A study on the impact of the distance types involved in protein structure determination by NMR

Simon B. Hengeveld,* T. Malliavin,† J.H. Lin,‡ L. Liberti,§ A. Mucherino,*

*IRISA, University of Rennes 1, Rennes, France.
simon.hengeveld@irisa.fr, antonio.mucherino@irisa.fr

†Institut Pasteur, Paris, France.
therese.malliavin@pasteur.fr

‡Academia Sinica, Taipei, Taiwan.
jhlin@gate.sinica.edu.tw

§École Polytechnique, Palaiseau, France.
liberti@lix.polytechnique.fr

***Abstract*—The Distance Geometry Problem (DGP) consists of finding the coordinates of a given set of points where the distances between some pairs of points are known. The DGP has several applications and one of the most relevant ones arises in the context of structural biology, where NMR experiments are performed to estimate distances between some atom pairs in a given molecule, and the possible conformations for the molecule are calculated through the formulation and the solution of a DGP. We focus our attention on DGP instances for which some special assumptions allow us to discretize the DGP search space and to potentially perform the complete enumeration of the solution set. We refer to the subclass of DGP instances satisfying such discretizability assumptions as the Discretizable DGP (DDGP). In this context, we propose a new procedure for the generation of DDGP instances where real data and simulated data (from known molecular models) can coexist. Our procedure can give rise to peculiar DDGP instances that we use for studying the impact of every distance type, involved in NMR protein structure determination, on the quality of the found solutions. Surprisingly, our experiments suggest that the distance types implying a larger effect on the solution quality are not the ones related to NMR data, but rather the more abundant, but much less informative, van der Waals distance type.**

I. Introduction

Given a simple weighted undirected graph $G = (V, E, d)$, the Distance Geometry Problem (DGP) in dimension 3 asks whether a realization $x : V \to \mathbb{R}^3$ exists such that all distance constraints

$$\forall \{u, v\} \in E, \quad ||x_u - x_v|| = d(u, v), \tag{1}$$

are satisfied [12], where $|| \cdot ||$ is the Euclidean norm. Several real-life applications can be formulated as a DGP [17]. In this work, we will focus on the very common application in structural biology, where vertices of G represent atoms of a given molecule, and the distance information associated to the edges of the graph either reflects its chemical structure, or it is experimentally obtained by Nuclear Magnetic Resonance (NMR) experiments [1]. In the context of this application, the employed distances come from different sources, and they can carry a different level of uncertainty on the actual distance value. In most of the cases, therefore, the equality constraints in equ. (1) are replaced with inequality constraints where the lower and the upper bounds on the distances are taken into account. Throughout the article, we will confuse the two terms *vertex* and *atom*, as well as the two terms *edge* and *distance*.

The Discretizable DGP (DDGP[1]) [16] is a subclass of the more general DGP class. Let E' be the subset of edges in G that are related to *exact* distances, as opposite to *interval* distances, where a lower and an upper bound are actually given. As a consequence, $E \setminus E'$ is the subset of edges in G that contain all the interval distances.

Definition 1 *A simple weighted undirected graph G represents a DDGP instance in dimension 3 if and only if there exists a vertex ordering on V such that the following two assumptions are satisfied:*

- **(a)** *$G[\{1, 2, 3\}]$ is a clique whose edges are in E';*
- **(b)** *$\forall v \in \{4, \ldots, |V|\}$, there exist u_1, u_2, and $u_3 \in V$ s.t.*
  - **(b.1)** *$u_1 < v$, $u_2 < v$, $u_3 < v$;*
  - **(b.2)** *$\{\{u_1, v\}, \{u_2, v\}\} \subset E'$ and $\{u_3, v\} \in E$;*
  - **(b.3)** *$d(u_1, u_3) < d(u_1, u2) + d(u_2, u_3)$,*

where $G[\cdot]$ is the subgraph induced by the subset of vertices of V given in argument.

While assumption **(a)** is able to fix the coordinate space where the molecular conformations will thereafter constructed, assumption **(b)** ensures that, for every atom $v \in V$ to be placed, there is only a small subset (under some conditions,

[1]Accordingly to the definitions in [12] and [9], we should actually name the problem the "interval" DDGP and use the acronym iDDGP. Throughout this paper we are however going to use the acronym DDGP, for two reasons: (i) it makes notations lighter; (ii) we prefer to think of the DDGP as a general problem comprising exact and interval distances, where an instance with only exact distances is actually a special case.

a discrete subset) of feasible positions for that atom. In fact, because of assumptions **(b.1)** and **(b.2)**, at least 3 other distinct atoms, preceding the current vertex v in the given vertex ordering, can be used as a reference for positioning the vertex v. Moreover, assumption **(b.2)** makes sure that at most one of the 3 available distances has a large enough degree of uncertainty to be considered as an interval distance, while at least two other distances can be considered as "exact". Finally, assumptions **(b.3)** ensures that the 3 reference atoms are not aligned in their assigned positions (the triangular inequality is strictly satisfied). Under these assumptions, the DDGP search space can be reduced to a discrete space having the structure of a tree [16]. Depending on the uncertainty associated to the third distance in assumption **(b.2)**, the nodes of the search tree can either contain singletons (i.e., exact locations for the current atom), or rather (relatively) small subsets of feasible positions for the atom v. For more information about the theoretical background concerning this discretization process for the search space, the reader is referred to the citations above, and others therein.

Since DDGP search spaces are trees, the complete enumeration of the solution space is potentially possible (this is a rather hard task when the search space is continuous). This is a point of high interest in the context of structural biology, because the identification of a possible conformation for a molecule, which satisfies all distance constraints, does not deny the existence of another (even completely different) conformation where all distances are still satisfied. A fundamental advantage in the DDGP formulation stands in the fact that multiple solutions can be identified at once, and that no solutions can be discovered thereafter, once a complete exploration of the search space has been performed.

The DDGP is NP-hard [8], as well as the more general DGP [21]. However, for the DDGP case, a specific algorithmic framework has been proposed in [10] to explore the search tree generated by the discretization process. This framework has been shown to work well in practice and is generally known under the name of Branch-and-Prune (BP) algorithm. The main idea is to exploit the distances that are available because of the assumptions in Def. 1 to construct the search tree recursively in a depth-first fashion, and to use additional distances that may be available for pruning purposes: whenever those additional distances are not satisfied by the candidate positions for the atoms, the currently explored branch of the tree is pruned, and the search is backtracked. This mechanism of alternating branching and pruning phases allows the search to focus on the parts of the search tree where it is more likely to discover solutions. More general details about the BP framework can be found in Section II, together with some specific information about the BP implementation that we will consider in the computational experiments below.

In previous works on the DDGP, NMR data were simulated from known molecular models extracted from PDB files [2]. As research went on (see for example [3, 9, 13, 19]), the considered DDGP instances approached more and more the *genuine* NMR data. The BP framework was extended, for example, to interval distances in [9]; it was associated to a coarse-grained representation (to better deal with uncertainty) in [19], and to a multi-threarding-like approach (to deal with larger instances) in [13]. Together with these NMR-derived distances, other distances, derived from the chemical structure of the considered molecule, are also included in the DDGP instances. These additional distances give a fundamental help in determining the final molecular conformations, by guiding the solvers towards feasible solutions.

In this work, we will make a considerable step forward, by using for the very first time real NMR data in our experiments. Moreover, we will present a computational study where we will mix real and simulated data, with the main aim of analyzing the impact of every involved type of distances on the obtained solutions. When all distance types are simulated except one specific type, then we expect to observe how important this specific type is for the determination of the molecular structure. As expected, our computational experiments seem to suggest that some distances actually have a larger impact than others on the quality of the obtained results. Unexpectedly, however, these are not NMR-related distances.

The rest of the paper is organized as follows. In Section II, we will briefly review the main ideas behind the BP framework. We pay particular attention to its implementation distributed under the name MDJEEP, where a coarse-grained representation of the solutions is introduced to deal with the uncertainty on the interval distances. In Section III, we will detail a procedure for the generation, from the original set of NMR raw data, of DDGP instances where additional distance information, obtained by analyzing the chemical structure of the molecule at hand, is also included. Finally, Section IV will be devoted to our computational experiments, where simulated and original NMR distances will coexist and form peculiar instances that we will use to study the impact of every distance type on the conformations that MDJEEP is able to find. Conclusions will be drawn in Section V, where some possible future works will be mentioned.

II. MDJEEP

MDJEEP[2] is a freeware implementation of the BP framework for DDGPs [18]. This main algorithmic framework can be implemented in the context of the DGP when the two discretization assumptions in Def. 1 are satisfied, because they ensure that the DGP search space has the structure of a tree (see Introduction). Algorithms based on the BP framework can therefore recursively construct this search tree (this is the "branching phase", which exploits the distances that are known by the assumptions), and to immediately verify the feasibility of newly generated tree branches (this second phase is named "pruning phase", because it actually allows for removing some branches from the search tree). While the pruning phase can be applied alike to both exact and interval distances, because the verification of one equality (exact case) is simply replaced by two inequalities (interval case) to verify, the outcome of the

[2] https://github.com/mucherino/mdjeep

branching phase is essentially different in the two situations. When all involved distances are exact, the set of feasible positions for a certain atom of the molecule (placed on a common layer of the search tree) is discrete and finite [11]. Instead, when interval distances are involved, disjoint and continuous subsets of feasible points can be identified for some atoms forming the molecule [9].

The use of a coarse-grained representation of the search space allows us to efficiently deal with the continuous feasible subsets of potential atomic positions while preserving the general tree structure [19, 20]. As its name suggests, this representation replaces every feasible subset of atomic positions, that is disjoint from all other possible positions related to the same atom, with a rough approximation of the subset. In this work, we approximate these subsets with three-dimensional boxes that are supposed to contain the *true* position of the atom (box shaped space regions were also employed in the preliminary experiments presented in [19]). A initial guess of the actual position of the atom inside its own box is attempted at the time the box is created; this position can subsequently be refined (via local optimization) for guaranteeing (when possible) the satisfaction of additional distances related to the current atom, including the ones that may come to play on further layers of the search tree.

In the experiment we will present in this work, we will consider the version `0.3.2` of MDJEEP. Since its version `0.3.0`, in fact, MDJEEP is integrated with a coarse-grained representation which allows for solving instances containing interval data [15]. For more information about this freeware, the reader is referred to the cited articles, as well as to the dedicated `GitHub` repository.

III. From NMR raw data to DDGP instances

Nuclear Magnetic Resonance (NMR) spectroscopy is able to provide raw data about the given molecule from which one can derive estimations of distances between pairs of atoms (mainly pairs of hydrogen atoms), together with some estimations on some of the typical torsion angles that can defined on the protein structure [5]. We can consider that these estimations are very rough: the distances between two given hydrogen atoms, for example, is given in ranges from 1.8 up to 5Å [24]. Moreover, these estimated distances generally only concern *short* distances, while the non-detection of a distance does not necessarily imply that it is longer than a "short" distance. Finally, it is sometimes hard to identify the correct pairs of atoms related to a given NMR distance, which can lead to the definition of distance constraints where a subset of potential atom pairs is given, and not only one unique pair [25, 26].

As already mentioned in the introductory part of this paper, we do not only include NMR data in our instances, but complete the overall distance information given by NMR with some additional distance information derived from analyzing different properties of the molecule. This additional distance information is fundamental to guide the solvers towards the feasible conformations. Moreover, in the specific case of the BP framework implemented by MDJEEP, it is important to point out that the assumptions in Def. 1 could not be satisfied without the explicit introduction of these additional distances [14].

In this section, we will present a procedure for creating DDGP instances by combining the NMR raw information with the additional distances that can be derived by analyzing the chemical structure of molecule. Our description will be focused on *proteins*. Listed below are four different types of distances that we will consider when building our DDGP instances:

1) lower bounds on the pairwise distances between all of the atoms, based on their van der Waals radii;
2) force field derived distances: the connectivity of the atoms in the molecular chemical structure can give us precious information on the relative distance between bonded atoms, and between two atoms bonded to another common atom;
3) torsion angle distances, derived either from the NMR measurements, otherwise computed so that to allow the corresponding torsion angle to take any angle value between 0 and $180°$;
4) NMR distances: these are interval distances, based on the measurements from NMR spectroscopy.

Our procedure is based on the idea to build-up the graph $G = (V, E, d)$ by adding more and more information (about atoms at first, about the various distance types then) in order to enrich it as much as possible. Figure 2 visualizes how the graph changes after each step in the process, showing the procedure for the first two amino acids of the protein `2jmy` in the Protein Data Bank (PDB) [2].

At the beginning, we are given an NMR file, and an empty graph G. This NMR file is in the form of a STAR file [23], which are files that are oftentimes available in the PDB. Generally, NMR files contain a string, describing the amino acid chain of the corresponding protein. Each amino acid has a defined structure and behaviour when forming peptide bonds in the backbone of a protein. Based on the protein structure, we begin by constructing the vertex set V of G, one amino acid at a time. When adding every amino acid, our procedure reflects the natural behaviour of the amino acid during the protein synthesis. Therefore, when constructing the graph, three total atoms are removed from the two amino acids forming the peptide bond, which thereafter form a water molecule, a byproduct of this event. The amino acids with missing atoms after the construction of the peptide bond are also referred to as *protein residues*. The first amino acid of the protein contains an additional N-terminus, and the final amino acid of the sequence contains the C-terminus. Compared to non-terminal residues, the amino acid which contains the N-terminus has two extra hydrogen atoms attached to the first Nitrogen atom, while the amino acid that contains the C-terminus has an extra

Oxygen atom[3]. Figure 1 shows an example of these special cases for a simple chain only containing two amino acids.

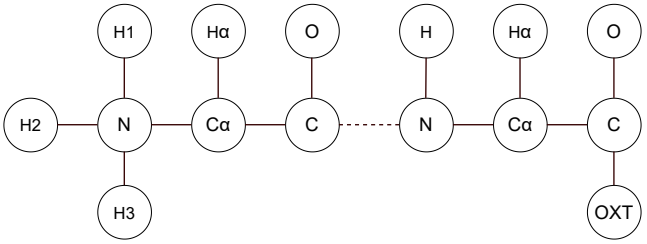


Fig. 1: Illustration of two amino acids forming a peptide bond (dashed). Side-chains are omitted for clarity.

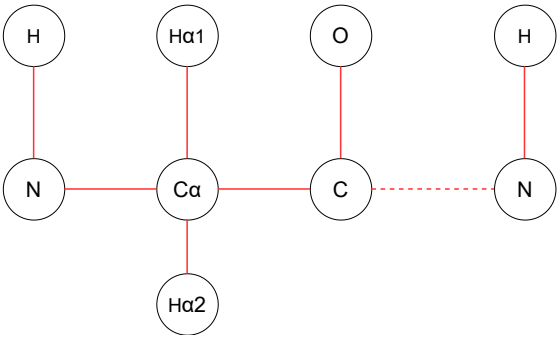
While adding these atoms to the graph, we make sure to partition the graph into different clusters. The clusters represent each individual amino acid, such that the combination of a cluster and an atom identifier is unique. This way, when we are adding distances from i.e. NMR files, we know which atoms in the NMR file pertain to which vertices in our graph. Once the vertex set V is filled, we start adding the various distance types, as detailed in the following four subsections. Tha partition of the graph in clusters representing protein residues allows us to easily identify the atoms related to the various distance types given below.

A. Lower bounds based on Van der Waals radii

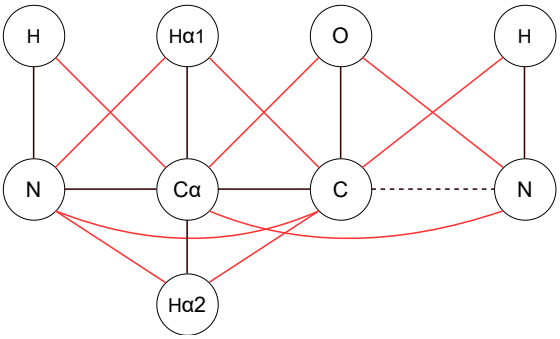
The van der Waals radius r_u, of an atom u is the radius of an imaginary hard sphere which represents the closest distance any other atom v can approach u. This means that these radii can be used to provide an expected lower bound to the distance between any pair of atoms. For any pair of two vertices $u, v \in V$, we add an edge e to G based on this van der Waals minimal distance. We sum the van der Waals radii r_u and r_v, and finally take 80% of the sum as the lower bound on the distance. This radius allows us to describes the electronic cloud around the nucleus of an atom as a sphere. However, due to polarisability, these clouds are never spheres. This means that sometimes a pair of atoms can be closer than the sum of their van der Waals radii, which is why use 80% of this sum as the lower bound on the distance.

Note that this is not a very strong distance, because it is only a lower bound. In practice, we create an interval distance which has a symbolic, very large upper bound. Technically, by adding these distances, the graph already looks like being fully connected. However, since these are only lower bounds and not very effective distances, we do not consider them when looking at the discretizability of the instance. We decided not to include these distances in Figure 2 for clarity.

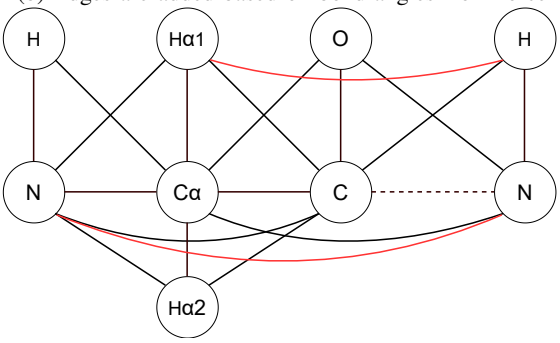
[3]This explanation may sound naïve to many people used to work in the context of structural biology. We decided to include this paragraph in this paper, however, because the bioinformatics community also includes pure computer scientists who may not be completely familiar with the biological mechanisms that our procedure attempts to reproduce by acting on G.



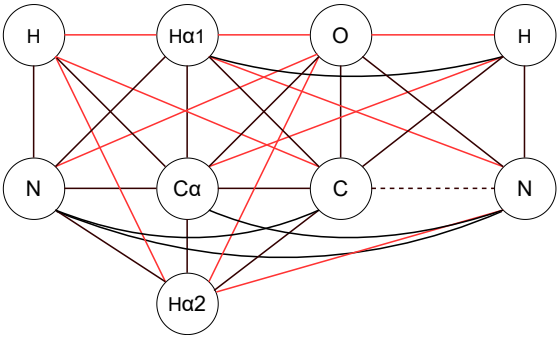
(a) New edges pertaining to bond-distances from force field data



(b) Edges are added based on bond-angles from force field data



(c) Two new edges from NMR data. One edge is a typical hydrogen NMR restraint. The second edge between the two nitrogen atoms is based on the restraints on a ψ torsion angle.



(d) Adding edges based on minimal and maximal torsion angles.

Fig. 2: The evolution of the distance graph G. The figure shows glycine and two atoms of a next amino acid in the sequence, connected by a peptide bond (dashed line). The edges that are added at each step are colored in red (light gray in gray scale); the pre-existing edges are instead marked in black. Note that we do not include the van der Waals lower bounds in this example.

B. Exact distances based on force fields

The first two proper types of distances that we can add, based on the chemical structure of the protein, are the distances that can be derived from the force fields. Force fields are computational methods that allow us to estimate the forces between atoms within molecules. Using these forces, we can obtain rather precise estimates of distances between bonded atoms. These distances are added as edges to G, and are shown in Figure 2a. Another useful piece of information that can be computed using force fields is the the angle between atoms bonded to a common atom. Using this bond angle, we can then also compute the distance between these two atoms. This can be done by using the law of cosines, as two sides and one angle are known. Examples of these distance edges are shown in Figure 2b, highlighted in red (light gray in gray scale).

C. Interval distances from raw NMR data

The next edges to be added are based on the NMR restraints found in NMR raw data. The NMR file type that we focused on in the practical implementation is the STAR file type [23]. This is predominantly because the NMR STAR files include a description of the residue chain. Other NMR files could also be used, as long as they are somehow paired with information about the sequence of amino acids. These NMR files generally include two types of restraints. The first type of restraints are direct constraints on distances. They typically describe distances between hydrogen atoms, and can only be measured for atoms that are not further away from each other than 5 to 6Å. Furthermore, these constraints lead to interval distances whereby the intervals can be quite large. An illustration of such a distance between hydrogen atoms can be found in Figure 2c, between the atoms H^1_α and H. In this work, when there is uncertanty on the distance assignment, we simply look at the corresponding PDB file and we fix the assignment to the right pair of atoms.

The second restraints that we can find in an NMR raw data file are bounds on different torsion angles within the protein backbone. A torsion angle, or *dihedral* angle, is defined by three consecutive bonds involving four atoms. The angle describes a rotation around the middle bond. Figure 3 gives an illustration of such a torsion angle γ. In the case of molecules, the points p_1 through p_4 describe four consecutive bonded atoms. The distances a, b and c then describe the bond distances. Finally, the angles α and β are bond angles that can be derived from force fields (see previous section).

NMR data files give empirical constraints on two types of torsion angles, ϕ and ψ. The ϕ angles pertain to the rotation around the N-Cα bonds while the ψ angles relate to the Cα-C bonds, both in the backbone. Using these torsion angles ($\gamma = \phi$ or ψ), and the distances (a, b) as well as the angles (α, β) obtained from the force field data, we can compute bounds on the distance between the first (p_1) and the last (p_4) atoms of the quadruplet. To do this, we compute the coordinates of the points p_1 and p_4 with the origin set at the midpoint of p_2p_3. We imagine the particular case where the torsion angle $\tau = 0$. Then, from this specific case we generalize for all other cases

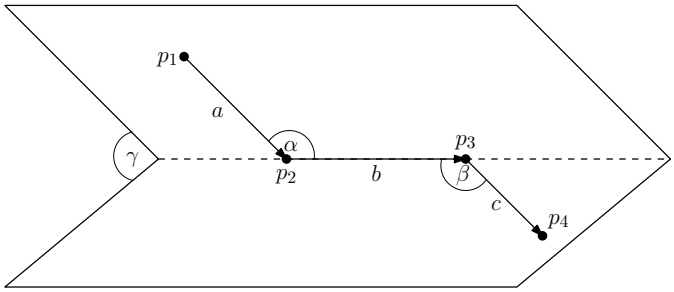


Fig. 3: An illustration of a dihedral angle given four points p_1, p_2, p_3 and p_4.

by rotating the vector $p_1\vec{p}_2$ around the x axis, with an angle of $\gamma/2$. We rotate $p_3\vec{p}_4$ in the opposite direction, with angle $-\gamma/2$. This gives us the below formulas for $p1$ and $p4$:

$$p_1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma/2 & -\sin\gamma/2 \\ 0 & \sin\gamma/2 & \cos\gamma/2 \end{pmatrix} \begin{pmatrix} -a\cos\alpha \\ +a\sin\alpha \\ 0 \end{pmatrix} - \begin{pmatrix} b/2 \\ 0 \\ 0 \end{pmatrix}$$

$$p_4 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma/2 & \sin\gamma/2 \\ 0 & -\sin\gamma/2 & \cos\gamma/2 \end{pmatrix} \begin{pmatrix} -c\cos\beta \\ +c\sin\beta \\ 0 \end{pmatrix} + \begin{pmatrix} b/2 \\ 0 \\ 0 \end{pmatrix}$$

All that remains is to compute the distance from $p1$ to $p4$ using the Euclidean distance formula:

$$\delta = \sqrt{(x_1 - x_4)^2 + (y_1 - y_4)^2 + (z_1 - z_4)^2}$$

An example of such a distance is shown between the two nitrogen atoms in Figure 2c. This interval distance is based on the ψ angle of the C_α-C bond.

D. Weak interval distances based on minimal and maximal torsion angles

Without considering the van der Waals distances for discretization purposes, we can notice that, even after adding the distance types described in the previous two sections, the graph G does not correspond yet to an instance that satisfies the assumptions in Def. 1. This is visible in Figure 2c, which shows that we are still missing some cliques to satisfy the assumptions.

To this aim, we add another type of interval distance: for every four atoms connected by three consecutive bonds, we can derive bounds on the distance between the first and the fourth atom of the sequence. We compute these bounds based on the minimal (0°) and maximal (180°) torsion angles of the sequence (if information about this torsion angle was not already exploited in the previous section). These intervals are rather large, as they cover all possible values for the torsion angle, but they can be useful in solution methods such as the BP algorithm for guiding towards feasible solutions. Examples of edges added to G corresponding to these distance intervals are shown in Figure 2d.

After adding this last distance type, the graph G does satisfy the assumptions allowing for the discretization of the search space. The solver MDJEEP (see Section II) can therefore be

invoked for the solution of the DDGP instance represented by the graph G.

IV. Computational experiments

In previous works on the DDGP (see for example [19]), the NMR instances were simulated by looking at some of the molecular models deposited on the PDB [2]. MDjeep was shown to perform quite well on those simulated data, and this fact strongly motivated the present work where real NMR data are going to be used. However, in our preliminary experiments, we soon noticed that the larger ranges for the intervals given by the real NMR data makes MDjeep often struggle to find solutions in a reasonable amount of time (experiments took more than one hour for the relatively small proteins considered below). Therefore, in order to revise and improve the performance of the solver, we carried the experiments that we are going to present below, where we will define DDGP instances mixing the real data with the simulated distances. This is done with the aim of studying the impact of each considered distance type, which is likely to give us important insights for future works on the development of MDjeep.

We point out that, when force fields are involved in the computation of the distances, we make reference to the AMBER force field [27]. Moreover, when simulated, the generation of the simulated distances strongly depend on the corresponding distance type [19]. Below, we outline how we simulate the distances for each involved type (when necessary, we extract information from the first molecular model that appears in the PDB files):

- for the two force field distances, based on bonds and angles, we include exact distances with precision up to 3 decimals;
- the NMR interval distances from NMR restraints and from NMR torsion angles were simulated with a 0.5Å interval, where the true value is placed randomly within this interval;
- for the intervals based on all possible distances derived from torsion angles, a 0.1Å interval is created, where the true value is placed randomly within this interval;
- finally, for all the remaining distances, which can have their lower bound set to the sum of the van der Waals radii, we replace this lower bound by the true value, while keeping the upper bound at a symbolically large number.

We point out that the generated DDGP instances all include both *backbone* atoms, as well as the *side chains* of the selected molecules.

Some of the DDGP instances that we will use in the experiments below only contain simulated distances; others are generated instead with the idea to mix those simulated data with real distances from NMR. In the latter situation, only one distance type per time is defined with real distances, while all the others are simulated. We repeated our experiments for 7 small proteins, selected because of their small size. The considered PDB models for these proteins are shown in Figure 4.

Our construction method for DDGP instances (see Section III) was implemented in Java programming language. It is able to read STAR-NMR files in input, and to give in output the DDGP instance text file formatted so that MDjeep can directly read it. We remark that, after the graph G has been constructed, a vertex order, which allows the assumptions in Def. 1 to be satisfied, is associated to the graph by simply running the greedy algorithm proposed in [7].

The experiments were ran on a computer with a 64-bit Linux Mint operating system using a 8-core Intel(R) Core i7-7700HQ CPU at 2800 Mhz and 16 GB of main memory. The results of the experiments, where MDjeep 0.3.2 is invoked to solve the constructed DDGP instances, are shown in Table I. The first row in the table indicates the experiments where all distance types were simulated; the last row is instead related to the experiments where only genuine non-simulated distances were included in the instance. Between the two extreme cases, the inner rows of the table show the results obtained with instances where only one distance type carries real distances, while all other distances are actually simulated.

After the solver MDjeep has found one (or several) solutions to the DDGP instances, in order to evaluate the quality of such solutions, we computed the lowest Root Mean Square Deviation (RMSD) between each solution and all available PDB models. The RMSD was computed after running our own implementation of the Kabsch alignment algorithm [6]. Our implementation also attempts a total reflection of the models to improve the alignment.

As expected, very satisfactory results can be obtained in general when all distances are simulated. And as expected, when at least one of the distance types is not simulated, but rather the real data are taken into account, then the quality of the results lowers. However, we can notice a different impact on the quality for each distance type. For example, distances and torsion angles derived from NMR belong to the distance types that do not make the quality of the solution drastically decrease when simulated data are replaced by the genuine distances. For the `2jmy` protein, MDjeep is actually able to obtain better results when using the real NMR data, instead of the simulated distances. One reason for this result is likely to be related to the quite scarce presence of NMR information. Differently from other distance types, in fact, NMR-derived distances (either pure distances or torsion angles) are generally much less abundant than van der Waals or force fields distances. It is important to point out, however, that the number of NMR-derived distances can actually change per each specific NMR experiment, and therefore their impact on the overall conformations can be more or less pronounced depending on the quantity of distances that can actually be derived by NMR.

Table I also shows that distances based on force fields have a rather light effect on the quality of the found solutions. For the `2fbu` protein, while the results obtained with real NMR-related data (both distances and torsion angles) are comparable with the results with simulated data, the table shows a reduction of about 50% on the RMSD value when genuine force field

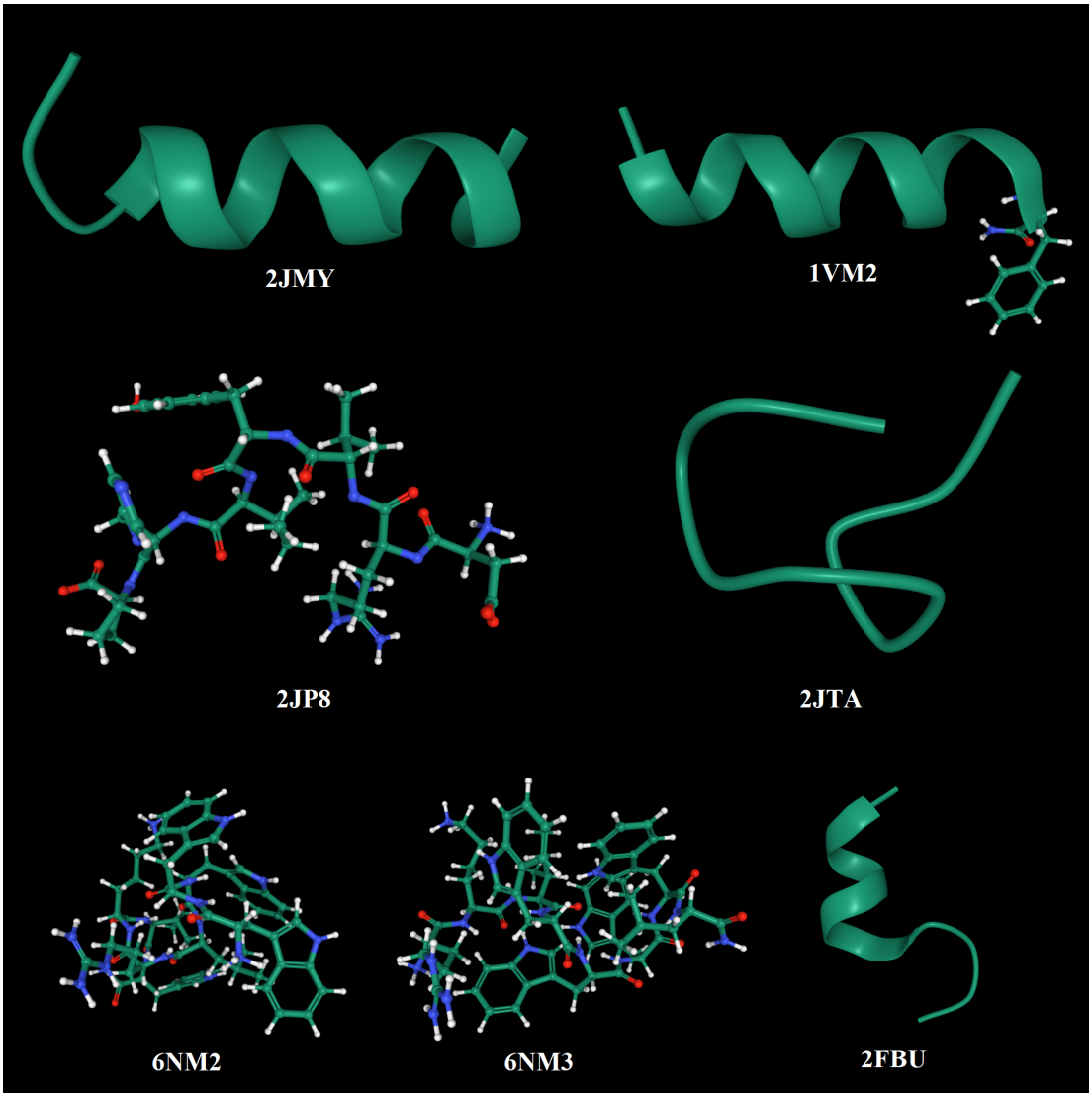


Fig. 4: The considered PDB model of the 7 considered proteins: `2jmy`, `1vm2`, `2jp8`, `2jta`, `6nm2`, `6nm3` and `2fbu` . The images were generated using Mol* [22], the PDB model viewer associated with the RCSB database. It is the viewer that automatically chose a more or less detailed representation for each molecule.

Raw data derived distance type	2jmy	1vm2	2jp8	2jta	6nm2	6nm3	2fbu
None	0.050	0.011	0.004	0.063	0.289	0.196	1.595
van der Waals	5.076	0.289	1.442	0.535	0.730	3.036	3.339
Force field bonds and angles	0.012	0.343	0.021	0.291	0.319	0.210	0.839
NMR restraints	0.005	0.568	0.009	0.771	0.462	0.355	0.612
NMR torsion angles	0.007	0.660	0.010	0.554	0.225	0.244	1.596
Min-max torsion angles	1.587	1.485	3.332	–	0.504	0.473	–
All types	4.181	3.850	3.529	4.522	3.419	4.326	4.94

TABLE I: The effect of each of the different distance types on the accuracy of the solution found by MDJEEP 0.3.2. For every experiment, the RMSD value (in Å) obtained when comparing the found solution and the original models in the PDB entry are reported. The symbol "–" indicates that MDJEEP was not able to find a solution within 1-hour CPU time.

parameters are used at the place of simulated data (which correspond to the set of distances extracted from a known PDB model in this case). This result must moreover be considered as more relevant of the one concerning NMR data, because at least 2 distances per atom, belonging to this type, are included in our DDGP instances.

One distance type which shows a larger impact on the experiments is the min-max torsion angle type. The reason for this is likely to be related to their total artificial nature.

Introduced in previous publications on the DDGP for guiding the search, they give a rather accurate information on all torsion angles that can be defined in the molecular conformation. However, only a few torsion angles can be estimated by NMR experiments, while no information can be obtained for the others when NMR gives to clues. The artificial nature of this distance type is reflected in our Table I, which shows that the set of simulated data deprived of this distance type give rise to DDGP instances that MDJEEP cannot solve in a computational time comparable to all the other presented experiments (which is, in less than 1 hour on our computer machine). This shows that those simulated data played an important role in guiding the search towards the feasible solutions. A very interesting remark is that, when no distances are simulated and this distance type is not included at all in the instance, then `MDjeep` can again find solutions in a comparable computational time (see last row in our table of experiments).

Finally, the distance type that seem to give a larger impact is, unexpectedly, the one related to van der Waals distances. Our instances are very rich of this distance type, because we can define one distance for every pair of atoms which has still no assigned distance. We remark however that this is the only type where the distances only indicate lower bounds, and where upper bounds are symbolically set to a sufficiently large values. When these distances are simulated, the actual distance value is taken for the lower bound; in genuine instances, it is rather set to the 80% of the sum of the two van der Waals radii. Even if these two possible lower bounds do not differ much, while the upper bounds remain unchanged, the effect on the real distances on the experiments is quite consistent. One main reason for this result is undoubtedly due to the very generous presence of this distance type in our instances.

V. Conclusions and future works

We studied the impact of different distance types on experiments of protein structure determination where the involved distances are either simulated or real. Our approach is based on the discretization of the search space of the corresponding distance geometry problem, which defines a special class of instances we have referred throughout our paper to as the DDGP. To perform this study, we have introduced a new procedure for the generation of DDGP instances that can take into consideration genuine data and, at the same time, simulated data obtained through known models of the molecules at hand. Unexpectedly, our experiments indicate that the distance types that can have a larger impact on the quality of the obtained solutions are not the ones related to NMR or force fields, but mostly the van de Waals distances. We conclude that this result is the direct consequence of the high abundance in our generated instances of van der Waals distances, which goes in contrast with the quite scarce presence of NMR-derived distances.

Future works will need to confirm our new empirical results with further experiments and a deeper study of the problem. To achieve this aim, we plan to work to improve the performances of MDJEEP so that it can deal with larger molecules and therefore to provide (at least a part of) potential solutions in a shorter time. One of the directions we will investigate consists in including in MDJEEP the energetic terms of the force fields which we are already using (one example is the Lennard-Jones energy, which was already integrated in the BP framework in a previous work [4]). Finally, an important step also consists in extending the solver so that it can consider distances with multiple (and therefore uncertain) assignments to some atom pairs.

Acknowledgments

We wish to thank the contributors of the several online forums for providing their very precious explanations. In particular we wish to thank the user Jean Marie[4] of Mathematics Stack Exchange for the explanations in the thread "Distance between two points in two different planes given the dihedral angle".

This work is partially supported by the international France-Taiwan project MULTIBIOSTRUCT funded by the ANR funding agency (ANR-19-CE45-0019) in France, and by the Ministry of Science and Technology (MoST 109-2923-M-001-004-MY3) of Taiwan.

References

[1] F.C.L. Almeida, A.H. Moraes, F. Gomes-Neto, *An Overview on Protein Structure Determination by NMR, Historical and Future Perspectives of the Use of Distance Geometry Methods*. In: [17], 377–412, 2013.

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

[3] A. Cassioli, B. Bardiaux, G. Bouvier, A. Mucherino, R. Alves, L. Liberti, M. Nilges, C. Lavor, T.E. Malliavin, *An Algorithm to Enumerate all Possible Protein Conformations verifying a Set of Distance Restraints*, BMC Bioinformatics **16**:23, 15 pages, 2015.

[4] D.S. Gonçalves, A. Mucherino, C. Lavor, *Energy-based Pruning Devices for the BP Algorithm for Distance Geometry*, IEEE Conference Proceedings, Federated Conference on Computer Science and Information Systems (FedCSIS13), Workshop on Computational Optimization (WCO13), Krakov, Poland, 335–340, 2013.

[5] R.K. Harris, *Nuclear magnetic resonance spectroscopy*, 1986.

[6] W. Kabsch, *A Solution for the Best Rotation to Relate Two Sets of Vectors*, Acta Crystallographica, 1976.

[7] C. Lavor, J. Lee, A. Lee-St.John, L. Liberti, A. Mucherino, M. Sviridenko, *Discretization Orders for Distance Geometry Problems*, Optimization Letters **6**(4), 783–796, 2012.

[8] C. Lavor, L. Liberti, N. Maculan, A. Mucherino, *The Discretizable Molecular Distance Geometry Problem*, Computational Optimization and Applications **52**, 115–146, 2012.

[9] C. Lavor, L. Liberti, A. Mucherino, *The interval Branch-and-Prune Algorithm for the Discretizable Molecular Distance Geometry Problem with Inexact Distances*, Journal of Global Optimization **56**(3), 855–871, 2013.

[10] L. Liberti, C. Lavor, N. Maculan, *A Branch-and-Prune Algorithm for the Molecular Distance Geometry Problem*, International Transactions in Operational Research **15**, 1–17, 2008.

[11] L. Liberti, C. Lavor, A. Mucherino, N. Maculan, *Molecular Distance Geometry Methods: from Continuous to Discrete*, International Transactions in Operational Research **18**(1), 33–51, 2011.

[12] L. Liberti, C. Lavor, N. Maculan, A. Mucherino, *Euclidean Distance Geometry and Applications*, SIAM Review **56**(1), 3–69, 2014.

[13] T.E. Malliavin, A. Mucherino, C. Lavor, L. Liberti, *Systematic Exploration of Protein Conformational Space using a Distance Geometry Approach*, Journal of Chemical Information and Modeling **59**(10), 4486–4503, 2019.

[4] `https://math.stackexchange.com/users/305862/jean-marie`

[14] A. Mucherino, *A Pseudo de Bruijn Graph Representation for Discretization Orders for Distance Geometry*, Lecture Notes in Computer Science **9043**, Lecture Notes in Bioinformatics series, F. Ortuño and I. Rojas (Eds.), Proceedings of the 3^{rd} International Work-Conference on Bioinformatics and Biomedical Engineering (IWBBIO15), Granada, Spain, 514–523, 2015.

[15] A. Mucherino, D.S. Gonçalves, L. Liberti, J-H. Lin, C. Lavor, N. Maculan, *MD-jeep: a New Release for Discretizable Distance Geometry Problems with Interval Data*, IEEE Conference Proceedings, Federated Conference on Computer Science and Information Systems (FedCSIS20), Workshop on Computational Optimization (WCO20), online event, 289–294, 2020.

[16] A. Mucherino, C. Lavor, L. Liberti, *The Discretizable Distance Geometry Problem*, Optimization Letters **6**(8), 1671–1686, 2012.

[17] A. Mucherino, C. Lavor, L. Liberti, N. Maculan (Eds.), *Distance Geometry: Theory, Methods and Applications*, 410 pages, Springer, 2013.

[18] A. Mucherino, L. Liberti, C. Lavor, *MD-jeep: an Implementation of a Branch & Prune Algorithm for Distance Geometry Problems*, Lectures Notes in Computer Science **6327**, K. Fukuda et al. (Eds.), Proceedings of the 3^{rd} International Congress on Mathematical Software (ICMS10), Kobe, Japan, 186–197, 2010.

[19] A. Mucherino, J-H. Lin, *An Efficient Exhaustive Search for the Discretizable Distance Geometry Problem with Interval Data*, IEEE Conference Proceedings, Federated Conference on Computer Science and Information Systems (FedCSIS19), Workshop on Computational Optimization (WCO19), Leipzig, Germany, 135–141, 2019.

[20] A. Mucherino, J-H. Lin, D.S. Gonçalves, *A Coarse-Grained Representation for Discretizable Distance Geometry with Interval Data*, Lecture Notes in Computer Science **11465**, Lecture Notes in Bioinformatics series, I. Rojas et al (Eds.), Proceedings of the 7^{th} International Work-Conference on Bioinformatics and Biomedical Engineering (IWBBIO19), Part I, Granada, Spain, 3–13, 2019.

[21] J. Saxe, *Embeddability of Weighted Graphs in k-Space is Strongly NP-hard*, Proceedings of 17^{th} Allerton Conference in Communications, Control and Computing, 480–489, 1979.

[22] D. Sehnal, S. Bittrich, M. Deshpande, R. Svobodová, K. Berka, V. Bazgier, S. Velankar, S.K. Burley, J. Koca, A.S. Rose, *Mol* Viewer: Modern Web App for 3D Visualization and Analysis of Large Biomolecular Structures*, Nucleic Acids Research, 2021.

[23] E.L. Ulrich, K. Baskaran, H. Dashti, et al. *NMR-STAR: Comprehensive Ontology for Representing, Archiving and Exchanging Data from Nuclear Magnetic Resonance Spectroscopic Experiments*, J Biomol NMR 73, 5–9, 2019.

[24] K. Wuthrich, *NMR in Structural Biology: A Collection of Papers by Kurt Wuthrich*, Vol. 5. World Scientific, 1995.

[25] M. Nilges, *Ambiguous distance data in the calculation of NMR structures*, Folding and Design, Volume 2, Supplement 1, Pages S53-S57, 1997.

[26] Linge JP, Habeck M, Rieping W, Nilges, *M. ARIA: automated NOE assignment and NMR structure calculation* Bioinformatics, 2003.

[27] R. Engh, R. Huber, *Accurate bond and angle parameters for X-ray protein structure refinement*, Acta Crystallographica Section A, Volume 47, 392-400, 1991