Protein Loop Closure using Orientational Restraints from NMR Data*

Chittaranjan Tripathy[1], Jianyang Zeng[1], Pei Zhou[2], and Bruce Randall Donald[1,2,**]

[1] Department of Computer Science, Duke University, Durham, NC 27708
[2] Department of Biochemistry, Duke University Medical Center, Durham, NC 27710

Abstract. Protein loops often play important roles in biological functions such as binding, recognition, catalytic activities and allosteric regulation. Modeling loops that are biophysically sensible is crucial to determining the functional specificity of a protein. A variety of algorithms ranging from robotics-inspired inverse kinematics methods to fragment-based homology modeling techniques have been developed to predict protein loops. However, determining the 3D structures of loops using global orientational restraints on internuclear vectors, such as those obtained from residual dipolar coupling (RDC) data in solution Nuclear Magnetic Resonance (NMR) spectroscopy, has not been well studied. In this paper, we present a novel algorithm that determines the protein loop conformations using a minimal amount of RDC data. Our algorithm exploits the interplay between the sphero-conics derived from RDCs and the protein kinematics, and formulates the loop structure determination problem as a system of low-degree polynomial equations that can be solved exactly and in closed form. The roots of these polynomial equations, which encode the candidate conformations, are searched systematically, using efficient and provable pruning strategies that triage the vast majority of conformations, to enumerate or prune all possible loop conformations consistent with the data. Our algorithm guarantees completeness by ensuring that a possible loop conformation consistent with the data is never missed. This data-driven algorithm provides a way to assess the structural quality from experimental data with minimal modeling assumptions. We applied our algorithm to compute the loops of human ubiquitin, the FF Domain 2 of human transcription elongation factor CA150 (FF2), the DNA damage inducible protein I (DinI) and the third IgG-binding domain of Protein G (GB3) from experimental RDC data. A comparison of our results versus those obtained by using traditional structure determination protocols on the same data shows that our algorithm is able to achieve higher accuracy: a 3- to 6-fold improvement in backbone RMSD. In addition, computational experiments on synthetic RDC data for a set of protein loops of length 4, 8 and 12 used in previous studies show that, whenever sparse RDCs can be measured,

* This work is supported by the following grants from National Institutes of Health: R01 GM-65982 to B.R.D. and R01 GM-079376 to P.Z.

** *Corresponding author*: Bruce Randall Donald, ✉brd+recomb11@cs.duke.edu, ☎919-660-6583, 🖷919-660-6519

our algorithm can compute longer loops with high accuracy. These results demonstrate that our algorithm can be successfully applied to compute loops with high accuracy from a limited amount of NMR data. Our algorithm will be useful to determine high-quality complete protein backbone conformations, which will benefit the nuclear Overhauser effect (NOE) assignment process in high-resolution protein structure determination.

1 Introduction

Protein loops are the segments of polypeptide chain that connect two secondary structure elements (SSEs) such as α-helices or β-strands. In addition to serving as linkers between SSEs, loops often play crucial roles in protein folding and stability pathways, and in many other important biological functions such as binding, recognition, catalytic activities and allosteric regulation [42,7,55,27].

While the *global fold*, i.e., the conformations and orientations of the SSEs of a protein, can often be determined with high accuracy via traditional experimental techniques such as X-ray crystallography or nuclear magnetic resonance (NMR) spectroscopy, modeling loops that seamlessly close the gap between two consecutive SSEs by satisfying the geometric, biophysical, and data constraints remains a difficult and open problem. In X-ray crystallography, for instance, the disorder in a protein crystal can render interpretation of the resulting electron density for loops difficult. As a result, protein structures found in the Protein Data Bank (PDB) [3] often have missing loops or disordered loops. The problem of computing loops that are biophysically reasonable and geometrically valid is called the *loop closure problem*. Since its introduction four decades ago in the classic paper by Gō and Scheraga [26], the loop closure problem has been an active area of research. In fact, modeling of loops can be regarded as an *ab initio* protein folding problem at a smaller scale. It is also an important problem in *de novo* protein structure prediction. Therefore, solutions and algorithms for accurate modeling of loops are highly desirable for understanding of the physical-chemical principles that determine protein structure and function.

Exploring the conformation space of a protein loop to identify low energy loop conformations is a difficult computational problem. Methods to identify such loops include database search and homology modeling [64,60,20], *ab initio* methods based on the minimization of empirical molecular mechanics energy functions [22,54,30], and robotics-inspired inverse kinematics and optimization-based methods [16,40,69,17,8,31]. These techniques work in two phases: first, the protein conformation space is explored to find a set of candidate loop constructs, which are then evaluated in the second phase using an appropriate empirical energy function to select the most promising set of loops.

Database methods [64,60,20] identify a set of candidate loops from a library of fragments derived from a protein structure database such as the PDB [3] that fit the anchor residues on either end of a loop. These loops are further ranked using criteria such as the sequence homology and conformational energy. Since these methods heavily rely on the statistical diversity of the structure database, the accuracy of loop predictions depends on how well the loop is represented in the

database. However, in general, database methods suffer from limited sampling of the loop conformations by the fragments in the database.

Ab initio loop modeling methods sample the conformation space randomly or use robotics-based sampling algorithms to generate a large number of loop conformations. Loop closure and energy minimization are done by using methods such as random tweak [54,21], analytical loop closure techniques [17,69], molecular dynamics simulation [5], Markov Chain Monte Carlo (MCMC) simulated annealing [13,22], and other optimization techniques [30]. The accuracy of loop prediction here depends on the efficacy of the conformational space exploration techniques used, and on the quality and proper parameterization (e.g., implicit or explicit solvent effects) of the force field employed to evaluate the conformational energy. These algorithms are computationally expensive due to a large number of random moves accompanied by repeated energy computations.

The protein loop closure problem is an *inverse kinematics* (IK) problem in computational biology, i.e., given the poses of terminal anchor residues, it asks to find all possible values of the degrees of freedom (DOFs) (i.e., the dihedrals ϕ and ψ) for which the fragment connects both the anchor residues. This problem has been studied widely in robotics and biology [16,40,69,17,8,31]. Tri-peptide loop closure, for which the number of DOFs is six and exactly six geometric constraints are stipulated due to the closure criterion, can be solved analytically [69,17,39,11] using exact IK solvers to give at most 16 possible solutions. For longer loops, the loop closure problem is underconstrained, so a continuous family of solutions are possible without additional constraints. Optimization-based IK solvers such as random tweak [54,21], and the cyclic coordinate descent (CCD) algorithm [8] have been successful in dealing with a large number of DOFs, and have found many applications [52,29,65]. These methods iteratively solve for the DOFs until the loop closure constraints are satisfied. However, the problem of loop closure subject to orientational restraints (e.g., from NMR data) has not been studied rigorously in the robotics or computational biology literature, and no practical deterministic algorithm exists to our knowledge.

Protein structure determination using nuclear Overhauser effect (NOE) distance restraints is NP-hard [50]. Traditional protein structure determination from solution NMR data starts with an elongated polypeptide backbone chain, and uses NOEs and dihedral angle restraints in a simulated annealing/simplified molecular dynamics (SA/MD) protocol [12,28,41,32,51] to compute the protein structure. Residual dipolar coupling (RDC) restraints are only incorporated in the final stages of the structure computation to refine the structures [6,51]. NOE-based structure determination protocols are known to be prone to local minima or lead to wrong convergence. To overcome the shortcomings of NOE-based methods, approaches in [18,46,4,56,25,1] have been proposed that primarily use RDC data, which provides precise global orientational restraints on internuclear vector orientations, to determine protein backbone structure. However, most of these approaches employ stochastic search, and therefore lack any algorithmic guarantee on the quality of the solution or running time. In recent work from our lab [66,68,19,71], polynomial-time algorithms have been proposed for high-

resolution backbone global fold determination from a minimal amount of RDC data. These algorithms represent the RDC equations and protein kinematics in algebraic form, and use exact methods in a divide-and-conquer framework to compute the global fold. In addition, these algorithms use a sparse set of RDC measurements (e.g., only two RDCs per residue), with the goal of minimizing the number of NMR experiments, hence the time and cost to perform them.

A high-resolution protein backbone is often a starting point for structure-based protein design [23,10,24,35]. An accurate backbone structure facilitates the assignment of NOESY spectra (i.e., *NOE assignment*), which is a prerequisite for high-resolution structure determination protocols, including side-chains. For example, the algorithms in [66,68,19,71] have been used in [67,73,72] to develop new algorithms for NOE assignment, based on which in [72] we recently developed a new framework for high-resolution protein structure determination, which was used prospectively to solve the solution structure of the FF Domain 2 of human transcription elongation factor CA150 (FF2) (PDB id: 2kiq). The global folds obtained by [66,68,19,71] have all the loops missing which requires a new algorithm that can compute the missing loops from RDCs. To alleviate this problem, a heuristic local minimization approach [51] for loops was used in [72].

In this paper, we give a solution to the loop closure problem. We present an efficient deterministic algorithm, POOL, that computes the missing loops from RDC data. Our algorithm exploits the interplay between protein backbone kinematics and the global orientational restraints derived from RDC data to naturally discretize the conformation space by polynomial-root solutions, and represents the candidate conformations using a tree. A systematic depth-first search of the conformation tree is used to enumerate all possible loop conformations that are consistent with the data. POOL uses efficient pruning strategies (Section 2.6) capable of pruning the majority of the conformations that are provably not part of a valid loop, thereby achieving a huge reduction in the search space. Unlike other algorithms, e.g. [4], that attempt to compute backbone structure using as many as 15 RDCs per residue recorded in two alignment media, our algorithm uses as few as 2 RDCs per residue in one alignment medium, which is often experimentally feasible. As we will show in Section 3.2, when given the same data, our algorithm performs better than traditional SA/MD-based approaches, e.g., [51]. Additional RDCs, and other data that provide constraints in torsion-angle space (e.g., TALOS [14,53] dihedral restraints) or in Euclidean space (e.g., sparse NOEs), whenever available, can directly be incorporated into our algorithm. In summary, we make the following contributions in this paper:

1. Derivation of quartic equations for backbone dihedrals ϕ and ψ from experimentally-recorded RDC sphero-conics and backbone kinematics, that can be solved *exactly* and in closed form;
2. Systematic search of the roots of the polynomial equations that encode the conformations, using efficient pruning methods to prune the vast majority of conformations;
3. Design and implementation of an efficient algorithm to determine the loop conformations from a limited amount of experimental RDC data;

Table 1. A ϕ-defining RDC is used to compute the backbone dihedral ϕ, and a ψ-defining RDC is used to compute the backbone dihedral ψ exactly and in closed form.

ϕ-defining RDC	C^{α}-H^{α}, C^{α}-C', C^{α}-C^{β}
ψ-defining RDC	N-H^{N}, C'-N, C'-H^{N}

4. Promising results from the application of our algorithm both on experimental NMR data sets for four proteins, and on simulated data sets for protein loops studied previously in [36,17,8].

2 Theory and Methods

2.1 Overview

POOL solves the following loop closure problem. Let the residues of the protein be numbered from 1 to n (from N- to C-terminus). Suppose the global fold of the protein has been determined from RDCs in a principal order frame (POF) of RDCs, as we showed was feasible in [66,68,19,72,71]. In principle, the global fold of proteins could also be computed using protein structure prediction [2], or homology modeling [33,34]; alternatively, X-ray structures (with missing loops) can be used. Given two consecutive SSEs with n_1 and n_2 being the last residue of the first SSE and first residue of the second SSE, respectively, the missing loop $[n_1, n_2]$ is defined as the fragment between residues n_1 and n_2 with both end residues included. The residues n_1 and n_2 that are part of the SSEs will be called the *stationary anchors*, and those of a candidate loop will be called the *mobile anchors*. We assume that the n_1 mobile anchor of the loop is attached to the n_1 stationary anchor of the first SSE. Then the loop closure problem is stated as follows: in the POF, given the poses of the stationary anchors n_1 and n_2 (points in $\mathbb{R}^3 \times SO(3)$), compute a complete set of conformations of fragments $[n_1, n_2]$ so that n_2 mobile anchor of each fragment in the set assumes the pose of the stationary anchor n_2, while satisfying the RDC data and standard protein geometry.

Our algorithm builds upon the initial work from our lab [19,68,72,71], where the authors developed polynomial time algorithms to compute high-resolution backbone global fold *de novo* from N-H^{N} and C^{α}-H^{α} RDCs in one alignment medium. These sparse-data algorithms have been extended to incorporate combinations of different types of RDCs (see Table 1) in one or two alignment media. The new generalized framework is called RDC-ANALYTIC [72,71]. POOL implements a novel algorithm to determine protein loop backbone structures from minimal amount of RDC data, and is a crucial addition to the RDC-ANALYTIC suite, which did not compute loops before.

Table 1 describes the RDC types that POOL uses to compute the backbone dihedrals exactly and in closed form (Section 2.3). A *ϕ-defining* RDC is used to compute the backbone dihedral ϕ, and a *ψ-defining* RDC is used to compute the backbone dihedral ψ. The input data to POOL include: (1) the global fold of the protein computed by [68,19,72]; (2) the alignment tensor, which generally can be computed from the global fold using [37,66]; (3) at least one ϕ-defining and one ψ-defining RDCs per residue, and optionally other data, e.g., TALOS [14,53] dihedral restraints and sparse NOEs; and (4) the primary sequence of the protein.

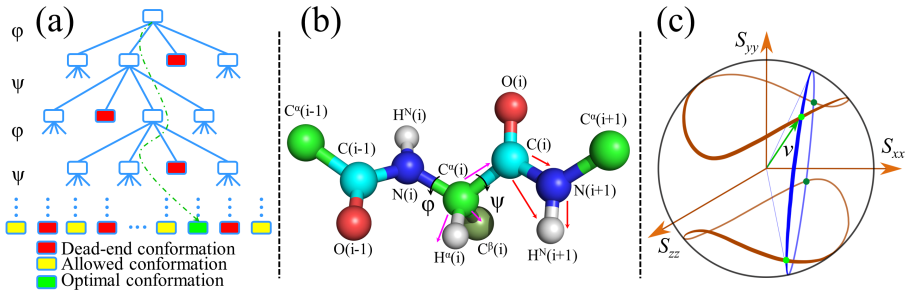

Fig. 1. (a) An example conformation tree. (b) The internuclear vectors (shown using arrows) for which RDCs are possible to measure. The magenta and red arrows represent ϕ-defining and ψ-defining RDCs, respectively. (c) The pringle-shaped RDC sphero-conic curves inscribed on a unit sphere constrain the internuclear vector $\mathbf{v}$ (green arrow) to lie on one of them. The kinematic circle (shown in blue almost edge-on) of $\mathbf{v}$ intersects the sphero-conic curves in at most four points (green dots) leading to a maximum of four possible orientations for the internuclear vector $\mathbf{v}$.

Solving a system of equations from RDCs, protein kinematics and loop closure constraints simultaneously is a difficult computational problem since it leads to solving a high-degree polynomial system. However, since RDCs are very precise measurements, an algorithm which is able to compute protein fragments by inductively solving low-degree polynomial equations derived from RDCs and backbone kinematics, and drives the computation to satisfy the loop closure criterion, will achieve the desired objective. Our algorithm POOL is based on this key insight. Starting from a stationary anchor, it solves each DOF sequentially using the equations derived in Sections 2.3 and 2.4. The discrete values of the DOFs computed from the polynomial roots, are represented by a conformation tree grown recursively as we solve for the DOFs progressively. An internal (i.e., non-leaf) node in the tree represents the conformation of a part of a candidate loop, and a leaf node represents a candidate loop conformation computed from RDCs. Figure 1 (a) illustrates a conformation tree for a loop. As each node is visited in a depth-first traversal of the tree, if the conformation represented by that node fails the conformation filters (Section 2.6), it is called a *dead-end* node, and the sub-tree rooted at that node is pruned. Dead-end nodes identified at lower levels (i.e., closer to the root) of the conformation tree prune more conformations than those identified at higher levels. Finally, all remaining unpruned conformations (leaf nodes) already close to the stationary anchor (since they satisfy the reachability criterion; see Section 2.6), are evaluated for loop closure. At this stage minimization techniques can be applied to improve the closure. Conformations satisfying the closure criterion are added to the final ensemble of loops. POOL enumerates all loop conformations that satisfy the RDC data and pass the conformation filters; therefore, it guarantees completeness.

2.2 RDC Sphero-Conics

The residual dipolar coupling r between two spin-$\frac{1}{2}$ nuclei a and b is given by

$$r = D_{\max} \mathbf{v}^T \mathbf{S} \mathbf{v}, \tag{1}$$

where $\mathbf{v}$ is the unit internuclear vector between a and b, $D_{\max}$ is the dipolar interaction constant, $\mathbf{S}$ is the *Saupe order matrix* [49], or *alignment tensor*, that specifies the ensemble-averaged anisotropic orientation of the protein in the laboratory frame. $\mathbf{S}$ is a 3×3 symmetric, traceless, rank 2 tensor with five independent elements [57,58,43,19]. The constant $D_{\max}$ is given by

$$D_{\max} = \frac{\mu_0 \hbar \gamma_a \gamma_b}{4\pi^2} \left\langle r_{ab}^{-3} \right\rangle, \tag{2}$$

where μ_0 is the magnetic permeability of vacuum, $\hbar$ is Planck's constant, γ_a and γ_b are the gyromagnetic ratios of the nuclei a and b, respectively, and $\left\langle r_{ab}^{-3} \right\rangle$ represents the vibrational ensemble-averaged inverse cube of the distance between the two nuclei. Letting $D_{\max} = 1$ (i.e., scaling the RDCs appropriately), and considering a global coordinate frame that diagonalizes the alignment tensor $\mathbf{S}$, often called the *principal order frame* (POF), Eq. (1) can be written as

$$r = S_{xx}x^2 + S_{yy}y^2 + S_{zz}z^2, \tag{3}$$

where S_{xx}, S_{yy} and S_{zz} are the three diagonal elements of a diagonalized alignment tensor $\mathbf{S}$, and x, y and z are, respectively, the x, y and z components of the unit vector $\mathbf{v}$ in a POF that diagonalizes $\mathbf{S}$. Since $\mathbf{v}$ is a unit vector, i.e.,

$$x^2 + y^2 + z^2 = 1, \tag{4}$$

an RDC constrains the corresponding internuclear vector $\mathbf{v}$ to lie on the intersection of a concentric unit sphere (Eq. (4)) and a quadric (Eq. (3)) [44]. This gives a pair of closed curves inscribed on the unit sphere that are diametrically opposite to each other (see Figure 1 (b), (c)). These curves are known as *sphero-conics* or *sphero-quartics* [9,47].

Using Eq. (4) in Eq. (3), we can rewrite Eq. (3) in the following form:

$$ax^2 + by^2 = c, \tag{5}$$

where $a = S_{xx} - S_{zz}$, $b = S_{yy} - S_{zz}$, and $c = r - S_{zz}$. Henceforth, we refer to Eq. (5) as the *reduced RDC equation*. For background on RDCs and RDC-based structure determination, the reader is referred to [57,58,43,19].

2.3 Analytic Solutions for Peptide Plane Orientations from ϕ-defining and ψ-defining RDCs in One Alignment Medium

The derivation below assumes standard protein geometry, which is exploited in the kinematics [66]. We choose to work in an orthogonal coordinate system defined at the peptide plane P_i with z-axis along the bond vector $\mathrm{N}(i) \rightarrow \mathrm{H}^{\mathrm{N}}(i)$, where the notation $a \rightarrow b$ means a vector from the nucleus a to the nucleus b. The y-axis is on the peptide plane i and the angle between y-axis and the bond vector $\mathrm{N}(i) \rightarrow \mathrm{C}^{\alpha}(i)$ is fixed. The x-axis is defined based on the right-handedness. Let $\mathbf{R}_{i,\mathtt{POF}}$ denote the orientation (rotation matrix) of P_i with respect to the POF. Then $\mathbf{R}_{1,\mathtt{POF}}$ denotes the relative rotation matrix between the coordinate system defined at the first residue of the current SSE and the principal order frame. $\mathbf{R}_{i,\mathtt{POF}}$ is used to derive $\mathbf{R}_{i+1,\mathtt{POF}}$ inductively after we compute the dihedral angles ϕ_i and ψ_i. $\mathbf{R}_{i+1,\mathtt{POF}}$, in turn, is used to compute the $(i+1)^{st}$ peptide plane.

Proposition 1. *Given the diagonalized alignment tensor components S_{xx} and S_{yy}, the peptide plane P_i, and a ϕ-defining RDC r for the corresponding internuclear vector of residue i, there exist at most 4 possible values of the dihedral angle ϕ_i that satisfy the RDC r. The possible values of ϕ_i can be computed exactly and in closed form by solving a quartic equation.*

Proof. Let the unit vector $\mathbf{v}_0 = (0, 0, 1)^T$ represent the N-H^N bond vector of residue i in the local coordinate frame defined on the peptide plane P_i. Let $\mathbf{v}_1 = (x, y, z)^T$ denote the internuclear vector for the ϕ-defining RDC for residue i in the principal order frame. We can write the forward kinematics relation between $\mathbf{v}_0$ and $\mathbf{v}_1$ as follows:

$$\mathbf{v}_1 = \mathbf{R}_{i,\texttt{POF}}\, \mathbf{R}_l\, \mathbf{R}_z(\phi_i)\, \mathbf{R}_r\, \mathbf{v}_0. \tag{6}$$

Here $\mathbf{R}_l$ and $\mathbf{R}_r$ are constant rotation matrices that describe the kinematic relationship between $\mathbf{v}_0$ and $\mathbf{v}_1$. $\mathbf{R}_z(\phi_i)$ is the rotation about the z-axis by ϕ_i.

Let c and s denote $\cos\phi_i$ and $\sin\phi_i$, respectively. Using this while expanding Eq. (6) we have

$$x = A_0 + A_1 c + A_2 s, \quad y = B_0 + B_1 c + B_2 s, \quad z = C_0 + C_1 c + C_2 s, \tag{7}$$

in which A_i, B_i, C_i for $0 \leq i \leq 2$ are constants. Using Eq. (7) in the reduced RDC equation Eq. (5) and simplifying we obtain

$$K_0 + K_1 c + K_2 s + K_3 cs + K_4 c^2 + K_5 s^2 = 0, \tag{8}$$

in which K_i, $0 \leq i \leq 5$ are constants. Using half-angle substitutions

$$u = \tan(\frac{\phi_i}{2}),\ c = \frac{1 - u^2}{1 + u^2},\ \text{and}\ s = \frac{2u}{1 + u^2} \tag{9}$$

in Eq. (8) we have

$$L_0 + L_1 u + L_2 u^2 + L_3 u^3 + L_4 u^4 = 0, \tag{10}$$

in which L_i, $0 \leq i \leq 4$ are constants.

Eq. (10) is a quartic equation which can be solved exactly and in closed form. Let $\{u_1, u_2, u_3, u_4\}$ denote the set of (at most) four real solutions of Eq. (10). For each u_i, the corresponding ϕ_i value can be computed using Eq. (9). □

Proposition 2. *Given the diagonalized alignment tensor components S_{xx} and S_{yy}, the peptide plane P_i, the dihedral ϕ_i, and a ψ-defining RDC r for the corresponding internuclear vector on peptide plane P_{i+1}, there exist at most 4 possible values of the dihedral angle ψ_i that satisfy the RDC r. The possible values of ψ_i can be computed exactly and in closed form by solving a quartic equation.*

Proof. The proof is provided in the supporting information (SI) **Appendix A** available online [61]. □

Proposition 3. *Given the diagonalized alignment tensor components S_{xx} and S_{yy}, the peptide plane P_i, a ϕ-defining RDC and a ψ-defining RDC for ϕ_i and ψ_i, respectively, there exist at most 16 orientations of the peptide plane P_{i+1} with respect to P_i that satisfy the RDCs.*

Proof. This follows directly from Proposition 1 and Proposition 2. □

2.4 Analytic Solutions for the ϕ Angle of Glycine from C^{α}-H^{α} RDC

The amino acid residue glycine (Gly) has two H^{α} atoms which we denote by H^{α_2} and H^{α_3}, respectively. The C^{α}-H^{α} RDC measured for Gly is the sum of the RDCs for these two bond vectors. We show that given the C^{α}-H^{α} RDC for a Gly residue we can compute all possible solutions for the dihedral ϕ.

Proposition 4. *Given the diagonalized alignment tensor components S_{xx} and S_{yy}, the peptide plane P_i, and the C^{α}-H^{α} RDC r for residue i which is a glycine, there exist at most 4 possible values of the dihedral angle ϕ_i that satisfy the C^{α}-H^{α} RDC r. The possible values of ϕ_i can be computed exactly and in closed form by solving a quartic equation.*

Proof. The proof is provided in the SI **Appendix B** available online [61]. □

2.5 Sampling the DOFs when RDCs are Missing

Theoretically, for a loop with n (> 6) DOFs, $n-6$ DOFs are redundant. Therefore, $n-6$ equality constraints are necessary to solve for the loop conformations so that the number of conformations is discrete. We systematically sample (at 5° resolution) the dihedrals from the Ramachandran map (and TALOS dihedral restraints if available) for the DOFs for which RDCs are missing, and use analytic equations to solve for the other dihedrals for which RDCs are available, to compute an ensemble of loops complete to the resolution of sampling. If RDCs can be recorded for the missing ones in a second alignment medium, POOL can use them (see the online SI **Appendix C** [61]). Table 2 shows that when as many as 5 RDCs are missing in a loop, POOL still could compute the loops accurately.

2.6 Pruning with Conformation Filters

Loop conformations are generated by traversing a conformation tree in a depth-first search order (Section 2.1). At each node, conformation filters are applied as *predicates*. If the node passes all the filters, then the subtree rooted at that node is visited; otherwise, the subtree is pruned. Failing a predicate at lower levels (closer to the root) of the conformation tree prunes more conformations than that detected at higher levels (farther from the root). In fact, pruning at depth i eliminates $O(b^{n-i})$ conformations, where b is the average number of branches in the conformation tree, and n is the height of the conformation tree. For loops with constrained work-space, substantial pruning can be achieved resulting in significant speedup. POOL uses the following conformation filters.

Real Solution Filter. While solving the equations derived in Sections 2.3 and 2.4 to compute the dihedrals, all non-real roots with the imaginary parts greater than a chosen threshold are discarded [72]. Also, multiplicities of the roots are eliminated, thereby pruning the subtrees rooted at the eliminated-roots.

Ramachandran and TALOS Filters. There exist regions in the Ramachandran map (*Rama-map*) that are forbidden for any biophysically relevant (ϕ, ψ) values for a given residue type. Therefore, any disallowed value for a dihedral suggested by the Rama-map, whenever it appears in the conformation tree, is pruned. We used the data from [38], and implemented a *residue-specific* Ramachandran filter.

Our implementation considers four residue types: Gly, Pro, pre-Pro, and other general amino acid types (called *general*). It has been specifically optimized for $O(1)$-time queries for the *favored* or *allowed* intervals for ϕ, and ψ given ϕ. If $M_{\mathcal{T}}$ is the Rama-map for residue type $\mathcal{T}$, and $I_{\mathcal{T}}$ is the set of all allowed ϕ-intervals for $\mathcal{T}$, we evaluate if $\phi \in I_{\mathcal{T}}$ for a computed ϕ. Similarly, when a ψ is computed, we evaluate if $\psi \in I_{\mathcal{T}}|_{\phi}$. TALOS [14,53] dihedral information, whenever available, are used as follows. If for the dihedral ϕ_i of the residue i of type $\mathcal{T}$, $I_{\mathcal{L}}$ is the TALOS-predicted interval, then for a computed ϕ for the residue i, we evaluate if $\phi \in I_{\mathcal{T}} \cap I_{\mathcal{L}}$. Similarly, for a computed ψ, the predicate $\psi \in I_{\mathcal{T}}|_{\phi} \cap I_{\mathcal{L}}$ is evaluated. The subtree rooted at the node representing the dihedral is pruned if any of these predicates fail. Further, in the absence of RDC data for a dihedral, finite-resolution uniform sampling of the Rama-map is used for that dihedral.

Steric Filter. We use our in-house implementation of the steric checker similar to that in [70]. During the depth-first search of the conformation tree, at each node corresponding to a newly added residue, the steric check is performed for (i) self-collision, i.e., if the fragment clashes with itself, and (ii) collision with the rest of the protein. If the clash score [70] is greater than a user-defined threshold, then the branch is pruned and the search backtracks.

Reachability Criterion. As each node of the conformation tree is visited, we test if the rest of the fragment, if grown using the best possible kinematic chain, can ever reach the stationary anchor. The node is pruned if this test fails. For long loops, this test prunes a large fraction of conformations, especially at the tree nodes at higher level (farther from the root).

Closure Criterion. When the distance between the mobile anchor (i.e., the conformation at a leaf node), and the stationary anchor is less than a user-specified threshold (chosen to be 0.2 Å), called the *closure distance*, and defined as the root-mean-square distance between the N, C^{α} and C' atoms of the mobile anchor and stationary anchor, the conformation is accepted and added to the ensemble of computed loops. Otherwise, the conformation is subject to a gradient-descent minimization over the last few dihedrals to improve the closure distance to below 0.2 Å while maintaining the user-defined RDC RMSD thresholds. If after minimization the closure is achieved, the conformation is accepted; otherwise, rejected. The RDC RMSD between back-computed and experimental RDCs is computed using the equation $\text{RMSD}_x = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(r^b_{x,i} - r^e_{x,i})^2}$, where x is either a ϕ-defining or a ψ-defining RDC type, n is the number of RDCs, $r^e_{x,i}$ is the experimental RDC, and $r^b_{x,i}$ is the corresponding back-computed RDC.

Pruning using unambiguous NOEs. When unambiguous backbone NOEs are available, they can be used as predicates to prune unsatisfying conformations.

3 Results and Discussion

To study the effectiveness of our algorithm, we applied it on experimental NMR data sets for four proteins. Further, to study the robustness of our algorithm to the variations in standard peptide geometry, we tested it on synthetic datasets for three sets of canonical loops of length 4, 8 and 12 residues that were investigated by three other protein loop closure algorithms [8,17,36].

Table 2. (*a*) The anchor residues are always included. (*b*) number of residues. (*c*) experimental RDCs used. The C^{α}-H^{α}, C^{α}-C' and N-H^{N} RDC RMSDs of loops computed by POOL are less than 2.0, 0.2 and 1.0 Hz, respectively. (*d*) Missing means unavailable. (*e*) Backbone RMSD computed vs. the NMR reference loops. The results show that the loops computed by POOL are more accurate than those computed by XPLOR-NIH [51].

Protein Loop[a]	Length[b]	Types of RDCs[c]	RDCs missing[d]	RMSD[e] (Å) (POOL)	RMSD[e] (Å) (XPLOR-NIH)
Ubiquitin 7-12	6	C^{α}-H^{α}, N-H^{N}	2	0.64	1.40
Ubiquitin 17-23	7	C^{α}-H^{α}, N-H^{N}	2	0.60	2.25
Ubiquitin 33-41	9	C^{α}-H^{α}, N-H^{N}	2	0.89	2.07
Ubiquitin 45-48	4	C^{α}-H^{α}, N-H^{N}	0	0.27	1.58
Ubiquitin 50-65	16	C^{α}-H^{α}, N-H^{N}	2	0.66	3.94
Ubiquitin 7-12	6	C^{α}-C', N-H^{N}	3	0.37	0.67
Ubiquitin 17-23	7	C^{α}-C', N-H^{N}	3	0.60	3.54
Ubiquitin 33-41	9	C^{α}-C', N-H^{N}	5	0.58	3.11
Ubiquitin 45-48	4	C^{α}-C', N-H^{N}	0	0.11	1.02
Ubiquitin 50-65	16	C^{α}-C', N-H^{N}	4	1.06	4.48
FF2 18-27	10	C^{α}-H^{α}, N-H^{N}	3	1.41	3.20
FF2 33-38	6	C^{α}-H^{α}, N-H^{N}	3	0.34	1.09
FF2 42-48	7	C^{α}-H^{α}, N-H^{N}	4	1.31	2.14
DinI 8-17	10	C^{α}-H^{α}, N-H^{N}	5	1.57	4.17
DinI 32-39	8	C^{α}-H^{α}, N-H^{N}	3	0.61	3.45
DinI 45-49	5	C^{α}-H^{α}, N-H^{N}	2	0.28	2.27
DinI 53-58	6	C^{α}-H^{α}, N-H^{N}	2	0.42	2.62
GB3 8-13	6	C^{α}-H^{α}, N-H^{N}	0	0.43	1.07
GB3 19-23	5	C^{α}-H^{α}, N-H^{N}	0	0.34	0.23
GB3 36-42	7	C^{α}-H^{α}, N-H^{N}	1	0.27	1.34
GB3 46-51	6	C^{α}-H^{α}, N-H^{N}	0	0.65	3.61

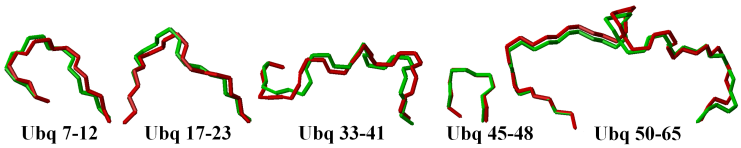

Fig. 2. Overlay of the loops (green) of ubiquitin computed by POOL using C^{α}-H^{α} and N-H^{N} RDCs vs. the corresponding loops (red) in the NMR reference structure (1d3z model 1) without any structural alignment.

3.1 Tests on Experimental NMR Data

We applied POOL to compute the loops of four proteins: FF2 (PDB id: 2kiq) [72], human ubiquitin (PDB id: 1d3z) [15], the DNA damage inducible protein I (DinI) (PDB id: 1ghh) [45], and the third IgG-binding domain of Protein G (GB3) (PDB id: 2oed) [62]. The RDC data for FF2 was recorded using Varian 600 and 800 MHz spectrometers at Duke University. Details of the NMR experimental procedures are provided in the SI **Appendix D** available online [61]. For ubiquitin, DinI and GB3, NMR data were obtained from BioMagResBank (BMRB) [63]. For each of these proteins, we used the NMR model 1 with loops removed as the respective test structures. RDCs were perturbed within the experimental-error window [66] to account for experimental errors.

Table 2 summarizes the results computed by POOL. For ubiquitin we used two different combinations of RDCs, viz. (C^{α}-H^{α}, N-H^{N}) and (C^{α}-C', N-H^{N}) to test the performance of our algorithm on different types of RDC data. In most cases, *sub-angstrom* RMSD loops were computed by POOL. Figure 2 shows the

overlay of the loops computed for ubiquitin using C^{α}-H^{α} and N-H^{N} RDCs with the corresponding loops from the NMR reference structure. For FF2, DinI and GB3, the results show that POOL is able to compute accurate loops when as many as 5 RDCs are missing.

The run-time analysis of POOL is similar to that in [68]. In practice, for short loops, POOL runs in minutes, and for longer loops (e.g., ubiquitin 50-65) it runs in hours on a 2.5 GHz dual-core processor Linux workstation.

3.2 Comparison vs. Traditional Structure Determination Protocols

To investigate whether traditional SA/MD-based structure determination protocols can compute accurate loop conformations using sparse data, we ran XPLOR-NIH [51] on the same input used by POOL for ubiquitin, FF2, DinI and GB3. Table 2 summarizes the results. In Figure 3, a comparison is made between the results obtained by applying POOL versus those obtained by applying XPLOR-NIH. The loops computed by POOL have much smaller (3- to 6-fold less for longer loops) backbone RMSD vs. the reference structures than those computed using XPLOR-NIH. For example, for ubiquitin loop 50-65, the loop computed by POOL has backbone RMSD 0.66 Å, a 6-fold decrease vs. the loop computed by XPLOR-NIH (3.94 Å). This shows that when given sparse data, our algorithm is able to compute more accurate loop conformations than the SA/MD-based protocols.

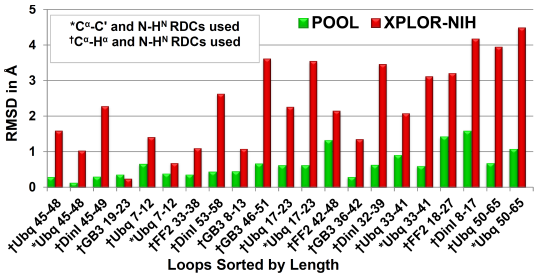

Fig. 3. POOL-computed loops achieve up to 6–fold improvement in backbone RMSD compared to XPLOR-NIH-computed loops.

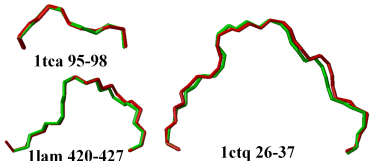

Fig. 4. Overlay of the lowest RMSD loop (green) computed by POOL for 4, 8 and 12-residue loops vs. the X-ray structures of the reference loops (red) without any structural alignment.

3.3 Comparison with Loop Prediction Algorithms

We compared the performance of POOL with three other loop prediction algorithms including the CCD method by Canutescu and Dunbrack [8], the CSJD algorithm by Coutsias *et al.* [17], and the self-organizing superimposition (SOS) algorithm by Liu *et al.* [36]. Unlike these algorithms, which do not use any data, POOL is a sparse data-driven algorithm. While CCD, CSJD and SOS algorithms have applications in protein structure prediction, none of them is specifically designed to incorporate geometric restraints from experimental NMR data. Our algorithm POOL provides an approach to fill this gap by being able to compute loops using sparse NMR data, specifically, RDCs.

In our study, we used the same test set as in [36,17,8]. This set consists of 10 loops each with 4, 8 and 12 residues long chosen from a set of nonredundant X-ray

Table 3. The minimum RMSD (Å) from X-ray structures for these four algorithms. The loops computed by POOL using only one ϕ-defining and one ψ-defining RDC per residue simulated using an alignment tensor estimated using PALES [76,75]. SOS, CSJD and CCD results were obtained from Table 1, Table 1 and Table 2 of [36], [17] and [8], respectively. These three methods do not use any experimental NMR data.

4-residue loops					**8-residue loops**					**12-residue loops**				
Loop	POOL	SOS	CSJD	CCD	Loop	POOL	SOS	CSJD	CCD	Loop	POOL	SOS	CSJD	CCD
1dvjA_20	0.74	0.23	0.38	0.61	1cruA_85	0.72	1.48	0.99	1.75	1cruA_358	1.54	2.39	2.00	2.54
1dysA_47	0.25	0.16	0.37	0.68	1ctqA_144	0.91	1.37	0.96	1.34	1ctqA_26	0.65	2.54	1.86	2.49
1eguA_404	0.42	0.16	0.36	0.68	1d8wA_334	0.28	1.18	0.37	1.51	1d4oA_88	1.83	2.44	1.60	2.33
1ej0A_74	0.18	0.16	0.21	0.34	1ds1A_20	0.70	0.93	1.30	1.58	1d8wA_46	0.93	2.17	2.94	4.83
1i0hA_123	0.27	0.22	0.26	0.62	1gk8A_122	0.87	0.96	1.29	1.68	1ds1A_282	1.50	2.33	3.10	3.04
1id0A_405	0.63	0.33	0.72	0.67	1i0hA_122	0.45	1.37	0.36	1.35	1dysA_291	0.76	2.08	3.04	2.48
1qnrA_195	0.47	0.32	0.39	0.49	1ixh_106	0.68	1.21	2.36	1.61	1eguA_508	1.25	2.36	2.82	2.14
1qopA_44	0.36	0.13	0.61	0.63	1lam_420	0.42	0.90	0.83	1.60	1f74A_11	0.76	2.23	1.53	2.72
1tca_95	0.12	0.15	0.28	0.39	1qopB_14	0.87	1.24	0.69	1.85	1qlwA_31	1.27	1.73	2.32	3.38
1thfD_121	0.25	0.11	0.36	0.50	3chbD_51	0.96	1.23	0.96	1.66	1qopA_178	0.87	2.21	2.18	4.57
Average	0.37	0.20	0.40	0.56	Average	0.69	1.19	1.01	1.59	Average	1.14	2.25	2.34	3.05

crystallographic structures from the PDB. Since there is no experimental RDC data available for these proteins, we simulated the RDCs using PALES [76,75]. Details of the RDC simulation are described in the SI **Appendix E** available online [61]. The alignment tensor, the RDC data, and the two anchor peptide planes of the loop were used by POOL to compute the loop conformations.

Table 3 summarizes the results for POOL, CCD, CSJD and SOS algorithms. In Figure 4, examples of minimum RMSD loop conformations determined by POOL are shown. For 4-residue loops the average minimum RMSD of the computed loops by POOL is larger than that for SOS, but smaller than that for CSJD and CCD. This can be explained by the fact that SOS allows slight deviations from standard protein geometry. For 8 and 12-residue loops POOL computes more accurate loops than other algorithms. For example, for 12-residue loops, the average minimum RMSD of the loops are 1.14, 2.25, 2.34 and 3.05 Å for POOL, SOS, CSJD and CCD, respectively, which shows a 2-fold improvement in accuracy by POOL. For five of these loops, POOL computed loops with *sub-angstrom accuracy*. Further, the reference loops in Table 3 have deviations from standard protein geometry; therefore, the RDCs simulated on them inherits these deviations, in addition to a Gaussian noise of 1 Hz added to account for experimental errors. These results suggest that POOL is robust to both experimental uncertainties in RDCs, and minor deviations from standard protein geometry assumptions. Therefore, POOL can be useful to compute longer loops with high accuracy using a minimal amount of RDC data.

4 Conclusions

While the global fold of a protein can often be determined from experimental NMR data [25,66,68,72,71], determining loop conformations from sparse experimental RDCs is a difficult problem. We described a novel, efficient, and practical deterministic algorithm, POOL, that determines accurate loop conformations from sparse RDC data. Empirical comparison with traditional structure determination protocols [51] demonstrates that POOL is able to achieve up to 6-fold improvement over the latter methods under sparse-data settings.

Since an accurate and complete protein backbone is a prerequisite for NOE-assignment algorithms [28,72] and side-chain resonance assignment methods [74] in traditional NMR structure determination protocols, POOL will be useful in high-resolution protein structure determination. Whenever RDCs can be collected for proteins with known X-ray structures containing missing loops, POOL can be used to determine the loop conformations.

Since RDCs also provide sensitive probes to protein conformational dynamics [59,48] over nano- to millisecond timescales, it will be interesting to extend our algorithm to capture and characterize the motional fluctuations, and deconvolve the dynamics from measured RDCs. In such cases, the ensemble of loops computed by POOL will effectively define a normal distribution of conformations centered at the experimentally-measured RDCs, and as such encode a unimodal dynamic ensemble about a protein's native fold. Our algorithm can even be a stepping stone to computing ensembles reflecting more complex dynamics.

Availability. The source code of our algorithm is available open-source under the GNU Lesser General Public License (Gnu, 2002).

Acknowledgments. We thank professors Jane and Dave Richardson, Dr. A. Yershova, Dr. A. Yan, Mr. V. Chen, Mr. J. MacMaster, Mr. C.-Y. Chen, Mr. J. Martin, Mr. P. Gainza, Mr. M. Hallen and Ms. S. Jain for many valuable suggestions. We thank all members of the Donald, Zhou and Richardson Labs for helpful discussions.

References

1. M. Andrec *et al. J Biomol NMR*, 21:335–347, 2004.
2. D. Baker and A. Sali. *Science*, 294:93–96, 2001.
3. H. M. Berman *et al. Nucleic Acids Res*, 28(1):235–242, 2000.
4. G. Bouvignies *et al. Angewandte Chemie*, 118:8346–8349, 2006.
5. R. E. Bruccoleri and M. Karplus. *Macromolecules*, 29:1847–1862, 1990.
6. A. T. Brünger. Yale University Press, New Haven, CT, 1992.
7. J. L. Buchbinder and R. J. Fletterick. *J Biol Chem*, 271(37):22305–22309, 1996.
8. A. A. Canutescu and R. L. Dunbrack, Jr. *Protein Sci*, 12(5):963–972, 2003.
9. J. Casey. *Proceedings of the Royal Society of London*, XIX:495–497, June 15, 1871.
10. C.-Y. Chen *et al. Proc Natl Acad Sci USA*, 106(10):3764–3769, 2009.
11. G. S. Chirikjian. In *Proceedings of IROS*, 2:1067–1073, 1993.
12. G. M. Clore *et al. J Magn Reson*, 131:159–162, 1998.
13. V. Collura *et al. Protein Sci*, 2:1502–1510, 1993.
14. G. Cornilescu *et al. J Biomol NMR*, 13:289–302, 1999.
15. G. Cornilescu *et al. J Am Chem Soc*, 120:6836–6837, 1998.
16. J. Cortés *et al. J Comput Chem*, 25(7):956–967, 2004.
17. E. A. Coutsias *et al. J Comput Chem*, 25:510–528, 2004.
18. F. Delaglio *et al. J Am Chem Soc*, 122:2142–2143, 2000.
19. B. R. Donald and J. Martin. *Prog NMR Spectrosc*, 55(2):101–127, 2009.
20. P. Du *et al. Protein Engineering*, 16(6):407–414, 2003.
21. R. M. Fine *et al. Proteins*, 1(4):342–362, 1986.
22. A. Fiser *et al. Protein Sci*, 9(9):1753–1773, 2000.
23. K. M. Frey *et al. Proc Natl Acad Sci USA*, 107(31):13707–13712, 2010.

24. I. Georgiev *et al.* *J Comput Chem*, 29:1527–1542, 2008.
25. A. W. Giesen *et al.* *J Biomol NMR*, 25:63–71, 2003.
26. N. Gō and H. A. Scheraga. *Macromolecules*, 3:178–187, 1970.
27. M. J. Gorczynski *et al.* *Chemistry & Biology*, 14(10):1186–1197, 2007.
28. P. Güntert. *Prog NMR Spectrosc*, 43:105–125, 2003.
29. X. Hu *et al.* *Proc Natl Acad Sci USA*, 104(45):17668–17673, 2007.
30. P. Koehl and M. Delarue. *Nat Struct Biol*, 2:163–70, 1995.
31. R. Kolodny *et al.* *Int J Robot Res*, 24:151–163, 2005.
32. J. Kuszewski *et al.* *J Am Chem Soc*, 126(20):6258–6273, 2004.
33. C. J. Langmead and B. R. Donald. In *Proceedings of CSB*, pages 209–217, 2003.
34. C. J. Langmead and B. R. Donald. In *Proceedings of CSB*, pages 278–289, 2004.
35. R. H. Lilien *et al.* *J Comput Biol*, volume 12(6), pages 740–761, 2005.
36. P. Liu *et al.* *PLoS Comput Biol*, 5(8):e1000478, 08 2009.
37. J. A. Losonczi *et al.* *J Magn Reson*, 138:334–342, 1999.
38. S. C. Lovell *et al.* *Proteins*, 50:437–450, 2003.
39. D. Manocha and J. F. Canny. *IEEE T Robotic Autom*, 10:648–657, 1994.
40. R. J. Milgram *et al.* *J Comput Chem*, 29(1):50–68, 2008.
41. C. Mumenthaler *et al.* *J Biomol NMR*, 10(4):351–362, 1997.
42. S. Pesce and R. Benezara. *Mol Cell Biol*, 13(12):7874–7880, 1993.
43. J. H. Prestegard *et al.* *Chemical Reviews*, 104:3519–3540, 2004.
44. B. E. Ramirez and A. Bax. *J Am Chem Soc*, 120:9106–9107, 1998.
45. B. E. Ramirez *et al.* *Protein Sci*, 9:2161–2169, 2000.
46. C. A. Rohl and D. Baker. *J Am Chem Soc*, 124:2723–2729, 2002.
47. G. Salmon. Longmans, Green and Company, London, 1912.
48. L. Salmon *et al.* *Angew Chem Int Edit*, 48(23):4154–4157, 2009.
49. A. Saupe. *Angewandte Chemie*, 7(2):97–112, 1968.
50. J. B. Saxe. In *Proc 17th Allerton Conf Comm, Ctrl Comput*, pages 480–489, 1979.
51. C. D. Schwieters *et al.* *J Magn Reson*, 160:65–73, 2003.
52. A. Shehu *et al.* *Proteins*, 65(1):164–179, 2006.
53. Y. Shen *et al.* *J Biomol NMR*, 44:213–223, 2009.
54. P. S. Shenkin *et al.* *Biopolymers*, 26(12):2053–2085, 1987.
55. L. Shi and J. A. Javitch. *Proc Natl Acad Sci USA*, 101(2):440–445, 2004.
56. F. Tian *et al.* *J Am Chem Soc*, 123:11791–11796, 2001.
57. N. Tjandra and A. Bax. *Science*, 278:1111–1114, 1997.
58. J. R. Tolman *et al.* *Proc Natl Acad Sci USA*, 92:9279–9283, 1995.
59. J. R. Tolman *et al.* *Nat Struct Biol*, 4(4):292–297, 1997.
60. S. C. E. Tosatto *et al.* *Protein Engineering*, 15(4):279–286, 2002.
61. C. Tripathy, J. Zeng, P. Zhou, and B. R. Donald. Supporting Information: `http://www.cs.duke.edu/donaldlab/Supplementary/recomb11/pool/` . 2011.
62. T. S. Ulmer *et al.* *J Am Chem Soc*, 125:9179–9191, 2003.
63. E. L. Ulrich *et al.* *Nucleic Acids Res*, 36(Database issue):D402–D408, 2008.
64. H. W. T. van Vlijmen and M. Karplus. *J Mol Biol*, 267:975–1001, 1997.
65. C. Wang *et al.* *J Mol Biol*, 373(2):503–519, 2007.
66. L. Wang and B. R. Donald. *J Biomol NMR*, 29(3):223–242, 2004.
67. L. Wang and B. R. Donald. In *Proceedings of CSB*, pages 189–202, 2005.
68. L. Wang *et al.* *J Comput Biol*, 13(7):1276–1288, 2006.
69. W. J. Wedemeyer and H. A. Scheraga. *J Comput Chem*, 20(8):819–844, 1999.
70. J. M. Word *et al.* *J Mol Biol*, 285:1711–1733, 1999.
71. A. Yershova *et al.* In *Proceedings of WAFR*, 68:355–372, 2010.
72. J. Zeng *et al.* *J Biomol NMR*, 45(3):265–281, 2009.
73. J. Zeng *et al.* In *Proceedings of CSB*, pages 169–181. ISBN 1752–7791, 2008.
74. J. Zeng *et al.* In *Proceedings of RECOMB*, Vol 6044 *LNCS*, pages 550–570, 2010.
75. M. Zweckstetter. *Nat Protoc*, 3:679–690, 2008.
76. M. Zweckstetter and A. Bax. *J Am Chem Soc*, 122(15):3791–3792, 2000.