DISCRETE APPROACHES FOR SOLVING MOLECULAR DISTANCE GEOMETRY PROBLEMS USING NMR DTA

Carlile Lavor,[*] Antonio Mucherino,[**] Leo Liberti,[***] and Nelson Maculan[****]

Abstract

The molecular distance geometry problem (MDGP) is the problem of finding the conformation of a molecule by exploiting known distances between some pairs of its atoms. Estimates of the distances between the atoms can be obtained through experiments of nuclear magnetic resonance (NMR) spectroscopy. The information on the distances, however, is usually limited, because only distances between hydrogens and shorter than 6 Å are usually available, and this makes the solution of the MDGP quite hard. In this paper, we focus our attention on protein backbones and we present a methodology for computing their full-atom conformations starting from NMR data. This task is performed by solving two MDGPs. First of all, only hydrogens are considered: we define an *artificial backbone of hydrogens* for which particular assumptions needed for the discretization of the problem are satisfied. This allows for solving the first MDGP with an *ad hoc* algorithm. Secondly, by exploiting the coordinates of the hydrogens and known bond lengths and bond angles, we compute the coordinates of the other atoms forming the protein backbone by using a polynomial-time algorithm. Computational experiments related to real proteins are presented.

Key Words

Distance geometry, discrete formulation, NMR data, hydrogens, Branch & Prune

1. Introduction

Proteins can be analysed by experiments of nuclear magnetic resonance (NMR) spectroscopy that is able to provide information from which the relative distances between some pairs of atoms forming the molecule can be estimated [1–3]. Such estimated distances can then be used for attempting a reconstruction of the three-dimensional conformation of the protein, which is of fundamental importance for understanding its dynamics, and, as a consequence, its function. The major difficulty to be faced during this task is that the information obtained through NMR is usually limited, because not all the possible relative distances are known, but only a small subset. Moreover, only distances between pairs of hydrogens are usually available, whereas proteins are also composed of carbon, nitrogen, oxygen and so on.

The distances estimated through NMR can be used for generating a set of constraints that the atoms forming the protein must satisfy in its three-dimensional conformation. This problem is known in the scientific literature as molecular distance geometry problem (MDGP), and different approaches for its solution have been proposed over the years (for a review, see [4, 5]). The most natural approach is to solve the problem by techniques for *constraint satisfaction*: a conformation for the atoms of the protein must be found so that all the available constraints are satisfied. However, the problem is usually reformulated as a *global continuous optimization problem*. In this approach, the set of constraints based on the distances is transformed into a penalty function which is able to give a measure of how much the available constraints are violated. When the penalty function is 0 for a given conformation, then this conformation satisfies all the constraints. Therefore, finding the global minimum of such a penalty function equals to find a conformation for a protein which is solution to the MDGP. Different penalty functions for this purpose have been proposed, and, unfortunately, they all have many local minima, which makes the search for their global optimum very hard [6].

Our aim is to use a recently proposed approach for solving MDGPs. In this approach, the domain of the penalty function is *discretized*, and the optimization problem to be solved is *combinatorial*. After the reformulation, the problem can be solved more easily. We refer to the reformulated problem as discretizable MDGP (DMDGP) [7]. A very important advantage in using this discrete formulation to the problem is that an *ad hoc* algorithm, called Branch & Prune (BP) [8], can be used for efficiently

[*] Departnment of Applied Mathematics (IMECC-UNICAMP), State University of Campinas, C.P. 6065, 13081–970, Campinas-SP, Brazil; e-mail: clavor@ime.unicamp.br

[**] INRIA, Lille Nord Europe, 59650 Villeneuve d'Ascq, France; e-mail: antonio.mucherino@inria.fr

[***] LIX, École Polytechnique, 91128 Palaiseau, France; e-mail: liberti@lix.polytechnique.fr

[****] Systems Engineering (COPPEUFRJ), Federal University of Rio de Janeiro, C.P. 68511, 21945-970, Rio de Janeiro-RJ, Brazil; e-mail: maculan@cos.ufrj.br

Recommended by Dr. L. Elnitski
(paper no. 210-1025)

solving it. The algorithm is based on the exploration of a binary tree of solutions, whose branches are pruned as soon as infeasibilities are found. Other algorithms having some relations with BP are given in [9–11].

The idea behind the discretization is the following. Let a be an atom with an unknown position. Let us suppose that the distances between a and other three atoms b_i, $i \in \{1, 2, 3\}$, are known, and that the position of each b_i is also known. Then, it can be proved that there are only two possible positions for the atom a. This discretizes the problem, because a binary tree of atomic positions can be defined, where the solutions to DMDGPs can be searched. Moreover, if there is at least a fourth atom b_4 with known position, independent from the others, and such that the distance between a and b_4 is known, then we are able to select only one feasible position for a between the two positions previously found. This feature allows to prune the binary tree very efficiently during the execution of BP. In this way, an exhaustive search on the remaining branches of the tree is not computationally expensive.

Naturally, the discretization is possible only when particular assumptions are satisfied. In the formal definition of the DMDGP, we require that, for each atom x_i, the distances between x_i and the three preceding atoms x_{i-3}, x_{i-2} and x_{i-1} must be known. Consequently, if the atoms are placed following their order, the atoms x_{i-3}, x_{i-2} and x_{i-1} already have a known position when a position for x_i is searched. As all the distances between the atoms of the quadruplet $\{x_{i-3}, x_{i-2}, x_{i-1}, x_i\}$ are known by hypothesis, the cosine of the torsion angle among these four atoms can be computed: there are only two corresponding torsion angles, and each of them brings to the definition of one position for x_i. The unrealistic case of aligned consecutive atoms (for which the earlier discussion on the torsion angles fails) is avoided by hypothesis. The interested reader can find the formal mathematical definition of the DMDGP in [7].

Unfortunately, the necessary assumptions for the discretization are not always satisfied. As an example, if we consider the hydrogens of a protein and we sort them amino acid per amino acid, following the increasing alphabetic order of their labels (H, H_α, H_β and so on), then it is quite impossible to have the assumptions satisfied. The same observation can be made if only the hydrogens of the protein backbone are considered. Indeed, in both the cases, not all the distances in the quadruplets $\{x_{i-3}, x_{i-2}, x_{i-1}, x_i\}$ are found by NMR because they are usually larger than 6 Å, i.e., threshold under which distances can be obtained through NMR. However, given a certain hydrogen H, because proteins are very compact objects, we know that there are other hydrogens surrounding H and close enough for having their distance detected by NMR. As a consequence, there might exist a particular ordering (and may be more than one ordering) for the hydrogens for which the necessary assumptions for the DMDGP are satisfied.

In this paper, we focus our attention on protein backbones. From the NMR, we expect to obtain distances between pairs of hydrogens of the protein backbone that are shorter than 6 Å. We will show how it is possible to define an *artificial backbone of hydrogens* where the hydrogens are sorted so that the necessary assumptions are satisfied. Then, we will show how to build the entire protein backbone (including the carbons and the nitrogens it contains) by exploiting the coordinates of the hydrogens just obtained and the information known *a priori* on bond lengths and bond angles. Therefore, we will show how the problem of finding the backbone of a protein can be divided in two subproblems, where only hydrogens are considered in the first one, and the other atoms of the backbone are considered in the second subproblem. These two subproblems are both MDGPs, having different properties.

Figure 1. An example of binary tree associated to an instance of the DMDGP with 6 atoms. The path marked by boxes represents a solution.

The remaining of the paper is organized as follows. In Section 2, we provide an overview of the BP algorithm which we use for solving instances of the DMDGP. In Section 3, we present the artificial backbone of hydrogens, where the particular ordering given to the hydrogen atoms makes the assumptions for the DMDGP satisfied, and we show a technique for computing the other atoms (N, C_α and C) of the protein backbone starting from the coordinates of the hydrogens. Computational experiments on a set of instances related to real proteins are shown in Section 4. Finally, conclusions are given in Section 5.

2. The Branch & Prune algorithm

When the assumptions of the DMDGP are satisfied, a binary tree of atomic positions can be defined and explored with the aim of finding solutions to the problem (see Fig. 1). As mentioned earlier, we consider the BP algorithm [8] for solving DMDGPs, which is strongly based on this tree structure. The binary tree of possible solutions is explored starting from its top by placing one atom per time.

Algorithm 1 provides a sketch of the BP algorithm. The first atom x_1 can be arbitrarily placed in the position $(0, 0, 0)$. The second one can be placed on one of the coordinate axes of the Cartesian system, so that the distance between x_1 and x_2 is the one known by hypothesis. Finally, the third atom x_3 can be placed on a plane formed by two coordinate axes, so that all the known distances are satisfied. Then, starting from the atomic position 4, Algorithm 1 is invoked iteratively. Its input parameters are i, the current atom whose position is searched; n, the total number of atoms; d, the set of known distances. One of the solutions to the problem is found when BP(n, n, d), the call to Algorithm 1 in which a position for x_n is searched, finds a feasible position for the last atom of the molecular conformation.

Algorithm 1 The BP algorithm.

```
0: BP($i$, $n$, $d$)
  for ($k=1$, $2$) do
    compute the $k$th atomic position for the $i$th atom: $x_i^{(k)}$;
    check the feasibility of the atomic position $x_i^{(k)}$:
    if (the atomic position $x_i^{(k)}$ is feasible) then
      if ($i = n$) then
        one solution is found;
      else
        BP($i+1$, $n$, $d$);
      end if
    else
      the current branch is pruned;
      end if
  end for
```

At each step of the algorithm, two possible positions for the current atom x_i are computed, and the search is branched. By recursing the search on each branch, the size of the binary tree increases exponentially, because two new branches are added to the tree at each step. For this reason, pruning tests are used to discover infeasible atomic positions. As soon as an atomic position is found to be infeasible, then the corresponding branch is pruned and the search is backtracked. The pruning phase usually reduces the tree within manageable sizes, so that an exhaustive search on the remaining branches is not computationally expensive.

There are different ways for checking the feasibility of computed atomic positions [12]. The most natural pruning test is the following. If x'_i is one possible position for x_i, then we can compare all the known distances between x_i and x_j, with $j < i$, to the corresponding distances that can now be computed between x'_i and each x_j. If known and computed distances match, then x'_i is feasible, otherwise it is infeasible. In this pruning test, it is very important to set up the tolerance ε accurately. Indeed, as it is impossible to test for real number equality using floating point arithmetic, the choice of epsilon plays an important role. Too small tolerances could force the pruning of all the atomic positions (and no solutions are found), whereas too large tolerances could allow too much positions to be accepted, with a consequence enlargement of the binary tree.

3. Computing protein backbones

Distances estimated by NMR are mostly between pairs of hydrogens. To reformulate the problem of finding the coordinates of these hydrogens as a DMDGP, we introduce an artificial backbone mainly formed by hydrogen atoms (see Fig. 2). The two hydrogens per amino acid of the protein backbones are both considered (the one bound to C_α and the one bound to N), and another hydrogen is borrowed from the side chain of the amino acid. Glycines only have one hydrogen in their side chains; all the other amino acids have at least one hydrogen bound to the carbon C_β, and any of them can be considered in the artificial backbone. Therefore, three hydrogens per amino acid are considered: H (bound to N), H_α (bound to C_α) and H_β (from the side chain).

Let $(V, <)$ be a subset of atoms of a protein containing the hydrogens H, H_α and H_β, where the symbol $<$ represents an ordering relation associated to the elements of V. In other words, for each element of V, we know which is its preceding atom, and which is its successive atom, being both defined by the ordering $<$. The proposed artificial backbone can be built as follows. Let us organize the atoms of V so that the following ordering is satisfied:

$$\mathrm{H}_\alpha^1, \mathrm{H}^1, \mathrm{H}^2, \mathrm{H}_\alpha^1, \dots, \mathrm{H}_\alpha^i, \mathrm{H}^i, \mathrm{H}_\beta^i, \mathrm{H}^{i+1}, \mathrm{H}_\alpha^i, \dots, \mathrm{H}_\alpha^n, \mathrm{H}^n, \mathrm{H}_\beta^n$$

where the superscripts indicate the amino acid to which the considered hydrogen belongs. Note that some of the hydrogens are considered twice: there are two elements in this sequence that refer to the same hydrogen. Then, let us

Figure 2. An artificial backbone of hydrogens related to the protein backbones. Note that some of the hydrogens are considered twice and that the considered ordering is specified by the labels associated to the edges.

add the three atoms N^1, C_α^1 and C^1 (in this order) at the beginning of the sequence of hydrogens, and let us add the hydrogen $H^{n'}$ (bound to the oxygen of the last amino acid) to its end. The obtained ordered subset $(V', <')$ represents the artificial backbone for which the assumptions for the DMDGP are satisfied. Naturally, $V \subset V'$, because we considered all the hydrogens in V, we duplicated some of them, and we added four more atoms (three at the beginning and one at the end).

The reason why the artificial backbone mainly contains hydrogen atoms is that relative distances between hydrogens can be found by NMR. Moreover, in [13], it has been formally proved that a necessary condition for having the needed assumptions for the DMDGP satisfied is that all the atoms of the artificial backbone with a rank larger than 3 must be hydrogen atoms. We added exactly three atoms (that are not hydrogens) at the beginning of the artificial backbone, because they define a Cartesian coordinate system, that we will use later during the computation of the other backbone atoms. These three added atoms correspond to the first three atoms of the protein backbone, belonging to the first amino acid of the protein. The needed distances between these three atoms and the following hydrogens are all known *a priori*, and therefore they do not need to be found by NMR.

A particular feature of the artificial backbone is that the same hydrogen can be considered twice in the sequence. Though this might appear nonsensical, it is very useful for reducing the relative distances in the quadruplets of consecutive atoms, and for making them shorter than the 6 Å threshold. Simple computations based on the known bond lengths and angles showed that the distances needed for having the discretization of the problem (the ones between the atoms of the quadruplets) are always shorter than 6 Å. Only in a few cases, some of such distances could be, in theory, larger than 6 Å, but this never occurred during our computational experiments. Therefore, the assumptions for the DMDGP are satisfied, and the problem of finding the coordinates of these atoms can be solved by applying the BP algorithm. This algorithm is able to provide very accurate solutions, containing the coordinates of all the hydrogens. Naturally, the coordinates of some hydrogens are provided twice, but, after the execution of the algorithm, the duplicated coordinates can just be discarded. Other details regarding the artificial backbone of hydrogens can be found in [13–15].

The coordinates of the atoms N, C_α and C of the protein backbones can be found by solving another MDGP. In this case, we do not exploit any information from the NMR. As some distances between hydrogens and the other backbone atoms are known *a priori* (they can be computed by using information on bond lengths and angles), supposing that the coordinates of the hydrogens H, H_α and H_β are already known, we can observe that this particular MDGP satisfies assumptions that are stronger than the ones of the DMDGP. In practise, not only we have the possibility to discretize the problem, but we can always associate only one position to each atom a. As explained in the Introduction, this is possible when the distances between an atom a and an atom b_i, with $i \in \{1, 2, 3, 4\}$, are known, and the coordinates of each b_i is also known.

To solve this problem, we follow the method presented in [16, 17]. Let d_{a,b_i} be the Euclidean distance between the atom a and the atom b_i, for all $i \in \{1, 2, 3, 4\}$. To find the coordinates of a, the following system needs to be solved:

$$\begin{cases} ||a - b_1|| = d_{a,b_1} \\ ||a - b_2|| = d_{a,b_2} \\ ||a - b_3|| = d_{a,b_3} \\ ||a - b_4|| = d_{a,b_4} \end{cases} \tag{1}$$

This is a quadratic system of four equations in three variables. However, as shown in [16, 17], if the system of linear equations:

$$Ax = b \tag{2}$$

where,

$$A = -2 \begin{bmatrix} (b_1 - b_2)^T \\ (b_1 - b_3)^T \\ (b_1 - b_4)^T \end{bmatrix}$$

$$x = a$$

$$b = \begin{bmatrix} (d_{a,b_1}^2 - d_{a,b_2}^2) - (||b_1||^2 - ||b_2||^2) \\ (d_{a,b_1}^2 - d_{a,b_3}^2) - (||b_1||^2 - ||b_3||^2) \\ (d_{a,b_1}^2 - d_{a,b_4}^2) - (||b_1||^2 - ||b_4||^2) \end{bmatrix}$$

is solved, its solution is also solution for the quadratic system (1). Thus, the MDGP related to the protein backbone can be solved by solving a sequence of 3 × 3 linear systems, each one in correspondence with an atom of the protein backbone to be placed.

For each atom N, C_α and C of the protein backbone, there are always four atoms b_i that can be considered in the linear system. For computing the position of the nitrogen N of the protein backbone, e.g., the following four atoms with known positions can be considered: C_α and C of the previous amino acid, the hydrogen H bound to N and the hydrogen H_α bound to the following C_α (see Fig. 3).

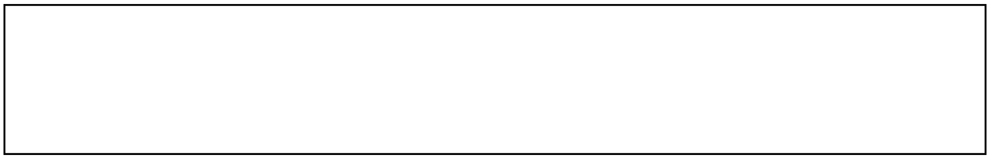

Figure 3. The atoms and the distances used in the three linear systems are used to determine the protein backbone.

The distances between C and N and between N and H are known because these two pairs of atoms are chemically bound. The distance between C_α and N is also known, because the bond lengths $C_\alpha - C$ and $C - N$ are known, and the angle among the three atoms $C_\alpha - C - N$ is also known. For the same reason, the distance between N and H_α is available. The solution of the linear system (2) allows to identify the coordinates of N. Similar observations can be made for the other two systems, related to the backbone atoms C_α and C. The reader is referred to Fig. 3 to find out which atoms and distances can be considered in each linear system.

4. Computational Experiences

We will show in this section how instances of the DMDGP related to artificial backbones of hydrogens can be efficiently solved by the BP algorithm, and how the solutions found by BP can be exploited for computing the other backbone atoms of a protein by solving a sequence of 3×3 linear systems. All the codes were written in C programming language and all the experiments were carried out on an Intel Core 2 CPU 6400 @ 2.13 GHz with 4 GB RAM, running Linux. The codes have been compiled by the GNU C compiler v.4.1.2 with the `-O3` flag.

Before presenting the computational experiments, we point out that, in the whole above discussion, we always supposed that the values of the known distances are exact. We are aware this is not true in general when dealing with data obtained by NMR. However, to show the correctness of our approach, we provide in this section some computational experiments where all the distances are considered as exact. Work is in progress for generalizing our approach to the case in which the considered distances are affected by errors and noise. Preliminary studies in this direction have been presented in [18, 19].

We generated a set of instances from the known conformations of some proteins, downloaded from the Protein Data Bank (PDB) [20], and we extracted from such conformations only the information regarding our artificial backbones. The atoms of the artificial backbones are all sorted in accordance with the special ordering discussed in Section 3, and only distances smaller than 6 Å are considered. For each amino acid, three hydrogen atoms are considered, and five in total are included in the instance, because two of them are considered twice. Apart from the first three atoms, the artificial backbone is composed of the hydrogens H, H_α and H_β.

All the instances we generated belong to the class of instances of the DMDGP, because the necessary assumptions are satisfied. We applied the BP algorithm for solving such instances, and the computational experiments are shown in Table 1. In the table, n is the number of atoms included in the instance. It is always a number which is divisible by 5, because each amino acid of the considered artificial backbone contains exactly five hydrogens (two of them are considered twice). m is the number of known distances (all these distances are shorter than 6 Å). We evaluated the quality of the obtained solutions by employing a commonly used penalty function, the largest distance error (LDE):

$$\mathrm{LDE}(\{x_1, x_2, \ldots, x_n\}) = \frac{1}{m} \sum_{\{i,j\}} \frac{|\, ||x_i - x_j|| - d_{ij} \,|}{d_{ij}}$$

Table 1
Results Obtained by the BP Algorithm on a Set of Artificial Backbones Obtained from Protein Conformations Downloaded from the PDB

Instance Name	n	m	LDE	CPU Time
1brv	95	994	1.12e−08	0.00
1a11	125	1,125	2.72e−09	0.00
1acw	145	1,462	2.03e−08	0.00
1bbl	185	1,562	3.12e−09	0.00
1erp	190	1,831	4.10e−09	0.00
1aqr	200	1,697	1.03e−08	0.00
1k1v	205	1,702	2.31e−09	0.00
1h1j	220	1,880	4.09e−09	0.00
1dv0	225	2,116	4.77e−09	0.00
1jkz	230	2,030	1.20e−08	0.00
1ahl	245	2,181	4.54e−08	0.17
1ccq	300	2,625	3.80e−08	0.01
1bqx	385	3,670	1.47e−08	0.01
1a2s	445	4,988	3.49e−08	0.00
1ag4	515	5,262	1.02e−07	0.00
1acz	540	4,776	2.77e−08	0.02
1itm	650	6,552	3.24e−08	1.99
1b4c	920	8,833	2.37e−08	0.01
1la3	940	8,335	1.21e−08	2.87
1a23	945	10,416	5.58e−08	0.24
1oy2	955	8,372	5.16e−08	46.12
1d8v	1,315	14,184	6.48e−08	1.14
1ezo	1,850	17,996	7.45e−08	86.60

In the definition of LDE, m is the total number of available distances, x_i is the generic atom of the conformation, d_{ij} is the known distance between x_i and x_j and $||x_i - x_j||$ is the computed distance between x_i and x_j. As there are distances equal to zero, we considered in the experiments a modified version of this function in which the terms containing divisions by zero are discarded (see [15] for more details). Finally, the CPU time (in seconds) is given for each experiment.

The experiments show that the BP algorithm is very efficient in finding solutions to the DMDGP in terms of quality of the solutions and CPU time, as already shown in previous works. In these experiments, each solution consists of the set of coordinates of the hydrogen atoms H, H_α and H_β of the artificial backbones.

These coordinates are then exploited in the linear systems for finding the coordinates of the other backbone atoms. In all the experiments, we obtained a correct protein backbone for each considered artificial backbone. Each experiment took less than 1 s of CPU time for computing the sequence of linear systems (in our implementation, the LAPACK library [21] is used for this purpose).

5. Conclusions

The discretization of MDGPs is possible only when particular assumptions are satisfied. We showed in this paper that these assumptions can be satisfied when considering data from NMR spectroscopy. To this aim, we reordered the hydrogens of the protein backbones in a special way, so that all the needed distances are supposed to be found by NMR. Moreover, we also showed that, after the identification of the coordinates of the hydrogens, the other backbone atoms can also be found by a discrete approach and computed by solving a sequence of linear systems.

In this paper, we focused our attention on protein backbones, but the same ideas can also be applied to whole protein conformations. What is needed to do is to extend the artificial backbone of hydrogens to all the hydrogens contained into the protein, and to define new linear systems to be used for computing the coordinates of atoms (which are not hydrogens) of the side chains.

Another important point is the management of the noise which can affect data obtained by NMR. In practise, exact distances are usually not available, but only their approximations, and this must be taken into account when working on the artificial backbones. The computation of the other backbone atoms by the sequence of linear systems, however, remains unchanged, because it is based on known values of bond lengths and bond angles.

These studies on the discretization of the MDGPs arising in biology are bringing to several interesting subproblems. Suitable strategies for their solutions are of increasing interest, because the discrete approach could allow the identification of protein conformations (or conformations of molecules in general) having a high precision. Future works will be addressed to this final aim.

Acknowledgements

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

References

[1] G.M. Crippen & T.F. Havel, *Distance geometry and molecular conformation* (New York: John Wiley & Sons, 1988).

[2] T.F. Havel, Distance geometry, in D.M. Grant & R.K. Harris (Eds.), *Encyclopedia of nuclear magnetic resonance* (New York: Wiley, 1995), 1701–1710.

[3] T. Schlick, *Molecular modelling and simulation: An interdisciplinary guide* (New York: Springer, 2002).

[4] C. Lavor, L. Liberti, & N. Maculan, Molecular distance geometry problem, in C. Floudas & P. Pardalos (Eds.), *Encyclopedia of optimization*, Second Edition (New York: Springer, 2009), 2305–2311.

[5] L. Liberti, C. Lavor, A. Mucherino, & N. Maculan, Molecular distance geometry methods: From continuous to discrete, *International Transactions in Operational Research*, 2010 (to appear).

[6] J.B. Saxe, Embeddability of weighted graphs in *k*-space is strongly NP-hard, *Proc. 17th Allerton Conference in Communications, Control, and Computing*, Monticello, IL, 1979, 480–489.

[7] C. Lavor, L. Liberti, & N. Maculan, Discretizable molecular distance geometry problem, Tech. Rep. q-bio.BM/0608012, arXiv, 2006.

[8] L. Liberti, C. Lavor, & N. Maculan, A Branch-and-Prune algorithm for the molecular distance geometry problem, *International Transactions in Operational Research*, *15*, 2008, 1–17.

[9] R.S. Carvalho, C. Lavor, & F. Protti, Extending the geometric buildup algorithm for the molecular distance geometry problem, *Information Processing Letters*, *108*, 2008, 234–237.

[10] D. Wu, Y. Yuan, & Z. Wu, The Solution of the Distance geometry problem for protein modeling via geometric buildup, *Biophysical Reviews and Letters*, *3*, 2008, 43–75.

[11] D. Wu, Z. Wu, & Y. Yuan, Rigid versus Unique Determination of protein structures with geometric buildup, *Optimization Letters*, *2*, 2008, 319–331.

[12] C. Lavor, L. Liberti, A. Mucherino, & N. Maculan, On a discretizable subclass of instances of the molecular distance geometry problem, *ACM Conference Proceedings, 24th Annual ACM Symposium on Applied Computing* (SAC09), Hawaii, USA, 2009, 804–805.

[13] A. Mucherino, C. Lavor, L. Liberti, & N. Maculan, On the definition of artificial backbones for the discretizable molecular distance geometry problem, *Mathematica Balkanica*, *23*, 2009, 289–302.

[14] C. Lavor, A. Mucherino, L. Liberti, & N. Maculan, An artificial backbone of hydrogens for finding the conformation of protein molecules, *IEEE Conference Proceedings*, *Computational Structural Bioinformatics Workshop* (CSBW09), Washington, DC, USA, 2009, 152–155.

[15] C. Lavor, A. Mucherino, L. Liberti, & N. Maculan, Computing artificial backbones of hydrogen atoms in order to discover protein backbones, *IEEE Conference Proceedings, International Conference IMCSIT09, Workshop on Combinatorial Optimization (WCO09)*, Poland, 2009, 751-756.

[16] Q. Dong & Z. Wu, A linear-time algorithm for solving the molecular distance geometry problem with exact inter-atomic distances, *Journal of Global Optimization*, *22*, 2002, 365–375.

[17] D. Wu & Z. Wu, An updated geometric build-up algorithm for solving the molecular distance geometry problem with sparse distance data, *Journal of Global Optimization*, *37*, 2007, 661–673.

[18] A. Mucherino & C. Lavor, The branch and prune algorithm for the molecular distance geometry problem with inexact distances, World Academy of Science, Engineering and Technology (WASET), *Proceedings of the "International Conference on Bioinformatics and Biomedicine"* (ICBB09), Italy, 2009, 349–353.

[19] A. Mucherino, L. Liberti, C. Lavor, & N. Maculan, Comparisons between an exact and a metaheuristic algorithm for the molecular distance geometry problem, *ACM Conference Proceedings*, *Genetic and Evolutionary Computation Conference* (GECCO09), Montréal, Canada, 2009, 333–340.

[20] H.M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T.N. Bhat, H. Weissig, I.N. Shindyalov, & P.E. Bourne, The Protein Data Bank, *Nucleic Acids Research*, *28*, 2000, 235–242.

[21] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, & D. Sorensen, *LAPACK users' guide*, Third Edition (Philadelphia, PA: SIAM, 1999).

Biographies

Carlile Lavor was born in Fortaleza, a city in the northeast part of Brazil. He is a mathematician and got his Ph.D. degree in Computer Science at Federal University of Rio de Janeiro, in 2001. In 2006, he obtained his Habilitation in Combinatorics at State University of Campinas, where he is currently an associate professor. He was a visiting professor at Politecnico di Milano, Universidad Politécnica de Madrid and École Polytechnique de Paris. His main research interests are molecular geometry optimization, bioinformatics and quantum information.

Antonio Mucherino was born in Caserta, Italy, a small city very close to Naples, where he made his studies. He is a mathematician and he has a Ph.D. degree in Computational Biology. After his Ph.D., he left Italy for the USA, and he spent 2 years as postdoc researcher at the Centre for Applied Optimization (University of Florida). Then he came back to Europe, and in particular to France, where he is currently working. He worked 18 months at the Ecole Polytechnique in Palaiseau, and, just recently, he started his current postdoc at INRIA, Lille Nord Europe. His interests are in global and combinatorial optimization, bioinformatics and data mining.

Leo Liberti was born in Milan, Italy. He is currently an associate professor at LIX, Ecole Polytechnique. He obtained his Ph.D. degree in Global Optimization at Imperial College, London in 2004, and worked as a postdoctoral fellow at Politecnico di Milano, Italy, during 2004–2005. His main research interests are reformulations in mathematical programming, global and combinatorial optimization with applications to complex industrial systems and bioinformatics.

Nelson Maculan was born in Londrina, Brazil. He is a mining and metallurgy engineer and got his Ph.D. degree in Operations Research at Federal University of Rio de Janeiro, in 1975. He obtained his Habilitation (HDR) in Management Science at University of Paris-Dauphine, in 1988. Since 1988, he is a full professor at Federal University of Rio de Janeiro. His main interests are in mathematical programming, numerical analysis and bioinformatics, with many papers published in these areas. He was a invited speaker of many important international conferences and visiting professor at prestigious universities around the world.