Determining protein structures using the discretizable molecular distance geometry problem

Silas Sallaume [a,1,2], Simone de Lima Martins [a,1,3], Luiz Satoru Ochi [a,1,4], Warley Gramacho da Silva [b,1,5], Carlile Lavor [c,1,6]

[a] *Departamento de Ciência da Computação, Universidade Federal Fluminense, Niterói - RJ, Brazil*

[b] *Campus Universitário de Palmas, Universidade Federal do Tocantins, Palmas - TO, Brazil*

[c] *Departamento de Matemática Aplicada, Universidade Estadual de Campinas, Campinas - SP, Brazil*

Abstract

One important problem in computational biology is the determination of the three-dimensional structure of proteins. Some information about protein structure can be obtained by using Nuclear Magnetic Resonance (NMR) techniques, but they provide only a sparse set of distances between atoms in a protein. The Molecular Distance Geometry Problem (MDGP) consists in determining the three-dimensional structure of a molecule using a set of known distances between some atoms. Generally, MDGP is expressed as a continuous optimization problem. Recently, a Branch and Prune (BP) algorithm was proposed to calculate the backbone of a protein, based on a discrete formulation for the MDGP. We present an extension of the BP algorithm that can calculate not only the protein backbone, but the whole three-dimensional structure of proteins. Since this new algorithm preserves the combinatorial approach of the BP algorithm, it can potentially find all the solutions of the problem (generally, the methods based on the continuous approach obtain just one solution). The proposed algorithm was able to efficiently find all the solutions of the problems associated to some of the most used proteins in the MDGP literature.

Keywords: discretizable molecular distance geometry problem, protein structure, computational biology

1 Introduction

The function of a protein is determined by its chemical and three-dimensional structures [2]. Some information about protein structure can be obtained by using Nuclear Magnetic Resonance (NMR) techniques, which are able to give a measure of the distance between pairs of atoms that are not greater than 6Å [11]. The problem of finding the atomic positions of a molecule, when only a given subset of atomic distances is known is called the Molecular Distance Geometry Problem (MDGP) [5]. In practice, the MDGP is solved by continuous optimization methods and they are usually capable to obtain just a single solution for the problem (for a survey on methods for the MDGP, see [6]).

In 2006, Lavor et al. [7] proposed a discrete formulation for the MDGP, called Discretizable Molecular Distance Geometry Problem (DMDGP), and presented a Branch and Prune (BP) algorithm that can calculate the backbone of a protein.

We present an extension of the BP algorithm that can calculate not only the protein backbone, but the whole three-dimensional structure of proteins. Since this new algorithm preserves the combinatorial approach of the BP algorithm, it can potentially find all the solutions of the problem. In the computational results, the proposed algorithm was able to efficiently find all the solutions of the problems associated to some of the most used proteins in the MDGP literature.

2 Calculating the whole protein structure

Formally, the MDGP can be defined as the problem of finding Cartesian coordinates $x_1, ..., x_n \in \mathbb{R}^3$ of atoms of a molecule such that

$$\| x_i - x_j \| = d_{i,j}, \qquad (i,j) \in S,$$

[1] We would like to thank FAPESP, FAPERJ, CNPq and CAPES for the financial support.
[2] Email: ssallaume@ic.uff.br
[3] Email: simone@ic.uff.br
[4] Email: satoru@ic.uff.br
[5] Email: wgramacho@uft.edu.br
[6] Email: clavor@ime.unicamp.br

where S is the set of pairs of atoms (i, j) whose Euclidean distances $d_{i,j}$ are known. If all distances are known, the problem can be solved in linear time [3]. In general, however, the problem is NP-hard [10].

In [7,8], it was showed that under the following assumptions (that are satisfied by most proteins), the MDGP can be formulated as a combinatorial problem, called DMDGP:

- all the distances $d_{i-3,i}, d_{i-2,i}, d_{i-1,i}$ must be known,
- the angles defined by each triplet of consecutive atoms cannot be equal to $k\pi$ (for $k \in \mathbb{Z}$),

for a given ordering of the atoms of a protein.

The BP algorithm proposed in [7,8] was applied to calculate just the backbone of artificial instances of the DMDGP. In fact, a real instance (protein molecule) is composed of a backbone and many side chains, which are always connected to one of the atoms of the backbone and appear systematically on each three atoms of the backbone [2].

It was empirically verified that the side chains can be considered as instances of the DMDGP, if they are treated isolated and by using an specific ordering of its atoms (in general, the atoms of the side chains are not linearly connected to each other) [9]. Thus, to determine the whole structure of a protein, several instances of the DMDGP should be solved. The difficulty lies in how to solve these instances in an efficiently and integrated manner.

The algorithm that we developed integrates the calculation of the positions of the atoms belonging to the protein backbone and also the positions of the atoms belonging to the side chains. To consider the atoms of the side chains as instances of the DMDGP, it was necessary to define an ordering of its atoms in order to satisfy the assumptions defined above. To do this, we formulate a linear programming problem that could be efficiently solved for any side chain.

The algorithm starts by fixing the position of the first three atoms of the protein backbone $(H, N, C_\alpha, C, N, C_\alpha, C, ..., N, C_\alpha, C)$. At the third atom (C_α), there is a side chain. So, a DMDGP is solved considering the chain formed by the first three atoms of the backbone and the atoms of the side chain, considering an ordering of its atoms previously determined. The known distances between the backbone atoms and the side chain atoms are used to eliminate some positions which are unfeasible. Then, DMDGP's are solved to find positions for the fourth, fifth and sixth atoms of the backbone and some of them are eliminated according to the known distances. At the sixth backbone atom, another DMDGP is solved to find positions for the side chain connected to the sixth backbone atom, considering the chain formed by the

fourth, fifth and sixth atoms of the backbone and the side chain atoms. This procedure follows in the same way until the positions for all the atoms of the protein backbone and all the side chains atoms are determined.

It was proved in [7] that for any solution found for solving the positions of the backbone atoms, there is another one that can be easily obtained without the application of the BP algorithm. We proved that this property also occurs when we apply our algorithm for finding the positions of all protein atoms. So, we used this property in the implementation of our algorithm in order to reduce computational costs by half.

3 Computational results

The real instances generated for the MDGP are extracted from the structures contained in the Protein Data Bank (PDB) [1]. The instances are generated by calculating the distances among all the atoms of a protein and discarding the distances that are above a cutoff value, which should be set to a value detectable by NMR techniques [4].

The code was written in C++ programming language by using the Standard Template Library and compiled by the Visual C++ 2005. All the experiments were carried out on an Intel Core 2, 1.6 GHz and 2GB RAM, running Windows XP with Service Pack 2.

Different metrics to measure the quality of solutions and different cutoff values are used in the generation of instances for the MDGP. All these factors hinder a fair comparison between the methods. Thus, for testing the proposed algorithm, we used some instances commonly found in the literature and the Largest Distance Error (LDE) as a measure of solution accuracy, defined by

$$LDE = \frac{1}{|S|} \sum_{(i,j) \in S} \frac{|\|x_i - x_j\| - d_{ij}|}{d_{ij}},$$

where S is the set of pairs of atoms (i, j) whose Euclidean distances d_{ij} are known and x_i, x_j are the Cartesian coordinates of atoms (i, j), respectively. The cutoff value for generating the instances was fixed in 6Å, which is the maximum value allowed to simulate data obtained through NMR [11].

The table below presents the obtained results. The column #Atoms indicates the number of atoms of the protein, the column #Sol shows the amount of found solutions, the column PDB indicates which of the found solutions has the greatest degree of similarity with the protein obtained from the PDB (using the RMSD value [8]), the column LDE shows the value in correspondence with the best found solution, and the column CPU shows the computational

Protein	*#Atoms*	*#Sol*	*PDB*	LDE	*CPU*
1brv	261	2	2	1.95e-17	0.5780
1ptq	402	8	2	3.91e-15	1.1400
1aqr	524	2	2	5.25e-17	1.9210
1hoe	558	4	2	6.93-e17	1.3590
1lfb	641	4	2	5.87e-17	1.5310
1ahl	684	2	1	3.97e-17	4.0000
1pht	811	8	2	6.41e-17	4.0930
1brz	859	2	1	5.29e-17	6.1710
1poa	914	32	9	7.50e-17	5.4210
1acz	1613	8	2	4.71e-17	17.078
1rgs	2015	4	1	1.58e-16	5.3590

time, in seconds, took by the method for finding all solutions.

The computational results show the very good performance of the algorithm and the high quality of the solution associated to the greatest degree of similarity with the protein obtained from the PDB. In fact, all the found solutions presented LDE values very small. As it was said, one of the main advantages of the combinatorial approach is the possibility to obtain all the solutions of the problem.

4 Conclusions

We presented a method to calculate the three-dimensional structure of a protein, using a set of known distances between some atoms of the protein obtained by NMR techniques. It was based on the BP algorithm, which can calculate just the protein backbone and was tested only on artificial instances. Now, it is possible to calculate the whole protein structure, including the side chains.

Since this new algorithm preserves the combinatorial structure of the BP algorithm, it was able to find all the solutions of the selected problems (generally, the methods based on the continuous approach obtain just on solution). In addition to this, the computational times for executing the algorithm were quite small and the quality of the generated solutions was very high.

References

[1] Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N., and Bourne, P.E., The Protein Data Bank, Nucleic Acids Research 28 (2000), 235-242.

[2] Creighton, T., Proteins: Structures and Molecular Properties (2nd edition), W.H. Freeman, New York, 1993.

[3] Dong, Q., and Wu, Z., A linear-time algorithm for solving the molecular distance geometry problem with exact interatomic distances, Journal of Global Optimization 22 (2002), 365-375.

[4] Gramacho, W., Algoritmos para o cálculo de estruturas de proteínas, Master Dissertation, Universidade Federal Fluminense (IC-UFF), 2008.

[5] Hendrickson, B.A., The molecule problem: exploiting structure in global optimization, SIAM Journal on Optimization 5 (1995), 835-857.

[6] Lavor, C., Liberti, L., and Maculan, N., Molecular distance geometry problem, Encyclopedia of Optimization (2nd edition), Springer, New York, 2305-2311, 2009.

[7] Lavor, C., Liberti, L., and Maculan, N. The discretizable molecular distance geometry problem, 2006, arXiv:q-bio/0608012v1.

[8] Liberti, L., Lavor, C., and Maculan, N., A branch-and-prune algorithm for the molecular distance geometry problem, International Transactions in Operational Research 15 (2008), 1-17.

[9] Sallaume, S. Cálculo de estruturas protéicas através da resolução do problema discreto de geometria das distâncias em moléculas, Master Dissertation, Universidade Federal Fluminense (IC-UFF), 2009.

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

[11] Schlick, T., Molecular Modeling and Simulation: An Interdisciplinary Guide, Springer, New York, 2002.