Chapter 13
Using a distributed SDP approach to solve simulated protein molecular conformation problems

Xingyuan Fang and Kim-Chuan Toh

Abstract This paper presents various enhancements to the DISCO algorithm (originally introduced by Leung and Toh [18] for anchor-free graph realization in $\mathbb{R}^d$) for applications to conformation of protein molecules in $\mathbb{R}^3$. In our enhanced DISCO algorithm for simulated protein molecular conformation problems, we have incorporated distance information derived from chemistry knowledge such as bond lengths and angles to improve the robustness of the algorithm. We also designed heuristics to detect whether a subgroup is well localized and significantly improved the robustness of the stitching process. Tests are performed on molecules taken from the Protein Data Bank. Given only 20% of the inter-atomic distances less than 6Å that are corrupted by high level of noises (to simulate noisy distance restraints generated from nuclear magnetic resonance experiments), our improved algorithm is able to reliably and efficiently reconstruct the conformations of large molecules. For instance, given 20% of inter-atomic distances which are less than 6Å and are corrupted with 20% multiplicative noise, a 5600-atom conformation problem is solved in about 30 minutes with a root-mean-square-deviation (RMSD) of less than 1Å.

13.1 Introduction

Determining protein structure is an important problem in biology. Majority of the protein structures are obtained by X-ray crystallography. However, some proteins could not be crystallized, and the information we have from its solution state is some pairwise atomic distance bounds (known as Nuclear Overhauser effect (NOE) dis-

Xingyuan Fang
Department of Operations Research and Financial Engineering, Princeton University, New Jersey, e-mail: xingyuan@princeton.edu

Kim-Chuan Toh
Department of Mathematics, National University of Singapore, Singapore-MIT Alliance, Singapore, e-mail: mattohkc@nus.edu.sg

tance restraints) estimated from nuclear magnetic resonance (NMR) spectroscopy experiments. From late 1970s, distance geometry algorithms have become increasingly used in the interpretation of experimental data on macro molecular conformation. Generally, we use these algorithms to determine the Cartesian coordinates of the atoms of a molecule which are consistent with a predetermined set of intramolecular distance constraints. Those constraints could come from experimental data and known chemistry knowledge such as bond lengths and angles. In 1984, the structure of the first protein determined in its native solution state from NMR data was computed by the algorithm DISGEO [15].

The mathematical setting of the molecular conformation problem is as follows. We wish to determine the coordinates of n atoms $\boldsymbol{x}_i \in \mathbb{R}^3$, $i = 1, \ldots, n$. The information that is available consists of measured distances or distance bounds for some of the pairwise distances $||\boldsymbol{x}_i - \boldsymbol{x}_j||$ with $(i, j) \in \mathcal{N}$, where $\mathcal{N}$ is the set of index pairs (i, j) for which inter-atomic distance information is available. Note that in our convention, we only consider the pair (i, j) with $i < j$. The molecular conformation problem is an instance of the graph realization problem where the atoms are the vertices of the graph, and the pairs $(i, j) \in \mathcal{N}$ are the edges with the weight of edge (i, j) specified by the given distance data $\tilde{d}_{ij}$. In a p-dimensional graph realization problem, one is interested in determining points in $\mathbb{R}^p$ such that $||\boldsymbol{x}_i - \boldsymbol{x}_j|| \approx \tilde{d}_{ij}$ for all $(i, j) \in \mathcal{N}$.

The molecular conformation problem is closely related to the *sensor network localization problem*, but much more challenging. In the sensor network localization problem, there are two classes of objects: anchors (whose locations are known a priori) and sensors (whose locations are unknown and to be determined). In practice, the anchors and sensors are able to communicate with one another if they are not too far apart (say within a certain cut-off range), to obtain an estimate of the distance for each communicable pair. For the molecular conformation problem, there are no anchors. And more importantly, not all pairs of atoms within the cut-off range have distance estimates. In this paper, we will use the term "conformation", "localization", and "realization" interchangeably.

Recently, semidefinite programming (SDP) relaxation techniques have been applied to the sensor network localization problem [3]. While this approach was successful for moderate-size problems with the number of sensors in the order of a few hundreds, it was unable to solve problems with a large number of sensors, due to computational limitations in SDP algorithms for solving large scale problems. To localize larger networks, a distributed SDP-based algorithm for sensor network localization was proposed in [6]. The critical assumption required for the algorithm in [6] to work well is that there exist anchors distributed uniformly throughout the physical space. As a result, the algorithm cannot be applied to the molecular conformation problem, since the assumption of uniformly distributed anchors does not hold in the case of molecular conformation.

In [4], a distributed SDP-based algorithm (called DAFGL) was proposed and tested for the molecular conformation problem. The performance of the DAFGL algorithm is satisfactory when given 50% of pairwise distances less than 6Å apart that are corrupted by 5% multiplicative noise. More recently, Leung and Toh [18]

proposed a new distributed approach, the DISCO (for DIStributed COnformation) algorithm, with a view towards applications in molecular conformation. The DISCO algorithm was demonstrated to be efficient and robust in solving simulated molecular conformation problems when given only 30% of the pairwise distances less than 6Å which are corrupted by 20% multiplicative noise. However, DISCO frequently fails to give good results when given only 20% pairwise distances less than 6Å apart.

In this paper, we describe the DISCO algorithm and the enhancements we made to make the algorithm work for the protein molecular conformation problem under the highly sparse distance data regime of using only 20% pairwise distances less than 6Å that are corrupted by high level of noise. We should mention that in this work, real NOE distance data from NMR experiments are not considered, although that is our future goal. Instead, the distance data we considered are simulated from known protein conformations in order to validate the results obtained by DISCO by comparing the reconstructed conformations to the original ones. The protein molecules we used in our experiments are downloaded from the Protein Data Bank (PDB). In our experiments, we discard all the hydrogen atoms in the molecules. The input distances (all in the unit of Å$= 10^{-10}$ meter) are given as lying in intervals $[\underline{d}_{ij}, \overline{d}_{ij}]$ for $(i,j) \in \mathscr{N}$. Here $\mathscr{N}$ denotes the set of index pairs (i,j) for which inter-atomic distance bounds are available. In our simulated protein molecular conformation problem, we consider two types of distance bounds. The first type of bounds come from known chemistry information such as bond lengths and angles, and the interval is given in the form:

$$\underline{d}_{ij} = (1-\varepsilon)d_{ij}, \quad \overline{d}_{ij} = (1+\varepsilon)d_{ij} \quad \forall\,(i,j) \in \mathscr{N}_c \tag{13.1}$$

where $\varepsilon \in (0, 0.1)$ is a parameter which can be chosen appropriately to reflect our confidence on the given distance d_{ij} derived from the chemistry information of the molecule. The index set $\mathscr{N}_c$ is used to denote the index pairs (i,j) for which distance bounds based on chemistry information are given. The second type of bounds are designed to simulate NOE restraints, and they are given as follows:

$$\underline{d}_{ij} = \max\left(2.0, (1-\sigma|\underline{z}_{ij}|)d_{ij}\right), \quad \overline{d}_{ij} = (1+\sigma|\overline{z}_{ij}|)d_{ij} \quad (i,j) \in \mathscr{N}_s, \tag{13.2}$$

where d_{ij} is the true distance between atoms i and j, $\underline{z}_{ij}, \overline{z}_{ij}$ are independent random variables such that $|\underline{z}_{ij}|, |\overline{z}_{ij}|$ have unit mean value. The parameter σ is the noise factor which we typically set to 20%. The number 2.0 in the expression for $\underline{d}_{ij}$ is a conservative lower bound on the shortest distance between two non-bonded (non-hydrogen) atoms. The index set $\mathscr{N}_s$ is used to denote the index pairs (i,j) for which the simulated distance bounds are given. Note that the overall index set $\mathscr{N}$ is the disjoint union of $\mathscr{N}_c$ and $\mathscr{N}_s$.

The main enhancements we made are as follows. First, we have included some distances derived from basic chemistry information such as bond lengths and bond angles into the distance data used in the conformation. We relied on the papers [17] and [1] for those basic chemistry information. To automatically derive those chemistry information for a protein molecule given its amino acids sequence, we find it

convenient to design a structure array data structure to code the information pertaining to each atom in the molecule. Second, we also designed effective heuristics to detect whether a subgroup of atoms is well localized, as well as substantially improved the robustness of the stitching process in the DISCO algorithm. We demonstrate that our enhanced DISCO algorithm is efficient and robust, and it works well under the highly sparse distance data regime we have targeted. Compared to the original DISCO algorithm, the RMSD errors of some reconstructed molecules have been significantly improved when given only 20% of pairwise distances (corrupted by 20% multiplicative noise) less than 6Å. For example, the average RMSD of the reconstructed conformations for the molecule `1RGS` is reduced from 4.5Å to 1.3Å over 10 random instances. We should mention that the 20% distances less than 6Å used to simulate NOE restraints is randomly selected from the set of all distances less than 6Å excluding those derived from chemistry knowledge.

As non-random test problems for evaluating sensor network localization algorithms are of interest to the sensor network community, we plan to make the distance data we have generated in this paper publicly available at the following web-site:

`http://www.math.nus.edu.sg/~mattohkc/disco.html`

The paper is organized as follows. Section 2 describes some existing molecular conformation algorithms; Section 3 details the mathematical models for molecular conformation based on SDP; Section 4 explains the design of DISCO and describes the enhancements we made; Section 5 describes the chemistry information we have incorporated into our simulated protein molecular conformation problems; Section 6 contains the experimental setup and numerical results; Section 7 gives the conclusion.

In this paper, we adopt the following notational conventions. Lower case letters, such as n, are used to represent scalars. Lower case letters in bold font, such as $\mathbf{s}$, are used to represent vectors. Upper case letters, such as X, are used to represent matrices. Upper case letters in calligraphic font, such as $\mathscr{D}$, are used to represent sets. Cell arrays will be prefixed by a letter "c", such as *cAest*. Cell arrays will be indexed by curly braces $\{\}$.

13.2 Related Work

Due to its great importance, there are quite a number of existing algorithms to tackle the molecular conformation problem. We discuss selected works in this section. Here we do not attempt to give a detailed survey of the existing work on the distance geometry approach for solving the molecular conformation problem. Our intention is only to highlight the most relevant work for which the experimental settings used bear the closest similarity with ours. For the existing algorithms which we mention in this section, we pay attention to each algorithm by the following aspects: the mathematics, its input data, and the results it is able to provide. In particular, we make a note of the largest molecule which each algorithm was able to solve and its

error (mostly measured by RMSD) in the tests done by the authors. This information is summarized in Table 13.1.

Before we begin, we note that from the theory of distance geometry [25, 26, 27], there is a natural correspondence between inner product matrices and Euclidean distance matrices. Thus it is quite common to solve the molecular conformation problem by working with an inner product matrix. If we denote the atom coordinates by column vectors $\boldsymbol{x}_i$, and let $X = [\boldsymbol{x}_1 \dots \boldsymbol{x}_n]$, then the inner product matrix is given by $Y = X^T X$. If we have a computed $\tilde{Y}$, then we can recover the approximate coordinates $\tilde{X}$ by taking the best rank-3 approximation based on the eigenvalue decomposition of $\tilde{Y}$.

The earliest distance geometry based algorithm for molecular conformation is the EMBED algorithm [14] developed by Havel, Kuntz and Crippen in 1983. The input data of EMBED consists of lower and upper bounds on some of the pairwise distances. EMBED uses the triangle and tetrangle inequalities to compute distance bounds for all pairs of points, followed by choosing random numbers within the bounds to form an estimated distance matrix $\tilde{D}$ (one should note that the triangle and tetrangle bounds are generally too weak to provide good estimates on the distances). It checks whether $\tilde{D}$ is close to a valid rank-3 Euclidean distance matrix by considering the three largest eigenvalues (in magnitude) of $\tilde{Y}$, the inner product matrix corresponding to $\tilde{D}$. In the fortunate case where the three eigenvalues are positive and are much larger than the rest, this would indicate that the estimated distance matrix $\tilde{D}$ is close to a true distance matrix, and the coordinates obtained from the inner product matrix are likely to be acceptable. In the unfortunate case where at least one of the three eigenvalues is negative, the estimated distance matrix $\tilde{D}$ is far from a valid distance matrix. In this case, EMBED repeats the step of choosing an estimated distance matrix until it obtains one that is close to a valid distance matrix. As a postprocessing step, the coordinates are improved by applying local optimization methods.

Subsequently, Havel and Wüthrich developed the DISGEO package [15] to improve the performance of EMBED. As the EMBED algorithm is unable to compute a conformation of the whole protein structure, due to the high dimensionality of the problem, DISGEO uses two passes of EMBED to overcome the problem. In the first pass, coordinates are computed for a subset of atoms subject to constraints inherited from the whole structure. In this step, EMBED uses data not only from experiments, but also from chemistry knowledge, including bond lengths, bond angles, hybridization theory, and so on. This step forms a ”skeleton” for the structure. The second pass of EMBED then computes coordinates for the remaining atoms, building upon the skeleton computed in the first pass. The authors tested the performance of DISGEO on the `BPTI` protein, which has 454 atoms. The input consists of distance (3290) and chirality (450) constraints needed to fix the covalent structure, and bounds (508) for distances between hydrogen atoms in different amino acid residues that are less than 4Å apart, to simulate the distance constraints available from a nuclear Overhauser effect spectroscopy experiment. Using a pseudostructure

Algorithm(s)	Largest molecule (No. of atoms)	Inputs	Output
EMBED (83), DISGEO (84), DG-II (91), APA (99)	454	All distance and chirality constraints needed to fix the covalent structure are given exactly. Some or all of the distances between hydrogen atoms less than 4Å apart and in different amino acid residues given as bounds.	RMSD 2.08Å
DGSOL (99)	200	All distances between atoms in successive residues given as lying in $[0.84d_{ij}, 1.16d_{ij}]$.	RMSD 0.7Å
GNOMAD (01)	1870	All distances between atoms that are covalently bonded given exactly; all distances between atoms that share covalent bonds with the same atom given exactly; additional distances given exactly, so that 30% of the distances less than 6Å are given; physically inviolable minimum separation distance constraints given as lower bounds.	RMSD 2–3Å(*)
MDS (02)	700	All distances less than 7Å were given as lying in $[d_{ij} - 0.01, d_{ij} + 0.01]$.	violations < 0.01Å
StrainMin (06)	5147	All distances less than 6Å are given exactly, a representative lower bound of 2.5Å is given for other pairs of atoms.	violations < 0.1Å
ABBIE (95)	1849	All distances between atoms in the same amino acid given exactly. All distances between pairs of hydrogen atoms less than 3.5Å apart, given exactly.	Exact
Geometric build-up (07)	4200	All distances between atoms less than 5Å apart given exactly.	Exact
DAFGL (07)	5681	70% of the distances less than 6Å were given as lying in $[\underline{d}_{ij}, \overline{d}_{ij}]$, where $\underline{d}_{ij} = (1 - 0.05\|\underline{z}_{ij}\|)_+ d_{ij}, \overline{d}_{ij} = (1 + 0.05\|\overline{z}_{ij}\|)d_{ij}$, and $\underline{z}_{ij}, \overline{z}_{ij}$ are standard normal random variables with zero mean and unit variance.	RMSD 3.16Å

Table 13.1 A summary of protein conformation algorithms. (*) The RMSD of 1.07Å reported by GNOMAD may be incorrect, and the true value should be about 2–3Å since the number reported in Figure 11 of [29] does not agree with that appears in Figure 8.

representation, they were able to solve for 666 geometric points[1]. Havel's DG-II package [13], published in 1991, improves upon DISGEO by producing from the same input as DISGEO five structures having an average RMSD of 1.76Å from the crystal structure.

The work in this paper is a continuation of the DISCO algorithm developed in 2009 [18]. DISCO differs from the previous methods in that it applies SDP relaxation methods to obtain the inner product matrix. In order to solve larger problems, it employs a divide-and-conquer approach for which each basis group is solved using SDP, and the overlapping groups are used to align the local solutions to form a global solution. Tests were performed on 14 molecules with number of atoms ranging from 400 to 5600. The input data consists of 30% of the distances below 6Å, given as lying in intervals $[\underline{d}_{ij}, \overline{d}_{ij}]$ which are generated from the true distances d_{ij} with 20% multiplicative noises added. Given such input, DISCO is able to produce a conformation for most molecules with an RMSD of 2–3Å.

Distributed algorithms (based on successive decomposition) similar to those in [4] were proposed for fast manifold learning in [30, 31]. In addition, those papers also considered recursive decomposition. The manifold learning problem is to seek a low-dimensional embedding of a manifold in a high dimensional Euclidean space by modeling it as a graph-realization problem. The resulting problem has similar characteristics as the molecular conformation problem (which is an anchor-free graph realization problem) we consider in this paper, but there are some important differences which we should highlight. For the manifold learning problem, exact pairwise distances between any pairs of vertices (atoms in our case) are available, but for the molecular conformation problem, only a very sparse subset of pairwise distances are assumed to be given and are only known within a given range. Such a difference implies that for the former problem, any local patch will have a "unique" embedding (up to rigid motion and certain approximation errors) computable via an eigenvalue decomposition, and the strategy to decompose the graph into sub-graphs is fairly straightforward. In contrast, for the latter problem, given the sparsity of the graph and the noise in the distances data, the embedding problem itself requires a new method, not to mention that sophisticated decomposition strategies also need to be devised.

The GNOMAD algorithm [29] by Williams, Dugan and Altman is a gradient descent based algorithm which attempts to satisfy the input distance constraints as well as minimum separation distance (MSD) constraints. Their algorithm applies to the situation when we are given sparse but exact distances. The knowledge of MSD constraints is useful in limiting the search space, but if they are not applied intelligently, they may trap the algorithm at an unsatisfactory local minimum. Since it is difficult to optimize all the atom positions simultaneously because of the high dimensionality of the problem, GNOMAD updates the positions of the atoms one atom at a time. The authors tested GNOMAD on the protein molecule 1TIM, which

[1] In NMR experiments, certain protons may not be stereospecifically assigned. For such pairs of protons, the upper bounds are modified via the creation of "pseudoatoms", as is the standard practice in NOE experiments. given 3798 distance and 450 chirality constraints, with three computed structures having an average RMSD of 2.08Å from the known crystal structure.

has 1870 atoms. Given all the covalent distances and distances between atoms that share covalent bonds to the same atom, as well as 30% of pairwise distances less than 6Å, they were able to compute a conformation with an RMSD of 2–3Å[2]. The experimental setting we considered in this paper is similar to that given in [29], but in the highly challenging regime of having very sparse and noisy distances. Also the algorithm we designed is completely different from GNOMAD. Our algorithm's input includes all the covalent distances and distances between atoms that share covalent bonds with the same atom. We also add 20% of pairwise distances less than 6Å (which are corrupted by high level of noise) to simulate distances derived from an NMR experiment.

In this brief discussion on related work, we have not touched on approaches based on global optimization and discrete optimization methods. For example, in [19], the authors developed a branch-and-prune method. We refer the reader to [19], [18] and [29] for more detailed review of various distance geometry based methods including those in [16], [9], [11], [21], proposed for the molecular conformation problem.

13.3 Optimization models for the molecular conformation problem

We begin this section with the optimization models we consider for the molecular conformation problem. Then we introduce the semidefinite programming (SDP) relaxations for these models, and describe the gradient descent method we adopt for improving the positions of the atoms by using the SDP solution as the starting point. Finally, we present the alignment problem for stitching (sometimes we will also use the words "merging" and "combining") two groups of atoms together using the overlapping atoms in the groups.

13.3.1 Semidefinite programming models

In the "measured distances" model, we have measured distances $\tilde{d}_{ij}$ for certain pairs of atoms, i.e.,

$$\tilde{d}_{ij} \approx ||\boldsymbol{x}_i - \boldsymbol{x}_j|| \qquad (i,j) \in \mathscr{N}. \tag{13.3}$$

In this model, the unknown positions $\{\boldsymbol{x}_i\}_{i=1}^n$ represents the best fit to the measured distances, obtained by solving the following nonconvex minimization problem:

$$\min \left\{ \sum_{(i,j)\in\mathscr{N}} \left| ||\boldsymbol{x}_i - \boldsymbol{x}_j||^2 - (\tilde{d}_{ij})^2 \right| \right\}. \tag{13.4}$$

[2] The RMSD of 1.07Å reported in Figure 11 in [29] is inconsistent with that appearing in Figure 8. It seems that the correct RMSD should be about 2–3 Å.

For convenience, we denote the measured inter-atomic distance matrix by $\tilde{D}$. In the "distance bounds" model, we have lower and upper bounds on the distances between certain pairs of atoms, i.e.,

$$\underline{d}_{ij} \leq ||\boldsymbol{x}_i - \boldsymbol{x}_j|| \leq \overline{d}_{ij} \qquad (i,j) \in \mathscr{N}. \tag{13.5}$$

In this model, the unknown positions $\{\boldsymbol{x}_i\}_{i=1}^n$ represent any feasible solution $X = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]$ satisfying the bound constraints. We denote the lower and upper bound distance matrices by $\underline{D}$, $\overline{D}$. Note that for the "distance bounds" model, we can naturally convert it to the "measured distance" model by taking $\tilde{d}_{ij} = (\underline{d}_{ij} + \overline{d}_{ij})/2$.

In order to proceed to the SDP relaxation of the problem, we consider the following matrix

$$Y := X^T X, \quad \text{where } X = [\boldsymbol{x}_1 \ldots \boldsymbol{x}_n]. \tag{13.6}$$

Let $\{\boldsymbol{e}_i\}_{i=1}^n$ be set of standard unit vectors in $\mathbb{R}^n$. By denoting $\boldsymbol{e}_{ij} = \boldsymbol{e}_i - \boldsymbol{e}_j$, we note that

$$||\boldsymbol{x}_i - \boldsymbol{x}_j||^2 = \boldsymbol{e}_{ij}^T Y \boldsymbol{e}_{ij}.$$

We can therefore conveniently express the constraints in (13.3) and (13.5) respectively as

$$(\tilde{d}_{ij})^2 \approx \boldsymbol{e}_{ij}^T Y \boldsymbol{e}_{ij} \qquad (i,j) \in \mathscr{N}$$

$$(\underline{d}_{ij})^2 \leq \boldsymbol{e}_{ij}^T Y \boldsymbol{e}_{ij} \leq (\overline{d}_{ij})^2 \quad (i,j) \in \mathscr{N}.$$

The SDP relaxation then consists in relaxing the constraint "$Y = X^T X$" in (13.6) into the constraint "$Y \succeq 0$", where the notation means that the $n \times n$ symmetric matrix Y is positive semidefinite.

The SDP relaxation of the measured distances model (13.4) is given by

$$\min \left\{ \sum_{(i,j) \in \mathscr{N}} \left| \boldsymbol{e}_{ij}^T Y \boldsymbol{e}_{ij} - (\tilde{d}_{ij})^2 \right| \mid Y \succeq 0 \right\}. \tag{13.7}$$

Similarly we can express the SDP relaxation of the distance bounds model (13.5) as finding an element in the following set:

$$\left\{ Y \mid (\underline{d}_{ij})^2 \leq \boldsymbol{e}_{ij}^T Y \boldsymbol{e}_{ij} \leq (\overline{d}_{ij})^2 \quad \forall\, (i,j) \in \mathscr{N},\, Y \succeq 0 \right\}. \tag{13.8}$$

Once we have obtained a matrix Y by solving either (13.7) or (13.8), we can estimate the atom positions $X = [\boldsymbol{x}_1 \ldots \boldsymbol{x}_n]$ by setting X to be the best rank-3 approximation of Y.

In [4], it has been shown that if the distance data is exact and the conformation problem is uniquely localizable, then the SDP relaxation (13.7) is able to produce the exact atom coordinates up to a rigid motion. We refer the reader to [4] for the definition of "uniquely localizable". Intuitively, it means that there is only one configuration in $\mathbb{R}^3$ (up to a rigid motion) that satisfies all the distance constraints. The

result (which is a variant of the result established by So and Ye [22] for graph realization with anchors) gives a strong indication that the SDP relaxation technique is a powerful relaxation. We can therefore hope that applying SDP relaxation to problems with sparse and noisy distance data will be effective.

We now discuss what typically would happen when the distance data is sparse and/or noisy, so that there is no unique realization. In such a situation, it is not possible to recover the true coordinates. Furthermore, the solution Y of the SDP (13.7) or (13.8) will generally have rank greater than 3, as we shall explain next. Suppose we have points in the plane, and certain pairs of points are constrained so that the distance between them is fixed. If the distances are perturbed slightly, then some of the points may be forced out of the plane in order to satisfy the distance constraints. Therefore, for noisy distance data, Y will tend to have a rank higher than 3. Another reason for Y to have a higher rank is that, if there are multiple optimal solutions in an SDP problem, interior-point methods used by many SDP solvers would converge to a solution with maximal rank [12].

This situation leads to potential issues. If Y has a rank higher than 3, then the best rank-3 approximation of Y is unlikely to give accurate positions for the atoms. To ameliorate this difficulty, we add the following regularization term into the objective function, i.e.,

$$-\gamma\langle I, Y\rangle \tag{13.9}$$

where γ is a positive regularization parameter. The motivation for introducing this term is to spread the atoms further apart so as to induce them to lie in a lower-dimensional space. Indeed, under the condition that $0 = \sum_{i=1}^{n} \boldsymbol{x}_i = X\boldsymbol{e}$, where $\boldsymbol{e} \in \mathbb{R}^n$ is the vector of all ones, we can easily show that $\sum_{i=1}^{n}\sum_{j=1}^{n} ||\boldsymbol{x}_i - \boldsymbol{x}_j||^2 = \langle I, Y\rangle/(2n)$ by using the definition that $Y = X^T X$. We refer interested readers to [3] for details on the derivation of the regularization term. Thus, for the measured distances model, the regularized SDP model for (13.7) becomes

$$\min\left\{ \sum_{(i,j)\in\mathcal{N}} |\boldsymbol{e}_{ij}^T Y \boldsymbol{e}_{ij} - (\tilde{d}_{ij})^2| - \gamma\langle I, Y\rangle \;\mid\; \boldsymbol{e}^T Y \boldsymbol{e} = 0,\, Y \succeq 0 \right\} \tag{13.10}$$

and the one related to the distance bounds model (13.8) becomes

$$\min\left\{ -\langle I, Y\rangle \mid (\underline{d}_{ij})^2 \le \boldsymbol{e}_{ij}^T Y \boldsymbol{e}_{ij} \le (\overline{d}_{ij})^2 \;\forall\, (i,j) \in \mathcal{N},\, \boldsymbol{e}^T Y \boldsymbol{e} = 0,\, Y \succeq 0 \right\}. \tag{13.11}$$

Note that we have added the constraint "$\boldsymbol{e}^T Y \boldsymbol{e} = 0$" to reflect the requirement for which the center of mass $X\boldsymbol{e}$ is fixed to the origin.

We should emphasize that as observed in [3], the inclusion of the regularization term in the SDP model can greatly improve the quality of the conformation solution X generated by the SDP model together with a subsequent gradient descent refinement. However, the choice of the regularization parameter γ is crucial. In our implementation of the enhanced DISCO algorithm, we adaptively adjust the value of γ based on the following separation ratio:

$$\texttt{sep\_ratio} = \frac{1}{|\mathscr{N}|}\sum_{(i,j)\in\mathscr{N}} s_{ij}, \quad \text{where } s_{ij} = \frac{||\boldsymbol{x}_i - \boldsymbol{x}_j||}{\tilde{d}_{ij}}. \tag{13.12}$$

Observe that, if the solution X generated by the SDP model is the correct conformation and $\tilde{d}_{ij}$ is the exact distance between atoms i and j for all $(i,j) \in \mathscr{N}$, then we must have `sep_ratio`=1. Of course, for noisy distances, `sep_ratio` would not be exactly 1, but we expect the value to be close to 1 if the computed conformation is not too different from the true one. In fact, from our extensive numerical experiments, we find that the value of `sep_ratio` should generally lie in the interval $[0.85, 1.1]$ in order for the conformation solution (generated from the SDP model) to have a reasonably good quality (measured in terms of the RMSD with respect to the true conformation). Based on such a criterion, we increase γ if `sep_ratio` is too small, and decrease it if `sep_ratio` is too big. If after 5 trials, we cannot get `sep_ratio` to lie in the required interval, we declare that the underlying conformation problem $\{\tilde{d}_{ij} | (i,j) \in \mathscr{N}\}$ is not localizable, and flag it as "bad".

Note that, in a distributed algorithm like DISCO, it is very important for us to design a reasonably good heuristic to detect whether a configuration is bad. This is because a bad configuration should not be stitched to a good (well localized) configuration for otherwise it will destroy the good one after the two configurations are aligned and stitched. A bad configuration should be separately handled after the majority of the atoms in the molecule have been localized.

13.3.2 Coordinate refinement via gradient descent

If we are given measured pairwise distances $\tilde{d}_{ij}$, then the atoms' coordinates can also be computed as the minimizer of the following nonconvex minimization problem:

$$\min f(X) := \sum_{(i,j)\in\mathscr{N}} \left(||\boldsymbol{x}_i - \boldsymbol{x}_j|| - \tilde{d}_{ij}\right)^2 - \beta \sum_{i=1}^{n}\sum_{j=1}^{n} ||\boldsymbol{x}_i - \boldsymbol{x}_j||^2. \tag{13.13}$$

Note that the above objective function is different from that of (13.4) because we want it to be differentiable. Observe that as in our SDP model, we have added a regularization term (with positive parameter β) in the objective function in (13.13). Similarly, if we are given bounds $\underline{d}_{ij}, \overline{d}_{ij}$ for pairwise distances, then the configuration can be computed as the solution of the following problem:

$$\min \sum_{(i,j)\in\mathscr{N}} \left(||\boldsymbol{x}_i - \boldsymbol{x}_j|| - \underline{d}_{ij}\right)_-^2 + \left(||\boldsymbol{x}_i - \boldsymbol{x}_j|| - \overline{d}_{ij}\right)_+^2 - \beta \sum_{i=1}^{n}\sum_{j=1}^{n} ||\boldsymbol{x}_i - \boldsymbol{x}_j||^2. \tag{13.14}$$

We can solve (13.13) or (13.14) by applying local optimization methods. For simplicity and computational efficiency, we choose to use a gradient descent method with backtracking line search. The algorithmic framework of this method is rather straightforward, so we shall omit the details. It is a simple exercise in calculus to

find the gradient of f with respect to the coordinate vector $\boldsymbol{x}_i$. However, we should emphasize that, to compute the gradient efficiently, the computation must be designed appropriately. In our implementation, we find it convenient to construct the $n \times |\mathcal{N}|$ sparse node-arc incidence matrix E for which the (i,j)-th column contains only two non-zero entries with 1 and -1 at row i and j, respectively. With E, we have that $XE = [\boldsymbol{x}_i - \boldsymbol{x}_j \mid (i,j) \in \mathcal{N}]$. Thus to calculate $[||\boldsymbol{x}_i - \boldsymbol{x}_j|| \mid (i,j) \in \mathcal{N}]$, one just needs to take the norm of the columns of XE.

The problems (13.13) and (13.14) are highly nonconvex problems with many local minimizers. Thus, if the initial iterate X^0 is not close to a good local minimizer, then it is extremely unlikely that the resulting X obtained from a local optimization method will be a good solution. In our case, however, when we set X^0 to be the conformation produced from solving the SDP relaxation, local optimization methods are often able to produce an X which improves upon the solution X^0 obtained from the SDP relaxation.

We should note that, while adding the regularization term in (13.13) and (13.14) (with a suitably chosen parameter β) would generally lead to a more accurate conformation solution X, it can sometimes (though rarely happen) give a much worse solution if the parameter β is chosen to be too large. In our implementation of the enhanced DISCO algorithm, we guard against such a bad case by examining the following expansion ratio:

$$\texttt{expansion\_ratio} = \max\{||\boldsymbol{x}_i||/||\boldsymbol{x}_i^0|| \mid i = 1,\ldots,n\},$$

where the columns of X^0 and X are translated to have their respective center of mass at 0. If the expansion ratio is larger than 3, we declare that the solution X is worse than X^0, and we discard the solution X from the refinement process. In our more sophisticated implementation, we would perform another pass of the gradient descent refinement by deleting the potentially "bad" atoms and their corresponding distances from the optimization model while also reducing the parameter β. By doing so, one hopes that the coordinates of the remaining atoms can be improved.

13.3.3 Alignment of configurations

A molecular configuration has translational, rotational, and reflective freedom. Nevertheless, we need to be able to compare two configurations to determine how similar they are. In order to do so, it is necessary to align them in a common coordinate system. Given $\{\boldsymbol{x}_i\}_{i=1}^n$ and $\{\boldsymbol{y}_i\}_{i=1}^n$, we can define the "best" alignment as the affine transformation T that minimizes the following problem:

$$\min_{\boldsymbol{c}\in\mathbb{R}^3, Q\in\mathbb{R}^{3\times 3}} \left\{ \sum_{i=1}^n ||T(\boldsymbol{x}_i) - \boldsymbol{y}_i||^2 \mid T(\boldsymbol{x}_i) = \boldsymbol{c} + Q(\boldsymbol{x}_i)\ \forall\, i,\ Q \text{ is orthogonal} \right\} \quad (13.15)$$

The functional form of T restricts it to be a combination of translation, rotation and reflection. In the special case when the columns of $X = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n]$ and $Y = [\boldsymbol{y}_1, \ldots, \boldsymbol{y}_n]$ are centered at the origin, (13.15) reduces to the following orthogonal procrustes problem

$$\min_{Q \in \mathbb{R}^{3\times 3}} \{||QX - Y||_F \mid Q \text{ is orthogonal}\}.$$

It is well known that the optimal Q can be computed from the singular value decomposition of XY^T.

13.4 The basic ideas of the DISCO algorithm

Here we present the basic ideas of the DISCO algorithm (for DIStributed COnformation). For the detail algorithm, we refer the reader to [18].

Before having DISCO, we could solve the molecular conformation problem by using the SDP relaxation technique and gradient descent refinement if the molecule was not too big (say the number of atoms was below 500). The aim of DISCO is to solve large-scale problems.

A natural idea to solve a large conformation problem is to employ a divide-and-conquer approach, which applies the following general framework. If the number of atoms is not too large, then solve the conformation problem via SDP, and apply gradient descent refinement to improve the coordinates; otherwise break the atoms into two subgroups, solve each subgroup recursively, and align and stitch them back together subsequently, again postprocessing the coordinates by applying gradient descent refinement after each stitching step.

We find that the use of the divide-and-conquer approach not only allow us to solve larger problem. It also allows us to design a more robust algorithm. For example, the repeated use of gradient descent refinement after each stitching step has certainly helped DISCO to become much more successful in producing accurate conformations. As a by-product of the divide-and-conquer process, we find that certain information collected during the SDP localization and stitching steps in DISCO can be used to help us to design a more robust algorithm.

Next we describe how DISCO (a) recursively divides a molecule into two subgroups of atoms; and (b) how to stitch (as well as how to decide whether to stitch) two processed subgroups together. The idea DISCO uses to tackle the first issue is to minimize the number of edges between the two subgroups. The reason is that when a group is split into two disjoint subgroups, the edges (distances) between the two subgroups are lost. In other words, some distance information is lost. Thus DISCO tries to minimize the information lost. For the second issue, DISCO's strategy is for the two subgroups to have overlapping atoms. If the overlapping atoms are accurately localized in the two subgroups, then they can be aligned for the purpose of stitching the two subgroups together. If not, it would not be a good idea to align and stitch them since a bad subgroup can destroy the good one when they are stitched

together. Therefore, DISCO designed a heuristic criterion for determining whether the overlapping atoms are accurately localized. In order to have a reliable alignment based on the overlapping atoms in the two subgroups, one of the most obvious criterion is that the RMSD of the coordinates of the overlapping atoms contained in the two subgroups must not be large, say less than 3Å; otherwise, it gives a strong indication that at least one of the two subgroups is not well localized. We have observed that the RMSD of the overlapping atoms used in the stitching of two subgroups provides valuable information on the quality of the larger stitched configuration. Thus, for each stitched configuration, we assign a quality index (`q_index`) as follows:

$$\texttt{q\_index} = \max\left\{ \begin{array}{l} \text{RMSD of} \\ \text{overlapping} \\ \text{atoms} \end{array}, \ \frac{1}{|\mathcal{N}'|} \sum_{(i,j)\in\mathcal{N}'} \max\left(s_{ij}^{5}, s_{ij}^{-5}\right) \right\} \tag{13.16}$$

where s_{ij} is defined as in (13.12), $\mathcal{N}'$ is the subset of $\mathcal{N}$ involved in the stitched configuration, but it excludes the large outlier values of more than 100 in $\max(s_{ij}^{5}, s_{ij}^{-5})$. The power of 5 in (13.16) is chosen based on empirical experience. At the basis level, the quality index is assigned based on the SDP solution Y obtained from (13.10) or (13.11) as follows:

$$\texttt{q\_index} = \left(\frac{1}{|\mathcal{N}'|} \sum_{(i,j)\in\mathcal{N}'} \max\left(s_{ij}^{5}, s_{ij}^{-5}\right) \right)^{1/2}$$

where s_{ij} is calculated based on the best rank-3 approximation $X = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_{n'}]$ of the SDP solution Y. If a basis configuration is already declared as "bad", we set its `q_index` to be ∞. Note that in the case of sensor network localization with exact distance data, it has been demonstrated in [5] that the quantity $Y_{ii} - ||\boldsymbol{x}_i||^2$ gives an error measure of the estimated position for the ith point. Unfortunately, when the distance data is highly noisy, that error measure has no obvious correlation to the underlying accuracy of the computed position $\boldsymbol{x}_i$. Thus we cannot use $\{Y_{ii} - ||\boldsymbol{x}_i||^2 \,|\, i = 1, \ldots, n\}$ to assign a value for `q_index`.

To summarize, suppose we have two subgroups with quality indices, $\texttt{q\_index}^1$ and $\texttt{q\_index}^2$, and that they have sufficient number (which we set a threshold of 8 atoms) of overlapping atoms with a dense underlying subgraph, roughly speaking, we will stitch the two subgroups together only if $\max\{\texttt{q\_index}^1, \texttt{q\_index}^2\} < 3$, and that the RMSD of the overlapping atoms in the two subgroups is less than 3Å.

Despite our rather effective heuristics to detect whether a subgroup is well localized, and to decide whether two subgroups should be stitched, we should point out that DISCO may still fail to work for some cases (we refer the reader to [18] for details). Thus, it is necessary to invest more effort to partition a group of atoms into localizable subgroups, and improve the heuristics for detecting badly localized subgroups. In our work, after incorporating chemistry information to the input distance data, we observe that it becomes slightly easier to partition a group of atoms into two localizable subgroups.

The pseudocode of the DISCO algorithm is presented in Algorithm 7. We illustrate how the DISCO algorithm solves a small molecule in Figure 13.1.

Algorithm 7 The DISCO algorithm

```
1: procedure DISCO(L,U)
2: if number of atoms < basis size then
3:     [cAest,cI] ← DISCOBASIS(L,U)
4: else
5:     [cAest,cI] ← DISCORECURSIVE(L,U)
6: end if
7: [cAest,cI] ← DISCOPATCH(L,U)
8: return cAest,cI
9: end procedure

1: procedure DISCOBASIS(L,U)
2: cI ← LIKELYLOCALIZABLECOMPONENTS(L,U)
3: for i = 1,...,LENGTH(cI) do
4:     cAest{i} ← SDPLOCALIZE(cI{i},L,U)
5:     cAest{i} ← REFINE(cAest{i},cI{i},L,U)
6: end for
7: return cAest,cI
8: end procedure

1: procedure DISCORECURSIVE(L,U)
2: [L1,U1,L2,U2] ← PARTITION(L,U)
3: [cAest1,cI1] ← DISCO(L1,U1)
4: [cAest2,cI2] ← DISCO(L2,U2)
5: cAest ← [cAest1,cAest2]
6: cI ← [cI1,cI2]
7: repeat
8:     [cAest,cI] ← COMBINECHUNKS(cAest,cI)
9:     [cAest,cI] ← REFINE(cAest,cI,L,U)
10: until no change
11: return cAest,cI
12: end procedure
```

13.5 Chemistry information

In this section, we describe the chemistry information we have added to the input distance data for DISCO.

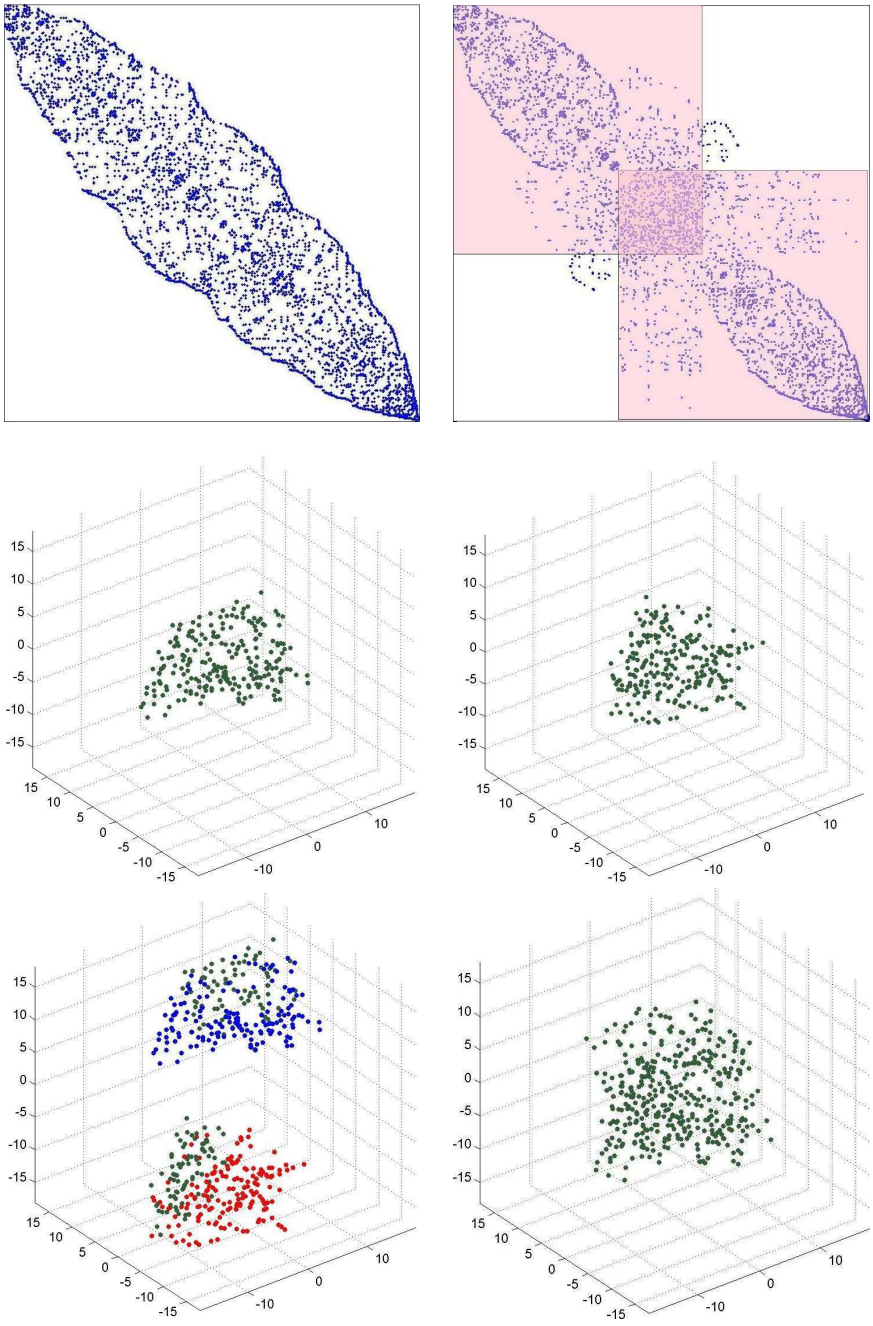

Fig. 13.1 (top left and right) Since the number of atoms is too large ($n = 402 >$ basis size $=$ 300), we divide the atoms into two subgroups. (middle left and right) We solve the subgroups independently. (bottom left) The subgroups have overlapping atoms, which are colored in green. (bottom right) The overlapping atoms allow us to align the two subgroups to form a conformation of the molecule.

13.5.1 Backbone distances

It well known that a protein molecule contains a backbone (which serves as the main "skeleton" of the molecule) from which the general shape of the molecule is determined; see Figure 13.2 for a schematic diagram of a backbone.

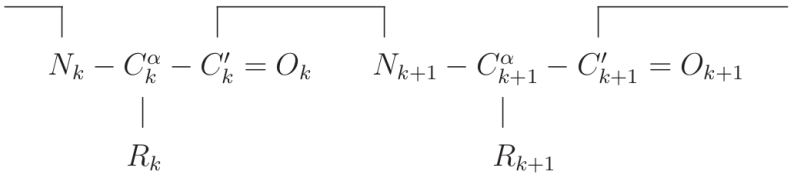

Fig. 13.2 A schematic diagram of a protein backbone, where R_k denotes the side chain of the kth amino acid.

The most commonly known distance information between atoms in a molecule are bond lengths and bond angles. Given bond lengths and bond angles, we can easily calculate the distance between two atoms which are bonded to the same atom by the cosine law. Our first attempt is to incorporate known distance information for atoms in the protein molecule into our simulated protein structure determination problem. Specifically, we incorporate the following known distances.

(a) Bond lengths of bonded pairs of atoms along the backbone, and distances between non-bonded atom pairs along the backbone which are derivable based on known bond lengths and bond angles by using the cosine law. The information we use comes mainly from the paper by R.Laskowski and D.Moss [17], and cross checked with the data in [10]. The mean distances between various atom pairs along the backbone are given in Table 13.5.1. Since the bond lengths and bond angles are not known perfectly, but within small standard deviations around some mean values, we add the distance information in the form of lower and upper bounds as follows:

$$\underline{d}_{ij} = d_{ij}(1-r), \quad \overline{d}_{ij} = d_{ij}(1+r), \tag{13.17}$$

where d_{ij} is the mean distance, and r is the standard deviation. For the distance coming from a bonded atom pair, we take r to be 1 percent; for a non-bonded pair, we take r to be 3 percent.
(b) Bond lengths of bonded pairs of atoms in the side chains. The main information we use is from a standard organic chemistry textbook [20] and a paper by W. Cornell and P. Cieplak [7]. The values we use are shown in Table 13.3. As there are too many kinds of bonds, we do not show all of them in the table. We also conduct experiments to find some bond lengths which are not found in the

literature. The way we did the experiments is as follows: for a particular bond, we calculate several such bond lengths from known conformations of molecules in PDB, and then take the average. The way we add the information to DISCO's input data is similar to (13.17). For the atoms which are mutually bonded, we take the standard deviation r to be 2 percent; for the atoms which are not mutually bonded, we take r to be 6 percent.

$d(C'-N)$	1.32	bond length	[17]
$d(C=O)$	1.24	bond length	[17]
$d(C'-C^{\alpha})$	1.52	bond length	[17]
$d(C^{\alpha}-N)$	1.46	bond length	[17]
$d(C^{\alpha}-C^{\beta})$	1.53	bond length	[17]
$\tau(O=C'-N)$	123°	bond angle	[17]
$\tau(O=C'-C^{\alpha})$	120°	bond angle	[17]
$\tau(N-C'-C^{\alpha})$	116°	bond angle	[17]
$\tau(N-C^{\alpha}-C')$	111°	bond angle	[17]
$\tau(N-C^{\alpha}-C^{\beta})$	110°	bond angle	[17]
$\tau(C^{\beta}-C^{\alpha}-C')$	111°	bond angle	[17]
$\tau(C'-N-C^{\alpha})$	121°	bond angle	[17]
$d(C_k^{\alpha},N_{k+1})$	2.41	cosine law	
$d(C_k',C_{k+1}^{\alpha})$	2.42	cosine law	
$d(O_k,C_{k+1}^{\alpha})$	2.76	cosine law	
$d(O_k,N_{k+1})$	2.25	cosine law	
$d(C_k^{\beta},N_{k+1})$	3.27	cosine law	

Table 13.2 Bond lengths and bond angles for atoms on a protein backbone [17]. The table also includes pairwise distances derivable from the known data. The subscript "k" refers to the k-th amino acid on the backbone.

Bond	Length (Å)
C-C	1.54
C=C	1.47
C=O	1.43
C-O	2.15
C-N	2.10

Table 13.3 A summary of the main bond lengths information we used for the side-chain atoms.

As mentioned in the Introduction, to automatically derive the chemistry information for a protein molecule given its amino acids sequence, we find it convenient to design a structure array data structure to code the information pertaining to each atom in the molecule. As an example, for the 402-atom protein molecule `1PTQ`, we use an 1×402 structure array (say `p`) with fields `'c'`, `'a'`, `'aa'`, `'am'` to store the information pertaining to the molecule. The first two elements of `p` are shown below:

```
p(1).c=[5.7208,-2.5088,10.2270],
p(1).a='N', p(1).aa='HIS', p(1).am='N'
p(2).c=[5.2388,-1.4878,9.2950],
p(2).a='C', p(2).aa='HIS', p(2).am='CA'
```

Here `p(1).c` refers to the known coordinates of the first atom; `p(1).a` refers to the type of atom; `p(1).aa` refers to the amino acid for which the first atom resides in; `p(1).am` refers to the atomic label of the first atom with respect to the amino acid it resides in.

13.5.2 van der Waals radii

We have also tried to add more lower bounds to our input data based on van der Waals radii. The van der Waals radius of an atom is half the minimum separation distance between two atoms (of the same type) which are not chemically related to each other. By carrying out empirical study using proteins from PDB, we found that van der Waals radii provide a good lower bound for the pairwise distance of atoms which are at least three bonds away in the molecule. Note that atomic radii can also be used to generate lower bounds for pairwise distances. But the van der Waals radii give better lower bounds as they are normally twice as large as atomic radii.

The van der Waals radii we used are from a standard inorganic textbook [2], which are given as follows: C (1.70Å), N (1.55Å), O (1.52Å), S (1.80Å).

However, after experimenting with additional lower bounds generated from van der Waals radii, we found that the results usually do not improve significantly. In addition, the time taken to solve the conformation problem becomes significantly longer because of the large number of additional lower bounds we have to handle.

The reason for not getting better results after adding the van der Waals radii might be as follow. For the input pairwise distances, though they are not exact, they are estimators of the true pairwise distances. But for van der Waals radii generated lower bounds, they are generally too weak to give useful information on the pairwise distances. Thus, adding van der Waals radii generated lower bounds are not really useful. Since it also increases the computational cost by doing so, we have decided not to add van der Waals radii generated lower bounds into our algorithm.

13.6 Numerical Experiments

Here we explain the computational issues in the DISCO algorithm. In Section 13.6.1, we present the experimental setup. In Section 13.6.2, we discuss the numerical results.

13.6.1 Experimental Setup

The source codes for our DISCO algorithm are written in MATLAB, and the SDPT3 software package of Toh, Todd and Tütüncü [24, 28, 23] is used to solve the SDP problems arising in the DISCOBASIS step of the algorithm.

We perform the numerical experiments on a dual-processor machine (3.2GHz Intel Core i5) with 4GB RAM, running MATLAB version 7.8 using only one processor.

We tested our algorithm using input distance data obtained from a set of 7 molecules taken from the Protein Data Bank. The conformations of these molecules are already known, so we can compare our computed conformations to the true conformations.

For the input distance data, we have two types of distance bounds. The first type of bounds come from chemistry information pertaining to the molecule and the second type of bounds are generated randomly to simulate NOE restraints. The sparsity of the simulated NOE distance bounds was modeled by choosing at random a proportion of all the short-range pairwise distances less than the cut-off range of 6Å, subject to the condition that the distance graph is connected[3]. The cut-off range of 6Å was selected because NMR techniques are able to give us distance information between some pairs of atoms only if they are less than approximately 6Å apart. We have adopted this particular input data model because it is simple and fairly realistic [29, 4]. In realistic molecular conformation problems, exact inter-atomic distances are not given, but only lower and upper bounds on the inter-atomic distances are known. Thus, after selecting a certain proportion of short-range inter-atomic distances, we add noise to the distances to give us lower and upper bounds. In this paper, we have experimented with a "normal" and a "uniform" noise model. The noise level is specified by a parameter σ, which indicates the expected value of the noise. When we say we have a noise level of 20%, what that means is that $\sigma = 0.2$. In the normal noise model, the bounds are specified by

$$\underline{d}_{ij} = \max\left(\alpha_{ij}, (1-\sigma|\underline{z}_{ij}|)d_{ij}\right), \quad \overline{d}_{ij} = (1+\sigma|\overline{z}_{ij}|)d_{ij},$$

where $\underline{z}_{ij}, \overline{z}_{ij}$ are independent normal random variables with zero mean and standard deviation $\sqrt{\pi/2}$. Consequently, the expected value of $|\underline{z}_{ij}|, |\overline{z}_{ij}|$ is 1, and the variance

[3] The interested reader may refer to the code for the details of how the selection is done.

is $\pi/2-1$. The positive scalar α_{ij} in $\underline{d}_{ij}$ is the minimum separation distance between atoms i and j, and we will discuss how it is chosen in the next paragraph.

In addition to the lower and upper bounds, which are available only for some atom pairs, we have minimum separation distances (MSDs) between all pairs of atoms. Due to physical reasons, two atoms i and j must be separated by a MSD α_{ij}, which depends on particular details such as the type of atoms (e.g. C-N, N-O), whether they are covalently bonded, etc. The MSD gives a lower bound for the distance between the two atoms. In our input distance data, for simplicity, we set $\alpha_{ij} = 1$Å for all covalently bonded atom pairs, regardless of the types of atoms, and $\alpha_{ij} = 2$Å for all non-bonded pairs. If we wished, we could also set α_{ij} to be the sum of the van der Waals radii (given in Section 13.5.2) of the corresponding atom pair, in the case in which the atoms are at least 3 bonds away in the molecule.

The error of the computed configuration is measured by the root mean square deviation (RMSD). If the computed configuration X is optimally aligned to the true configuration X^* using the procedure of Section 13.3.3, then the RMSD is defined by the following formula

$$\text{RMSD} = \frac{1}{\sqrt{n}} \left(\sum_{i=1}^{n} ||\boldsymbol{x}_i - \boldsymbol{x}_i^*||^2 \right)^{1/2}.$$

The RMSD basically measures the "average" deviation of the computed atom positions to the true positions.

13.6.2 Results and discussion

To help the reader to appreciate the difficulty of the molecular conformation problem under the setup we have just described, we solved a small conformation problem using sparse and noisy distances. This information is presented in Table 13.4. Even if we solve the conformation problem in a centralized fashion without divide-and-conquer, due to the sparsity and noise in the distance data, we can only get an approximate solution.

Input data: 20% distances $\leq$ 6Å					
Molecule	n	20% normal noise		20% uniform noise	
		RMSD (Å)	ℓ	RMSD (Å)	ℓ
1PTQ	402	1.08	4	0.84	4

Table 13.4 A conformation problem with sparse and noisy distance data solved in a centralized fashion without divide-and-conquer. In the table, ℓ is the number of atoms with degree less than 4.

The performance of our enhanced DISCO algorithm is listed in Tables 13.5 and 13.6. We report the average RMSDs of the conformations obtained for various molecules over 10 random instances of input distance data.

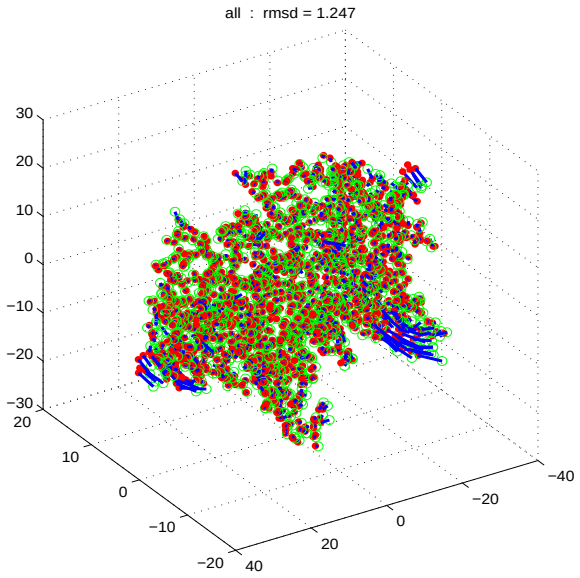

Fig. 13.3 The conformation of the molecule `1F39` corresponding to the first random input distance data in Table 13.5. In the plot, green circles depict the true positions, red dots give the computed positions, and blue line segments are the error vectors.

Finally, we would like to add that the enhanced DISCO algorithm can also improve the performance of DISCO on the 3D anchor-free graph realization problems considered in [8]. For the "bridge-donut" and "PACM" graphs considered in that paper, we are able to obtain the results shown in Table 13.7, which are comparable or better than the reconstruction results obtained by the 3D-ASAP divide-and-conquer algorithm in [8]. In Table 13.7, "ANE" denotes the average normalized error which is defined by $\sqrt{\sum_{i=1}^{n}||\boldsymbol{x}_i-\boldsymbol{x}_i^*||^2}/\sqrt{\sum_{i=1}^{n}||\boldsymbol{x}_i^*||^2}$, assuming that the true configuration $\{\boldsymbol{x}_i^* \mid i=1,\ldots,n\}$ has center of mass at the origin.

The RMSD plots across the molecules, with 10 runs given different random distance data, are shown in Figure 13.4. The plots show that our enhanced DISCO algorithm is able to produce accurate conformations ($<$ 2 Å) for all the molecules over different random inputs. Note that for each molecule, we only generate about 3.0–3.7n simulated distance bounds (to simulate the NOESY distance restraints) to be used in order to construct the conformations. Thus, the number of simulated distance bounds supplied is extremely sparse compared to the total number $(n(n-2)/2)$ of possible pairwise distances.

Input data: 20% distances $\leq$ 6Å, corrupted by 20% normal noise					
Molecule	n (l)	RMSD (Å)	Time (s)	nnz_chem/n	nnz_noe/n
1PTQ	402 (5)	0.86	23.3	2.6	3.0
1AX8	1003 (2)	1.48	110.9	2.6	3.2
1F39	1534 (5)	1.25	182.6	2.7	3.2
1RGS	2015 (10)	1.33	386.6	2.7	3.2
1KDH	2923 (14)	1.35	515.8	2.6	3.4
1BPM	3672 (9)	0.99	764.3	2.6	3.6
1MQQ	5681 (29)	0.87	1665.6	2.6	3.7

Table 13.5 The average RMSDs of the computed conformations for various molecules corresponding to 10 random instances of input distance data generated by the normal noise model. In the table, l is the average number of atoms with less than 4 neighbors; nnz_chem is the number of distance bounds generated based on chemistry information; nnz_noe is the number of distance bounds generated randomly to simulate the NOESY distance restraints.

Input data: 20% distances $\leq$ 6Å, corrupted by 20% uniform noise					
Molecule	n (l)	RMSD (Å)	Time (s)	nnz_chem/n	nnz_noe/n
1PTQ	402 (5)	0.77	21.4	2.6	3.0
1AX8	1003 (2)	1.37	106.0	2.6	3.2
1F39	1534 (5)	1.12	168.7	2.7	3.2
1RGS	2015 (10)	1.22	360.5	2.7	3.2
1KDH	2923 (14)	1.12	473.1	2.6	3.4
1BPM	3672 (9)	0.83	696.5	2.6	3.6
1MQQ	5681 (29)	0.75	1589.1	2.6	3.7

Table 13.6 Same as Table 13.5 but for input distance data generated by the uniform noise model.

Before the current enhancements, DISCO did not perform so well, for example, on the molecule 1RGS, which has a less rigid structure. But now we can see from the plots in Figure 13.4 that our enhanced DISCO algorithm is able to solve the problems robustly and accurately. When given 20% of the short-range distances, corrupted by 20% noise. The computed conformations have RMSD between 1.0 and 1.8 Å. We believe the RMSDs we obtained are the best numbers which we could hope for, and we present an intuitive explanation of why this is so. For simplicity, let us assume that the mean distance of any given edge is 3.75Å. This is reasonable because the maximum given distance is about 6Å and the smallest distance is about 1.5Å. Given 20% noise, we give a bound of about 3.0–4.5Å for that distance. Thus the true distance is only estimated to within the range of 0.75Å. Therefore we should expect the ideal RMSD to be about 0.75Å.

To give the reader an idea of how the computed conformations look like generally, we show in Figure 13.3 the conformation of the molecule 1F39 corresponding to the input data in Table 13.5. As we may observe from the plot, the atoms in the core region are accurately localized, but those on the peripheral region are less well localized.

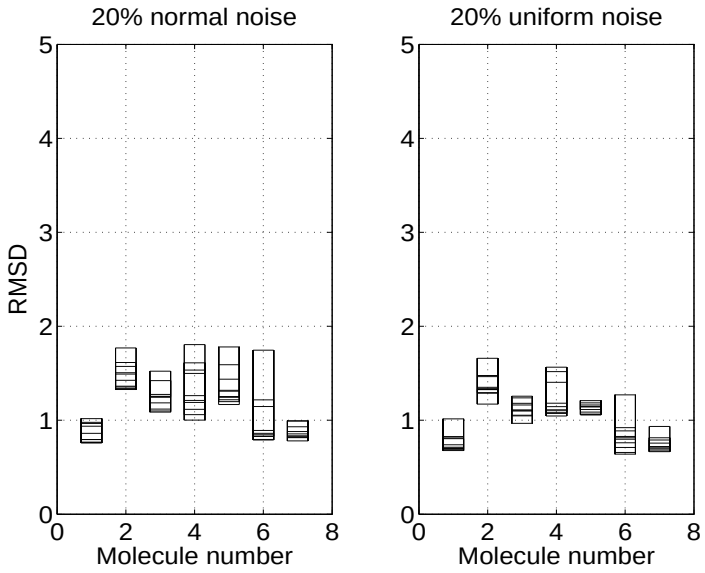

Fig. 13.4 For each molecule, ten random inputs were generated with different random number seeds. We plot the RMSDs of the ten structures produced by DISCO against the molecule number. (left) 20% short-range distances, 20% normal noise; (right) 20% short-range distances, 20% uniform noise.

	bridge-donut ($n = 500$)			PACM ($n = 799$)		
noise level	ANE	RMSD	Time (s)	ANE	RMSD	Time (s)
0	6.11e-03	1.78e-02	42.67	1.77e-02	9.42e-02	91.42
5	1.02e-02	2.95e-02	42.77	2.37e-02	1.26e-01	97.27
10	3.28e-02	9.53e-02	40.79	8.00e-02	4.27e-01	93.46
15	4.11e-02	1.19e-01	42.01	4.43e-02	2.36e-01	104.22
20	4.52e-02	1.31e-01	45.96	4.95e-02	2.64e-01	98.75
25	8.42e-02	2.45e-01	44.58	7.91e-02	4.22e-01	98.94
30	9.39e-02	2.73e-01	55.33	9.24e-02	4.93e-01	99.51
35	8.53e-02	2.48e-01	69.11	2.00e-01	1.07e-00	136.53
40	1.79e-01	5.21e-01	73.04	9.52e-02	5.08e-01	161.13
45	1.43e-01	4.16e-01	60.40	1.95e-01	1.04e-00	198.45
50	2.13e-01	6.19e-01	46.74	2.14e-01	1.14e-00	246.29

Table 13.7 Results obtained by the enhanced DISCO algorithm on the "bridge-donut" and "PACM" 3D graph realization problems considered in [8].

13.7 Conclusion

We have proposed a novel divide-and-conquer, SDP-based algorithm for the molecular conformation problem. Our numerical experiments demonstrate that the algorithm is able to solve very sparse and highly noisy protein molecular conformation

problems with simulated data accurately and efficiently. The largest molecule with more than 5000 atoms was solved in about 30 minutes to an RMSD of 1.0Å, given only 20% of pairwise distances less than 6Å which are corrupted by 20% multiplicative noise.

In this work, we have only dealt with simulated data. The next step forward would be to adapt our enhanced DISCO algorithm to tackle molecular conformation problems with real MNR experimental data, as was done in [15].

References

1. T. Ashida, Y. Tsunogae, I. Tanaka, and T. Yamane, *Peptide chain structure parameters, bond angles and conformationai angles from the cambridge structural database*, Acta Crystallographica **B43**, 212–218, 1987.
2. P. Atkins, *Inorganic Chemistry*, Oxford, 2006.
3. P. Biswas, T.-C. Liang, K.-C. Toh, T.-C. Wang, and Y. Ye, *Semidefinite programming approaches for sensor network localization with noisy distance measurements*, IEEE Transactions on Automation Science and Engineering **3**, 360–371, 2006.
4. P. Biswas, K.-C. Toh, and Y. Ye, *A distributed SDP approach for large scale noisy anchor-free graph realization with applications to molecular conformation*, SIAM Journal on Scientific Computing **30**, 1251–1277, 2008.
5. P. Biswas and Y. Ye, *Semidefinite programming for ad hoc wireless sensor network localization*, Proceedings of the third international symposium on Information processing in sensor networks, ACM Press, 46–54, 2004.
6. P. Biswas and Y. Ye, *A distributed method for solving semidefinite programs arising from ad hoc wireless sensor network localization*. In: "Multiscale Optimization Methods and Applications", W.W. Hager (Ed.), Springer, 69–84, 2006.
7. W. Cornell and P. Cieplak, *A second generation force field for the simulation of proteins, nucleic acids and organic molecules*, Journal of the American Chemical Society **117**, 5179–5197, 1995.
8. M. Cucuringu, A. Singer, and D. Cowburn, *Eigenvector synchronization, graph rigidity and the molecule problem*, arXiv:1111.3304v3, 2012.
9. Q. Dong and Z. Wu, *A geometric build-up algorithm for solving the molecular distance geometry problems with sparse distance data*, Journal of Global Optimization, **26**, 321–333, 2003.
10. R. Engh and R. Huber, *Accurate bond and angle parameters for x-ray protein structure refinement*, Acta Crystallographica, **A47**, 392–400. 1991.
11. I.G. Grooms, R.M. Lewis, and M.W. Trosset, *Molecular embedding via a second-order dissimilarity parameterized approach*, SIAM Journal on Scientific Computing **31**, 2733–2756, 2009.
12. O. Güler and Y. Ye, *Convergence behavior of interior point algorithms*, Mathetical Programming **60**, 215–228, 1993.
13. T.F. Havel, *A evaluation of computational strategies for use in the determination of protein structure from distance constraints obtained by nuclear magnetic resonance*, Progress in Biophysics and Molecular Biology **56**, 43–78, 1991.
14. T.F. Havel, I.D. Kuntz and G.M. Crippen, *The combinatorial distance geometry approach to the calculation of molecular conformation*, Journal of Theoretical Biology **104**, 359–381, 1983.
15. T.F. Havel and K. Wüthrich, *A distance geometry program for determining the structures of small proteins and other macromolecules from nuclear magnetic resonance measurements of 1h-1h proximities in solution*, Bulletin of Mathematical Biology **46**, 673–698, 1984.

16. B. Hendrickson, *The molecule problem: exploiting structure in global optimization*, SIAM Journal of Optimization **5**, 835–857, 1995.
17. R. Laskowski and D. Moss, *Main-chain bond lengths and bond angles in protein structures*, Journal of Molecular Biology **231**, 1049–1067, 1993.
18. N-H. Leung and K-C. Toh, *An sdp-based divide-and-conquer algorithm for large scale noisy anchor-free graph realization*, SIAM Journal on Scientific Computing **31**, 4351–4372, 2009.
19. L. Liberti, C. Lavor, and N. Maculan, *A branch-and-prune algorithm for the molecular distance geometry problem*, International Transactions in Operational Research **15**, 1–17, 2008.
20. J. McMurry, *Organic Chemistry*, Thompson, 2008.
21. J.J. Moré and Z. Wu, *Distance geometry optimization for protein structures*, Journal on Global Optimization **15**, 219–234, 1999.
22. A.M.-C. So and Y. Ye, *Theory of semidefinite programming for sensor network localization*, Proceedings of the sixteenth annual ACM-SIAM symposium on discrete algorithms (SODA), 405–414, 2005.
23. K-C. Toh, M.J. Todd, R.H. Tutuncu, *The SDPT3 web page*: http://www.math.nus.edu.sg/ mattohkc/sdpt3.html.
24. K.C. Toh, M.J. Todd, and R.H. Tutuncu, *SDPT3—a MATLAB software package for semidefinite programming*, Optimization Methods and Software **11**, 545–581, 1999.
25. M.W. Trosset, *Applications of multidimensional scaling to molecular conformation*, Computing Science and Statistics **29**, 148–152, 1998.
26. M.W. Trosset, *Distance matrix completion by numerical optimization*, Computational Optimization and Applications **17**, 11–22, 2000.
27. M.W. Trosset, *Extensions of classical multidimensional scaling via variable reduction*, Computational Statistics **17**, 147–163, 2002.
28. R.H. Tutuncu, K-C. Toh and M.J. Todd, *Solving semidefinite-quadratic-linear programs using SDPT3*, Mathematical Programming Series B **95**, 189–217, 2003.
29. G.A. Williams, J.M. Dugan, and R.B. Altman, *Constrained global optimization for estimating molecular structure from atomic distances*, Journal of Computational Biology **8**, 523–547, 2001.
30. Z. Zhang and H. Zha, *Principal manifolds and nonlinear dimension reduction via local tangent space alignment*, SIAM Journal of Scientific Computing **26**, 313–338, 2004.
31. Z. Zhang and H. Zha, *A domain decomposition method for fast manifold learning*, Proceedings of Advances in Neural Information Processing Systems, 18, 2006.