Asociación Argentina
de Mecánica Computacional

Mecánica Computacional Vol XXXI, págs. 2757-2763 (artículo completo)
Alberto Cardona, Paul H. Kohan, Ricardo D. Quinteros, Mario A. Storti (Eds.)
Salta, Argentina, 13-16 Noviembre 2012

SOLVING MOLECULAR DISTANCE GEOMETRY PROBLEMS WITH UNCERTAIN DATA

Rodrigo Lima[a], Carlile Lavor[b] and José Mario Martínez[c]

[a]*Instituto de Ciências Exatas, Universidade Federal de Itajubá, Itajubá, Minas Gerais, Brasil, rodlima@unifei.edu.br, http://www.unifei.edu.br*

[b]*Instituto de Matemática, Estatística e Computação Científica, Universidade Estadual de Campinas, Barão Geraldo, São Paulo, Brasil, clavor@ime.unicamp.br, http://www.ime.unicamp.br*

[c]*Instituto de Matemática, Estatística e Computação Científica, Universidade Estadual de Campinas, Barão Geraldo, São Paulo, Brasil, martinez@ime.unicamp.br, http://www.ime.unicamp.br*

Keywords: distance geometry, uncetain data, molecules, computational experiments.

Abstract. The Molecular Distance Geometry Problem consists in finding the positions in three dimensional space of atoms of a molecule, given some inter-atomic distances. We formulate this problem as a nonlinear optimization problem and solve some instances using a continuous optimization routine. To carry out the experiments, we assume initially that the distances have precise values and then add errors in order to simulate the real data provided by Nuclear Magnetic Resonance Data.

1 INTRODUCTION

The Molecular Distance Geometry Problem (MDGP) consists in finding the three dimensional structure of a molecule using only some distances between its atoms. In order to solve this problem, we need to obtain a set of n points $\{x^1, x^2, \ldots, x^n\} \subset \mathbb{R}^3$ such that $\|x^i - x^j\| = \hat{d}_{ij}$, where $\|\cdot\|$ is the Euclidean norm and $\hat{d}_{ij}$ is the Euclidean distance between the atoms i and j (Liberti et al., 2008, 2010; Lavor et al., 2012). If all the values $\hat{d}_{ij}$ are known exactly, the problem can be solved in linear time (Dong and Wu, 2002). However, the most interesting situation occurs when some distances $\hat{d}_{ij}$ contain errors. In this case, we say that these values are corrupted.

In this work we formulate the task of finding three dimensional structures as a continuous optimization problem as follows:

$$\begin{aligned} &\text{minimize} \quad \textstyle\sum_{i,j} (\|x^i - x^j\| - \hat{d}_{ij})^2, \\ &\text{subject to} \quad x^i \in \mathbb{R}^3, i = 1, 2, \ldots, n. \end{aligned} \tag{1}$$

A possible difficulty that arises in this formulation is the non-differentiability of the objective function when $x^i = x^j$ for all $i \neq j$. However, Jan de Leeuw proved in (De Leeuw, 1984) that if $\hat{d}_{ij} > 0$ for all i, j, the local minimizers of (1) are configurations that do not contain coincident points and a minimization algorithm that uses first derivatives can be applied to solve the problem.

When the distances between atoms of a protein are obtained without errors, the objective function of the formulation (1) has many global minimizers. In fact, this happens because any configuration of points that differs from the original structure by a rigid motion or a rigid motion composed with a reflection, can be a solution of the problem.

2 COMPUTATIONAL EXPERIMENTS

To carry out the experiments with the formulation (1), we use an optimization routine named *GENCAN* (Birgin and Martínez, 2002). This routine, available at www.ime.usp.br/~egbirgin/tango, is able to find approximate solutions to minimization problems with box constraints. We assume initially that all inter-atomic distances have precise values. After, we add errors in some distances to simulate real data and we try to investigate how these errors can affect the structures obtained. All experiments have been carried out on a single core of an Intel Core 2 CPU 2.4GHz with 2GB RAM running MAC OS X 10.5 and the codes are written in Fortran 77.

In order to find numerically an adequate set of points with lower value of the objective function we employ a multistart strategy: we solve the same instance of the problem (1) several times using a different initial point in each run. Each solution found by *GENCAN* is compared with the true structure through an alignment technique that we describe as follows.

2.1 COMPARING STRUCTURES WITH AN ALIGNMENT PROCEDURE

We represent each protein in a simplified way using only the 3D coordinates of nitrogen N, carbon C and alpha-carbon C_α presented in each amino acid. This representation captures the main features of the three-dimensional arrangements of amino acids in the molecule structure. The Figure 1 presents the generic sketch of an amino acid, where the atoms N (left) , C_α (center) and C (right) are shown and the letter G represents an organic substituent.

The most common way to compare structures is to superimpose them in some optimal manner and looking for their similarities and discrepancies after superimposition. We use

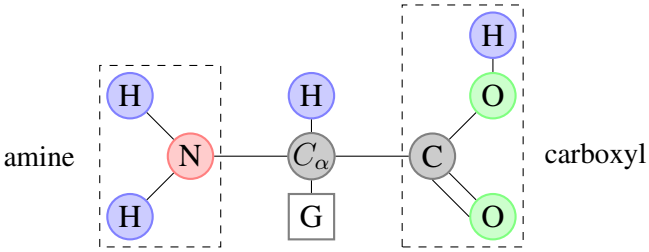


Figure 1: Sketch of an amino acid.

LOVOALIGN (Andreani et al., 2007, 2008) to compare structures. This routine, available at www.ime.unicamp.br/~martinez/lovoalign, measures the degree of similarity between two structures by maximizing the *Structal Score*

$$s = \sum \frac{20}{1 + (d/2.24)^2} - 10n_g, \tag{2}$$

where d is the Euclidean distance between C_α atoms of each compared structure and n_g is the number of gaps. According to (2), if two structures are identical, the *Structal Score* is given by $s = 20n$, where n is the number of atoms. In this case, if we divide s by n we obtain a normalized score. In the experiments we use this normalized score to decide if two proteins have some degree of similarity. The proteins that appear in our computational tests were extracted from *Protein Data Bank* (www.rcsb.org/pdb) and are shown in the Table 1. The proteins with the biggest number of atoms are featured by blue color.

1ACZ	1AHL	1AQR	1BVP	1BRV
1BRZ	1CRN	1EPW	1F39	1FS3
1HOE	1JK2	1LFB	1M40	1MBN
1MQQ	1N4W	1PHT	1POA	1PTQ
1RGS	1RWH	2E7Z	2ERL	3B34

Table 1: Proteins used in the computational experiments.

We summarize the main steps of our experiments as follows. To each protein in the Table 1, we take all distances between pairs of atoms and we add errors in some values to simulate real data, in a random way. Then, we solve the problem (1) hundred times employing a different initial point. In each run, the solution obtained by *GENCAN* is compared with the true structure using the routine *LOVOALIGN*. If the normalized score obtained after the comparison is approximately equal to 20, we declare success, otherwise, a new initial point is generated and the problem (1) is solved again with the same data. The results shown in the next tables correspond to runs where we obtain the highest values of normalized scores.

2.2 RESULTS

In order to investigate the effect of errors in the resolution of the problem (1), we carry out three sets of experiments with the selected proteins. In each set, we extract some values $\hat{d}_{ij}$ from the distance matrix associated to each protein and then, we add errors in such a way that the final (corrupted) values belong to the interval $[\hat{d}_{ij} - 2, \hat{d}_{ij} + 2]$. In the two first sets of experiments, we adopt the criterion used by Bonnie Berger *et al* (Berger et al., 1999) to select entries from the distance matrices. In the first set, we take only a small fraction of total distances to add

errors. To each protein, this fraction was determined according to the relation

$$d_{\text{err}} = \left\lfloor \frac{1}{2} n(1-\epsilon) \right\rfloor, \tag{3}$$

where n is the total number of atoms and ϵ is a random real number in $[0, \frac{1}{2})$. In the second set, we employ (3) to generate randomly wrong values by row in the distance matrix. It was proved that, in both cases, it is possible to reconstruct the original structure (Berger et al., 1999). Finaly, in the last set we suppose that 10% of the total distances are obtained with errors. In this case we choose the values $\hat{d}_{ij}$ in a random way to add errors.

The first three columns in Table 2 show, respectively, the name of each protein, the number of atoms considered and the total of inter-atomic distances. To each set of experiments d_{err} means the number of corrupted distances (with errors), $f(x)$ indicates the final value of objective function attained by *GENCAN* and s is the normalized score obtained after the comparison between the true structure and the numerical solution. Despite of final values of $f(x)$ do not be small in the tests, we get good values to normalized scores. This means that the numerical and true structures are quite similar and the errors do not affect much the quality of solutions obtained. In the table below, the proteins with the largest number of atoms are written with blue color.

			Set 1			Set 2			Set 3		
prot	atoms	d_t	d_{err}	$f(x)$	s	d_{err}	$f(x)$	s	d_{err}	$f(x)$	s
1ACZ	324	52326	51	6.22E+01	20.000	25477	3.32E+04	19.916	5232	6.90E+03	19.980
1AHL	147	10731	10	1.72E+01	19.999	5185	6.79E+03	19.782	1073	1.32E+03	19.952
1AQR	120	7140	3	5.17E+00	20.000	3437	4.39E+03	19.775	714	9.41E+02	19.961
1BPV	312	48516	38	4.42E+01	20.000	23617	3.11E+04	19.889	4851	6.27E+03	19.973
1BRV	57	1596	24	2.14E+01	19.998	753	8.50E+02	19.562	159	1.99E+02	19.900
1BRZ	159	12561	35	4.28E+01	19.999	6075	7.83E+03	19.807	1256	1.60E+03	19.970
1CRN	138	9453	68	7.98E+01	19.996	4552	5.83E+03	19.782	945	1.21E+03	19.949
1EPW	3861	7451730	672	9.13E+02	20.000	3649435	4.86E+06	19.989	745173	9.92E+05	19.998
1F39	303	45753	30	4.13E+01	20.000	22269	2.90E+04	19.895	4575	5.90E+03	19.976
1FS3	372	69006	168	2.24E+02	19.999	33626	4.38E+04	19.917	6900	9.04E+03	19.983
1HOE	222	24531	102	1.20E+02	19.999	11908	1.54E+04	19.855	2453	3.20E+03	19.974
1JK2	270	36315	18	2.02E+01	20.000	17658	2.31E+04	19.844	3631	4.66E+03	19.961
1LFB	232	26796	76	9.10E+01	19.999	13013	1.67E+04	19.816	2679	3.39E+03	19.971
1M40	1224	748476	361	4.79E+01	20.000	366145	4.87E+05	19.975	74847	9.96E+04	19.995
1MBN	459	105111	56	7.31E+01	20.000	51276	6.77E+04	19.936	10511	1.40E+04	19.986
1MQQ	2032	2063496	316	3.76E+02	20.000	1010105	1.34E+06	19.984	206349	2.75E+05	19.997
1N4W	1610	1295245	735	9.90E+02	20.000	633872	8.40E+05	19.981	129524	1.72E+05	19.996
1PHT	249	30876	6	6.02E+02	20.000	15006	1.98E+04	19.885	3087	3.99E+03	19.977
1POA	354	62481	108	1.27E+02	20.000	30440	3.96E+04	19.889	6248	8.17E+03	19.978
1PTQ	150	11175	7	1.13E+01	20.000	5402	6.78E+03	19.785	1117	1.44E+03	19.965
1RGS	792	313236	45	6.00E+01	20.000	153092	2.03E+05	19.957	31323	4.14E+04	19.991
1RWH	2265	2563980	970	1.32E+03	20.000	1255227	1.67E+06	19.986	256398	3.42E+05	19.997
2E7Z	2907	4223871	675	9.01E+02	20.000	2068257	2.76E+06	19.990	422387	5.63E+05	19.998
2ERL	120	7140	56	6.71E+01	19.995	3437	4.41E+03	19.690	714	8.82E+02	19.926
3B34	2790	3890655	1121	1.51E+03	20.000	1905038	2.53E+06	19.989	389065	5.18E+05	19.998

Table 2: Reconstructing 3D structures with errors in the distances.

The performance of routine *GENCAN* in the three sets of experiments are shown in the Table 3. In this table *iter* is the number of total iterations, *evalf* is the total of evaluations of objective function and t is the CPU time in seconds. We remember that the values in this table correspond to runs where we obtained the highest values of normalized scores. We can note

that the numbers of iterations and evaluations of function are small and the values of time are reasonable in all tests.

	Set 1			Set 2			Set 3		
prot	iter	evalf	t	iter	evalf	t	iter	evalf	t
1ACZ	17	38	0.320	19	38	0.284	19	42	0.352
1AHL	19	52	0.064	23	49	0.088	19	46	0.068
1AQR	16	37	0.036	16	38	0.044	15	38	0.036
1BPV	15	25	0.268	17	33	0.268	17	30	0.208
1BRV	17	42	0.012	19	40	0.012	13	22	0.008
1BRZ	14	20	0.076	16	41	0.064	17	31	0.080
1CRN	15	25	0.064	15	21	0.052	18	33	0.056
1EPW	20	34	41.843	21	39	53.675	22	45	51.700
1F39	18	24	0.292	18	40	0.236	18	45	0.320
1FS3	19	38	0.280	19	42	0.384	17	35	0.324
1HOE	15	24	0.112	16	32	0.108	15	26	0.112
1JK2	18	49	0.236	20	46	0.320	27	75	0.576
1LFB	26	65	0.248	18	33	0.192	18	26	0.160
1M40	16	28	3.732	19	38	2.748	17	40	2.968
1MBN	17	37	0.500	16	36	0.472	16	39	0.556
1MQQ	19	37	8.373	26	85	12.645	22	61	11.420
1N4W	16	36	4.864	15	36	4.332	19	37	5.204
1PHT	17	26	0.204	16	34	0.148	14	28	0.164
1POA	18	33	0.340	17	33	0.320	17	41	0.440
1PTQ	14	22	0.056	20	32	0.056	15	27	0.036
1RGS	18	32	1.460	19	43	1.872	18	37	1.720
1RWH	22	57	16.553	21	50	11.697	18	35	10.510
2E7Z	22	50	17.189	20	34	16.441	24	54	22.720
2ERL	16	21	0.040	17	27	0.040	15	29	0.040
3B34	21	58	20.565	20	43	15.685	20	45	21.060

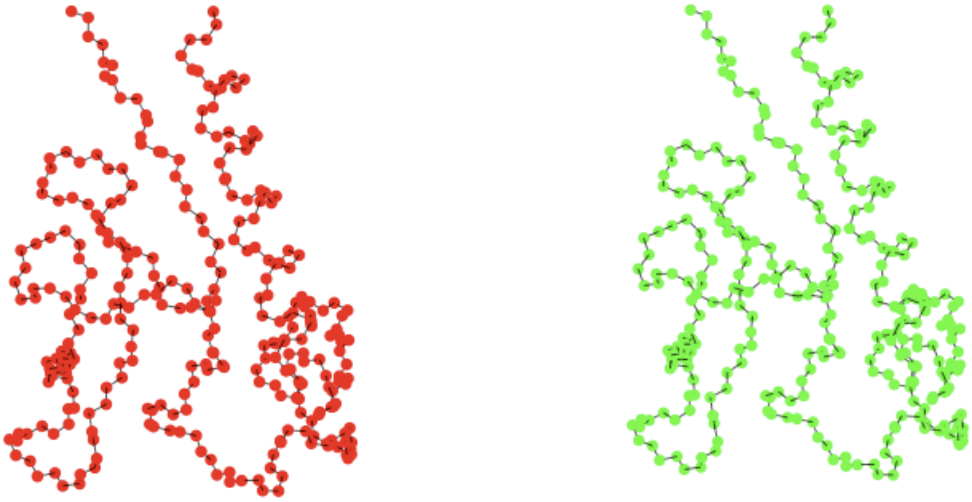
Table 3: Performance of routine *GENCAN*.

In order to illustrate some results of the tables presented before, we considered the protein named 1EPW. The Figure 2 a) shows the true structure and the Figure 2 b) indicates the numerical solution obtained by *GENCAN* in the experiment of Set 3. Both figures were constructed using only the first 300 atoms of each structure (N, C_α, C). The atoms were represented by points and consecutive atoms were joined by line segments. We also compute all distances between pairs of atoms in each structure and then we plot a graph to investigate the results. The graph of Figure 3 shows, respectively, the true distances $\hat{d}_{ij}$ (without errors) in the x axis and the final distances between points $\|x^i - x^j\|$ of the numerical solution in the y axis.

We remember that the problem (1) was solved using the values $\hat{d}_{ij}$ with errors. We can observe in this test that despite of the fact that the final value of objective function is not small ($\approx 9 \times 10^5$), the final distances are reasonably adjusted to true distances (without errors). In fact, the value of the normalized score obtained in this experiment is approximately equal to 20. This means that we obtained a structure that has great similarity with the true configuration.

3 CONCLUSIONS

In this work we proposed an optimization approach to solve some instances of the Molecular Distance Geometry Problem (MDGP). We considered the case where some distances between atoms have errors and then we solved a minimization problem to recover the true structure using only information about interatomic distances. The problems were solved with a routine named *GENCAN*.



a) true structure b) numerical structure

Figure 2: Comparing structures.

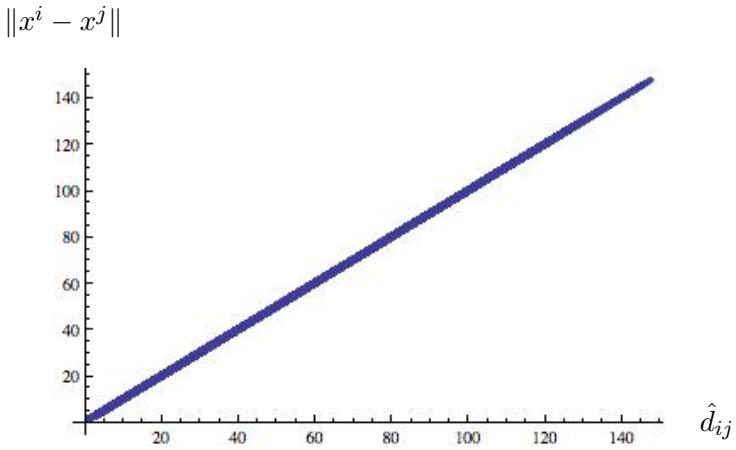


Figure 3: True distances without errors *versus* numerical distances.

In order to investigate the quality of obtained solutions, we compared the true structure with the numerical one employing an alignment procedure. The degree of similarity after the comparison was measured by the normalized *Structal Score*, a real number in the interval $[0, 20]$. According to numerical experiments, we observed that in all tests the obtained configurations had great similarities with the true structures. We know that the criterion of comparison adopted in this work is not realistic. In fact, we used the true structure to evaluate the quality of numerical solutions. However, in a future work we intend to investigate other methods to predict if the solutions found are good or not without realize any comparison with the true structure.

REFERENCES

Andreani R., Martínez J.M., and Martínez L. Continuous optimization methods for structure alignment. *Mathematical Programming*, 112:93 – 124, 2008.

Andreani R., Martínez L., and Martínez J.M. Convergent algorithms for protein structural alignment. *BMC Bioinformatics*, 8:1 – 15, 2007.

Berger B., Kleinberg J., and Leighton T. Reconstructing a three-dimensional model with arbitrary errors. *Journal of ACM*, 46:212–235, 1999.

Birgin E.G. and Martínez J.M. Large-scale active-set box-constrained optimization method with spectral projected gradients. *Computational Optimization and Applications*, 23:101–125, 2002.

De Leeuw J. Differentiability of kruskal's stress at a local minimum. *Psychometrika*, 49:111–113, 1984.

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

Lavor C., Mucherino A., Liberti L., and Maculan N. Recent advances on the discretizable molecular distance geometry problem. *European Journal of Opertational Research*, 219:698–706, 2012.

Liberti L., Lavor C., and Maculan N. A branch-and-prune algorithm for the molecular distance problem. *Iternational Transactions in Operational Research*, 15:1–17, 2008.

Liberti L., Lavor C., Mucherino A., and Maculan N. Distance geometry methods: from continuous to discrete. *Iternational Transactions in Operational Research*, 18:33–51, 2010.