Influence of Pruning Devices on the Solution of Molecular Distance Geometry Problems

A. Mucherino[1], C. Lavor[2], T. Malliavin[3], L. Liberti[4], M. Nilges[3], and N. Maculan[5]

[1] CERFACS, Toulouse, France, mucherino@cerfacs.fr
[2] IMECC, UNICAMP, Campinas, SP, Brazil, clavor@ime.unicamp.br
[3] Institut Pasteur, Paris, France, {terez,michael.nilges}@pasteur.fr
[4] LIX, École Polytechnique, Palaiseau, France, liberti@lix.polytechnique.fr
[5] COPPE, Federal University of Rio de Janeiro, Rio de Janeiro, RJ, Brazil, maculan@cos.ufrj.br

Abstract. The Molecular Distance Geometry Problem (MDGP) is the problem of finding the conformation of a molecule from inter-atomic distances. In some recent work, we proposed the *interval* Branch & Prune (*i*BP) algorithm for solving instances of the MDGP related to protein backbones. This algorithm is based on an artificial ordering given to the atoms of the protein backbones which allows the discretization of the problem, and hence the applicability of the *i*BP algorithm. This algorithm explores a discrete search domain having the structure of a tree and prunes its infeasible branches by employing suitable pruning devices. In this work, we use information derived from Nuclear Magnetic Resonance (NMR) to conceive and add new pruning devices to the *i*BP algorithm, and we study their influence on the performances of the algorithm.

1 Introduction

Proteins are important molecules formed by chains of smaller molecules called amino acids. Several experimental techniques, as Nuclear Magnetic Resonance (NMR), are able to provide some information on interatomic distances in protein molecules which can be exploited for obtaining the three-dimensional conformation of the protein. As the protein conformation often enables to give good clues about the protein function, the conformation determination is of fundamental importance. The problem of finding the protein conformation from a list of inter-atomic distances is known in the scientific literature as the Molecular Distance Geometry Problem (MDGP) [4]. By nature, the MDGP is a constraint satisfaction problem, but its solution is usually attempted by employing global optimization techniques [10]. It usually requires a search in a continuous space which is a subset of $\mathbb{R}^{3n}$, where n is the number of atoms forming the molecule. It has been proved that the MDGP is an NP-hard problem [16].

Since 2006 we have been working on a combinatorial reformulation of the MDGP. Under suitable assumptions, we are able to discretize the problem and to reduce the search on a discrete search domain. Even though the problem is still

NP-hard after the discretization [3], it can be efficiently solved by employing a Branch & Prune (BP) algorithm [9]. It is important to remark that this algorithm is able to find *all* solutions to the problem, differently from other algorithms based on continuous formulations and/or heuristics [10, 15].

We refer to this combinatorial reformulation of the MDGP as Discretizable MDGP (DMDGP) [3]. Let $G = (V, E, d)$ be a weighted undirected graph representing an instance of the problem: each vertex in V corresponds to an atom, and there is an edge in E between two vertices if and only if the distance between the corresponding atoms is known (the distance value is given by the associated weight d). In order to have the combinatorial reformulation, we need two assumptions to be satisfied for a given ordering on the vertices in V. By Assumption 1, the edge set E must contain all cliques on quadruplets of consecutive vertices, that is,

$$\forall i \in \{4, \ldots, n\} \; \forall j, k \in \{i-3, \ldots, i\} \qquad (\{j, k\} \in E)$$

and, by Assumption 2, the following strict triangular inequality

$$\forall i = 2, \ldots, n-1, \quad d_{i-1,i+1} < d_{i-1,i} + d_{i,i+1},$$

must hold.

Assumption 1 ensures that the distances between each possible pair of atoms in any quadruplet of consecutive atoms are known. Moreover, if Assumption 2 holds, there cannot be triplets of consecutive atoms that are perfectly aligned. Supposing that positions for the atoms are searched by following the same ordering given to the vertices of V, there exist at most two possible positions in which each atom can be placed if these two assumptions are satisfied. This leads to the definition of a discrete search domain, which has the structure of a tree. This tree can be constructed in the practice by exploiting distances that must be known by Assumption 1. Moreover, the considered instance can also contain other distances, that we can use for pruning branches of the tree in order to focus our searches on its feasible branches only. This is the main idea behind the BP algorithm [9].

The basic version of this algorithm has however two main limitations. First of all, exact distances should be available in order to construct the discrete search domain, whereas real-life NMR experiments are usually noisy, so that lower and upper bounds on the distances are actually known. Moreover, given any atom of the protein, there must be at least 3 distances concerning this atom, otherwise Assumption 1 cannot be satisfied. This property is quite difficult to be satisfied by NMR instances, because the number of available distances is usually not sufficient, and only distances related to particular atoms, mainly pairs of hydrogens, are actually available. Therefore, even though the BP algorithm is extremely efficient in its basic version, it is unfortunately mainly suitable for simulated instances of the DMDGP, and not for NMR instances.

We recently overcame these two issues by introducing a hand-craft ordering for the atoms of the protein backbones, and by proposing an extension of the BP algorithm which is based on such an ordering. This ordering allows us to discretize a full class of MDGPs, the one which is related to protein backbones,

even if only noisy distances between pairs of hydrogens are available. This is possible because all distances used in the construction of the discrete search domain can be computed a priori by information on the chemical composition of the protein backbones. The distances obtained by NMR are only used for pruning purposes. We shall refer to this extension of the BP algorithm as *interval* BP (*i*BP) [6].

The *i*BP algorithm is able to consider NMR instances related to protein backbones. However, in our previous publications [5, 6], we only presented computational experiments where simulated data were considered. Indeed, when we firstly tried to solve NMR instances by *i*BP, we found out that the available information on the distances was not sufficient for efficiently pruning the search domain. For this reason, we decided to conceive new pruning devices, with the aim of identifying sooner during the search infeasible parts of the search domain. These new pruning devices are all based on other information (rather than distances) that NMR experiments can provide. We also analyze the influence of each newly added pruning device on the performances of the *i*BP algorithm.

The rest of the paper is organized as follows. In Section 2, we give a brief description of the *i*BP algorithm and of the artificial ordering for the protein backbones which allows the discretization of the problem. New pruning devices are presented in details in Section 3, and computational experiments on NMR instances are given in Section 4. Conclusions are drawn in Section 5.

2 The interval Branch & Prune

In order to solve DMDGPs where interval data are considered, we recently defined an artificial ordering for the atoms of the protein backbones. In this section, we describe this particular artificial ordering and we discuss the *i*BP algorithm, that is based on this ordering. For more details, the interested reader is referred to [5, 6].

Let us start by assigning the following ordering to the atoms of the first amino acid of the considered protein:

$$r^1_{PB} = \{N^1, H^1, H^0, C^1_\alpha, N^1, H^1_\alpha, C^1_\alpha, C^1\}.$$

Note that the superscripts indicate the amino acid to which each atom belongs. One of the hydrogens bound to N^1 (in general, there is only one hydrogen) is indicated by the symbol H^0. The carbon C^1_α and the nitrogen N^1 appear twice in the sequence. This is done in order to reduce the relative distances between pairs of atoms in the ordering, and also in order to consider the distances between copies of the same atom (that must be equal to 0). The other carbon of the first amino acid, the atom C^1, is considered, in this case, only once. Let us now assign the following ordering to the atoms of the second amino acid:

$$r^2_{PB} = \{N^2, C^2_\alpha, H^2, N^2, C^2_\alpha, H^2_\alpha, C^2, C^2_\alpha\}.$$

This sequence of atoms is used for building a *bridge* between the first amino acid, and the third one, from which a generic ordering is considered. In fact, the

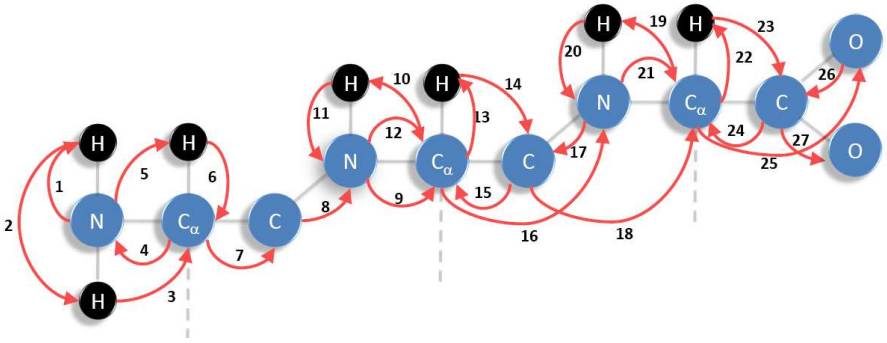

Fig. 1. The hand-craft artificial ordering r_{PB}.

ordering defined on the second amino acid is quite similar to the generic one. Atoms are considered more than once, and, in particular, the carbon C_α^2 appears in the sequence 3 times. This is the ordering for the generic amino acid (from the third to last but one):

$$r_{PB}^i = \{N^i, C^{i-1}, C_\alpha^i, H^i, N^i, C_\alpha^i, H_\alpha^i, C^i, C_\alpha^i\}.$$

The nitrogen N^i is considered twice, the carbon C_α^i is considered 3 times, and the carbon C^{i-1} belonging to the previous amino acid is repeated among the atoms of the amino acid i. In total, for each amino acid, we have 4 copies of atoms that already appeared somewhere else in the sequence. Note that hydrogen atoms are never duplicated. Since the last amino acid contains a few atoms more, this is the ordering that we consider:

$$r_{PB}^p = \{N^p, C^{p-1}, C_\alpha^p, H^p, N^p, C_\alpha^p, H_\alpha, C^p, C_\alpha^p, O^p, C^p, O^{p+1}\}.$$

Note that this is the only case in which oxygen atoms appear. The two oxygens O^p and O^{p+1} present in the last residue r_{PB}^p correspond to the two oxygens of the C-terminal carboxyl group COO^- of the protein.

Let us indicate by the symbol r_{PB} the defined artificial ordering on the whole protein backbone:

$$r_{PB} = \{r_{PB}^1, r_{PB}^2, \ldots, r_{PB}^i, \ldots, r_{PB}^p\}.$$

Fig. 1 shows the hand-craft ordering for a small protein backbone formed by 3 amino acids. It is constructed so that, for each atom $v \in V$, the three edges $(v-3, v)$, $(v-2, v)$ and $(v-1, v)$ are always contained in E. The corresponding distances are obtained from known bond lengths and bond angles, that only depend from the kind of bound atoms. The two edges $(v-2, v)$ and $(v-1, v)$ are always associated to exact distances, whereas only the edge $(v-3, v)$ may be associated to an interval distance. In particular, there are three different possibilities. If $d(v-3, v) = 0$, then v represents a duplicated atom, and therefore

Algorithm 1 The *i*BP algorithm.

```
1: iBP(j, r, d, D)
2: if (r_j is a duplicated atom) then
3:    iBP(j + 1, r, d, D);
4: else
5:    if (d(r_j − 3, r_j) is exact) then
6:       b = 2;
7:    else
8:       b = 2D;
9:    end if
10:   for k ∈ {1, ..., b} do
11:      compute the k-th atomic position x^k_{r_j} for the r_j-th atom;
12:      check the feasibility of position x^k_{r_j} using pruning devices;
13:      if (x^k_{r_j} is feasible) then
14:         if (j = |r|) then
15:            a solution x is found, print it;
16:         else
17:            iBP(j + 1, r, d, D);
18:         end if
19:      end if
20:   end for
21: end if
```

the only feasible coordinates for v are the same of its previous copy. If $d(v-3, v)$ is an exact distance, the standard discretization process can be applied, and two possible positions for v can be computed. Finally, if $d(v-3, v)$ is represented by an interval, we discretize the interval and take D sample distances from it. For each sample distance, we apply the standard discretization process. In this case, $2 \times D$ possible atomic positions can be computed for v. As a consequence, the discrete search domain is a tree, which is not necessarily binary (this would require that all distances $d(v-3, v)$ are exact) [5].

Algorithm 1 is a sketch of the *interval* BP (*i*BP) [6]. It essentially requires 4 input arguments: the index j (in the ordering given to V) of the current atom to be placed, the artificial ordering r, the set of distances d (which can be either exact or represented by intervals), and the number D of sample distances used for discretizing interval distances. The main focus of this paper is on line 12 of Algorithm 1: the pruning devices that are used for discovering infeasible atomic positions.

3 Pruning devices

Pruning devices can be used in the *i*BP algorithm for pruning away infeasible branches of the discrete search domain. In this work, we study the influence of pruning devices on the performances of the algorithm. Each of such pruning devices is based on a different kind of information which can be obtained

through NMR experiments. The Direct Distance Feasibility (DDF) device (see Section 3.1) considers the available lower and upper bounds on the distances between hydrogen atoms. The Torsion Angle Feasibility (TAF) device (see Section 3.2) considers instead the lower and upper bounds on the protein backbone torsion angles. Finally, the Secondary Structure Feasibility (SSF) device (see Section 3.3) is based on the so-called *chemical shift index* of spin nuclei of the atoms C_α and H_α of each amino acid. Indeed, as shown in [11, 18], these indices are strongly related to the secondary structures to which each amino acid belongs. The technique described in [17], for example, is able to compute torsion angle restraints in secondary structures from chemical shift indices, with a precision of about 10°.

3.1 Direct Distance Feasibility (DDF)

NMR experiments are able to provide a list of lower and upper bounds on some distances between pairs of hydrogen atoms of the molecule. The Direct Distance Feasibility (DDF) pruning device is based on the idea of pruning atomic positions for which these lower and upper bounds are not satisfied. DDF has been widely used in our previous publications: even though it represents a very basic test, and it is easy to implement, DDF allows us to discard large parts of the discrete search domain very efficiently on sets of artificial instances [5, 7, 8, 12, 14]. However, when we tried to consider real NMR data, we noticed that the range defined by these lower and upper bounds is so large that DDF is not able anymore to sufficiently prune branches of the tree. This causes the multiplication of the solutions found by *i*BP, where some infeasible solutions are also contained. This is the reason why we needed to add new pruning devices in order to consider NMR instances.

3.2 Torsion Angle Feasibility (TAF)

Along with the list of lower and upper bounds on the distances, NMR experiments can also provide information on the torsion angles of protein backbones. Three different torsion angles can be defined along the backbone main chain $N - C_\alpha - C - N - \ldots$:

$$\begin{array}{rcl} \phi & \equiv & \{C, N, C_\alpha, C\}, \\ \psi & \equiv & \{N, C_\alpha, C, N\}, \\ \omega & \equiv & \{C_\alpha, C, N, C_\alpha\}. \end{array}$$

The angle ϕ, for example, is the angle defined by the two planes $\{C, N, C_\alpha\}$ and $\{N, C_\alpha, C\}$. The torsion angle ω is usually very close to π, because there is a peptide bond that does not allow this subset of atoms to take any other configuration. The other two angles ϕ and ψ, instead, can vary in larger ranges.

Even if the *i*BP algorithm is not based on the torsion angle representation of the protein backbone, but rather on an atomic representation, the torsion angles ϕ and ψ can be easily computed every time the four atoms needed for

their computation are available. As soon as the value for one of these angles is obtained, we can check if it satisfies the known lower and upper bounds provided by NMR: the last positioned atom can be pruned if the computed angle does not satisfy this constraint. We shall call this pruning device Torsion Angle Feasibility (TAF). Note that it is useless to consider the torsion angle ω because, by construction, our artificial backbone always satisfies the constraint $\omega = \pi$.

3.3 Secondary Structure Feasibility (SSF)

Subsets of atoms of a protein can fold in local structures which are very typical in proteins. Such local structures are referred to as *secondary structures*, and they are mainly represented by α-helices and β-sheets. In both cases, these secondary structures are stabilized through hydrogen bonds between pairs of amino acids. More precisely, given a pair (a_i, a_j) of amino acids belonging to the same secondary structure, there is a hydrogen bond between the hydrogen H (the one bound to N) of amino acid a_i and the oxygen O (bound to C) of amino acid a_j. This hydrogen bond forces the involved atoms, and in particular the hydrogen H of a_i and the oxygen O of a_j, to be very close to each other.

As a consequence, the torsion angles ϕ and ψ are constrained to vary in predefined ranges when the corresponding amino acids fold in α-helix or β-sheet. The bounds on the torsion angles can therefore be refined by using this information. Moreover, in the case of α-helices, it is known that the amino acid a_j is always a_{i+4}. Therefore, the two atoms which need to be closer than a certain threshold are known a priori: a new distance (between the hydrogen H of a_i and the oxygen O of a_{i+4}) can be added to the list of known distances. The possibility to add this new distance for each amino acid in α-helices reflects the strong regularity of this secondary structure; β-sheets are instead less regular: for each a_i, we do not know a priori the corresponding a_j.

In order to reject conformations which do not satisfy the restrictions given by the protein secondary structures, we use the *chemical shift index* described above to predict the subset of amino acids that are supposed to fold in α-helix or in β-sheet. As mentioned above, the technique described in [17] is able to find good estimates of the torsion angles related to amino acids having a given chemical shift index. However, since we do not need in general tight bounds on the torsion angles, we just consider intervals that are centered in $-60°$ for both ϕ and ψ (typical values for α-helices), or centered in $135°$ and $-120°$ (typical values for ϕ and ψ, respectively, in β-sheets) [1].

The Secondary Structure Feasibility (SSF) pruning device is therefore based on the idea of refining bounds for the torsion angles and/or adding new distances to the considered instance. This is done by exploiting information obtained by NMR on the chemical shift index of each amino acid. When the secondary structure is an α-helix, the oxygen O bound to the carbon C is needed for verifying the hydrogen bond distance. Note that this oxygen is not included in our artificial ordering (see Fig. 1). However, we can easily compute its coordinates when the positions for the atom C (which is bound to O), for the atom N (which is bound to C) and for the atom H (which is bound to N) are known. Because of

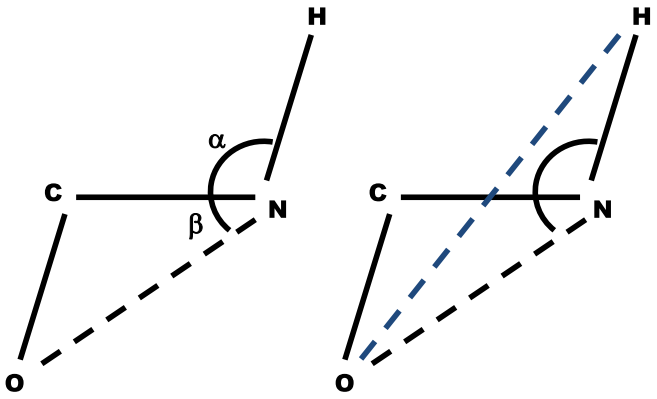

Fig. 2. Distances and angles that allow to compute the distance between the oxygen O bound to C and the hydrogen H bound to N. The angle β can be computed from known information regarding the triangle $\widehat{OCN}$. α is instead a bond angle. The distance between O and H can be computed by solving the triangle $\widehat{ONH}$, where the angle in N is $\alpha + \beta$. Note that the four atoms must lie on the same plane. The same procedure can also be applied for the other possible configuration, when the torsion angle is 0.

the presence of a peptide bond (the same which forces the torsion angle ω to be equal to π), the four atoms O, C, N and H lie on the same plane. Bond lengths are known, and, since bond angles are also known, the distance between O and N, as well as the distance between C and H, can be computed. Finally, Fig. 2 shows how to compute the distance between O and H. By exploiting all these distances and the coordinates for the atoms C, N and H, the coordinates for O can be uniquely computed.

The coordinates of the oxygen O can be computed by intersecting three spheres which are centered in the three atoms C, N and H, and having as radii the corresponding distances from O. This intersection of spheres can be computed by solving two linear systems: this same idea is also exploited in a generalization of the DMDGP presented in [13], and the interested reader can find many details about this procedure in the reference paper. We just remark that the procedure can provide in general two possible sets of coordinates for the oxygen O. Because of numerical errors, this may happen even if only one position for O is actually feasible. In our implementation, the pruning device SSF is applied only if the computed coordinates of O are not affected by numerical errors.

instance name	n_{aa}	n	D	iBP calls	#DDF	#TAF	#SSF	CPU time
2jmy	15	134	15	4724652	2356670	-	-	39
2jmy	15	134	15	10482	5244	2695	-	1
2jmy	15	134	15	31986247	15206046	-	6189223	248
2jmy	15	134	15	33709275	16017742	1069321	5156934	298
2ppz	36	323	20	98807	48586	-	-	1
2ppz	36	323	20	91466	43568	41600	-	2
2ppz	36	323	20	414926692	142727215	-	70158539	10263
2ppz	36	323	20	58296108	18941155	10111249	615926	1471
2jwu	56	503	22	6528633	6715391	-	-	117
2jwu	56	503	22	11159985	28183553	1029437	-	396
2jwu	56	503	22	20119294	14742376	-	1601915	432
2jwu	56	503	22	44795676	19494850	9450743	5313997	1363

Table 1. Experiments on NMR instances. #DDF, #TAF and #SSF provide the number of times each pruning device found and discarded an infeasible atomic position. The symbol "–" indicates that the corresponding pruning device was not used in the experiment.

4 Computational experiments

The iBP algorithm has been implemented in C programming language and compiled by the GNU C compiler v.4.1.2 with the `-O3` flag. We performed the experiments presented in this section on an Intel Core 2 CPU 6400 @ 2.13 GHz with 4GB RAM, running Linux.

For the first time since we started to work on this topic, we are able to present computational experiments where real data from NMR are managed by using an algorithm based on a discrete search. The data related to protein conformations having different features (number of amino acids, secondary structures) have been downloaded from the Protein Data Bank (PDB) [2], along with the corresponding conformations already obtained by other methods. For each instance, we study the influence of the implemented pruning devices on the iBP algorithm. We point out that, in general, NMR instances may not contain the necessary information for applying all three pruning devices. Distances are always available, and therefore the pruning device DDF can always be considered. Bounds on torsion angles and chemical shift indices might be omitted. In the following, we consider instances where all necessary information is supposed to be available.

Table 1 shows the results of our experiments for a selected subset of instances. The instance name is the PDB code of the molecule. `2jmy` is a small peptide completely folded in α-helix. `2ppz` is a protein containing only α-helices as secondary structure, whereas `2jwu` contains both secondary structures. All considered instances contain information on distances and torsion angles. Only distances regarding the hydrogens H and H_α of each amino acid have been considered: all others (mainly related to the amino acid side chains) have been discarded. The information regarding the secondary structures have been ob-

tained from the conformations downloaded from the PDB (we plan to include a procedure to automatically interpreting the chemical shift index associated to each amino acid in future versions of *i*BP, see for example [17]). In the table, n_{aa} is the number of amino acids forming the protein, whereas n is the length of the artificial ordering r_{PB} (hence, it consists of the number of atoms in the protein backbone, including duplicated atoms). The number D of sample distances which are considered for discretizing intervals is also specified for each experiment. The behavior of the *i*BP algorithm is evaluated through the number of times the algorithm recursively calls itself before finding the first solution, and through the number of times each pruning device is able to identify and prune an infeasible atomic position. If this information is absent (the symbol "–" is used in the table), it means that the pruning device was not applied in the given experiment. For each protein, we performed 4 experiments, where the following combinations of pruning devices were considered: DDF, DDF+TAF, DDF+SSF, DDF+TAF+SSF. The *i*BP algorithm is stopped as soon as the first solution is found. For each experiment, we provide the CPU time in seconds.

We consider the number of *i*BP calls as a valid measure of the influence of the newly added pruning devices. When the number of *i*BP calls decreases, the added pruning devices were able to discard infeasible atomic positions that DDF was not able to recognize, and lead the search towards feasible positions sooner. In this case, the CPU time decreases. Moreover, when the number of *i*BP calls instead increases, atomic positions previously considered as feasible are declared infeasible by the new pruning devices, and therefore the search is focused on different parts of the search domain. In this case, the CPU time may increase, but there is a gain on the quality of the obtained solution.

Both situations can be seen in Table 1. For the helix `2jmy`, the number of *i*BP calls decreases of two orders of magnitude when the pruning device TAF is added to the standard DDF. Indeed, #DDF decreases, because TAF was able to recognize infeasible atomic positions earlier during the search and was able to prune larger parts of the search domain. This also happens when TAF is added to DDF alone or DDF+SSF in the experiments related to the protein `2ppz`. Otherwise, the second situation is more common in these experiments: when a new pruning device is added, the number of atomic positions pruned by these devices while working in cooperation increases. Therefore, they are able to lead the search towards better solutions, i.e. towards solutions where the constraints related to all pruning devices are satisfied.

We remark that only one solution to the problem is computed in these experiments. At this stage of our work, we cannot analyze yet the influence of the pruning devices on the whole set of solutions, because the considered pruning devices, even if they are used all together, are not able to keep under control the combinatorial explosion due to the recursive calls to *i*BP. For the same reason, we cannot judge yet on bio-related aspects of the found solutions in comparison to the employed pruning devices. We plan to do so in the future by including the amino acid side chains in our artificial backbone.

5 Conclusions

The *i*BP algorithm for the MDGP is the first algorithm implementing a discrete search which is able to manage interval data. It can be currently applied to MDGPs related to protein backbones, for which we identified a particular artificial ordering for their atoms that allows us to discretize the problem. In this work, we studied the influence of pruning devices on a set of NMR instances, i.e. instances where real data from NMR are contained. The pruning devices are based on different information that can be obtained through NMR experiments: a list of bounds on the distances between pairs of atoms of the molecule, a list of bounds on the torsion angles of the protein backbones, and finally information regarding the protein secondary structures. The presented experiments showed that the newly added pruning devices are actually able to prune away large parts of the discrete search domain, so that the search can be focused on feasible parts of the domain. Next step is to consider information regarding the amino acid side chains. This could allow us to identify only a few feasible solutions for each considered instance.

Acknowledgments

The authors wish to thank the Brazilian research agencies FAPESP and CNPq, the French research agency CNRS, Institut Pasteur, and École Polytechnique, for financial support.

References

1. J.M. Berg, J.L. Tymoczko, L. Stryer. *Biochemistry*. W.H. Freeman publications (6^{th} edition), 2006.
2. H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, and P.E. Bourne. The protein data bank. *Nucleic Acid Research*, 28:235–242, 2000.
3. C. Lavor, L. Liberti, and N. Maculan. The discretizable molecular distance geometry problem. Technical Report q-bio/0608012, arXiv, 2006.
4. C. Lavor, L. Liberti, and N. Maculan. Molecular distance geometry problem. In C. Floudas and P. Pardalos, editors, *Encyclopedia of Optimization*, pages 2305–2311. Springer, New York, 2^{nd} edition, 2009.
5. C. Lavor, L. Liberti, and A. Mucherino. On the solution of molecular distance geometry problems with interval data. IEEE conference proceedings, International Workshop on Computational Proteomics, International Conference on Bioinformatics & Biomedicine (BIBM10), Hong Kong, 77–82, 2010.
6. C. Lavor, L. Liberti, and A. Mucherino. The *i*BP algorithm for the discretizable molecular distance geometry problem with interval data. submitted. Available on Optimization Online.
7. C. Lavor, A. Mucherino, L. Liberti, and N. Maculan. On the computation of protein backbones by using artificial backbones of hydrogens. to appear in *Journal of Global Optimization*, 2011. Available online from July 24, 2010.

8. C. Lavor, A. Mucherino, L. Liberti, and N. Maculan. Discrete approaches for solving molecular distance geometry problems using nmr data. *International Journal of Computational Biosciences* 1(1):88–94, 2010.
9. L. Liberti, C. Lavor, and N. Maculan. A branch-and-prune algorithm for the molecular distance geometry problem. *International Transactions in Operational Research*, 15:1–17, 2008.
10. L. Liberti, C. Lavor, A. Mucherino, and N. Maculan. Molecular distance geometry methods: from continuous to discrete. *International Transactions in Operational Research* 18(1):33-51, 2010.
11. S.P. Mielke, V.V. Krishnan. An evaluation of chemical shift index-based secondary structure determination in proteins: influence of random coil chemical shifts. *Journal of Biomolecular NMR* 30(2):143–196, 2004.
12. A. Mucherino and C. Lavor. The branch and prune algorithm for the molecular distance geometry problem with inexact distances. In *Proceedings of the International Conference on Computational Biology*, volume 58, pages 349–353. World Academy of Science, Engineering and Technology, 2009.
13. A. Mucherino, C. Lavor, L. Liberti. The Discretizable Distance Geometry Problem. submitted.
14. A. Mucherino, L. Liberti, C. Lavor, and N. Maculan. Comparisons between an exact and a metaheuristic algorithm for the molecular distance geometry problem. In F. Rothlauf, editor, *Proceedings of the Genetic and Evolutionary Computation Conference*, pages 333–340, Montreal, 2009. ACM.
15. M. Nilges, A.M. Gronenborn, A.T. Brunger, and G.M. Clore. Determination of three-dimensional structures of proteins by simulated annealing with interproton distance restraints. application to crambin, potato carboxypeptidase inhibitor and barley serine proteinase inhibitor 2. *Protein Engineering* 2:27–38, 1988.
16. J.B. Saxe. Embeddability of weighted graphs in k-space is strongly NP-hard. *Proceedings of 17th Allerton Conference in Communications, Control and Computing*, 480–489, 1979.
17. Y. Shen, F. Delaglio, G. Cornilescu, A. Bax. TALOS+: a hybrid method for predicting protein backbone torsion angles from NMR chemical shifts. *Journal of Biomolecular NMR* 44(4):213–236, 2009.
18. D.S. Wishart, B.D. Sykes, F.M. Richards. The chemical shift index: a fast and simple method for the assignment of protein secondary structure through NMR spectroscopy. *Biochemistry* 31(6), 1647–1698, 1992.