Tuning interval Branch-and-Prune for protein structure determination

Bradley Worley, Florent Delhommel, Florence Cordier, Thérèse Malliavin, Benjamin Bardiaux, Nicolas Wolff, Michael Nilges, Carlile Lavor, Leo Liberti

J. Glob. Optim. manuscript No.
(will be inserted by the editor)

Tuning interval Branch-and-Prune for protein structure determination

Bradley Worley · Florent Delhommel · Florence Cordier · Thérèse E. Malliavin · Benjamin Bardiaux · Nicolas Wolff · Michael Nilges · Carlile Lavor · Leo Liberti

Received: date / Accepted: date

Abstract The interval Branch and Prune (*i*BP) algorithm for obtaining solutions to the interval Discretizable Molecular Distance Geometry Problem (*i*DMDGP) has proven itself as a powerful method for molecular structure determination. However, substantial obstacles still must be overcome before *i*BP may be employed as a tractable general-purpose alternative to existing structure determination algorithms. This work introduces an iterative variant of the *i*BP algorithm that leverages existing knowledge of protein structures in order to reduce the size of the effective search space by many orders of magnitude. These improvements are included in a newly released implementation of the *i*BP software that aims to provide a solid platform for both research and application of the *i*DMDGP.

Keywords Distance geometry · Protein structure · Nuclear Magnetic Resonance · Branch-and-Prune

B. Worley · T. E. Malliavin · B. Bardiaux · M. Nilges
Unité Bioinformatique Structurale, Institut Pasteur
Paris, France
E-mail: bradley.worley@pasteur.fr
E-mail: therese.malliavin@pasteur.fr
E-mail: michael.nilges@pasteur.fr

F. Delhommel · F. Cordier · N. Wolff
Unité Résonance Magnétique Nucléaire des Biomolecules, Institut Pasteur
Paris, France
E-mail: florent.delhommel@pasteur.fr
E-mail: florence.cordier@pasteur.fr
E-mail: nicolas.wolff@pasteur.fr

C. Lavor
IMECC-UNICAMP, University of Campinas
Campinas, SP, Brazil
E-mail: clavor@ime.unicamp.br

L. Liberti
LIX, École Polytechnique
Palaiseau, France
E-mail: liberti@lix.polytechnique.fr

1 Introduction

Within biology, it is well-established that the biochemical function of a molecule is strongly related to its three-dimensional structure. As a direct consequence, substantial effort is devoted within the field of structural biology to the problem of molecular structure determination: given the chemical composition and topology of a molecule, we seek its conformation in $\mathbb{R}^3$. For the specific problem of protein structure determination, the composition and topology of a protein molecule are completely specified by its amino acid sequence,[1] and we again seek its conformation(s) in three dimensions.

Proteins are highly flexible polymer chains of amino acids, and consequently may have multiple conformations for a given amino acid sequence. Therefore, additional geometric measurements are required in order to obtain a reasonable number of solutions to the protein structure determination problem. Nuclear Magnetic Resonance (NMR) experiments are frequently employed to obtain these measurements. In an NMR experiment, the interaction between the nucleus of each atom in a molecule and a strong magnetic field is measured using precisely timed pulses of radio-frequency radiation [15]. The interaction energy of a nucleus as a fraction of the total magnetic field energy is known as its chemical shift. Chemical shifts are highly sensitive reporters of the local electronic environments of their nuclei, making them excellent tools for structure determination. For proteins, whose polymer backbones are formed by repeated N–C_α–C' units, chemical shifts are dominated by local backbone geometry (Fig. 1), in particular the backbone dihedral angles ϕ, ψ and ω. The ϕ and ψ angles are historically referred to as the Ramachandran angles [19]. For amino acid i of a protein, the ϕ_i angle is defined by the atoms $C'^{(i-1)}$–$N^{(i)}$–$C_\alpha^{(i)}$–$C'^{(i)}$, the ψ_i angle is defined by $N^{(i)}$–$C_\alpha^{(i)}$–$C'^{(i)}$–$N^{(i+1)}$, and the ω_i angle is defined by $C_\alpha^{(i)}$–$C'^{(i)}$–$N^{(i+1)}$–$C_\alpha^{(i+1)}$. The ω_i dihedral angle is usually fixed to 180° due to the known near-planarity of the peptide bond [1]. When all ω dihedrals are fixed to 180°, and the local ge-

[1] Additional post-translational modifications may alter the chemical composition of a protein molecule, but their effects must be detected by further chemical analysis.

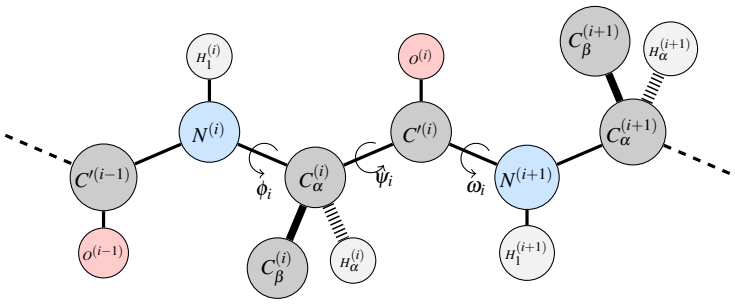

Fig. 1 Depiction of a single amino acid unit i of a protein backbone, including any atoms of flanking units $i-1$ and $i+1$ necessary for defining the backbone dihedral angles ϕ_i, ψ_i and ω_i. While the repetition vertex orders introduced in sec. 2.3 do not include C_β atoms, the NMR chemical shifts of these atoms are important indicators of the local dihedral angles (cf. 1).

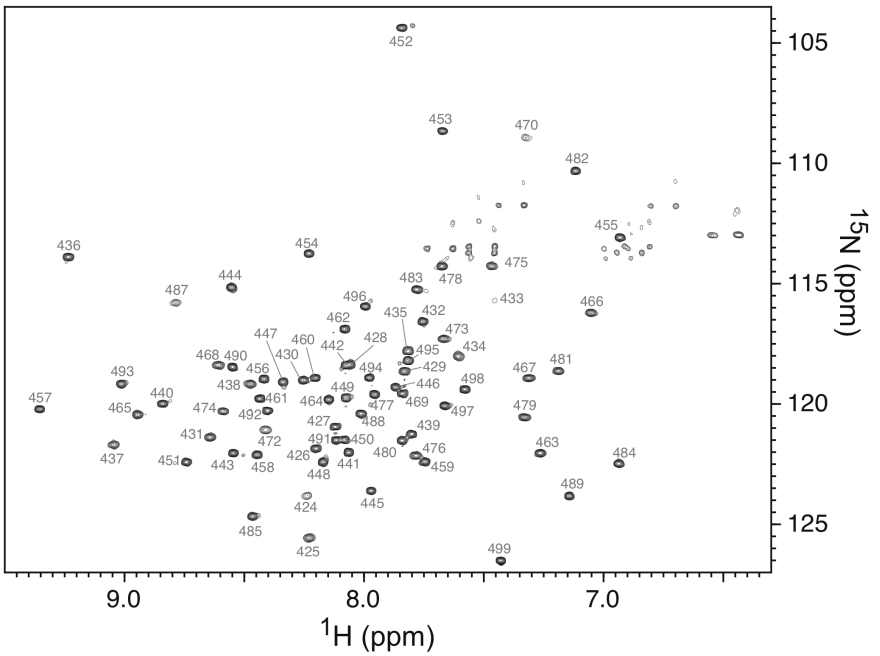

Fig. 2 NMR spectrum obtained from the ^{1}H–^{15}N heteronuclear single quantum coherence (HSQC) experiment for the HHD2 protein sample described in sec. 4. Peaks in the spectrum labeled with a number correspond to pairs of bonded $H_1^{(i)}$ and $N^{(i)}$ atoms in the protein, where the number corresponds to the index of the amino acid unit (i). The protein studied here is a short sub-sequence of a much longer amino acid sequence, so the values of i indicate the position of the synthesized amino acids within that longer sequence.

ometry is assumed to be fixed [10], the conformation of a protein's backbone may be completely specified by supplying the Ramachandran angles (ϕ, ψ) for all of its amino acid units. These angles may be predicted from the chemical shifts of the N, C_α, C_β, C', H_1 and H_α atoms using TALOS-N, an artificial neural network trained on a large database of previously determined NMR protein structures [23]. Multiple predictions from the neural network are used to estimate a mean and standard deviation for each backbone dihedral angle, resulting in fairly reliable Ramachandran angle intervals. However, when some chemical shift measurements are missing, the accuracy of the predicted dihedral angles can suffer. A set of multidimensional NMR experiments involving different nuclei [11] are required to determine the chemical shift of each atom, using a process known as sequential assignment (cf. §4). Fig. 2 illustrates one such experimental result in protein NMR, a two-dimensional spectrum that is used to correlate the chemical shifts of backbone $H_1^{(i)}$ and $N^{(i)}$ atoms.

In addition to the local geometic information provided by chemical shifts, NMR may also be used to obtain global geometric information in the form of distances between pairs of atoms. In such experiments, the intensity of the NMR signal at the

chemical shifts of two atoms, e.g. $H^{(i)}$ and $H^{(j)}$, is approximately proportional to the inverse sixth power of their distance, $d(H^{(i)},H^{(j)})^{-6}$. Because the signal intensity decays rapidly with increasing distance, this information is usually only available for atoms that are within 5–6 Å in the structure. Additional effects, including inevitable molecular motion during the measurement, can also result in a change of signal between nearby atoms. Finally, these signal intensity measurements are often perturbed by strong systematic measurement errors, so they are generally converted into interval distances using "rules of thumb" developed by NMR spectroscopists [27].

Conventional NMR protein structure determination protocols combine known distances and angles between bonded atoms with predicted Ramachandran angles and interval distance measurements into a non-convex objective function that smoothly penalizes any deviations of the structure from the target geometry [4,18]. As this objective contains many local optima, metaheuristics such as simulated annealing with multiple random initializations are employed. Nevertheless, there is no guarantee that the resulting structures are within the feasible set specified by the distance, angle and dihedral constraints.

The protein structure determination problem may be recast into a DISTANCE GEOMETRY PROBLEM (DGP) by converting all aforementioned geometric constraints into distance constraints between pairs of atoms. Formally, the protein is represented as a graph $G=(V,E,d)$, where V represents the atoms and E holds the atom pairs which have a known distance. The distances are either exact values or intervals, so there exists a partitioning of the edge set $E=E_D\cup E_I$, where E_D and E_I hold exact and interval distances, respectively. We denote the set of positive real valued intervals as $\mathbb{IR}_+$. Exact distances in E_D are given by known bond lengths, angles, and dihedrals in protein structures. Three-atom angles and four-atom dihedral angles that are known from protein chemistry shall be collected into the sets Θ and Ω, respectively. Given this information, we may formally re-introduce the INTERVAL DISCRETIZABLE MOLECULAR DISTANCE GEOMETRY PROBLEM [16] as the following,

INTERVAL DMDGP: given a simple weighted undirected graph $G=(V,E,d)$ where $E=E_D\cup E_I$ and,

$$d:\begin{cases}E_D\to\mathbb{R}_+\\ E_I\to\mathbb{IR}_+\end{cases}$$

and a vertex ordering $R=(v_1,\ldots,v_n)$ on V satisfying the following requirements:

- The subgraph of G induced by $V_0=\{v_1,v_2,v_3\}$ is a clique with all edges in E_D
- For all $j\in R\setminus V_0$ we have
  1. $d_{j-1,j},\ d_{j-2,j}\in E_D$
  2. $d_{j-3,j}\in E$
- For all $j\in R\setminus\{v_1,v_2\}$ the strict triangular inequality is satisfied by $d_{j-1,j}$, $d_{j-2,j}$ and $d_{j-2,j-1}$

is there an embedding $\mathbf{x}:V\to\mathbb{R}^3$ such that $\|\mathbf{x}_u-\mathbf{x}_v\|=d_{uv}\ \ \forall\{u,v\}\in E_D$ and $\|\mathbf{x}_u-\mathbf{x}_v\|\in[\underline{d}_{uv},\overline{d}_{uv}]\ \ \forall\{u,v\}\in E_I$?

In general, the vertex ordering R may visit each vertex in the graph one or more times. We define such a REPETITION ORDER [6] as a sequence $R:\mathbb{N}\to V\cup 0$ with

length $|R| \in \mathbb{N}$ (such that $R_i = 0$ for all $i > |R|$) when it satisfies the requirements of the *i*DMDGP. A repetition order entry r_i is referred to as *repeated* when there exist one or more elements v_j with $j < i-2$, such that $v_j = v_i$. As shown in [6, §4], there exist *i*DMDGP instances whose vertex orders *must* contain repetitions.

The interval Branch-and-Prune (*i*BP) algorithm is a combinatorial method for finding solutions to the *i*DMDGP. The *i*DMDGP solution space is a search tree, within which each path from the root node to a leaf node represents a distinct protein conformation. The *i*BP algorithm, explained in detail previously [13,5,12], recursively traverses the search tree in order to enumerate solutions to a given problem instance. If at any point in the search, the partial solution becomes infeasible with respect to the constraints, then the leaves of the current search tree node are "pruned" and *i*BP backtracks until a solution is identified. By searching the entire tree, *i*BP is capable of systematically exploring the feasible set of any given problem instance.

This work introduces a variant of *i*BP that uses an iterative tree traversal algorithm based on embedding equations derived from Clifford algebra [12]. This iterative variant is ideal for *i*DMDGP instances that have highly repetitive vertex orders [6], as it requires no extra matrix computations for repeated atoms in the order. We then pair our iterative *i*BP algorithm with such a highly repetitive discretization vertex order that directly uses information on dihedral angles during tree traversal. When many backbone dihedral angles are known to high precision in a given problem instance, this method and vertex order produce significantly smaller search trees than prior methods. For the purposes of this work, the proposed algorithm and vertex order enable *i*BP to obtain solutions to problem instances containing strong backbone dihedral information. For such problem instances, it is challenging for *i*BP to obtain *any* solutions using previously introduced vertex orders that do not branch on backbone dihedrals. We show that our new implementation solves this family of instances efficiently, even when previous *i*BP implementations fail to obtain any solutions.

The remainder of this paper is organized as follows. In Section 2, we introduce the iterative *i*BP algorithm with its embedding equations, along with new repetition vertex orders for protein structures. In Section 3, we describe features of the new iterative *i*BP implementation that simplify its use in routine applications while retaining the remarkable flexibility and generality of *i*BP. In Section 4 we present some computational results, specifically focusing on benchmarking the ability of *i*BP to find a single solution. Finally, we conclude the paper in Section 5.

2 Algorithmic considerations

2.1 Iterative embedding relations

*i*BP is most naturally defined as a recursive tree traversal algorithm [13,7] that uses recursively defined affine transformation matrices [24] for computing embedded coordinates. However, recent use of Clifford algebra has yielded embedding equations that are non-recursive [12], the key results of which are recalled here.

For any vertex j in a repetition order R, we seek the embedded coordinate $\mathbf{x}_j \in \mathbb{R}^3$, given distances to the three preceding vertices $j-1$, $j-2$ and $j-3$. From the

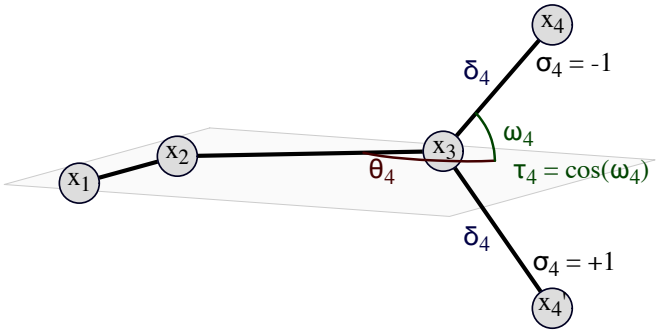

Fig. 3 Embedding relations for computing the position of a vertex $\mathbf{x}_4$ from its predecessors $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3)$, *via* the parameters $(\delta_4, \theta_4, \tau_4, \sigma_4)$.

properties of the *i*DMDGP, the distances $d_{j-1,j}$, $d_{j-2,j}$ and $d_{j-3,j}$ are known, where $d_{j,j-3}$ is potentially an interval. The remaining distances in the clique formed by the four vertices $(j-3, j-2, j-1, j)$ are calculable from the coordinates $\mathbf{x}_{j-1}$, $\mathbf{x}_{j-2}$ and $\mathbf{x}_{j-3}$. We shall introduce the quantities δ_j, θ_j, τ_j and σ_j for embedding vertex j, where $\delta_j \triangleq d_{j-1,j}$ (cf. Fig. 3). The angle θ_j is uniquely defined, and is obtained from the cosine law using the relevant distances,

$$\theta_j \triangleq \cos^{-1}\left(\frac{d_{j-1,j}^2 + d_{j-2,j-1}^2 - d_{j-2,j}^2}{2 d_{j-1,j}\, d_{j-2,j-1}}\right) \tag{1}$$

From the requirement that the vertex order satisfies the strict triangular inequality, θ_j is strictly within $(0, \pi)$, ensuring the vertices $(j-3, j-2, j-1)$ are non-collinear. The variable $\tau_j \triangleq \cos\omega_j$ is related to the dihedral angle ω_j formed by the vertices $(j-3, j-2, j-1, j)$. When ω_j is known from either protein chemistry or measurement, such that $\omega_j \in \Omega$, we may directly compute τ_j, as well as the sign variable $\sigma_j \in \{-1, +1\}$:

$$\sigma_j \triangleq \begin{cases} \frac{\sin\omega_j}{|\sin\omega_j|} & \text{if } \sin\omega_j \neq 0 \\ 1 & \text{if } \sin\omega_j = 0 \end{cases}$$

In cases where $\omega_j \notin \Omega$, the *i*DMDGP nevertheless guarantees that the distance $d_{j-3,j}$ is available, and we may compute τ_j from the cosine law for a trihedron [12]:

$$\tau_j = \frac{2 d_{j-2,j-1}^2 \left(d_{j-3,j-2}^2 + d_{j-2,j}^2 - d_{j-3,j}^2\right) - d_{j-3,j-2,j-1} d_{j-2,j-1,j}}{\sqrt{4 d_{j-3,j-2}^2 d_{j-2,j-1}^2 - d_{j-3,j-2,j-1}^2}\sqrt{4 d_{j-2,j-1}^2 d_{j-2,j}^2 - d_{j-2,j-1,j}^2}} \tag{2}$$

where

$$d_{j-3,j-2,j-1} \triangleq d_{j-3,j-2}^2 + d_{j-2,j-1}^2 - d_{j-3,j-1}^2$$
$$d_{j-2,j-1,j} \triangleq d_{j-2,j-1}^2 + d_{j-2,j}^2 - d_{j,j-1}^2$$

Given δ_j, θ_j, τ_j, and σ_j, the embedded coordinates of vertex j are given by the following equation:

$$\mathbf{x}_j = \mathbf{p}_1 + \tau_j \mathbf{p}_2 + \sigma_j \sqrt{1-\tau_j^2}\, \mathbf{p}_3$$

where $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3 \in \mathbb{R}^3$ depend only on $\mathbf{x}_{j-1}$, $\mathbf{x}_{j-2}$, $\mathbf{x}_{j-3}$, δ_j and θ_j,

$$\begin{aligned}
\mathbf{p}_1 &= -\left(\frac{\delta_j}{\|\mathbf{r}_{12}\|}\right)\left(\left(\cos(\theta_j) - \frac{\|\mathbf{r}_{12}\|}{\delta_j}\right)\mathbf{x}_{j-1} - \cos(\theta_j)\,\mathbf{x}_{j-2}\right) \\
\mathbf{p}_2 &= -\left(\frac{\delta_j}{\|\mathbf{r}_{12}\|}\right)\left(\frac{\sin(\theta_j)}{\|\mathbf{r}_{12}\times\mathbf{r}_{23}\|}\right)\left(\|\mathbf{r}_{12}\|^2\,\mathbf{r}_{23} - (\mathbf{r}_{12}\cdot\mathbf{r}_{23})\,\mathbf{r}_{12}\right) \\
\mathbf{p}_3 &= -\left(\frac{\delta_j}{\|\mathbf{r}_{12}\|}\right)\left(\frac{\sin(\theta_j)}{\|\mathbf{r}_{12}\times\mathbf{r}_{23}\|}\right)\|\mathbf{r}_{12}\|\,(\mathbf{r}_{12}\times\mathbf{r}_{23})
\end{aligned}$$

and we have introduced $\mathbf{r}_{12}, \mathbf{r}_{23} \in \mathbb{R}^3$ for notational simplicity,

$$\begin{aligned}
\mathbf{r}_{12} &= \mathbf{x}_{j-1} - \mathbf{x}_{j-2} \\
\mathbf{r}_{23} &= \mathbf{x}_{j-2} - \mathbf{x}_{j-3}
\end{aligned}$$

By explicitly parameterizing each embedding with the sign σ_j, we obtain an unambiguous solution for the coordinate $\mathbf{x}_j$. An *i*BP search tree that uses these equations therefore stores a set of values $(\delta, \theta, \tau, \sigma, \mathbf{x})$ at each of its nodes. Thus, whenever ω_j—and consequently σ_j and τ_j—is known for vertex j, this yields a single branch at that level in the tree.

In the present *i*BP implementation [5] that uses recursively defined affine transformation matrices, *i*BP must compute and store a matrix for each repeated vertex in a repetition order. To compute the matrix of a repeated vertex, the sign parameter σ_j must be determined by systematic search during branching: the value of σ_j that yields an embedded coordinate nearest to the originally determined coordinate is used to build the transformation matrix. For vertex orders containing a large degree of repetition, this introduces unnecessary computations. Using the above equations, embedding a new vertex $\mathbf{x}_j$ requires only $(\delta_j, \theta_j, \omega_j, \sigma_j)$ and the embedded coordinates of the last three vertices in the order, so repeating a vertex in a discretization order introduces a copy of the original vertex coordinate, but requires no additional computations.

Like the *i*BP implementation using affine transformation matrices, our implementation requires the calculation of τ_j from distances by eq. (2) when $\omega_j \notin \Omega$. While this can lead to numerical instability in large instances with imperfect distance data, the fact that our implementation directly computes τ_j when $\omega_j \in \Omega$ effectively mitigates this instability when dihedral constraints are available.

Thus far, the discussion of the Clifford algebraic embedding equations has focused on situations when either ω_j or $d_{j,j-3}$ are known exactly, resulting in a single value of τ_j and either one or two values of σ_j, respectively. When $d_{j-3,j}$ is an interval distance, such that $\{j-3, j\} \in E_I$, we shall denote its lower and upper bounds as $\underline{d}_{j-3,j}$ and $\overline{d}_{j-3,j}$, respectively. Similarly, when ω_j is an interval dihedral, i.e. $\omega_j \in \Omega_I$, we denote its bounds as $\underline{\omega}_j$ and $\overline{\omega}_j$. In a process that is described in more detail below, these intervals are then *discretized* through linear interpolation in order to obtain values that may be used in the embedding equations.

Algorithm 1 The *advance* method

Input: Index $\mathbf{I}$ and tree size $\mathbf{N}$ in $\mathbb{N}^{|R|}$, incremented element position j
Output: Least modified element position j^*
$I_k \leftarrow 1 \quad \forall k \in \{j+1,\ldots,|R|\}$
for all $k \in \{j, j-1,\ldots,1\}$ **do**
    $I_k \leftarrow I_k + 1$
    $j^* \leftarrow k$
    if $I_k > N_k$ **then**
        $I_k \leftarrow 1$
    else
        return j^*
    end if
end for
return j^*

2.2 Iterative tree traversal

In order to obtain solutions to the *i*DMDGP, *i*BP discretizes the search space by breaking each interval into a finite set of values, resulting in a search tree. We introduce the multi-index $\mathbf{N} \in \mathbb{N}^{|R|}$, referred to as the tree size, which holds the number of branches at each level of this tree. Herein R is the repetition order associated with the *i*DMDGP instance. The number of branches at level j of the tree is given by N_j, and depends on the geometric information available for vertex j. When j is a repetition of a previously embedded vertex, we have $N_j = 1$. When $d_{j-3,j}$ is an exact distance, there are two possible embeddings of vertex j, so $N_j = 2$. Finally, when $d_{j-3,j}$ is an interval distance, $N_j = 2B$ where B is a user-specified discretization factor. In cases where ω_j is known—as opposed to knowing only $\cos\omega_j$ *via* $d_{j-3,j}$—the branch count N_j is reduced by a factor of two, as σ_j takes a single value.

By default, interval discretization in *i*BP is done uniformly, with each of the B discrete points equally spaced upon its arc formed by the three-sphere intersection sub-problem. Due to discretization, *i*BP is a heuristic method. For any *i*DMDGP instance having a non-empty feasible set, there is no guarantee that *i*BP will select discretization points which satisfy the original *i*DMDGP [9]. However, when sufficiently large values of B are used in concert with a small tolerance during distance feasibility pruning, the chances of obtaining solutions is increased. Thus, in the absence of alternative methods for discretizing the interval d_j and ω_j values, we defer to uniform subdivision.

In the following, we shall employ multi-index notation to describe the iterative tree traversal routine, with multi-indices denoted by capitalized boldface letters. Given a tree size $\mathbf{N} \in \mathbb{N}^{|R|}$, we introduce the partial order operator for all $|R|$-indices $\mathbf{I} \in \mathbb{N}^{|R|}$ as follows:

$$\mathbf{I} \leq \mathbf{N} \rightarrow \begin{cases} \textbf{true} & \text{if } I_j \leq N_j \quad \forall j \in \{1,\ldots,|R|\} \\ \textbf{false} & \text{otherwise} \end{cases}$$

An index $\mathbf{I}$ describes a valid—though potentially infeasible—path through the tree if it satisfies $\mathbf{1} \leq \mathbf{I} \leq \mathbf{N}$, where $\mathbf{1}$ is a vector of ones.

In the iterative formulation of *i*BP described in this paper, paths through the search tree are enumerated by *advancing* an index $\mathbf{I}$ (Algorithm 1). Advancing an index $\mathbf{I}$ at its last element $I_{|R|}$ yields a sequence of indices—and thus paths through a tree—that implicitly effects a depth-first search in that tree. On the other hand, incrementing an index $\mathbf{I}$ at some element I_j, with $j < |R|$, is equivalent to pruning the tree at that level, as it skips all indices that would have been produced by incrementing at $|R|$. The term *index* is used to describe $\mathbf{I}$ due to the similarity it shares to the indices of multidimensional arrays of shape $\mathbf{N}$. In this analogy, the leaves of the search tree are each given one element in a multidimensional array, which has $|R|$ dimensions and N_j elements along dimension j. Alternatively, the act of advancing an index may be considered equivalent to a set of combined operations on a stack S.

Proposition 1 (Stack equivalence) *An index $\mathbf{I} \in \mathbb{N}^{|R|}$, combined with a level $j \leq |R|$, represents a stack S, such that $S = (I_1, I_2, I_3, \ldots, I_j)$. By introducing the following equivalence between index operations and stack operations,*

(i.) $I_k \leftarrow z$, $k > j$: *no operation.*
(ii.) $I_k \leftarrow z$, $k \leq j$:
  (a.) pop S until $|S| = k - 1$,
  (b.) push z onto S.
(iii.) $j \leftarrow k$, $k > j$: *push* 1 *onto S until $|S| = k$.*

we see that the advance *method maintains a stack within the first j elements of its index* $\mathbf{I}$.

Proof (Stack equivalence)

(i.) As S only contains the first j elements of $\mathbf{I}$, the modification of an index element I_k for $k > j$ does not modify S.
(ii.) This applies to lines 5 and 8 of Algorithm 1, and ensures that $S_k = I_k$ for $k \leq j$. Note that this changes the size of the stack, which is reflected in the algorithm by the least modified index j^*
(iii.) This ensures that the initialization ($j \leftarrow 1$) and the act of moving to the next tree level ($j \leftarrow j+1$) are equivalent to pushing the next available tree node onto the stack S. □

Using these indices, we may construct an iterative variant of *i*BP that implicitly traverses and prunes a search tree by advancing an index (Algorithm 2). The *calcAngle* and *calcTorsion* methods in Algorithm 2 calculate the bond angle θ_i and the cosine of the dihedral angle, τ_i, using equations 1 and 2 above, using the provided vertex coordinates to compute the required distances.

2.3 Ramachandran-defined vertex orders

In order to leverage known backbone dihedral angles when enumerating solutions to the *i*DMDGP, a new set of vertex orders was introduced. We define an initial order

Algorithm 2 The iterative *i*BP algorithm

1: **Input**: *i*DMDGP instance with graph G, repetition order R, and tree size $\mathbf{N}$.
2: **Output**: Set of solutions $\mathscr{X}$.
3: Initialize $\mathbf{I} \leftarrow \mathbf{1}$, $j = 1$, and $\mathscr{X} \leftarrow \emptyset$
4: **while** $\mathbf{I} \leq \mathbf{N}$ **do**
5: **while** $j \leq |R|$ **do**
6: **if** r_j is a repetition **then**
7: Copy $\mathbf{x}_j$ from the original embedding $\mathbf{x}_i$ {where $i < j$ s. t. $r_i = r_j$}
8: **else**
9: $\delta \leftarrow d_{j-1,j}$
10: $\theta \leftarrow$ calcAngle($\mathbf{x}_{j-1}$, $\mathbf{x}_{j-2}$, $d_{j-1,j}$, $d_{j-2,j}$)
11: **if** $\exists \omega_j \in \Omega$ **then**
12: $\ell_\omega \leftarrow \frac{I_j}{N_j}(\overline{\omega}_j - \underline{\omega}_j) + \underline{\omega}_j)$
13: $\sigma \leftarrow \sin\ell_\omega / |\sin\ell_\omega|$ {Or 1, if $\sin\ell_\omega = 0$}
14: $\tau \leftarrow \cos\ell_\omega$
15: **else**
16: $\sigma \leftarrow 2(I_j \bmod 2) - 1$
17: $\ell_d \leftarrow \frac{\lfloor I_j/2 \rfloor}{N_j/2}(\overline{d}_{j-3,j} - \underline{d}_{j-3,j}) + \underline{d}_{j-3,j}$
18: $\tau \leftarrow$ calcTorsion($\mathbf{x}_{j-1}$, $\mathbf{x}_{j-2}$, $\mathbf{x}_{j-3}$, $d_{j-1,j}$, $d_{j-2,j}$, ℓ_d)
19: **end if**
20: Embed $\mathbf{x}_j$ given $\mathbf{x}_{j-1}$, $\mathbf{x}_{j-2}$, $\mathbf{x}_{j-3}$, δ, θ, τ, σ
21: **if** vertex $\mathbf{x}_j$ is infeasible **then**
22: $(\mathbf{I}, j) \leftarrow$ advance($\mathbf{I}$, $\mathbf{N}$, j)
23: **break** {Jump to outermost **while** (line 4).}
24: **end if**
25: **end if**
26: **if** $j = |R|$ **then**
27: $\mathscr{X} \leftarrow \mathscr{X} \cup \mathbf{x}$ {The current embedding $\mathbf{x}$ is a solution}
28: **end if**
29: $j \leftarrow j + 1$
30: **end while**
31: $(\mathbf{I}, j) \leftarrow$ advance($\mathbf{I}$, $\mathbf{N}$, $|R|$)
32: **end while**

R_1, an inner order R_i and a final order R_n for the first, inner, and last amino acid units of protein graphs, respectively. The vertex orders are as follows:

$$R_1 = \left\{N^{(1)},\, H_1^{(1)},\, H_2^{(1)},\, C_\alpha^{(1)},\, N^{(1)},\, H_\alpha^{(1)},\, C_\alpha^{(1)},\, C'^{(1)}\right\}$$

$$\begin{aligned} R_i = \big\{&N^{(i)},\, O^{(i-1)},\, C_\alpha^{(i-1)},\, C'^{(i-1)},\, N^{(i)},\, C_\alpha^{(i)},\, C'^{(i)},\, N^{(i+1)},\\ &C^{(i-1)},\, N^{(i)},\, C_\alpha^{(i)},\, H_1^{(i)},\, N^{(i)},\, C_\alpha^{(i)},\, C'^{(i)},\, H_\alpha^{(i)},\, C'^{(i)},\, C_\alpha^{(i)}\big\} \end{aligned}$$

$$\begin{aligned} R_n = \big\{&N^{(n)},\, O^{(n-1)},\, C_\alpha^{(n-1)},\, C'^{(n-1)},\, N^{(n)},\, C_\alpha^{(n)},\\ &C'^{(n)},\, C'^{(n-1)},\, N^{(n)},\, C_\alpha^{(n)},\, H_1^{(n)},\, N^{(n)},\, C_\alpha^{(n)},\, C'^{(n)},\\ &H_\alpha^{(n)},\, C'^{(n)},\, C_\alpha^{(n)},\, O_1^{(n)},\, C'^{(n)},\, O_2^{(n)}\big\} \end{aligned}$$

where i denotes the amino acid unit index and n denotes the total number of amino acid units within this context. These orders make extensive use of repeated vertices in order to achieve direct branching on the backbone dihedrals ϕ, ψ and ω.

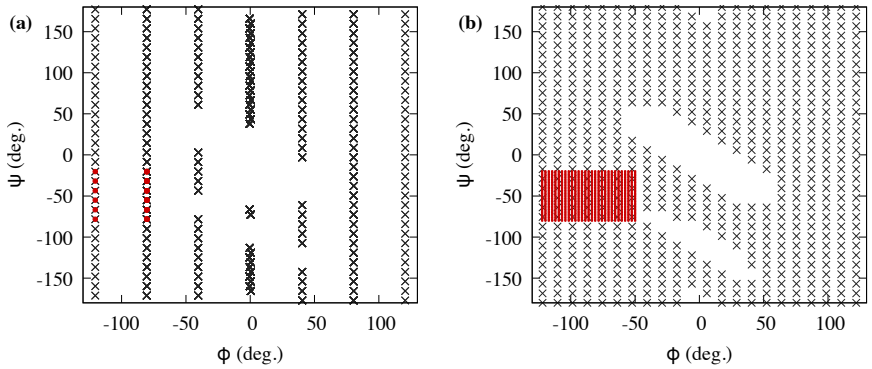

Fig. 4 Reorder-dependent differences in sampling of Ramachandran space, using identical branching factors ($B = 64$). Black crosses and red dots represent conformations sampled with and without application of dihedral restraints to (ϕ, ψ), respectively. **a.** Old repetition orders. **b.** New Ramachandran-defined repetition orders.

Fig. 4 illustrates how the new orders more directly and uniformly sample Ramachandran space than previously published vertex orders [7]. An *i*BP calculation was performed, using both vertex orders, to find solutions to the *i*DMDGP instance of a backbone-only tetrapeptide, with the direct distance feasibility (DDF, [5]) and van der Waals (VDW) pruning devices enabled. Because the previous orders do not branch on the ϕ or ω backbone dihedral angles of proteins, they do not enable *i*BP to cover this space in a regular way, and often sample nearby points in (ϕ, ψ)-space more than once during tree traversal (Fig. 4a). An even stronger difference in behaviour is observed when constraints are placed on an instance's ϕ and ψ angles (Fig. 4, red dots). Using previously published orders, *i*BP treats the dihedral constraints as extra pruning edges, thus maintaining the same irregular sampling pattern and pruning any partial conformations that fall outside the constraints. Using the new orders that directly branch on protein backbone dihedral angles, *i*BP adaptively changes its sampling pattern (Fig. 4b, red dots) to better cover the region specified by the constraints. The red rectangle in the second panel of Fig. 4 is indeed a regular grid of sampled conformations. The central regions of (ϕ, ψ)-space were not sampled by either vertex order, as they were pruned by the VDW device.

3 Implementation-specific technical innovations

3.1 CHARMM-syntax force fields

The basic input to protein structure determination procedures is the amino acid sequence of the protein of interest. From the sequence, software packages draw information from "force field" libraries in order to construct the topology (graph structure) and parameters (graph edge weighting) of the target molecule. Routinely employed software packages [26,2,3,22] employ a common force field syntax derived from the *Chemistry at HARvard Molecular Mechanics* (CHARMM) software package [2].

To ensure extensibility, flexibility and interoperability with these packages, our implementation of *i*BP also uses CHARMM-syntax force fields to construct *i*DMDGP instances. In addition, *i*BP accepts a superset of the CHARMM parameter file syntax that allows users to specify interval distances, angles and dihedrals within force fields. The force field used by *i*BP was derived from the protein based force field of the Crystallography and NMR System (CNS, [3]) with minor additions. For example, the topology information of a protein backbone is defined in *i*BP using the following notation:

```
! BBI: interior (1<i<n) backbone-only residue
residue BBI
 group
  atom N  type=NH1  charge=-0.36 end
  atom HN type=H    charge= 0.26 end
  atom CA type=CH1E charge= 0.00 end
  atom HA type=HA   charge= 0.10 end
  atom C  type=C    charge= 0.48 end
  atom O  type=O    charge=-0.48 end
 bond N  HN      bond N  CA      bond CA HA
 bond CA C       bond C  O
 dihedral HN N CA HA
end
```

Vertex orders in *i*BP are also specified using a custom syntax that retains the flexibility of CHARMM-style force fields. Each amino acid unit defined in the force field topology files is given a corresponding order. As an example, the Ramachandran-defined vertex order of a protein backbone corresponding to the above topology entry is specified as follows:

```
reorder BBI
 N, -O, -CA, -C, N, CA, C, +N,
 -C, N, CA, H1, N, CA, C, HA, C, CA
end
```

where `-O` denotes atom `O` of the previous amino acid unit, and `+N` denotes atom `N` of the next. At runtime, *i*BP constructs the *i*DMDGP instance of a protein from its sequence using this combined topology, parameter, and vertex order information. The validity of the resulting problem instance is then checked, and tree traversal is initiated *via* Algorithm 2. Therefore, it is now straightforward to use *i*BP within routine structure determination efforts. In addition, this flexibility enables *i*BP to handle structure determination involving post-translational modifications and non-natural amino acids, provided a repetition order can be crafted.

3.2 Pruning and timing statistics

Our implementation of *i*BP admits the inclusion of new pruning devices *via* a system of callback functions. On tree initialization, each pruning device registers its callbacks at each level of the tree. During tree traversal, *i*BP executes each registered

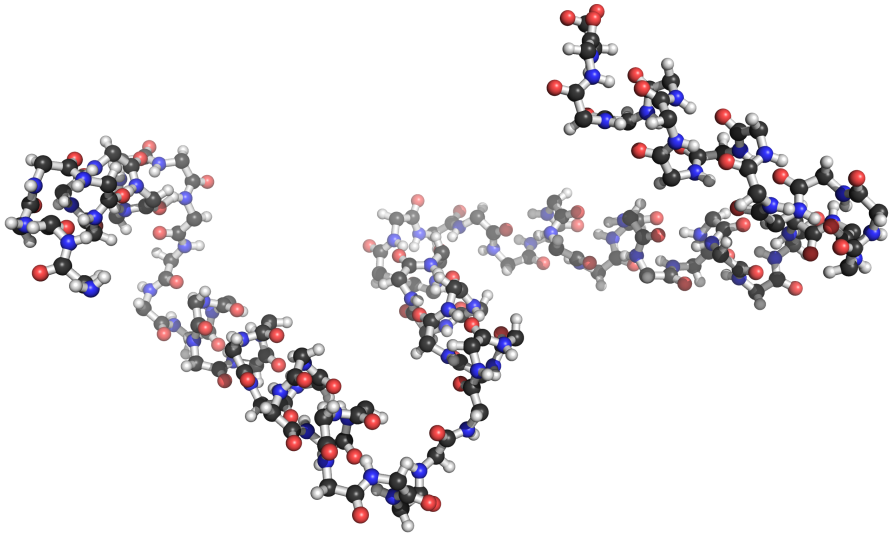

Fig. 5 The first solution returned by *i*BP for the HHD2 *i*DMDGP instance, using only dihedral constraints predicted from NMR chemical shifts.

callback function to determine the feasibility of newly embedded vertices. Finally, when traversal is completed or terminated prematurely by the user, each pruning device outputs vertex-specific pruning results. This is useful for identifying any distance, angle or torsion constraints that are geometrically inconsistent, for example.

Furthermore, the index-based traversal algorithm enables *i*BP to roughly estimate its runtime. Given an index $\mathbf{I}$ in a tree of size $\mathbf{N}$, we define the "width" of the sub-tree from $\mathbf{1}$ to $\mathbf{I}$ as the number of leaves in that sub-tree,

$$w(\mathbf{I},\mathbf{N}) = \sum_{j=1}^{|R|}(I_j - 1)\prod_{k=j+1}^{|R|} N_k$$

Therefore, the width of the entire tree is $w(\mathbf{N},\mathbf{N})$+1, and the width of the yet-untraversed portion of the tree is given by $w(\mathbf{N}-\mathbf{I}+\mathbf{1},\mathbf{N})$. Assuming the rate of tree traversal—defined as the number of leaf nodes traversed per unit time—is relatively constant on average, we estimate the remaining runtime of an *i*BP tree traversal as follows:

$$t_r = \frac{w(\mathbf{N}-\mathbf{I}+\mathbf{1},\mathbf{N})}{w(\mathbf{I},\mathbf{N})}t_e$$

where t_r and t_e are the remaining and elapsed runtimes, respectively. Using these equations, *i*BP may output tree size and periodic timing information to the user. This assumption of a constant traversal—and thus pruning—rate of *i*BP is ensured by measuring $w(\mathbf{N}-\mathbf{I}+\mathbf{1},\mathbf{N})$ at sufficiently spaced time intervals.

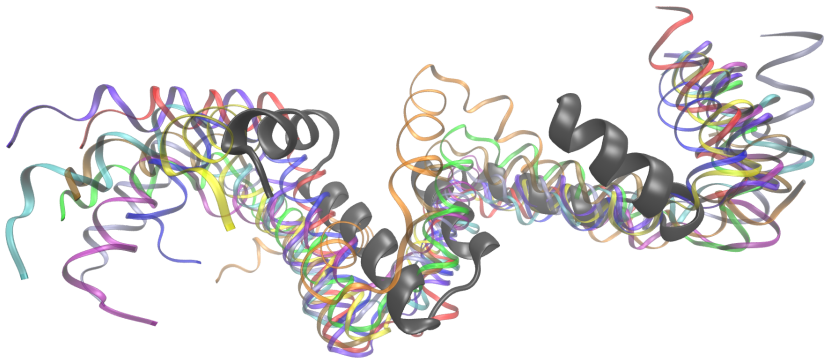

Fig. 6 Comparison of the first solution returned by *i*BP for the HHD2 *i*DMDGP instance (Fig. 5, shown here in dark grey) with the ten lowest-energy structures from ARIA/CNS (colored, thin traces), using the larger initial structure set and slower annealing rate.

4 Computational results

4.1 Experimental data

The vertex orders in this work are ideally suited to pairing with backbone dihedral angles, such as those predicted from measured NMR chemical shifts (e.g. by TALOS-N, [23]). To illustrate this, the structure of an HHD2 protein domain (residues G420-R499 of Whirlin isoform-4) was solved by constraining all backbone dihedral angles to the intervals predicted by TALOS-N (Fig. 5). NMR chemical shifts were measured from a uniformly ^{15}N,^{13}C-labeled protein sample that was prepared at a concentration of 300 μM in 250 μL of a buffer solution (150 mM NaCl, 50 mM Tris-HCl, 5% D2O, pH 7.5). All data were collected at 25°C on a Bruker Avance III 900 MHz spectrometer with a three-channel cryogenically cooled probe. The following experiments were performed, yielding chemical shifts for the following types of atomic nuclei:

- BT-HSQC [14]: H_1, N. (cf. Fig. 2)
- HNHA [25]: H_1, N, H_α.
- BT-HNCO [14], BT-HNCO+ [8]: H_1, N, C'.
- BT-HNCACB, BT-HNCOCACB [14]: H_1, N, C_α, C_β.

From these experiments, about 99% of the expected backbone chemical shifts were assigned (99% of H_1, N, C_α, C_β and C'; 91% of H_α). The assigned chemical shifts, along with the amino acid sequence of HHD2, were then used to obtain predicted intervals for ϕ and ψ backbone dihedral angles using TALOS-N. The ω backbone dihedral angles were fixed to 180°, following standard practice in the structural bioinformatics field. The resulting *i*DMDGP instance contained 464 vertices, and had a vertex order length $|R| = 1378$.

4.2 Structure calculations

To obtain the structure illustrated in Fig. 5, the iterative *i*BP variant described in Algorithm 2 was run using the Ramachandran-defined vertex orders described in Section 2.3. A branching factor of $B = 16$ and a branch epsilon of $\varepsilon = 0.01$ Å were used, resulting in an effective branching factor $B_{\text{eff},j}$ that varies at each level of the tree according to the following equation:

$$B_{\text{eff},j} \triangleq \min\left\{B, \left\lfloor \frac{\overline{d}_{j,j-3} - \underline{d}_{j,j-3}}{\varepsilon} \right\rfloor\right\}$$

The direct distance feasibility (DDF, [5]) and van der Waals (VDW) pruning devices were enabled during *i*BP tree traversal, as well as a new pruning device that tested for direct dihedral angle feasibility. In short, dihedral feasibility pruning ensures that all quartets of atoms with corresponding dihedral angles in Ω are consistent with their respective dihedral constraints. This dihedral feasibility pruning device effectively generalizes previously developed *i*BP pruning devices related to proteins, including the α-helix and chirality pruning devices introduced in [5].

To provide a basis for comparison, conventional NMR structure calculations were performed by molecular dynamics simulated annealing (MDSA) using ARIA/CNS [3,20], which was recently evaluated as one of the most effective software tools for NMR structure determination [21]. In MDSA, the motion of each protein structure is simulated by numerically solving Newton's equations of motion from various initial velocities, during which the temperature of the system is reduced from $\sim 10^4$ K to ~ 50 K. The structures from simulated annealing are then subjected to a round of local descent. For both the dynamics and descent stages, the objective function is an approximation of the molecule's potential energy, which includes terms for the known local geometry and the NMR-derived geometric constraints [18].

Within ARIA/CNS, default parameters were used to randomly generate two sets of 20 and 100 initial structures, based solely on the predicted (ϕ, ψ) angles from TALOS-N. The smaller set of structures was subjected to local descent using a short MDSA calculation, whereas the larger set was subjected to a longer MDSA calculation with a ten-fold slower annealing rate.

While complete traversal of the search tree, which contained 10^{147} leaves, was estimated to require 10^{140} minutes, the solution illustrated in Fig. 5 was obtained by *i*BP within a few seconds, thanks to the strong constraints supplied from TALOS-N. In contrast, average times of the short and long MDSA computations were 29 seconds and 186 seconds, respectively. However, none of the structures produced by short MDSA were in the dihedral angle feasible set, and only 71% of structures from long MDSA were feasible. A comparison of the overall folds produced by *i*BP and the long MDSA run of ARIA/CNS is given in Fig. 6; while the secondary structures and general fold are the same among all structures, only the *i*BP structure is guaranteed to be feasible in the *i*DMDGP.

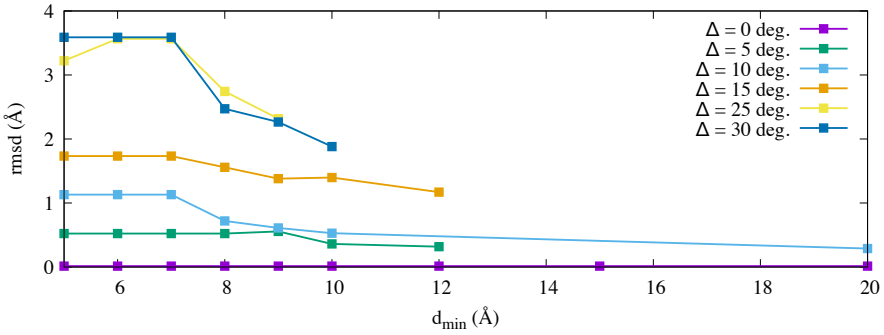

Fig. 7 Summary of root mean square deviations (rmsd) between the first solution produced by *i*BP from the TALOS instance and the target structure, as a function of dihedral uncertainty (Δ) and distance threshold d_{min}. Full results are listed in Tables 1, 2, and 3.

4.3 Sensitivity to uncertainty

In general graphs, *i*BP exhibits exponential worst-case complexity [17]. Fortunately, the fact that pruning edges of similar length are distributed sufficiently uniformly over the vertex order ensures that branches do not occur frequently, resulting in tractable instances. However, when intervals are employed for branching in *i*BP, discretization can potentially spoil the favorable properties of the method [9]. Lower discretization factors B increase the probability that no path through the search tree remains feasible after applying the discretization. Conversely, increasing the discretization factor grows the search tree exponentially, resulting in dramatically longer runtimes.

To analyze the sensitivity of the proposed *i*BP algorithm and vertex order to interval uncertainty, a set of experiments was performed using the first solution from the TALOS instance (the *target structure*) as a basis. First, dihedral restraints were obtained by computing the backbone ϕ and ψ dihedral angles from the target structure, and varying degrees of uncertainty (Δ) were added to the dihedrals of residues 11–13, 30–34, 46–49, and 61–65. These residues were found in "loop" regions of the target structure, which are generally more flexible in protein structures. In addition, long-range distance restraints were obtained by collecting the following set of distances from the target structure:

$$\mathscr{D} = \{\|\mathbf{x}_i - \mathbf{x}_j\| < d_{max} \; ; \; |i-j| \geq 5\}$$

where d_{max} specifies the maximum admissible distance in the set. The final distance restraints were obtained by forming intervals 0.5Å wide around each distance in $\mathscr{D}$. We shall refer to the size of the set $\mathscr{D}$ as m_{dist} in all further discussions. For the problem instances resulting from each pair of values (Δ, d_{max}), *i*BP was run using identical parameters to those used to solve the original TALOS instance, with the exception that the DDF tolerance was expanded to 0.1Å. The time required for *i*BP to obtain a single solution, subject to a time limit of 5 hours, was recorded as t_1 for each instance, and the root mean square deviation of each obtained solution to the

target structure was also recorded (Fig. 7). The complete set of results of this analysis is given in Tables 1, 2, and 3.

This analysis illustrates several general behaviors of *i*BP for solving protein instances. When dihedrals are specified with high precision (Table 1), the conformational space is small, and *i*BP rapidly obtains a solution. As Δ is increased, *i*BP still rapidly obtains a solution when only a few long-range distances are supplied. However, their deviations from the target structure increase with Δ, and supplying more distances decreases the deviations at the expense of increased computation time. Finally, the inclusion of a large number of distance restraints tends to increase runtime past the 5 hour limit, especially when the conformational space is large due to large Δ. While one may expect runtimes to decrease as more distances are added, due to the graph becoming more complete [17], the opposite effect is observed. This is a result of the negative interplay between interval discretization and distance pruning in *i*BP, and might be alleviated using error-tolerant pruning devices such as mean distance error (MDE) [9].

Finally, it is important to note that none of the artificial instances produced from the original TALOS structure generated solutions when running *i*BP with previously defined vertex orders [7], due to the irregular sampling patterns of Ramachandran space that those orders produce.

5 Conclusions

This paper introduces a new implementation of the *interval* Branch-and-Prune algorithm for molecular structure determination [13], and describes several modifications that provide enhanced performance for the specific task of protein structure elucidation. Comparison against an existing state of the art method in the field demonstrates the advantages of *i*BP in practical structure determination problems. Our new *i*BP implementation is completely open source, and is available under the MIT license at `http://github.com/geekysuavo/ibp-ng`.

Acknowledgements Bradley Worley thanks the Pasteur Foundation for postdoctoral support. The authors wish to thank Institut Pasteur, Ecole Polytechnique, CNRS (TGIR-RMN-THC FR3050), and the French research agency (ANR-10-BINF-03-08 “Bip:Bip”) for financial support. Carlile Lavor wishes to acknowledge the Brazilian research agencies CNPq and FAPESP for support.

Table 1 Results of solving the TALOS instance under low dihedral angle uncertainties ($\Delta = 0^\circ, 1^\circ, 2^\circ$).

Δ (°)	d_{max} (Å)	m_{dist}	t_1 (s)	rmsd (Å)
0	5	0	1.7	0.014
0	6	1	1.7	0.014
0	7	6	1.7	0.014
0	8	29	1.7	0.014
0	9	81	1.7	0.014
0	10	163	1.7	0.014
0	12	314	1.7	0.014
0	15	625	1.7	0.014
0	20	1150	1.7	0.014
1	5	0	1.7	0.014
1	6	1	1.7	0.014
1	7	6	1.7	0.014
1	8	29	1.6	0.014
1	9	81	1.7	0.014
1	10	163	1.7	0.014
1	12	314	1.7	0.014
1	15	625	1.7	0.014
1	20	1150	1.7	0.014
2	5	0	1.8	0.014
2	6	1	1.7	0.014
2	7	6	1.7	0.014
2	8	29	1.7	0.014
2	9	81	1.7	0.014
2	10	163	1.7	0.014
2	12	314	1.7	0.014
2	15	625	1.7	0.014
2	20	1150	1.7	0.014

Table 2 Results of solving the TALOS instance under moderate dihedral angle uncertainties ($\Delta = 5°, 10°, 15°$).

Δ (°)	d_{max} (Å)	m_{dist}	t_1 (s)	rmsd (Å)
5	5	0	1.7	0.522
5	6	1	1.7	0.522
5	7	6	1.7	0.522
5	8	29	1.6	0.522
5	9	81	1.7	0.555
5	10	163	2.1	0.360
5	12	314	96.4	0.317
5	15	625	–	–
5	20	1150	–	–
10	5	0	1.3	1.130
10	6	1	1.2	1.130
10	7	6	1.2	1.130
10	8	29	1.7	0.720
10	9	81	79.1	0.611
10	10	163	638.0	0.529
10	12	314	–	–
10	15	625	–	–
10	20	1150	1378.0	0.289
15	5	0	0.8	1.732
15	6	1	0.8	1.732
15	7	6	0.8	1.732
15	8	29	1.2	1.557
15	9	81	1.3	1.381
15	10	163	18.9	1.398
15	12	314	1400.8	1.169
15	15	625	–	–
15	20	1150	–	–

Table 3 Results of solving the TALOS instance under high dihedral angle uncertainties ($\Delta = 20°, 25°, 30°$).

Δ (°)	d_{max} (Å)	m_{dist}	t_1 (s)	rmsd (Å)
20	5	0	1.4	2.028
20	6	1	1.3	2.028
20	7	6	1.5	2.028
20	8	29	2.8	1.334
20	9	81	4.4	2.130
20	10	163	8.2	1.632
20	12	314	1004.2	1.389
20	15	625	–	–
20	20	1150	–	–
25	5	0	1.4	3.222
25	6	1	1.3	3.566
25	7	6	0.8	3.566
25	8	29	3.2	2.742
25	9	81	3.3	2.317
25	10	163	–	–
25	12	314	–	–
25	15	625	–	–
25	20	1150	–	–
30	5	0	1.6	3.589
30	6	1	2.1	3.589
30	7	6	1.9	3.589
30	8	29	17.1	2.472
30	9	81	20.7	2.266
30	10	163	106.5	1.882
30	12	314	–	–
30	15	625	–	–
30	20	1150	–	–

References

1. Berkholz, D.S., Driggers, C.M., Shapovalov, M.V., Dunbrack, R.L., Karplus, P.A.: Nonplanar peptide bonds are common and conserved but not biased toward active sites. Proc. Natl. Acad. Sci. USA **109**, 449–453 (2012)
2. Brooks, B.R., Bruccoleri, R.E., Olafson, B.D., States, D.J., Swaminathan, S., Karplus, M.: CHARMM: A program for macromolecular energy, minimization, and dynamics calculations. J. Comp. Chem. **4**, 187–217 (1983)
3. Brünger, A.T., Adams, P.D., Clore, G.M., DeLano, W.L., Gros, P., Grosse-Kunstleve, R.W., Jiang, J.S., Kuszewski, J., Nilges, M., Pannu, N.S., Read, R.J., Rice, L.M., Simonson, T., Warren, G.L.: Crystallography and NMR System: A new software suite for macromolecular structure determination. Acta Cryst. **D54**, 905–921 (1998)
4. Brünger, A.T., Nilges, M.: Computational challenges for macromolecular structure determination by X-ray crystallography and solution NMR-spectroscopy. Q. Rev. Biophys. **26**, 49–125 (1993)
5. Cassioli, A., Bardiaux, B., Bouvier, G., Mucherino, A., Alves, R., Liberti, L., Nilges, M., Lavor, C., Malliavin, T.E.: An algorithm to enumerate all possible protein conformations verifying a set of distance constraints. BMC Bioinformatics **16**, 23–37 (2015)
6. Cassioli, A., Günlük, O., Lavor, C., Liberti, L.: Discretization vertex orders in distance geometry. Discrete Appl. Math. **197**, 27–41 (2015)
7. Costa, V., Mucherino, A., Lavor, C., Cassioli, A., Carvalho, L.M., Maculan, N.: Discretization orders for protein side chains. J. Glob. Optim. **60**, 333–349 (2014)
8. Gil-Caballero, S., Favier, A., Brutscher, B.: HNCA+, HNCO+, and HNCACB+ experiments: improved performance by simultaneous detection of orthogonal coherence transfer pathways. J. Biomol. NMR **60**, 1–9 (2014)
9. Goncalves, D.S., Mucherino, A., Lavor, C., Liberti, L.: Recent advances on the interval distance geometry problem. J. Glob. Optim. pp. 1–21 (2017)
10. Hinsen, K., Hu, S., Kneller, G.R., Niemi, A.J.: A comparison of reduced coordinate sets for describing protein structure. J. Chem. Phys. **139**, 124,115 (2013)
11. Ikura, M., Kay, L.E., Bax, A.: A novel approach for sequential assignment of ^{1}H, ^{13}C and ^{15}N spectra of proteins: heteronuclear triple-quantum resonance three-dimensional NMR spectroscopy. Application to calmodulin. Biochemistry **15**, 4659–4667 (1990)
12. Lavor, C., Alves, R., Figueiredo, W., Petraglia, A., Maculan, N.: Clifford Algebra and the Discretizable Molecular Distance Geometry Problem. Adv. Appl. Clifford Algebras **25**, 925–942 (2015)
13. Lavor, C., Liberti, L., Mucherino, A.: The interval Branch-and-Prune algorithm for the discretizable molecular distance geometry problem with inexact distances. J. Glob. Optim. **56**, 855–871 (2013)
14. Lescop, E., Schanda, P., Brutscher, B.: A set of BEST triple-resonance experiments for time-optimized protein resonance assignment. J. Magn. Reson. **187**, 163–169 (2007)
15. Levitt, M.H.: Spin Dynamics: Basics of Nuclear Magnetic Resonance. Wiley (2008)
16. Liberti, L., Lavor, C., Maculan, N., Mucherino, A.: Euclidean Distance Geometry and Applications. SIAM Rev. **56**, 3–69 (2014)
17. Liberti, L., Lavor, C., Mucherino, A.: The Discretizable Molecular Distance Geometry Problem seems Easier on Proteins. Springer New York
18. Mucherino, A., Lavor, C., Liberti, L., Maculan, N.: Finding low-energy homopolymer conformations by a discrete approach. In: Global Optimization Workshop 2012. Natal, Brazil (2012)
19. Ramachandran, G.N., Ramakrishnan, C., Sasisekharan, V.: Stereochemistry of polypeptide chain configurations. J. Mol. Biol. **7**, 95–99 (1963)
20. Rieping, W., Habeck, M., Bardiaux, B., Bernard, A., Malliavin, T.E., Nilges, M.: ARIA2: automated NOE assignment and data integration in NMR structure calculation. Bioinformatics **23**, 381–382 (2007)
21. Rosato, A., Vranken, W., Fogh, R.H., Ragan, T.J., Tejero, R., Pederson, K., Lee, H.W., Prestegard, J.H., Yee, A., Wu, B., Lemak, A., Houliston, S., Arrowsmith, C.H., Kennedy, M., Acton, T.B., Xiao, R., Liu, G., Montelione, G.T., Vuister, G.W.: The second round of Critical Assessment of Automated Structure Determination of Proteins by NMR: CASD-NMR-2013. J. Biomol. NMR **62**, 2728–2733 (2013)
22. Schwieters, C.D., Kuszewski, J.J., Tjandra, N., Clore, G.M.: The Xplor-NIH NMR Molecular Structure Determination Package. J. Magn. Reson. **160**, 66–74 (2003)
23. Shen, Y., Bax, A.: Protein backbone and sidechain torsion angles predicted from NMR chemical shifts using artificial neural networks. J. Biomol. NMR **56**, 227–241 (2013)

24. Thompson, H.B.: Calculation of Cartesian Coordinates and Their Derivatives from Internal Molecular Coordinates. J. Chem. Phys. **47**, 3407–3410 (1967)
25. Vuister, G.W., Bax, A.: Quantitative J correlation: a new approach for measuring homonuclear three-bond J_{HNH_α} coupling constants in ^{15}N-enriched proteins. J. Am. Chem. Soc. **115**, 7772–7777 (1993)
26. Weiner, P.K., Kollman, P.A.: AMBER: Assisted model building with energy refinement. A general program for modeling molecules and their interactions. J. Comp. Chem. **2**, 287–303 (1981)
27. Wüthrich, K.: NMR of Proteins and Nucleic Acids. Wiley (1986)