Journal of Mathematical Chemistry Vol. 37, No. 3, April 2005 (© 2005)
DOI: 10.1007/s10910-004-1466-4

Molecular conformation search by distance matrix perturbations

Ioannis Z. Emiris*

Department of Informatics & Telecommunications, National University of Athens, Panepistimiopolis, Athens 15784 Greece

E-mail: emiris@di.uoa.gr

Theodoros G. Nikitopoulos

Department of Computer Science, University of Crete, Heraklio, Greece

Received 20 February 2004; revised 25 April 2004

Three-dimensional molecular structure is fundamental in chemical function identification and computer-aided drug design. The enumeration of a small number of feasible conformations provides a rigorous way to determine the optimal or a few acceptable conformations. Our contribution concerns a heuristic enhancement of a method based on distance geometry, typically in relation with experiments of the NMR type. Distance geometry has been approached by different viewpoints; ours is expected to help in several subtasks arising in the process that determines 3D structure from distance information. More precisely, the input to our algorithm consists of a set of approximate distances of varying precision; some are specified by the covalent structure and others by Nuclear Magnetic Resonance (NMR) experiments (or X-ray crystallography which, however, requires crystallization). The output is a valid tertiary structure in a specified neighborhood of the input. Our approach should help in detecting outliers of the NMR experiments, and handles inputs with partial information. Moreover, our technique is able to bound the number of degrees of freedom of the conformation manifold. We have used numerical linear algebra algorithms for reasons of speed, and because they are well-implemented, fully documented and widely available. Our main tools include, besides distance matrices, structure-preserving matrix perturbations for minimizing singular values. Our MATLAB (or SCILAB) implementation is described and illustrated.

KEY WORDS: distance geometry, local conformation search, matrix perturbation, singular value

AMS subject classification: 92E10 Molecular structure, 92C40 Biochemistry, molecular biology, 65F15 Eigenvalues, eigenvectors, 15A18 Eigenvalues, singular values, and eigenvectors

*Corresponding author.

1. Introduction

Structural proteomics is today a major challenge in computational chemistry and molecular biology. Drug design and discovery relies increasingly on structure-based methods in order to improve efficiency and accuracy. Three-dimensional geometric (i.e., tertiary) structure is essential in function identification, docking of small flexible ligands to macromolecules as well as pharmacophoric pattern matching.

If we were given all exact pairwise distances between a set of points, their 3D coordinates could be immediately obtained. So these are equivalent expressions of the tertiary structure. Furthermore, distances provide an excellent model for studying 3D molecular conformations because they can capture the geometry as determined by the torsion dihedral angles about bond axes. The hypothesis is that, at a first approximation, conformations depend only upon the dihedral angles, whereas bond angles and bond lengths can be considered as rigid. This is valid because distortions of bond angles and lengths require much more energy than changes of the dihedral angles; e.g., Ref. [1]. Therefore, when talking about degrees of freedom in the rest of this paper, we shall refer to varying dihedral angles. In the case of proteins, the conformation is determined by the structure of the polypeptide chain. Rotation is permitted around the angles ϕ, ψ and ω, each triplet corresponding to one amino-acid residue. However, the last one is usually at an angle of $\omega = \pi$, said to be in a *trans* state, and rarely at $\omega = 0$ if at a *cis* state. So, in most cases, we focus on the first two angles.

Distance data can, at least partially, be provided by the contact map, which is precisely what Nuclear Magnetic Resonance (NMR) experiments offer. This input data is obtained by exploiting the Nuclear Overhauser Effect (NOE), a powerful and mature technology, which is further improving nowadays. Remark that distances between neighboring atoms are readily computable. Section 2 formalizes the notions of distance geometry, and discusses existing work related to our approach in more detail.

Today, new distance geometry methods are sought (e.g., Refs. [2–5]) in order to contribute in the quest of massive-throughput structure determination, e.g., Refs. [6,7]. One factor for this is the automatization of spectrum assignment in NMR experiments, which shall increase significantly the available distance data. One application of distance geometry is to assign a level of confidence to distance information, which may be imprecise for the new needs of computational chemistry and molecular biology. This can be achieved by treating some inputs as accurate while allowing to perturb inputs that seem false or are simply not yet available due to the length of the phase of spectrum assignment in NMR. Certain methods consider entire families of proteins with homological similarities, and try to treat them with the minimal possible distance information. The latter may come from an on-line NMR process, where some of the intervals are inaccurate or simply not known.

We expect that our use of distance geometry shall help in several subtasks arising in the procedures that predict 3D structure from distance information. Typically, one applies a sequence of algorithms in order to refine the input and then filter the set of allowable conformations through a series of tests. Our algorithm should work in conjunction, and possibly in alternation, with certain other processes such as bound smoothing (usually by application of the triangle and tetrangle inequality), segmentation into substructures, and outlier elimination. Another stage typically enforces energy minimization and a number of chemical conditions on the current candidate conformations.

In our setting, the primary structure is considered as known, which enables us to deduce certain distances. The bond angles can usually be determined from the covalent structure, while for fixed bond lengths there is a one-to-one relation between the bond angle and the geminal distance so that these distances can also be determined. For proteins, this information represents the approximate distances between certain pairs of backbone Carbon atoms. The distances across rotatable bonds usually vary within their *cis*/*trans* limits, and all the distances within any known rigid group of atoms (e.g., amino-acid residues or phenyl rings) are constrained to their known values. Distances that are unknown and not given by the experimental data are constrained by the triangle (and tetrangle) inequality and must satisfy certain obvious bounds, such as the one corresponding to van der Waal's forces.

Our algorithm uses local search to identify molecular conformations when a partial set of pairwise Cartesian distances between atoms is known with some error. If it is given a known valid conformation, our technique can explore nearby conformations lying on the same manifold of all allowable conformations, hence also topologically close to each other. This answers the need of biased sampling in order to avoid previously sampled configurations. Our technique offers the freedom to choose the direction of exploration. Furthermore, it is able to bound the dimension of the manifold.

Distance matrices contain the pairwise distances between atoms, so we formulate the problem of computing conformations as a *structured singular value (or eigenvalue) minimization* problem. This is an optimization problem involving eigenvalues, which are values of a function defined in the matrix subspace to be specified below; this subspace lies in the set of real symmetric matrices. Given distance approximations (or interval constrains, respectively), the aim is to find values near the given approximations (or in these intervals) so that the structure can be embedded in 3D Euclidean space. Although the algorithm is numerical for reasons of speed, it guarantees its output under certain assumptions. It has been implemented on MATLAB and on SCILAB. It outputs backbone conformations, such as the one shown in figure 1, for the example of a cyclic molecule. To give an example of its performance, our code can determine a conformation of a molecule with 20 degrees of freedom in 3.79 s on a 500-MHz PENTIUM-III processor. We implemented the triangle inequality in order to preprocess the given intervals iteratively.

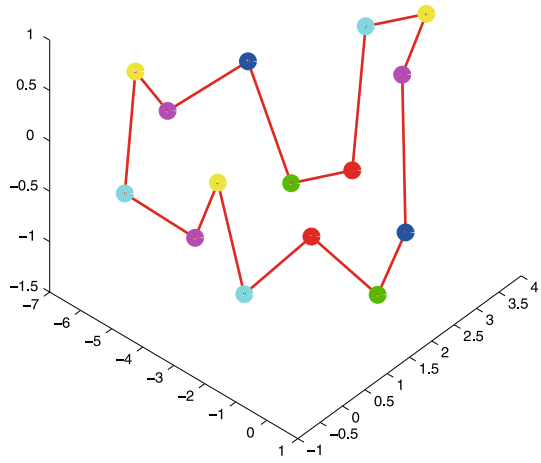

Figure 1. One backbone conformation computed by our program, for a cyclic molecule with 15 degrees of freedom.

The article is organized as follows. The next section overviews the theory of distance matrices and how it is used in conformation search. Section 3 contains the background on structured matrix perturbations and existing work in the area. Section 4 elaborates on our algorithm and sketches our MATLAB and SCILAB implementations. Section 5 applies distance geometry and our approach to cycloalkanes, whereas the following section reports on experimental results for more general molecules. Section 7 sketches further work.

2. Distance matrices

We review techniques related to distance geometry and introduce distance matrices; then, we formalize certain algebraic properties and the problem to be treated.

The problem of identifying the conformation of proteins of known amino-acid sequence, by using a model of residue–residue energy-like potential, was the underlying motivation in exploring the theory of distance geometry [8,9]. Distance geometry applications have been quite successful (e.g., Refs. [5,9–12]), contributing in the conformational analysis of molecules with about 200 residues [13]. Given an incomplete set of distances, the question of 3D embedding is a global optimization problem; see, e.g., Refs. [9,11,14].

Distance geometry has been implemented in certain packages. Most often, it constitutes one phase in a series of algorithms applied to refine the experimental data and filter the set of predicted conformations until a few valid candidates are produced. One package that is currently maintained and freely available is DGSOL [15], which relies on continuation methods for global

optimization in order to trace the minimizing configurations. A package that used to be very popular is EMBED, which explores the conformation space by random sampling. The key idea lies in minimizing an error function, which measures the total violation of the distance constraints after a best-fit embedding of the structure in Euclidean space. Since there is a lot of freedom in choosing this function, it is possible to make it smooth and well-behaved for optimization. A number of different conformations have been obtained for molecules with about 100 atoms or more [9]. To ensure completeness, linearized embedding uses the metric matrix, which contains the inner products between vectors defining local coordinate systems within the molecule [16]. Other packages for molecular conformations using distance geometry include HELIX, DGEOM, DPSACE, VEMBED, and DYANA [3]. DYANA relies on local information, hence it handles well nearby atoms, but may encounter difficulties with those lying far apart on the chain. It is quite fast on many classes of inputs, depending on the way the constraints are distributed along the chain. DYANA is an example of using local (spherical) coordinates, which offers an interesting general approach (see, also Ref. [16]). This and other software is found at Refs. [17,18].

The speed of modern hardware has revived an interest into algebraic methods, which may handle efficiently substructures of small size as part of larger problems in computational chemistry and structural biology. Hence, algebraic techniques have been applied to conformational search, since they offer completeness, raise no issues of convergence, and can certify their results, e.g., [16, 19–22]. Modeling the molecular problem in algebraic terms is achieved, in a general manner, by distance geometry. However, all of these methods have complexity exponential in the number of degrees of freedom, so they are limited to small molecules, say with at most a dozen of degrees of freedom. The goal of this paper is to exploit the power of distance matrices while studying molecules of larger sizes, by employing numerical linear algebra techniques.

It is time now for a formal presentation of distance geometry; for further details see Refs. [8,9,19]. Suppose that there are n points; these shall correspond to the backbone atoms allowed to rotate. Let $d_{ij}, i, j \in \{1, \ldots, n\}$, denote the Euclidean distance between the corresponding points, with $d_{11} = \cdots = d_{nn} = 0$. We define the corresponding symmetric *distance matrix (or Cayley–Menger matrix)* by

$$D(1, \ldots, n) := \begin{bmatrix} 0 & \frac{1}{2}d_{12}^2 & \ldots & \frac{1}{2}d_{1n}^2 & 1 \\ \frac{1}{2}d_{12}^2 & 0 & \ldots & \frac{1}{2}d_{2n}^2 & 1 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \frac{1}{2}d_{1n}^2 & \frac{1}{2}d_{2n}^2 & \ldots & 0 & 1 \\ 1 & 1 & \ldots & 1 & 0 \end{bmatrix},$$

which contains, besides the adjacency matrix as a (principal) submatrix, an additional row and column of units, with the diagonal being zero.

Theorem 2.1 [8]. A necessary and sufficient condition for the distance matrix $D(1, \ldots, n)$ to express a point set embeddable in m-dimensional Euclidean space E_m, $m \geqslant 3$ is, for some ordering of the points, to have (i) for $k = 1, \ldots, m$,

$$(-1)^{k+1} \det D(1, \ldots, k+1) > 0,$$

and (ii) for any points u, v, $m+1 \leqslant u$, $v \leqslant n$,

$$\det D(1, \ldots, m+1, u) = \det D(1, \ldots, m+1, v) = \det D(1, \ldots, m+1, u, v) = 0.$$

Let us illustrate this theorem with some examples. Part (i) for $k = 1$ becomes $d^2_{i_1 i_2} > 0$, which ensures that points $i_1, i_2 \in \{1, \ldots, n\}$ are distinct. For $k = 2$, condition (i) becomes

$$4 \det (D(i_1, i_2, i_3)) = \left(d^2_{i_2 i_3} - d^2_{i_1 i_2} - d^2_{i_1 i_3}\right)^2 - 4 d^2_{i_1 i_2} d^2_{i_1 i_3} < 0$$

for any three points $i_1, i_2, i_3 \in \{1, \ldots, n\}$. This is violated, since the determinant vanishes, when the points indexed by i_1, i_2, i_3 are collinear. The condition is satisfied if the respective distances satisfy the triangle inequality. We generalize this inequality, and derive it from the law of cosines, at the end of section 4, expression (3). Part (ii) is satisfied if matrix $D(1, \ldots, n)$ has rank $m+2$.

Corollary 2.2. With the notation of the previous theorem, $n \geqslant 4$ distinct points (not all coplanar) are embeddable in E_3 if and only if $\operatorname{rank}(D(1, \ldots, n)) = 5$.

This corollary provides the foundation of our approach. For an independent proof of this fact, the reader may refer to Ref. [19]. So, the problem of mapping the input points to E_3, is equivalent to perturbing, or completing, matrix $D(1, \ldots, n)$ so that its rank becomes 5. In fact, we may use the matrix containing simply the squared distances d^2_{ij}, zeros on the diagonal and units on the last column and last row. Our discussion has thus reduced the problem of identifying a 3D structure to an algebraic test.

One variant of the problem can be stated as follows: Given an incomplete, undirected, weighted graph G, the *molecular Euclidean embedding* problem is that of mapping the nodes (or vertices) of G to points in the 3D Euclidean space E_3 so that any two nodes with an edge between them are mapped to points whose Euclidean distance equals the weight of that edge. Then, the point set, or the corresponding distance matrix, is said to be embeddable. This problem is NP-hard with respect to E_k for any $k \geqslant 2$, even if all given distances lie in $\{1, 2\}$ [23]. It was one of the first geometric problems shown to be in this class.

Fundamental work exists concerning general distance matrices. Let $\| \cdot \|_2$ stand for the 2-norm (or Euclidean norm) of vectors or of matrices; this is the

default norm, understood when none is specified. For vector $v = (v_i)_i$ and for matrix M we have

$$\|v\|_2 = \left(\sum_i v_i^2 \right)^{1/2}, \quad \|M\|_2 = \max_{v \neq 0} \frac{\|Mv\|_2}{\|v\|_2}.$$

A relevant property of distance matrices is the following.

Theorem 2.3 [24]. Let $e^T = [1, \ldots, 1]$ be the vector of units. For any vector s such that $s^T e = 1$, and any square matrix M, define the norm

$$|M|_s := \| (I - es^T) M (I - es^T)^T \|_2,$$

where I is the identity matrix with the same dimension as M. Given a distance matrix D and any vector s, we can construct a new distance matrix D' embeddable in E_3 such that $|D - D'|_s$ is minimized.

This construction is based on the truncation of the matrix of singular values, hence its computation is relatively fast. This has been implemented but it is only a secondary tool in our approach, since it does not respect the information which is available in the form of distance intervals or other prior information on the entries of D.

3. Matrix perturbations

This section presents numerical linear algebra approaches for the problems related to distance matrices, namely rank reduction while preserving structure. We shall thus describe the algorithmic basis of our technique.

Reducing a specific subset of eigenvalues and bringing them close to zero has been addressed in numerical analysis in different contexts (see [4, 25–27] and the references thereof). We shall focus on the latter approach, which studies the minimization of the last singular value, because it is possible to devise *structured rank-reducing perturbations* which preserve (at least) symmetry, reality and zero diagonal by modifying only certain entries. Wicks and Decarlo [27] propose a modified Newton-type iteration in order to avoid instabilities near the minimum, where the derivative vanishes. Moreover, this modification ensures global convergence at a nearly quadratic rate, including in the case of arbitrary complex rectangular matrices. We shall extend these methods in order to reduce more than one singular values, while maintaining the structure. It is possible to specify the set of perturbable entries and the perturbation magnitude per entry, hence defining the direction of the perturbation in the search space. If, moreover, we limit the magnitude of the perturbation per entry, we are able to search in a neighborhood of our choice.

The rest of the section presents the notions of linear algebra required; for further information (see [28–30]). Let us consider an $N \times N$ matrix M. The *eigenvalues* of M are the scalar solutions of λ in the vector equation $Mv = \lambda v \Leftrightarrow (M - \lambda I)v = 0$, for some column nonzero vector $v \in \mathbb{R}^N$, which is called the associated right eigenvector, where I is the $N \times N$ identity matrix and 0 stands for the N-dimensional zero vector. It is possible to write the above equations using the left associated column eigenvector u, namely $u^T M = \lambda u^T \Leftrightarrow u^T(M - \lambda I) = 0$. When M is real and symmetric, its eigenvalues λ are real and its (left or right) eigenvectors form an orthonormal basis of $\mathbb{R}^N$. The real symmetric eigenvalue, or spectral decomposition, problem is equivalent to solving matrix equation $M = Q^T \Lambda Q$, where $Q^T = Q^{-1}$ contains the eigenvectors as columns of Q, and diagonal matrix Λ contains the eigenvalues. This decomposition is numerically unstable, therefore hard to compute, and shall thus be avoided.

A more useful matrix decomposition is the Singular Value Decomposition (SVD), which writes $M = Q_1 \Sigma Q_2^H$, where $Q_i^H = Q_i^{-1}$, $i = 1, 2$, and Q_i^H stands for the transposed conjugate matrix, and Σ is a diagonal matrix containing the *singular values* of M. The columns of Q_1, Q_2 are the left and right singular vectors associated to the respective singular values. The SVD decomposition is unique if we require that the singular vectors be of unit length. The absolute values of the eigenvalues are the singular values. For a real symmetric M, both Q_i are real. Moreover, the associated left and right singular vectors are either equal or opposite to each other; the latter case occurs exactly when the corresponding eigenvalue is negative. The singular vectors are equal to the corresponding eigenvectors within sign. The *rank* of a matrix is the number of nonzero eigenvalues, or the number of nonzero singular values. If the rank is smaller than the minimum of the row or column dimension, then the matrix is said to be singular. For square matrices, this is equivalent to the vanishing of the determinant. Rank computations rely on the SVD because it is in practice faster and more stable numerically.

Our method makes use of the Moore–Penrose *pseudo-inverse* of a matrix M. This is the unique matrix M^+ satisfying

$$MM^+M = M, \quad M^+MM^+ = M^+, \quad (MM^+)^H = MM^+, \quad (M^+M)^H = M^+M.$$

If M is $m \times n$ and has full rank then M^+ equals either $(M^H M)^{-1} M^H$ for $m \geqslant n$, or $M^+(MM^+)^{-1}$ for $m \leqslant n$. If M is Hermitian then $M^+M = MM^+$ [28]. There are efficient and accurate implementations for the pseudo-inverse. In particular, the computation based on the SVD, for a real symmetric matrix M is as follows. Consider that $M = Q_1 \Sigma Q_2^T$ with $\Sigma = \mathrm{diag}(\sigma_1, \ldots \sigma_r, 0, \ldots, 0)$, then $M^+ = Q_2 \Sigma^+ Q_1^T$, where

$$\Sigma^+ = \mathrm{diag}\left[\frac{1}{\sigma_1}, \ldots, \frac{1}{\sigma_r}, 0, \ldots, 0\right], \quad r = \mathrm{rank}(M).$$

Here, diag$[a_1, \ldots, a_N]$ stands for a diagonal matrix with entries $a_1, \ldots, a_N$, and the singular values are ordered: $\sigma_1 \geqslant \cdots \geqslant \sigma_r > 0$.

Let u and v be vectors of reals with $u^T v \neq 0$. Then, $\left(u^T M v\right)/\left(u^T v\right)$ is a *Rayleigh quotient*. If either u or v is (respectively, near) an eigenvector corresponding to an eigenvalue λ of M, then the Rayleigh quotient reproduces (resp. approximates) that eigenvalue. Iterative algorithms for eigenvalue computation and, in particular, spectral decompositions (like the power method), use the Rayleigh quotients to iteratively improve a numerical approximation of λ.

Proposition 3.1 (Extremal property of Rayleigh quotients) [29]. Let σ_j, v_j denote respectively the ordered singular values and (right) singular vectors of M. Then,

$$\sigma_n = \min_{\|x\|_2=1} x^T M x, \quad x \in \mathbb{R}^N,$$

provided $x^T v_j = 0$ for each $n+1 \leqslant j \leqslant N$.

Now we state the main property concerning the singular values (and eigenvalues) of matrices under small perturbations.

Theorem 3.2 [30, Thm.IV.2.3]. Let R be a matrix with the same dimensions as matrix M. If $\sigma_k(\cdot)$ denotes the kth singular value of some matrix, then the function

$$f(\xi) := \sigma_k(M - \xi R),$$

is differentiable with respect to real variable ξ, as long as $\sigma_k(M - \xi R)$ is distinct from all other singular values, for all ξ. Moreover, for M real and symmetric, $f'(\xi) = -u_k^T R v_k$, where u_k, v_k are the singular vectors of $M - \xi R$ associated to $\sigma_k(M - \xi R)$.

An analogous result applies to eigenvalues and eigenvectors. The theorem yields, immediately, the following Newton-type iteration for minimizing the smallest singular value and, eventually, driving it to zero.

$$\xi \leftarrow \xi + \frac{u_N^T (M - \xi R) v_N}{u_N^T R v_N} = \frac{u_N^T M v_N}{u_N^T R v_N},$$

where u_N, v_N are the singular vectors of $M - \xi R$ associated with singular value $\sigma_N(M - \xi R)$. Then, the Nth singular value of $M - \xi R$ approaches zero with a quadratic rate.

So far, we have discussed essentially one-dimensional perturbations, since there is a single variable controlling the matrix change. In Ref. [27], this is generalized to structured perturbations, i.e., where several entries are perturbed in an independent fashion; the approach is also extended to rectangular complex

matrices. Let R_{ij} be a *perturbation matrix* having the same dimensions as M and zeros in all entries except of units at entries (i, j) and (j, i), where $1 \leqslant i < j \leqslant N$. The number of independent R_{ij} is denoted by p. Let $\mathcal{R}$ be a subspace of symmetric square matrices of dimension p generated by the $R_{k_i k_j}$, $1 < k_i < k_j \leqslant N$:

$$\mathcal{R} = \left\{ \sum_{k=1}^{p} \xi_k R_{k_i k_j} \ : \ [\xi_1, \ldots, \xi_p] \in \mathbb{R}^p \right\}.$$

Now, $(u^T M v)/(u^T R_{ij} v)$ is called the *generalized Rayleigh quotient* with respect to M. The algorithm in Ref. [27, secion 4], specialized to a square real symmetric $N \times N$ matrix M, has the following input, output, and procedure:

Algorithm 3.3. Input: a square real symmetric $N \times N$ matrix M, and a tolerance $\tau > 0$ that determines numerical zero.
Output: a square real symmetric $N \times N$ matrix whose last (i.e., smallest) singular value is smaller than τ and which lies in the neighborhood of M. The output matrix is singular within τ.

0. Initialize the perturbation matrix $R \in \mathcal{R}$, possibly to the zero matrix.
1. Compute the SVD decomposition $M - R = U \Sigma V^T$, where the Nth singular value and vectors are denoted by σ_N, u_N, v_N, respectively.
2. Let perturbation matrix $\Delta \in \mathcal{R}$ have minimum norm such that it minimizes $\|u_N^T \Delta v_N - u_N^T (M - R) v_N\|$.
3. Define quantities $\alpha, \gamma \in \mathbb{R}$ as follows. Let

$$\alpha \leftarrow \ \| \ u_N^T \Delta (I - v_N v_N^T)(M - R)^+ \Delta \ \|, \qquad \gamma \leftarrow \ \min \left\{ \ 1, \frac{\|u_N^T \Delta v_N\|}{4 \alpha \sigma_N} \ \right\},$$

and set $R \leftarrow R + \gamma \Delta$.
4. If $\gamma \|\Delta\| < \tau \|M - R\|$, then the algorithm terminates; otherwise, go to step 1.

In steps 2–4, the 2-norm is used. Step 2 reduces to finding vector $\xi \in \mathbb{R}^p$, which defines Δ in the basis of the R_{ij}, assuming $\|\Delta\| = \|\xi\|$. Equivalently, the algorithm must compute $\xi = E^+ F$, where E is the p-dimensional vector $[\ldots, v_N^T R_{ij}^T u_N, \ldots]$, where the (i, j) range over all entries of M to be perturbed independently, and $F = u_N^T (M - R) v_N$.

Step 3 is designed so that the algorithm achieves nearly quadratic rate of convergence as it approaches the minimum. In implementing it, we can simplify the calculations by using relation

$$(I - v_N v_N^T) \ (M - R)^+ = [v_1, \ldots, v_{N-1}, 0] \ \mathrm{diag} \left[\frac{1}{\sigma_1}, \ldots, \frac{1}{\sigma_{N-1}}, 0 \right] \ U^T.$$

Theorem 3.4 [27]. Suppose that $\|\Delta\|$ is always bounded; for this, it suffices that $\|E^+\|$ remains bounded. Then, Algorithm 3.3 makes $\sigma_N(M-R)$ approach zero, unless $u_n^T R v_N$ tends to zero for all $R \in \mathcal{R}$. The algorithm has global convergence at a nearly quadratic rate, even as the last (i.e., smallest) singular value approaches its minimum value.

4. Computing conformations

We extend the above algorithm in order to further reduce the rank of the matrix to $n-1$, instead of $N-1$, where n indexes henceforth the largest among the singular values that must be minimized and, eventually, reduced to zero. In practice, $n = 6$ in order for the given matrix to be perturbed to a valid Cayley–Menger distance matrix.

We shall use an iteration similar to the Rayleigh quotient iteration, for producing structured rank-reducing perturbations. The idea is essentially to inverse the procedure of the standard Rayleigh quotient. Suppose $N \times N$ matrix M is close enough to being embeddable in E_3. At each step of Newton's iteration for minimizing its nth singular value (and all smaller singular values), supposing the singular value is distinct, it suffices to compute $\Delta \in \mathcal{R}$ such that $u_n^T \Delta v_n = u_n^T (M-R) v_n$ and set the perturbation matrix $R \leftarrow R + \Delta \in \mathcal{R}$, where vectors u_n, v_n are associated to singular value $\sigma_n(M-R)$. This leads to a heuristic way of computing Δ since we have no formal manner to define a quantity like γ in Algorithm 3.3.

A better approach uses further necessary conditions to facilitate the optimization process. Let the Dirac function be δ_{ij} such that

$$\delta_{ij} = 1 \Leftrightarrow i = j \quad \text{and} \quad \delta_{ij} = 0 \text{ otherwise.}$$

To identify a new conformation, all singular values smaller than σ_{n-1} must be close to zero. The following algorithm uses the necessary conditions of this fact.

Algorithm 4.1. Input: an interval $N \times N$ distance matrix and starting values in each one of the intervals such that a valid distance matrix exists in their neighborhood; a perturbation space $\mathcal{R}$ corresponding to $N \times N$ matrices; an index n (typically 6); a tolerance $\tau > 0$.
Output: a valid $N \times N$ distance matrix satisfying the intervals of the input matrix such that the nth and all smaller singular values are smaller than τ.

Procedure: Let the approximate distance matrix of the starting values be D.

It suffices to compute $\Delta \in \mathcal{R}$ such that

$$u_n^T \Delta v_j = \delta_{nj} u_n^T (D-R) v_j, \quad n \leqslant j \leqslant N, \tag{1}$$

and set in the next step $R \leftarrow R + \Delta$, where singular vector u_n is associated to singular value $\sigma_n(D - R)$. Moreover, (1) should hold for each u_i, $n \leqslant i \leqslant N$, so the above relation becomes:

$$u_i^T \Delta v_j = \delta_{ij} u_i^T (D - R) v_j, \quad n \leqslant i, j \leqslant N. \tag{2}$$

Satisfying the stated conditions is equivalent to solving a linear system $E\xi = F$.

The algorithm iterates until the nth singular value drops below τ.

Once we have fixed the basis $\mathcal{R}$ of the perturbation space, defining Δ is equivalent to finding vector $\xi \in \mathbb{R}^p$, where p stands for the dimension of $\mathcal{R}$. The above conditions lead to the solution of a dense linear system $E\xi = F$, where

$$E = \begin{bmatrix} \vdots & & \vdots \\ v_j^T R_{i_1 j_1} u_i, & \ldots, & v_j^T R_{i_p j_p} u_i \\ \vdots & & \vdots \end{bmatrix}, \qquad F = \begin{bmatrix} \vdots \\ \delta_{ij} u_i^T (D - R) v_j \\ \vdots \end{bmatrix},$$

where each pair i, j, for $i = n, \ldots, N$, $j = i, \ldots, N$ defines the corresponding row in matrix E and vector F. For $N \times N$ Cayley–Menger matrices corresponding to $N - 1$ points, the number of perturbable entries is $p \leqslant (N-1)(N-2)/2$. The row dimension of E equals $\sum_{l=n}^{N} (N + 1 - l)$, for $N \geqslant n$, whereas its column dimension is p.

For $i \neq j$, condition (2) becomes $u_i^T \Delta v_j = 0$, and most relations will be of this type. They have the effect of keeping Δ small.

If E is square, then QR (or LU) decomposition is applied for computing ξ, by solving the square system $E\xi = F$. If the linear system $E\xi = F$ is overdetermined, we use the Moore–Penrose *pseudo-inverse*. This yields the solution $\xi = E^+ F$, optimal in a *least-squares* sense [29]. In other words, it minimizes the sum of squares of the values of the equations at the chosen solution.

Example 4.2. Let us consider the cyclohexane molecule, to be fully examined in Section 5. The goal is to minimize the 6th singular value, which implies that the first 5 singular values can be nonzero but all other singular values are close to zero, hence the rank becomes 5. Then F is a 3×1 vector, and E is the following 3×3 matrix:

$$E = \begin{bmatrix} v_6^T R_{25} u_6 & v_6^T R_{36} u_6 & v_6^T R_{47} u_6 \\ v_7^T R_{25} u_6 & v_7^T R_{36} u_6 & v_7^T R_{47} u_6 \\ v_7^T R_{25} u_7 & v_7^T R_{36} u_7 & v_7^T R_{47} u_7 \end{bmatrix}.$$

The algorithm actually takes an additional parameter as input, which bounds the number of iterations. It stores the candidate matrix of lowest rank at all times. So, the algorithm outputs this candidate, in case it cannot compute a distance matrix of rank 5 in the prescribed number of iterations.

Remark 4.3. If some Cayley–Menger matrix is sufficiently close (in terms of Newton's iteration) to a given approximate distance matrix D, then a Cayley–Menger matrix exists and is unique if and only if the solution of (1), stated as part of Algorithm 4.1, exists and is unique.

How is this remark justified? Formally, D must be in an attractive region of a valid Cayley–Menger distance matrix in terms of Newton's iteration. Since Newton's iteration converges, matrix Δ should exist if and only if a valid Cayley–Menger distance matrix exists. Matrix Δ represents the direction of approaching the embeddable matrix so having a unique direction is equivalent to a unique completion, in other words a unique Cayley–Menger distance matrix.

The number of columns of E depends on the number of perturbation matrices. Since we seek a unique solution, the number of these columns should be at most equal to the number of rows of E, namely $\sum_{l=6}^{N}(N - l + 1)$ for $n = 6$. The common case is the number of columns to be exactly equal to the number of its rows, analogous to the fact that a linear system with unique solution is typically square. It is possible, of course, to have uniqueness with an overdetermined system, though this is improbable, under exact computation. For a random overdetermined system, there is no solution that satisfies all equations, hence we strive for a solution vector that minimizes some criterion on the set of equations. We choose the least-squares criterion, which minimizes the sum of squares of the values of the given equations at the solution vector.

More formally, if N stands for the dimension of the distance matrix, each iteration must apply SVD on an $N \times N$ matrix. This has complexity in $O(N^3)$, and the hidden constant is roughly bounded above by 20. On the other hand, the linear system $E\xi = F$ has dimension that grows in the worst case asymptotically like N^2, because this is the growth of the number p of columns in E, as well as the growth of the number of rows in E. Hence, the complexity of solving for ξ is in $O(N^6)$ and clearly dominates the overall complexity per step. The hidden constant is small, usually less than 2. Of course, the number of columns and of rows in E can be smaller than the maximum possible, namely in $O(N)$, in which case the total complexity lies in $O(N^3)$. This is certainly true when a small number of matrix entries are perturbable. This happens soon after the overall algorithm (which applies Algorithm 4.1 but also enforces the given interval constraints) starts execution, because several entries have reached the extreme values allowed by the corresponding input intervals.

Since a unique completion of a given incomplete distance matrix depends upon the number of perturbation matrices, the unspecified entries can be freely chosen within a set of compatible distances. Based on this information, we estimate the dimension of the space of all possible conformations. This result is related to a more general, though looser, bound from Ref. [31].

Remark 4.4. Let p stand for the number of all unknown or unspecified entries in D. If the approximate (or incomplete) distance matrix D leads to an embeddable Cayley–Menger matrix in E_3, which is sufficiently close to D, then at least as many as

$$\max\left\{p - \sum_{l=6}^{N}(N-l+1),\ 0\right\}, \qquad N \geqslant 6,$$

of these entries can be freely perturbed; the other entries are then determined. Moreover, this number bounds from below the dimension (degrees of freedom) of the conformation manifold.

We have described the main heuristic for finding a valid conformation in the neighborhood of a given invalid distance matrix. But how to produce the latter from a set of interval constraints? We may compute several possible starting points by systematically sampling the input intervals. For molecules or molecular substructures with few degrees of freedom (less than 15), we are able to fully enumerate all realizable 3D conformations. Our method is not able to avoid certain obstacles, such as those related to the requirement of distinctness of the singular values, as they are perturbed, hence running the risk of getting trapped into local minima. Thus, different starting points can be used to circumvent this. In general, a regular sampling may be suboptimal, because some solutions have bigger attractive regions in the space of starting parameters, and there can even be fractal boundaries between attractive regions [32]. In fact, optimizing the sampling is a question of independent interest and part of our future work.

The section concludes with an ancillary technique implemented for refining the initial intervals, namely triangular inequalities, though more work is required in this important direction. The underlying principle is the following. Let d_{ij} express Euclidean distance between point masses i and j. The relation between the geminal distance d_{13} and the bond angle θ between the two consecutive bonds of atom 2 (with atoms 1 and 3, respectively) is given explicitly by the *law of cosines*:

$$d_{13}^2 = d_{12}^2 + d_{23}^2 - 2d_{12}d_{23}\cos(\theta), \tag{3}$$

where $|d_{12} - d_{23}|$ and $d_{12} + d_{23}$ are called the lower and upper triangle inequality limits respectively. The problem of generating and refining all triangle inequality bounds is equivalent to a shortest-path problem, which is of polynomial complexity [12,33].

5. Cycloalkanes

The algorithm and the observations in the previous sections make no assumption about the geometry of molecular chains. Here, we consider the case

of cycloalkanes since it is a problem of conformational calculations with many strong geometric constraints. This problem may arise as a subtask when studying proteins. For an illustration, we examine molecules with 6–8 backbone degrees of freedom, which are rotations about carbon-carbon bonds in the aromatic ring.

The *cyclohexane* has an infinite number of geometrically possible conformations due to its symmetry. Besides two rigid chair conformations, it can assume any conformation in a closed one-dimensional loop manifold; this manifold contains two embedded points corresponding to boat conformations [19].

The Cayley–Menger matrix is

$$D = \begin{bmatrix} 0 & b & c & u_1 & c & b & 1 \\ b & 0 & b & c & u_2 & c & 1 \\ c & b & 0 & b & c & u_3 & 1 \\ u_1 & c & b & 0 & b & c & 1 \\ c & u_2 & c & b & 0 & b & 1 \\ b & c & u_3 & c & b & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix},$$

where bond lengths $b = 1.54^2$ Å, bond angles remain fixed at 109.47° (thus, using the rule of cosines, $c = 2.51^2$ Å) and u_1, u_2, u_3 represent unknown values. Once these unspecified entries are determined, we can recover the geometry of the molecule up to global translations, rotations and chirality. Thus, the starting point is to identify the symmetric perturbation matrices of D. In this case, the subspace of matrices $\mathcal{R}$ has a basis comprised of R_{14}, R_{25}, R_{36}. Assume that we already know a conformation of cyclohexane in the one-dimensional manifold. This conformation corresponds to a unique set of u_1, u_2, u_3 values. The values of u_2 and u_3 depend on u_1. Thus, by removing one perturbation matrix (e.g., R_{14}) from $\mathcal{R}$, we can find a new *unique* Cayley–Menger matrix for each value of u_1, using our algorithm. This is equivalent to setting u_1 to a certain number of values. In the cyclohexane's case, we used a fixed step value of 0.05 Å, to explore the entire conformation manifold.

Besides computing all conformations on the manifold, our method was applied to enumerate all distinct types of conformations. By altering just dihedral angles it is impossible to pass between the boat and the chair geometries, whereas changing some angles between bonds can do it. We have applied a small perturbation (i.e., in the range of 10%), of interatomic distances in order to destroy the molecule's symmetry and produce a finite number of conformations, thus allowing us a "global" view of conformation space. Our method gives results as good as fully rigorous algebraic methods in that we obtain at most four solutions, as in Ref. [1,19,21]. The four isolated conformations correspond to two chair and two boat conformations, which correspond to the conformations most encountered in nature and hence minimizing energy. See figure 2 for these four conformations, as computed in Ref. [19]. The number of solutions

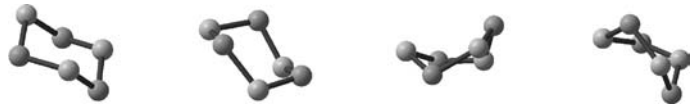

Figure 2. Chair and boat conformations of cyclohexane from Ref. [19].

upper bounds the number of connected components of the manifold, provided the input is generic (in practice, random).

Now let us refer to matrix E and vector F of the previous section: Since R_{14} is removed from $\mathcal{R}$, the dimensions of E, F are 3×2 and 3×1, respectively. Thus, the optimization involves the solution of an overdetermined system of linear equations. After three iterations, we have $\|E_k\xi - F_k\|_2 < 10^{-15}$, and this is zero within the precision of 16 decimal digits used by modern day hardware when employing double-precision floating-point arithmetic. Therefore the algorithm stops. This is an instance of a *certified answer* in the context of numerical computation.

For the *cycloheptane*, the Cayley–Menger matrix has seven unknown entries:

$$D = \begin{bmatrix} 0 & b & c & u_1 & u_2 & c & b & 1 \\ b & 0 & b & c & u_3 & u_4 & c & 1 \\ c & b & 0 & b & c & u_5 & u_6 & 1 \\ u_1 & c & b & 0 & b & c & u_7 & 1 \\ u_2 & u_3 & c & b & 0 & b & c & 1 \\ c & u_4 & u_5 & c & b & 0 & b & 1 \\ b & c & u_6 & u_7 & c & b & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix},$$

where matrices R_{14}, R_{15}, R_{25}, R_{26}, R_{36}, R_{37}, R_{47} form a basis of $\mathcal{R}$. Starting with a given conformation, we extract one perturbation matrix from $\mathcal{R}$ (e.g., R_{14}, corresponding to fixing u_1) and proceed with the singular value optimization. In this case, E_k and F_k define a 6×6 system. We were able to completely explore each one-dimensional manifold with a step of 0.05 Å, obtaining for example, more than 50 valid conformations with u_1 in the range $[8.586, 11.290]$ Å. We see that the matrix entry u_1 is constrained by $u_1 < 11.29$ Å. While exploring the one-dimensional manifold and after some iterations, if u_1 is increased beyond this bound, then condition (1), stated as part of Algorithm 4.1, cannot be satisfied. This is a case of incompatible constraints, and matrix E_k becomes singular implying there is no possible ξ.

After extracting one more perturbation matrix from $\mathcal{R}$, it is not possible to obtain any solutions, so the manifold dimension cannot be larger than one. Hence our method confirms what is known about the cycloheptane, i.e., that there are two one-dimensional conformation manifolds [16].

For the *cyclooctane*, the Cayley–Menger matrix has 12 unknowns:

$$D = \begin{bmatrix} 0 & b & c & u_1 & u_2 & u_3 & c & b & 1 \\ b & 0 & b & c & u_4 & u_5 & u_6 & c & 1 \\ c & b & 0 & b & c & u_7 & u_8 & u_9 & 1 \\ u_1 & c & b & 0 & b & c & u_{10} & u_{11} & 1 \\ u_2 & u_4 & c & b & 0 & b & c & u_{12} & 1 \\ u_3 & u_5 & u_7 & c & b & 0 & b & c & 1 \\ c & u_6 & u_8 & u_{10} & c & b & 0 & b & 1 \\ b & c & u_9 & u_{11} & u_{12} & c & b & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 \end{bmatrix}.$$

We are interested in the conformation manifold. A direct consequence of Remark 4.4 bounds the dimension from below by $12-10=2$. By extracting certain perturbation matrices from $\mathcal{R}$, just as for the cycloheptane above, we could bound the dimension from above by two. Hence its dimension is 2, confirming what is known [1,22].

6. Computational performance

This section reports on experiments with 6–12 degrees of freedom in cycloalkanes, in table 1. We also apply our implementation to molecular rings of up to 20 degrees of freedom, not necessarily cycloalkanes; see table 2.

Our software is based on MATLAB Version 5.3 or, alternatively, on SCILAB. The advantages of the latter package include its flexibility in code development, and the fact that it is freely distributed and simple to install; the two systems have almost identical syntax. We have used MATLAB to generate C code which, when compiled, gives faster timings than those reported in tables 1 and 2. If the described methods become the bottleneck in the entire process of

Table 1
Method's performance for computing one cycloalkane conformation.

Molecule	DOF	Initial 6th SV	Final 6th SV	Iterations	(sec)	(KFlop)
Cyclohexane	6	1.56e − 01	3.72e − 08	3	0.02	26
Cycloheptane	7	1.45e − 01	1.31e − 08	3	0.03	38
Cyclooctane	8	1.11e − 01	4.65e − 07	3	0.05	54
Cyclononane	9	1.24e − 01	3.31e − 08	3	0.08	80
Cyclodecane	10	1.64e − 01	6.86e − 07	3	0.12	119
Cycloendecane	11	2.10e − 01	1.43e − 06	3	0.18	183
Cyclododecane	12	1.32e − 01	7.41e − 08	5	0.27	281

conformational search with NMR data, it is of course possible to implement all routines in C or C++. In this case, the linear algebra routines can be provided by the LAPACK [34] software library, whose algorithms are used by MATLAB and SCILAB. Besides the distance geometry and matrix perturbations methods described above, we have implemented interval refinement by means of the triangle inequality.

In the tables are shown the initial and final values for the 6th singular value (SV), for molecules of a varying number of degrees of freedom (DOF). The input is created by perturbing a known conformation, then our code computes a nearby valid conformation. The initial perturbation is limited, so the 6th singular value is initially smaller than 1; this reveals the local nature of our optimization. The step size of our experiments is typically $\geqslant 0.05$ Å, though this can vary. It is interesting that three iterations suffice for all inputs. This implies that these are relatively small inputs and that the complexity of our method does not stem from the number of degrees of freedom, but rather from the existence of local minima. In other words, as long as the initial and final configurations are sufficiently close, the algorithm is pretty fast.

Both tables give results averaged over three runs, computed on a 500MHz PENTIUM-III architecture. The time complexity, expressed in terms of seconds and thousands of floating-point operations (KFlops), is meant as a rough indication of the algorithm's performance. In table 2, the last column shows the normalized ratio of running time over N^6, where N stands for the matrix dimension, therefore $N-1$ equals the number of degrees of freedom. Since this ratio fluctuates

Table 2
Method's performance for computing one conformation of a structure with a ring backbone.

DOF	Initial 6th SV	Final 6th SV	Iterations	(sec)	(KFlop)	Ratio
7	2.98e − 02	6.64e − 14	3	0.01	36	3.81
8	2.57e − 02	4.43e − 12	3	0.05	49	9.41
9	2.10e − 02	6.29e − 11	3	0.05	73	5.00
10	2.38e − 02	2.95e − 13	3	0.11	109	6.21
11	3.16e − 02	2.60e − 12	3	0.16	165	5.36
12	8.13e − 02	1.20e − 07	3	0.22	282	4.56
13	8.09e − 02	8.49e − 08	3	0.30	450	3.98
14	3.72e − 02	6.04e − 13	3	0.49	606	4.31
15	3.53e − 02	2.02e − 14	3	0.77	940	4.59
16	3.78e − 02	1.72e − 12	3	1.15	1404	4.76
17	3.83e − 02	1.70e − 13	3	1.54	2082	4.53
18	3.53e − 02	3.93e − 13	3	2.14	3039	4.55
19	3.80e − 02	4.59e − 14	3	2.91	4344	4.55
20	4.00e − 02	7.09e − 13	3	3.79	6136	4.42

Table 3
Point mass coordinates for the molecule in figure 1.

Atom	X	Y	Z
1.	2.9542	−1.4439	0.2325
2.	3.4683	−0.3184	−0.6155
3.	2.6029	0.9049	−0.5484
4.	2.0938	1.1841	0.8346
5.	1.1711	2.3655	0.8891
6.	−0.2595	2.0011	0.6237
7.	−0.5796	1.9223	−0.8397
8.	−2.0068	1.5446	−1.1051
9.	−2.7251	1.0769	0.1259
10.	−3.4377	−0.2276	−0.0748
11.	−2.8818	−1.3326	0.7738
12.	−1.9387	−2.2259	0.0239
13.	−0.6922	−1.5182	−0.4183
14.	0.4935	−1.8305	0.4457
15.	1.7375	−2.1024	−0.3472

very little as N increases, we confirm the theoretical prediction that the time complexity of our algorithm is in $O(N^6)$.

In figure 1 we present a molecule with 15 degrees of freedom, as computed by our software on MATLAB. Here all bond lengths are equal to 1.5 Å, as induced by table 3, which contains the Cartesian coordinates of all backbone atoms, regarded as point masses. The shown conformation satisfies all bond length constraints, as well as the bond angles constraints. This computation first finds the center of mass, from the given Cayley–Menger distance matrix. Then, it computes the position of each point with respect to this center. Hence, the result is unambiguous within rotations and translations. Our techniques cannot distinguish between mirror symmetries, so it would require a postprocessing step, where one decides on chirality by homology arguments based on the existing databases.

Behind MATLAB and SCILAB lies the software library LAPACK [34], of which we heavily use its tridiagonal eigensolver. In particular, the orthogonalization by the routine "DSTEIN" uses more than 90% of the time to compute the eigenpairs by tridiagonalization. Still, the main bottleneck of our algorithm is (dense) linear system solving, which could benefit from specialized software and from exploiting structure.

7. Future research

Our algorithm uses, in each iteration, a symmetric eigenvalue decomposition and linear system solving. This is a local optimization procedure, whereas the problem is essentially one of global optimization. To give a complete set

of conformations, improved sampling techniques must be applied. We have also experimented with interval analysis in order to exclude regions that contain no conformation. This will provide candidate regions which are small enough to be explored by our methods. Our preliminary tests applied the interval capabilities of MAPLE, and package ALIAS, implemented in C/C++ [35].

Bound smoothing is a standard technique in refining distance intervals obtained indirectly, namely by successively applying the triangle and tetrangle inequalities [36]. Although the former has been implemented, the latter has not been fully exploited yet. The optimal use of these inequalities is an important open question. We have observed a systematic overestimation of these intervals; one way to reduce them is through heuristics based on information from the Protein Data Bank. Using a local coordinate system may offer better constraint propagation.

The complete solution of some perturbed system (as for the cyclohexane) is part of future work, when it comes to larger molecules. This would yield a set of isolated solutions, whose cardinality would bound the number of connected components. For the cyclooctane, this cardinality constitutes an interesting open question today; it is believed [16] that there are two or three connected components, but no proof exists.

Last but not least, we expect some improvements in accuracy and efficiency if we use the notion of a cluster of singular values, for example, in the algorithm of Ref. [37].

Acknowledgments

The first author started this work at INRIA Sophia–Antipolis, France, where he was a full-time Tenured Researcher. His work at the National University of Athens has been partially supported by Project 70/4/6452 of the Research Council of the National University of Athens, and by the bilateral project "Calamata" of an Associated Team with INRIA Sophia-Antipolis, funded by INRIA.

The second author thanks Gordon Crippen and Timothy Havel for insightful discussions.

References

[1] N. Gō and H.A. Scheraga, Macromolecules 3 (1979) 178.
[2] Q. Dong and Z. Wu, J. Global Optimization 26 (2003) 321.
[3] P. Guntert, C. Mumenthaler and K. Wuthrich, J. Mol. Biol. 273 (1997) 283.
[4] S. LeGrand, A. Elofsson and D. Eisenberg, in: *Protein Folds: A Distance Based Approach*, eds. H. Bohr and S. Brunak (CRC Press, Boca Raton, FL, 1996) pp. 105–113.
[5] National Research Council, *Mathematical Challenges from Theoretical/Computational Chemistry* (National Academy Press, Washington, DC, 1995).
[6] C. Bailey-Kellogg, A. Widge, J.J. Kelley III M.J. Berardi, J.H. Bushweller and B.R. Donald, J. Comp. Biol. 7 (2000) 537.
[7] T.E. Malliavin, P. Barthe and M.A. Delsuc, Theor. Chem. Acc. 106 (2001) 91.

[8] L.M. Blumenthal, *Theory and Applications of Distance Geometry*, 2nd ed., Vol. 15 (Chelsea Publishing Company, Bronx, NY, 1970).
[9] T.F. Havel, in: *Encyclopedia of Computational Chemistry*, eds. P. von Ragué, P.R. Schreiner, N.L. Allinger, T. Clark, J. Gasteiger, P.A. Kollman and H.F. Schaefer III (Wiley, New York, 1998) pp. 723–742.
[10] J.M. Blaney, G.M. Crippen, A. Dearing and J.S. Dixon, DGEOM: program #590, Quantum Chemistry Program Exchange, 1990. http://qcpe.chem.indiana.edu.
[11] I.D. Kuntz, J.F. Thomason and C.M. Oshiro, in: *Methods in Enzymology*, Vol. 177, eds. N.J. Oppenheimer and T.L. James (Academic Press, Boston, 1993) pp. 159–204.
[12] A.W.M. Dress and T.F. Havel, Discr. Appl. Math. 19 (1988) 129.
[13] T. Malliavin and F. Dardel, in: *Sciences Fondamentales*, Vol. AF (Techniques de l'Ingénieur, Paris, 2002) pp. 6608 (1–18).
[14] P.M. Pardalos and X. Lin, in: *New Trends in Mathematical Programming*, eds. F. Giannessi, S. Komlósi and T. Rapcsák (Kluwer, Boston, 1997).
[15] J. Moré and Z. Wu, J. Global Optimization 15 (1999) 219.
[16] G.M. Crippen, J. Comp. Chem. 13 (1992) 351.
[17] Quantum Chemistry Program Exchange, http://qcpe.chem.indiana.edu.
[18] Conformational Searching and Analysis Software, http://www.netsci.org/resources/software/modeling/conf.
[19] I.Z. Emiris and B. Mourrain, Algorithmica, special issue on algorithms for computational biology 25 (1999) 372.
[20] T.F. Havel and I. Najfeld, in: *Computer Algebra in Science and Engineering*, eds. J. Fleischer, J. Grabmeier, W. Hehl and W. Küchlin (World Scientific Publishing, Singapore, 1995) pp. 243–259.
[21] D. Manocha, Y. Zhu and W. Wright, Comp. Appl. Biol. Sci. 11 (1995) 71.
[22] W.J. Wedemeyer and H.A. Scheraga, J. Comput. Chem. 20 (1999) 819.
[23] J.B. Saxe, in: *Proceedings of the 17th Allerton Conference on Communications, Control and Computing* (1979) pp. 480–489.
[24] R. Mathar, Linear Algebra Appl. 67 (1985) 1.
[25] M. Bakonyi and C.R. Johnson, SIAM J. Matrix Anal. Appl. 16 (1995) 646.
[26] J.W. Demmel, SIAM J. Matrix Anal. Appl. 13 (1992) 10.
[27] M.A. Wicks and R.A. Decarlo, SIAM J. Matrix Anal. Appl. 16 (1995) 123.
[28] D. Bini and V.Y. Pan, *Polynomial and Matrix Computations*, Vol. 1, *Fundamental Algorithms* (Birkhäuser, Boston, 1994).
[29] G.H. Golub and C.F. Van Loan, *Matrix Computations*, 3rd ed. (The Johns Hopkins University Press, Baltimore, Maryland, 1996).
[30] G.W. Stewart and J. Sun, *Matrix Perturbation Theory* (Academic Press, Boston, 1990).
[31] A.I. Barvinok, Discr. Comput. Geom. 13 (1995) 189.
[32] Y.Z. Xu, Q. Ouyang, J.G. Wu, J.A. Yorke, G.X. Xu, D.F. Xu, R.D. Soloway and J.Q. Ren, J. Comput. Chem. 21 (2000) 1101.
[33] J. Kuszewski, M. Nilges and A.T. Brünger, J. Biomolecular NMR 2 (1992) 33.
[34] E. Anderson, Z. Bai, C. Bischof, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney, S. Ostrouchov and D. Sorensen, *LAPACK Users' Guide*, 2nd ed. (SIAM, Philadelphia, 1995).
[35] J-P. Merlet, in: *Proc. Conf. Systèmes d'Equations Algébriques* (Toulouse, 2000).
[36] P.L. Easthope and T.F. Havel, Bull. Math. Biol 51 (1989) 173.
[37] I. Dhillon, G. Fann and B. Parlett, in: *Proc. 8th SIAM Conf. Parallel Proces. Scient. Comp.*, eds. M. Heath, V. Torczon, G. Astfalk, P.E. Bjørstad, A.H. Karp, C.H. Koebel, V. Kumar, R.F. Lucas, L.T. Watson and D.E. Womble (SIAM, Philadelphia, 1997) pp. 383–389.