first two anchor points of the stem at the C-terminal side of the loop. The fragments encode common angles of protein structure; the rotational angles in chains built from the library fragments are "protein-line" or common in the PDB. Overlapping the last two atoms guarantees that the angle at the attachment point is "protein-like" too. A two-dimensional analog of this option is depicted in Figure 2. The total number N of chains of length l to consider is

$$N = L^{\lceil l/(f-3)\rceil+1}, \tag{2}$$

where L is the number of fragments in the library and f is the size of the fragment (five in this study; Kolodny et al. 2002). Only a small fraction of N corresponds to valid loops, i.e., chains that solve the loop closure problem. Note that the same strategy can be used to construct a loop from the C-terminus toward the N-terminus.

Strategy B: bidirectional construction (for long loops). While strategy A is well adapted for short loops ($l < 9$), it fails for longer loops because of the combinatorial explosion in the number N of chains to generate (see eq. (2)). One solution to overcome this problem is to generate half-loops starting from the two stems in the framework, and then assemble the half-loops that (approximately) overlap at their end points. Using this procedure, we enumerate chains of length half the length of the loop, resulting in a reduction

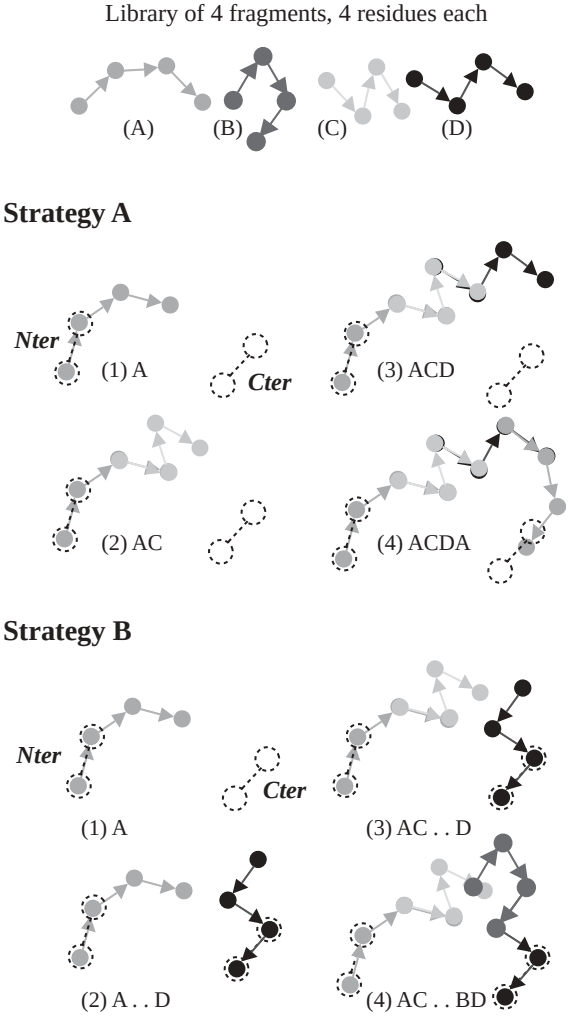

Fig. 2. Loop construction using a library of fragments. For clarity, we consider a two-dimensional example in which we build a seven-residue loop from a library containing four fragments, each four residues long. The stems of the loop in the framework are shown with dashed lines (two anchor points on each side), and labels Nter and Cter, corresponding to the N-terminal and C-terminal ends of the loop, respectively. Strategy A: construction of the loop starts from the left anchor points. In each extension, the first two residues of the new fragment (picked from the four fragments A, B, C or D in the library) are superimposed on the last two residues of the loop. In the example shown, the loop ACDA ends (approximately) on the first right-hand anchor point. Strategy B: the loop is built from both side of the gap in the framework. In the example shown, the last residue of extension AC and the first residue of extension BD (approximately) overlaps. Notice that library fragments can be reused, and that the chain has a direction. In the two-dimensional case shown here, positioning a a new fragment on an existing extension requires two residues, and loop closure is checked over one residue. In three dimensions, positioning requires three residues, and loop closure is checked over two residues.

of the running time by a factor of 2 in the exponent. Two half-loops meet when the last two C_α atoms of the first half-loop are approximately positioned on the first two C_α atoms of the second half-loop. Overlapping two atoms, guarantees that the positioning of the residues around the meeting point has "protein-like" structure. A two-dimensional analog of this type of construction is described in Figure 2. Clearly, storing all half-loops is impractical. Instead, we generate loops using three steps, as follows.

1. Mark all positions in space that are end points of the last two C_α atoms of a half-loop generated from the N-terminal stem. Since the data points are noisy, voxels with 1 Å resolution suffice to describe the three-dimensional space. The amount of memory used to store these end pairs can be further reduced if we take into account the fact that the distance between two consecutive C_α atoms is fixed.

2. Enumerate all half-loops generated from the C-terminal stem, and store those that are part of a valid loop. These are the half-loops with end points that fall in the same voxel as previously marked positions. These marked points are flagged as "valid".

3. Regenerate the first half-loops corresponding to the "valid" marked points, and assemble the loops.

We implemented both methods and observed similar results for short loops.

2.3. Accuracy of Loop Predictions

There will generally be a range of accuracy in predicting loop conformations for different loops. It is therefore necessary to assess the quality of a method by testing it on a large selection of loops. Our test set, which we refer to as the SALI set, contained 427 loops selected from the list reported by Sali and co-workers (Fiser, Do, and Sali 2000). The length of the loops ranges from four to 14 residues. The initial SALI set contained 40 proteins for each loop length; it was filtered to remove proteins that are now considered obsolete in the PDB. The residue numbers that define the target loops were taken from Fiser, Do, and Sali (2000).

The accuracy of a single loop prediction is measured by comparing it to its native conformation. A large variety of criteria for comparing loop conformations exist. They range from cRMS measures on different sets of atoms (C_α only, or all main-chain atoms), to dihedral angle and dihedral angle class comparison. In this study, we rely on cRMS, computed over Cartesian coordinates. Our measure is equivalent to the "global" RMS defined by Fiser, Do, and Sali (2000). It is computed by finding the optimal superposition (Kabsch 1976) of the stem residues of the test loop and native loop, respectively, and summing the subsequent differences in the positions of the C_α of the two loops. The global RMS provides both a measure of the local fitness of the candidate loop with respect to the native loop, and a measure of the quality of its positioning in the framework.

3. Results

Here, we test two different methods for protein loop building based on information extracted from known protein structures. The first approach searches a database of protein structures for loop candidates of the correct size, and evaluates these candidates based on geometric fit of their flanking residues ("stem residues"). In the second approach, we explore systematically a discrete approximation of the conformational space accessible to the loop considered. In this approximation, protein loops are built from small libraries of protein fragments of five residues. These libraries have been shown to provide good approximations of protein structures (Kolodny et al. 2002). Both approaches are tested on 427 target loops, varying in length from four residues to 14 residues. We are concerned with the sampling properties of our loop building procedures. Note that the tests described in this study do not reproduce real homology modeling experiments as we use the native conformation of the protein as the framework to build the loop.

3.1. Selecting Loop Conformations from a Database of Protein Structures

We use a non-redundant subset of SCOP1.63 domains as a database for searching candidate template loops that fit in a given target loop (see Section 2 for the definition of the non-redundant subset). To test the completeness of this database, we plot in Figure 3, for all target loops, the cRMS of the five best-fitting template loops. Since most of the proteins of our test set are present in our database, the search of candidates for each target loop was performed over segments from proteins unrelated to the target protein (i.e., such that the FASTA E-value for the alignment of the corresponding sequences is greater than 0.01). There are large variations in the quality of the best-fitting loops between target loops of the same length. For example, the five best-matching loops for the eight-residue loop at position 50-57 in 1BTL (a beta-lactamase from *E. Coli*) are better than 1.5 Å, whereas for the eight-residue loop at position 606-613 in 1GOF (an oxydoreductase from *dactylium dendroides*), none of the database-derived loops is better than 4.6 Å (see Figure 4). The native loop at position 50-57 of 1BTL resembles an antiparallel β-sheet, with a short end-to-end distance (7.48 Å between the C_α of residues 49 and 58). Our database search selects protein segments that consistently share the same geometry as the native loop. The native loop at position 606-516 of 1GOF, on the other hand, has a coil conformation, with an end-to-end distance of 14.2 Å. Many types of protein segments of

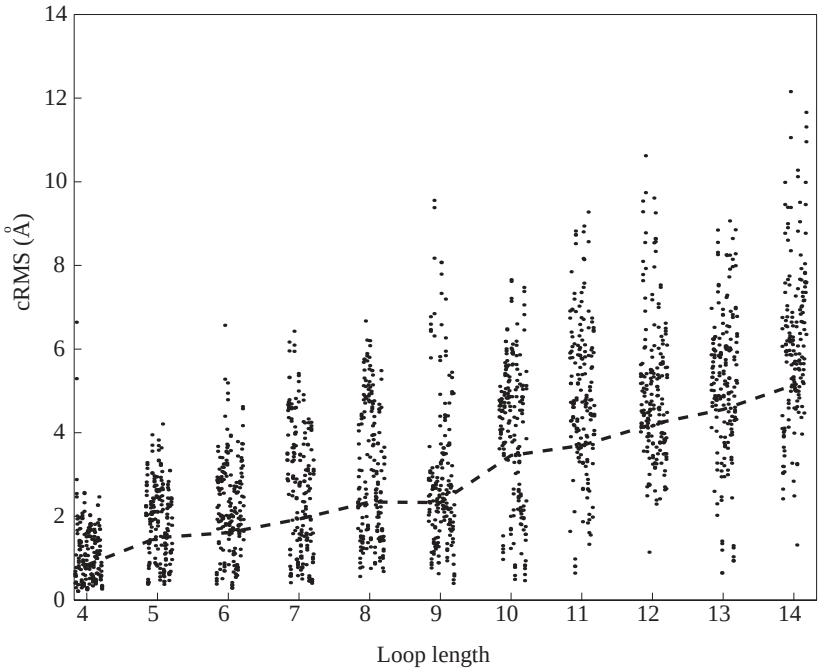

Fig. 3. For all 427 target loop searches, the five best template loops extracted from unrelated proteins are shown. Results are sorted according to loop length. Every column represents one target loop. The dashed line plots the average best cRMS values as a function of the loop length, where the average is computed over all loops of a given length.

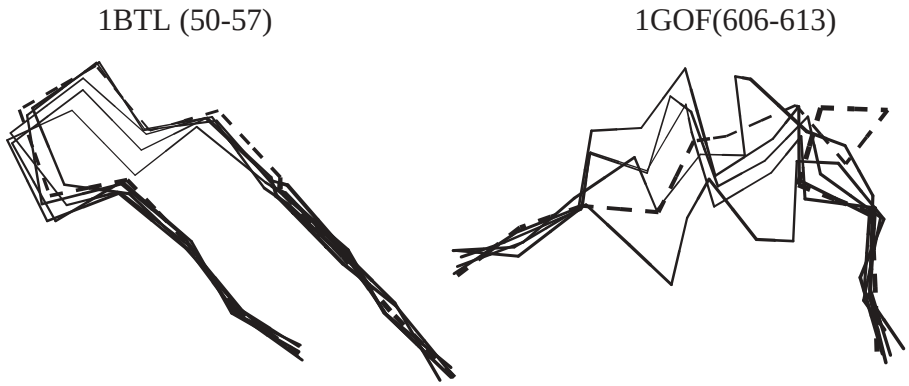

Fig. 4. Modeling of medium-sized loops (eight residues in length). For both proteins, the native conformation of the loop is shown as a dashed line, and the C_α trace of the five best template loops are shown as solid lines. The cRMS of the five best template loops for 1BTL range from 0.56 to 1.41 Å, while the five best template loops for 1GOF are spread between 4.5 and 5.76 Å. This figure was generated with MOLSCRIPT (Kraulis 1991).

length eight residues could fit in this gap. The diversity of the selected template loops is larger, and these loops are of lower quality than those found for 1BTL.

Figure 3 shows that we cannot expect to find reliably template loops with cRMS better than 3 Å for loops of more than nine residues. This result is consistent with the analysis of PDB-based protein loop prediction by vanVlijmen and Karplus (1997), and highlights the weaknesses of methods that search for long loops in the PDB.

3.2. Generating Loop Conformations Using Short Protein Building Blocks

We have shown in a recent study that it is possible to generate good approximations of protein structures using small libraries of small protein fragments (Kolodny et al. 2002). For example, we were able to construct models for a set of 145 structurally diverse proteins (Park and Levitt 1995) such that the average cRMS distance (computed over the C_α trace) between the test set structures and their approximations varies from 1.85 Å for a library of 20 fragments of five residues to 0.99 Å for a library of 100 fragments of length five residues. In this subsection, we assess the performance of a modified version of this approach applied to the loop building problem on the 427 target loops of our test sets. We used libraries of fragments of length five residues, each containing between 20 and 100 such fragments. Loops were constructed using strategy B, which generates all possible half-chains starting simultaneously from each side of the target loops, retaining those half-chains that meet at their end points (see Section 2). Figure 5 shows the average cRMS of the best-fitting fragment-based loop computed over all target loops of a given length l, as a function of l, for different libraries of fragments. Table 1 gives the range of cRMS for best-fitting loops over all target loops of a given length. Finally, Table 2 shows the average number of fragment-based loops generated within 3 and 4 Å of their target loops, where the average is computed over all target loops of the same length.

It was not possible to generate fitting loops for all combinations of library size and loop length. For short loops and small fragment libraries, most if not all candidate chains do not achieve loop closure, within the 1 Å tolerance we have set (see Section 2). For example, we can only build loops of four residues with libraries of 60 or more fragments. Figure 5 shows that the quality of the fragment-based loops improves as the size L of the fragment library increases. This improvement, however, comes at a cost. For long loops, the use of large libraries results in combinatorial explosion, as the total number of candidate loops generated by this procedure is a high-order function of L (see eq. (2)).

For very short loops ($l < 6$), the database approach described in Section 3.1 selects better loops than those built from the fragments (see Figure 5). This result is not surprising, as the database search provides a better sampling for these short loops than the fragment building procedure. Conversely, the fragment-based approach to loop building performs much better than the database approach for loops larger than six residues, and this occurs even for small fragment libraries. Table 1 shows also that systematic sampling provides more consistent quality of the best-fitting loops between target loops of the same length. For example, we have seen (Figure 4) that a database search finds native-like loops for the target loop 1BTL(50-57), with a best cRMS of 0.56 Å, but fails on the target loop 1GOF(606-613), in which case the cRMS of the best loop is 4.74 Å. In comparison, the best loops found by systematic sampling using the library of 80 fragments for the same targets have cRMS values with respect to the corresponding native loops of 1.2 and 1.44 Å, respectively.

Table 2 illustrates that even with libraries of small size, the systematic search reliably builds candidate loops within 3 Å of their target, for loop length up to 14 residues. The corresponding computing times are given in Table 3.

4. Discussion

Loop closure is an essential element of many protein structure prediction problems. For example, it is nearly always needed in homology modeling, where the framework for the structure of the target protein is derived from the highly homologous regions in a template protein, leaving gaps between these regions that need to be filled in with protein segments (see Figure 1). It is very reminiscent of the inverse kinematics problem, with many occurrences in the field of robotics. Analytical solutions to the inverse kinematics problem exist for kinematic chains with six degrees of freedom or less (Go and Scheraga 1970; Bruccoleri and Karplus 1985; Palmer and Scheraga 1991; Manocha and Canny 1994; Manocha, Zhu, and Wright 1995; Wedemeyer and Scheraga 1999). Unfortunately, these methods do not extend to longer chains. In this paper, we have described a new method for solving the loop closure problem for proteins, which combines a systematic search with database derived information on proteins.

The wealth of information available in the PDB, the depository of all protein structures experimentally determined, makes the protein loop closure problem different from standard inverse kinematics problem. The first loop building procedures designed to use this information search the PDB for loop candidates, based on geometric fitness criteria (Summers and Karplus 1990). This approach assumes that there exists at least one segment from a known protein structure that matches the target loop to be modeled. Fidelis et al. (1994) had shown that unfortunately this assumption was not valid for loops longer than four residues. The results described in Section 3.1 are considerably more promising, in that we show that a database approach performs well for loops up to nine residues long. For example, from the data shown in Figure 3 we see that for 80% of the target loops of length 9, we can find

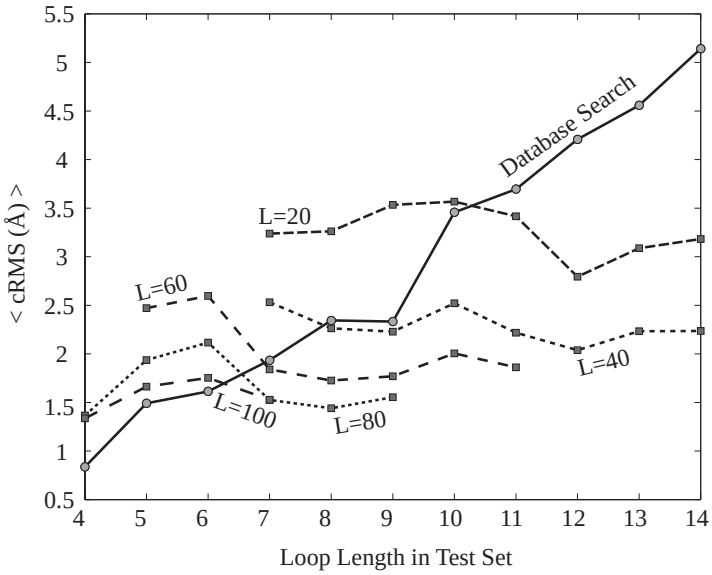

Fig. 5. A fragment-based approach to the loop building problem. Candidate loops for each of the 427 target loops in our data set were built using small libraries of protein fragments of five residues. The quality of each candidate loop is defined as its cRMS distance to the target loop, computed over all C_α atoms of the loops after superposition of the flanking regions of the native and candidate loops. For each target loop T, the cRMS of the best-fitting loop is stored as best(T). The average of best(T) computed over all target loops of length l provides a measure of the performance of the fragment-based approach for generating loops of length l. We denote this average as ⟨cRMS⟩. We plot ⟨cRMS⟩ as a function of loop length, for different library size L. For comparison, the solid line shows the performance of the database approach described in Section 3.1.

Table 1. Range of cRMS (Å) for the Best Fragment-Based Loops

Loop	Number of Fragments in Library (L)					
Length	20	40	60	80	100	DB
4	NA	NA	NA	0.31–4.75	0.32–4.21	0.21–2.47
5	NA	NA	0.93–4.87	0.54–4.66	0.28–3.02	0.38–3.33
6	NA	NA	0.46–4.60	0.34–3.90	0.38–3.42	0.28–3.51
7	0.43–4.84	0.56–4.49	0.45–2.85	0.34–2.54	0.41–2.69	0.40–4.79
8	1.15–4.97	0.53–3.89	0.47–2.75	0.37–2.45	NA	0.57–4.74
9	1.51–4.98	0.76–3.44	0.75–2.68	0.82–2.62	NA	0.40–6.58
10	1.92–4.95	1.01–3.55	0.82–2.68	NA	NA	0.47–6.45
11	1.78–4.89	1.29–3.52	1.14–2.51	NA	NA	0.65–6.90
12	1.41–4.18	1.05–2.83	NA	NA	NA	1.14–8.33
13	2.17–4.72	1.04–3.50	NA	NA	NA	0.65–7.37
14	2.15–4.28	1.46–3.10	NA	NA	NA	1.32–7.66

Table 2. Average Number of Fragment-Based Loops Within 3 Å (4 Å) of Their Target Loop

Loop	Number of Fragments in Library (L)				
Length	20	40	60	80	100
4	NA	NA	NA	40 (57)	66 (91)
5	NA	NA	4 (7)	12 (26)	36 (71)
6	NA	NA	1.7 (3.9)	5.6 (14)	14 (32)
7	0.3 (0.3)	2.5 (6.7)	18 (46)	56 (162)	123 (368)
8	1.3 (3.8)	27 (108)	258 (1016)	1354 (5355)	NA
9	0.8 (2.6)	14 (82)	163 (800)	845 (4358)	NA
10	0.3 (1.3)	4.2 (34)	49 (339)	NA	NA
11	0.5 (4.1)	26 (220)	203 (1640)	NA	NA
12	6.8 (53)	420 (4448)	NA	NA	NA
13	1.7 (20)	129 (2135)	NA	NA	NA
14	0.8 (13)	63 (1257)	NA	NA	NA

Table 3. Average CPU Time Needed to Reconstruct One Loop Using the Fragment-Based Approach

Loop	Number of fragments in library (L)				
Length	20	40	60	80	100
4	NA	NA	NA	0.0303	0.0435
5	NA	NA	0.0202	0.0307	0.0448
6	NA	NA	0.0210	0.0320	0.0462
7	0.0182	0.0818	0.0973	0.1525	0.1847
8	0.0359	0.0896	0.1639	0.3534	NA
9	0.0382	0.1074	0.1850	0.4415	NA
10	0.0376	0.1022	0.1753	NA	NA
11	0.0778	0.3803	2.2904	NA	NA
12	0.1387	2.6720	NA	NA	NA
13	0.1439	3.5776	NA	NA	NA
14	0.1579	4.9761	NA	NA	NA

Computing times are given in min, on a PC with Xeon processor at 2.8 GHz, running Linux.

at least one template loop with a cRMS better than 3 Å. This number drops to 30% for loops of length 10, and to 12% for loops of length 14. These results are consistent with the results of vanVlijmen and Karplus (1997). In a recent study, Du, Andrec, and Levy (2003) have shown that for a protein chain of 15 residues, there is a 91% probability of finding a nonhomologous protein segment in the PDB within 2 Å cRMS. Based on these results, they concluded that the database approach to loop building should perform well up to 15 residues. We argue that, in fact, their results are similar to our results for loops of nine residues. The search in the PDB of candidate segments that fit in a gap of a protein is successful when the segment found is structurally similar to the native conformation of the target loop, and if the flanking regions of the segment in the protein to which it belongs matches the stem regions on both sides of the gap. The second condition defines the orientation of the segment in the framework. It is the basis of the geometric filter applied to candidate segments (see Section 2). A successful segment for a loop of nine residues must therefore match the target loop and its stem regions, giving a total of 15 residues in our procedure.

Improvements of the PDB segment search approaches to the protein loop prediction problem compared to the study of Fidelis et al. (1994) can undoubtedly be assigned to the much larger protein structure database currently available. It is not clear however that further significant progress can be expected in the future, even with the steady increase in size of these databases. For example, our results based on the current database are not better than those of vanVlijmen and Karplus (1997), which were obtained based on the smaller

databases available six years ago (we apply the geometric filter for selecting loops used by vanVlijmen and Karplus). The size of the conformational space available to a protein loop grows exponentially with the number of degrees of freedom that define the loop. For long loops (i.e., longer than nine residues), this length might be too large to be sampled properly by the PDB, even if we were to multiply its size by an order of magnitude. The results described in Section 3.2 concerning our fragment-based approach to the loop building problem indicate that there is in fact no need to wait for much better statistics in the distribution of long protein segments. The results reported by Du, Andrec, and Levy (2003) indicate that the current PDB provides a good sampling of the conformations of protein fragments of five residues. In a previous study, we have shown that these conformations can be clustered into small libraries of fragment representatives, and that these libraries can be used to model with adequate accuracy most protein structures (Kolodny et al. 2002). In this paper, we have used these libraries to build protein loops. As these libraries contain only a small number of fragments, it is possible to enumerate exhaustively all loop conformations obtained by concatenating these fragments. The ability of these libraries to generate satisfactory models for complete proteins (Kolodny et al. 2002). extends to the generation of satisfactory models of protein loops (see Figure 5). The size of the library is an important parameter of our approach. It controls the trade-off between feasibility and accuracy. Loop building based on libraries with a large number of fragments generates accurate loops of length up to eight residues. For longer loops, the procedure is very costly in computing time. We have shown, however, that smaller libraries still provide satisfactory loops, at a very reasonable computing cost (see Tables 2 and 3).

The libraries of protein fragments can be replaced by libraries of set of dihedral angles. Our strategy involves the superposition of the first three C_α atoms of the newly added fragment onto the last three C_α atoms of the chain already constructed. This is equivalent to adding two residues to the chain each time a new fragment is added. As such, our method bears similarity with the PDB-based loop prediction method of Sudarsanam et al. (1995), which is based on a ϕ_{i+1}, ψ_i dimer database. Sudarsanam et al. did not cluster their large database of dimers, and consequently could not perform exhaustive construction of candidate loops. They report results on the construction of short target loops of five residues, which are comparable in quality to those reported here.

There are two technical aspects of the dimer method that are worth discussing in relation to our work. First, the database of ϕ_{i+1}, ψ_i dimers was binned into 400 categories, corresponding to all possible amino acid pairs. In contrast, we did not generate sequence specific libraries for two reasons. There are no indications that short protein sequences adopt the same conformation in the different proteins in which they occur; in fact, there is evidence of the converse for fragments up to nine residues (Mezei 1998), and more specific attempts to classify loop conformations have not yet been successful in finding a method of predicting a loop conformation based on its sequence (Ring et al. 1992). Secondly, the database of ϕ_{i+1}, ψ_i dimers was built selectively from residues belonging to loop regions in proteins, removing all residues that belong to an α-helix or β-sheet. We have tried a similar approach by designing loop-specific fragment libraries, which are used in turn to generate candidate loops that can fit in a gap in a protein framework. These structure-specific libraries however did not perform better than the general libraries used above (results not shown).

Overall, we have presented a new method for solving the protein structure analogue of the inverse kinematics problem, predicting protein loop conformation. We generate all possible loop conformations that satisfy the loop closure criteria, using libraries of small protein fragments of length five residues. Our approach has the advantages of *ab initio* methods since we are able to enumerate all candidate loops in the discrete approximation of the conformational space accessible to the target loop. It also has the advantages of a database search approach, as our use of fragments of known protein structures guarantees that the backbone conformations are physically reasonable. By varying the size of the library of fragments used to build the loops, we control the trade-off between accuracy and feasibility in terms of computing time. Previous database approaches to the loop prediction problem were limited to loops of up to nine residues. Results from this study are more optimistic, and extend the application of the database-based approach to loops of 14 residues.

Acknowledgments

We are grateful to Jack Snoeyink for helpful discussions. This work was supported in part by National Science Foundation Grant CCR-00-86013. ML and PK also acknowledge support from the National Institute of Health (GM 63817).

References

Altschul, S. F., Gish, W., Myers, E., Miller, W., and Lipman, D. J. 1990. A basic local alignment search tool. *Journal of Molecular Biology* 215:403–410.

Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N., and Bourne, P. E. 2000. The protein data bank. *Nucleic Acids Research* 28:235–242.

Bernstein, F. C., Koetzle, T. F., Williams, G. J. B., Meyer, E. F., Brice, M. D., Rodgers, J. R., Kennard, O., Shimanouchi, T., and Tasumi, M. 1977. Protein data bank – computer-based archival file for macromolecular structures. *Journal of Molecular Biology* 112:535–542.

Bonneau, R., and Baker, D. 2001. *Ab initio* protein structure prediction: progress and prospects. *Annual Review of Biophysics and Biomolecular Structure* 2:173–189.

Brenner, S. E., Koehl, P., and Levitt, R. 2000. The astral compendium for protein structure and sequence analysis. *Nucleic Acids Research* 28:254–256.

Browne, W. J., North, A. C. T., Philipps, D. C., Brew, K., Vanaman, T. C., and Hill, R. C. 1969. A possible three-dimensional structure of bovine α-lactalbumin based on that of hen's egg-white lysozyme. *Journal of Molecular Biology* 42:65–86.

Bruccoleri, R. E., and Karplus, M. 1985. Chain closure with bond angle variations. *Macromolecules* 18:2767–2773.

Bruccoleri, R. E., and Karplus, M. 1990. Conformational sampling using high-temperature molecular-dynamics. *Biopolymers* 29:1847–1862.

Canutescu, A. A., and Dunbrack, R. L. 2003. Cyclic coordinate descent: a robotics algorithm for protein loop closure. *Protein Science* 12:963–972.

Carlacci, L., and Englander, S. W. 1993. The loop problem in proteins – a Monte Carlo simulated annealing approach. *Biopolymers* 33:1271–1286.

Chandonia, J. M., Walker, N. S., Conte, L. L., Koehl, P., Levitt, M., and Brenner, S. E. 2002. Astral compendium enhancements. *Nucleic Acids Research* 30:260–263.

Collura, V., Higo, J., and Garnier, J. 1993. Modeling of protein loops by simulated annealing. *Protein Science* 2:1502–1510.

Deane, C. M., and Blundell, T. L. 2000. A novel exhaustive search algorithm for predicting the conformation of polypeptide segments in proteins. *Proteins: Structure, Function and Genetics* 40:135–144.

Deo, A. S., and Walker, I. D. 1995. Overview of damped least-squares methods for inverse kinematics of robot manipulators. *Journal of Intelligent Robotics Systems* 14:43–68.

Du, P. C., Andrec, M., and Levy, R. M. 2003. Have we seen all structures corresponding to short protein fragments in the protein data bank? an update. *Protein Engineering* 16:407–414.

Dudek, M. J., and Scheraga, H. A. 1990. Protein-structure prediction using a combination of sequence homology and global energy minimization, 1. Global energy minimization of surface loops. *Journal of Computational Chemistry* 11:121–151.

Fidelis, K., Stern, P. S., Bacon, D., and Moult, J. 1994. Comparison of systematic search and database methods for constructing segments of protein-structure. *Protein Engineering* 7:953–960.

Fine, R. M., Wang, H., Shenkin, P. S., Yarmush, D. L., and Levinthal, C. 1996. Predicting antibody hyper-variable loop conformations, II. Minimization and molecular dynamics studies of mcp603 from many randomly generated loop conformations. *Proteins: Structure, Function and Genetics* 1:342–362.

Finkelstein, A. V., and Reva, B. A. 1992. Search for the stable state of a short chain in a molecular-field. *Protein Engineering* 5:617–624.

Fiser, A., Do, R. K. G., and Sali, A. 2000. Modeling of loops in protein structures. *Protein Science* 9:1753–1773.

Go, N., and Scheraga, H. A. 1970. Ring closure and local conformational deformation of chain molecules. *Macromolecules* 3:178–186.

Jones, T. A., and Thirup, S. 1986. Using known substructures in protein model-building and crystallography. *EMBO Journal* 5:819–822.

Kabsch, W. 1976. Solution for best rotation to relate 2 sets of vectors. *Acta Crystallographica, Section A* 32:922–923.

Khatib, O., Yokoi, K., Chang, K., Ruspini, D., Holmberg, R., and Casal, A. 1996. Vehicle/arm coordination and multiple mobile manipulator decentralized cooperation. *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Minneapolis, MN, pp. 546–553.

Kolodny, R., Koehl, P., Guibas, L., and Levitt, M. 2002. Small libraries of protein fragments model native protein structures accurately. *Journal of Molecular Biology* 323:297–307.

Kotay, K., Rus, D., Vona, M., and McGray, C. 1998. The self-reconfiguring robotic molecule: design and control algorithms. *Proceedings of the 3rd International Workshop on Algorithmic Foundations of Robotics*, Houston, TX, March 5–7 pp. 375–386.

Kraulis, P. J. 1991. Molscript: a program to produce both detailed and schematic plots of protein structures. *Journal of Applied Crystallography* 24:946–950.

Lavalle, S. M., Finn, P. W., Kavraki, L. E., and Latombe, J. C. 2000. A randomized kinematics-based approach to pharmacophore-constrained conformational search and database screening. *Journal of Computational Chemistry* 21:731–747.

McGarrah, D. B., and Judson, R. S. 1993. Analysis of the genetic algorithm method of molecular conformation determination. *Journal of Computational Chemistry* 14:1385–1395.

Manocha, D., and Canny, J. F. 1994. Efficient inverse kinematics for general 6r manipulators. *IEEE Transactions on Robotics and Automation* 10:648–657.

Manocha, D., and Zhu, Y. S. 1994. Kinematic manipulation of molecular chains subject to rigid constraints. *Proceedings of the 2nd International Conference on Intelligent Systems for Molecular Biology*, Stanford, CA, Vol. 2, pp. 285–293.

Manocha, D., Zhu, Y. S., and Wright, W. 1995. Conformational-analysis of molecular chains using nano-kinematics. *Computer Application of Biological Sciences* 11:71–86.

Mezei, M. 1998. Chameleon sequences in the PDB. *Protein Engineering* 11:411–414.

Moult, J., and James, M. N. N. 1986. An algorithm for determining the conformation of polypeptide segments in protein by systematic search. *Proteins: Structure, Function and Genetics* 1:146–163.

Moult, J., Fidelis, K., Zemla, A., and Hubbard, T. 2003. Crit-

ical assessment of methods of protein structure prediction (casp) round v. *Proteins: Structure, Function and Genetics* 53:334–339.

Murzin, A. G., Brenner, S. E., Hubbard, T., and Chothia, C. 1995. Scop – a structural classification of proteins database for the investigation of sequences and structures. *Journal of Molecular Biology* 247:536–540.

Nielsen, J., and Roth, B. 1999. On the kinematic analysis of robotic mechanisms. *International Journal of Robotics Research* 18:1147–1160.

Palmer, K. A., and Scheraga, H. A. 1991. Standard-geometry chains fitted to X-ray derived structures: validation of the rigid geometry approximation, I. Chain closure through a limited search of "loop" conformations. *Journal of Computational Chemistry* 12:505–526.

Park, B. H., and Levitt, M. 1995. The complexity and accuracy of discrete state models of protein structure. *Journal of Molecular Biology* 249:493–507.

Park, J. Y., Chang, P. H., and Yang, J. Y. 2003. Task-oriented design of robot kinematics using the grid method. *Advanced Robotics* 17:879–907.

Ring, C. S., and Cohen, F. E. 1994. Conformational sampling of loop structures using genetic algorithms. *Israel Journal of Chemistry* 34:245–252.

Ring, C. S., Kneller, D. G., Langridge, R., and Cohen, F. E. 1992. Taxonomy and conformational-analysis of loops in proteins. *Journal of Molecular Biology* 224:685–699.

Rosenbach, D., and Rosenfeld, R. 1995. Simultaneous modeling of multiple loops in proteins. *Protein Science* 4:496–505.

Shenkin, P. S., Yarmush, D. L., Fine, R. M., Wang, H. J., and Levinthal, C. 1987. Predicting antibody hypervariable loop conformation, 1. Ensembles of random conformations for ring-like structures. *Biopolymers* 26:2053–2085.

Subbiah, S., Laurents, D. V., and Levitt, M. 1993. Structural similarity of DNS-binding domains of bacte-riophage repressors and the globin core. *Current Biology* 3:141–148.

Sudarsanam, S., Dubose, R. F., March, C. J., and Srinivasan, S. 1995. Modeling protein loops using a $\phi - i + 1, \psi - i$ dimer database. *Protein Science* 4:1412–1420.

Summers, N. L., and Karplus, M. 1990. Modeling of globular-proteins – a distance-based data search procedure for the construction of insertion deletion regions and pro reversible non-pro mutations. *Journal of Molecular Biology* 216:991–1016.

Tolani, D., Goswami, A., and Badler, N. I. 2000. Real-time inverse kinematics techniques for anthropomorphic limbs. *Graphical Models* 62:353–388.

Vajda, S., and DeLisi, C. 1990. Determining minimum energy conformations of polypeptides by dynamic-programming. *Biopolymers* 29:1755–1772.

vanVlijmen, H. W. T., and Karplus, M. 1997. PDB-based protein loop prediction: parameters for selection and methods for optimization. *Journal of Molecular Biology* 267:975–1001.

Wampler, C. W. 2004. Displacement analysis of spherical mechanisms having three or fewer loops. *Journal of Mechanical Design* 126:93–100.

Wang, L. T., and Chen, C. C. 1991. A combined optimization method for solving the inverse kinematics problem of mechanical manipulators. *IEEE Transactions on Robotics and Automation* 7:489–499.

Wang, Y. F., and Chirikjian, G. S. 2004. Workspace generation of hyper-redundant manipulators as a diffusion process on se(n). *IEEE Transactions on Robotics and Automation* 20:399–408.

Wedemeyer, W. J., and Scheraga, H. A. 1999. Exact analytical loop closure in proteins using polynomial equations. *Journal of Computational Chemistry* 20:819–844.

Zhao, J. M., and Badler, N. I. 1994. Inverse kinematics positioning using nonlinear programming for highly articulated figures. *ACM Transactions on Graphics* 13:313–336.

Zheng, Q., and Kyle, D. J. 1996. Accuracy and reliability of the scaling relaxation method for loop closure: an evaluation based on extensive and multiple copy conformational samplings. *Proteins: Structure, Function and Genetics* 24:209–217.

Zheng, Q., Rosenfeld, R., Vajda, S., and DeLisi, C. 1993a. Loop closure via bond scaling and relaxation. *Journal of Computational Chemistry* 14:556–565.

Zheng, Q. A., Rosenfeld, R., Vajda, S., and DeLisi, C. 1993b. Determining protein loop conformation using scaling-relaxation techniques. *Protein Science* 2:1242–1248.

Zheng, Q., Rosenfeld, R., DeLisi, C., and Kyle, D. J. 1994. Multiple copy sampling in protein loop modeling—computational-efficiency and sensitivity to dihedral angle perturbations. *Protein Science* 3:493–506.