A Pseudo de Bruijn Graph Representation for Discretization Orders for Distance Geometry

Antonio Mucherino

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

Abstract. Instances of the distance geometry can be represented by a simple weighted undirected graph G. Vertex orders on such graphs are discretization orders if they allow for the discretization of the K-dimensional search space of the distance geometry. A pseudo de Bruijn graph B associated to G is proposed in this paper, where vertices correspond to $(K+1)$-cliques of G, and there is an arc from one vertex to another if, and only if, they admit an overlap, consisting of K vertices of G. This pseudo de Bruijn graph B can be exploited for constructing discretization orders for G for which the consecutivity assumption is satisfied. A new atomic order for protein backbones is presented, which is optimal in terms of length.

1 Introduction

The ordering associated to the atoms of a given molecule plays a fundamental role in the discretization of Molecular Distance Geometry Problems (MDGPs) [15,19]. The MDGP is the problem of finding suitable three-dimensional conformations for a given molecule by exploiting the information concerning known distances between atom pairs. A simple weighted undirected graph $G = (V, E, d)$ can be formally used for representing an MDGP instance, where vertices u and $v \in V$ represent atoms, and there is an edge $(u, v) \in E$ between u and v if the corresponding distance is known. The weights associated to the edges provide the numerical values for such distances. These values can be either exact or represented by a real-valued interval. The MDGP basically asks whether the graph G can be embedded in dimension $K = 3$. Notice, however, that the same problem can be defined for any dimension $K > 0$.

The discretization of the MDGP allows for reducing the search conformational space of the problem to a tree [16]. While atoms can generally take any position in a continuous portion of the space (e.g. a (hyper)sphere containing the entire molecule), the discretization makes it possible to consider a discrete and finite subset of possible positions for each atom of the molecule. This space reduction does not decrease the problem complexity (which is NP-hard [21]), but it allows for the development of ad-hoc algorithms on search trees for discovering one solution (or even several solutions) to the problem [14].

A *discretization order* is an order given to the vertices of the graph G that allows for the discretization [11]. In previous works, discretization orders have been either handcrafted [7,13] or automatically generated [11,18]. When handcrafted, the orders have

F. Ortuño and I. Rojas (Eds.): IWBBIO 2015, Part I, LNCS 9043, pp. 514–523, 2015.

been particularly designed for an important class of molecules: the proteins. These molecules, in fact, perform several (often vital) functions in living beings. They are chains of *amino acids*, which are bonded to each other through peptide bonds. The simple examination of the known chemical structure of the 20 amino acids involved in the protein synthesis (which implies knowledge on distances) allowed for the identification of discretization orders for the protein backbone [13] and its side chains [7]. These handcrafted orders can also be seen as sequences of overlapping cliques of atoms (see Section 2 for more details): the possible positions for an atom u can be computed by using the information related to the atoms that immediately precede u in the order (these "reference" atoms belong to a common clique). This class of discretization orders satisfies the so-called *consecutivity assumption*, because all reference atoms are consecutive and they immediately precede u in the order.

The problem of finding a discretization order satisfying the consecutivity assumption is NP-complete [4]. When this assumption is relaxed, so that an atom u can have, as a reference, atoms that are not its immediate predecessors in the order, then the problem of finding the order has polynomial complexity [11]. A greedy algorithm for an automatic detection of discretization orders that do not necessarely satisfy the consecutivity assumption was proposed in [11,18].

When the consecutivity assumption is satisfied, it is possible to verify in advance whether the discretization distances (that are, for a given atom, grouped in the same clique) are compatible and are able to provide a finite number of positions for an atom of the molecule. For each u, since the reference atoms belong to a common clique, all relative distances are a priori known, so that their feasibility can be verified. This is not true anymore when the assumption is not satisfied: not all distances, necessary for the feasibility verification, may be available. On the one hand, therefore, orders satisfying the consecutivity assumption should be favored; on the other hand, however, the problem of identifying such orders is NP-complete.

In this work, the problem of finding discretization orders with the consecutivity assumption is studied, and, to this purpose, a *pseudo de Bruijn* graph representation [2] for cliques contained in MDGP instances is proposed. This novel representation allows in fact for an easier search for this kind of discretization orders. In this representation, cliques of G are vertices of a directed graph $B = (V_B, A_B)$, where there is an arc from the vertex b to the vertex $c \in V_B$ when the two corresponding cliques overlap. As a consequence, a discretization order can be seen as a *path* on the graph B, such that every atom of G appears at least once in the sequence of cliques. Orders induced from these paths on B are discretization orders satisfying the consecutivity assumption.

By exploiting the proposed pseudo de Bruijn graph representation, a new discretization order for protein backbones was identified. In comparison to the order previously proposed in [13], this new ordering contains fewer atomic repetitions, and it is optimal in terms of length. The de Bruijn graph representation provides a support for the identification of this particular class of orders. When no ordering can be found by exploiting this representation, then orders that the greedy algorithm in [11,18] is able to identify can be considered as valid alternatives, even if this algorithm cannot ensure that the consecutivity assumption be satisfied.

Two different graphs will be considered throughout the paper. The graph $G = (V, E, d)$ represents an instance of the MDGP, where vertices u, v, etc., are *atoms* and weighted edges are *distances*. The edge set E is partitioned in E' and its complement $E \setminus E'$, where E' only contains edges referring to exact distances. The graph $B = (V_B, A_B)$ represents the pseudo de Bruijn graph containing cliques of G, where the arc from the vertex b to the vertex c indicates that the two corresponding cliques overlap (see Section 2 for the rigorous definitions).

The rest of the paper is organized as follows. In Section 2, the pseudo de Bruijn graph representation for cliques in MDGP instances is presented and commented in details. By exploiting this novel representation, new discretization orders for the MDGP where the consecutivity assumption is satisfied can be found. Section 3 will present one of such orders for the protein backbones, that will result to be optimal in length. Section 4 will conclude the paper with a discussion.

2 de Bruijn Graph Representation

Graphs of *de Bruijn* are widely employed for formalizing problems related to DNA assembly [5,6,8,9]. New generation technologies are able to provide researches with subsequences of DNA (named *reads*) that need to be successively assembled into one unique sequence, which is the final DNA molecule. The best way to formalize this problem is to consider a graph where vertices represent reads, i.e. the subsequences, and where there is arc from a vertex to another when the ending of the former coincides with the beginning of the latter (there is an overlap).

The graph B considered in this work is an extension of the classical de Bruijn graph [2] which is used in the DNA application. If G represents an instance of the MDGP, the vertices of the *pseudo* de Bruijn graph $B = (V_B, A_B)$ are $(K+1)$-cliques of the graph G, where K is the dimension of the search space. A vertex $b \in V_B$ can be seen as a subsequence of $K+1$ atoms admitting an internal ordering.

In the standard de Bruijn graph, there is an arc from b to c if there is an overlap. In other words, if the ending of the subsequence b coincides with the beginning of the subsequence c, then the arc (b, c) is added in A_B. In this application, since the vertices in V_B cannot be considered as static objects (the internal order of their atoms is not constant), the standard definition of de Bruijn graph needs to be extended. Consider for example that $c \in V_B$ is a $(K+1)$-clique composed by exact edges (all distances are exact): in this case, the $K+1$ atoms in the clique can be reordered $(K+1)!$ times. If instead $b \in V_B$ contains one interval distance, there are $2(K-1)!$ permutations of the atoms that allow the extremes of the interval distance to be the first and the last atom in the clique (see Def. 2.5). In this application, it is necessary for the overlap to have length equal to K. Notice that, even if the main application of this work is to biological molecules, the theory presented in the following holds for any dimension $K > 0$.

Definition 2.1. *There is a K-overlap from the vertices b to the vertex c of V_B if there exists an internal order for the atoms in b and an internal order for the atoms in c for which the K-suffix of b coincides with the K-prefix of c.*

Algorithm 1. An algorithm for constructing an induced order r for the vertices of G from a total K-valid path on the pseudo de Bruijn graph B associated to G.

```
1: find_induced_order  in: P = {p_1, p_2, ..., p_n}  out: order r
2: i = 1
3: for all u ∈ p_1 in the internal order do
4:     r_i = u; i++
5: end for
6: for (j = 2, n) do
7:     u = last vertex in internal order of p_j
8:     r_i = u; i++
9: end for
```

Notice that this definition applies to any kind of clique (either consisting of exact distances, or containing interval data).

The interest in constructing the graph $B = (V_B, A_B)$ from the graph G is evident. When a set of coordinates has already been assigned to its first K atoms (in a given internal order), each $(K+1)$-clique allows for computing a finite set of possible positions for its last atom [12]. When all the distances in the clique are exact, there are only 2 possible positions for the atom; when the distances between the first and the last atom is represented by a real-valued interval, the positions for the last atom lie on two arcs, which can be discretized [13]. Each clique in the suitable *path* on B gives therefore the necessary information for computing a finite set of possible positions for each atom of the molecule. A path of K-overlapping $(K+1)$-cliques naturally implies a sorted sequence of atoms, i.e. an order for the vertices of the graph G.

Definition 2.2. *A K-valid path $P = \{p_1, p_2, \ldots, p_n\}$ on B is a sequence of K-overlapping cliques p_i where the internal order of each clique is preserved when referring to p_{i-1} and p_{i+1}. When every atom $u \in V$ is included in at least one clique p_i, then the path is said "total".*

Notice that the condition on the clique internal order is not necessary when standard de Bruijn graphs are concerned.

A total K-valid path on B implies the definition of an order $r : \mathbb{N}_+ \longrightarrow V \cup \{\clubsuit\}$ with length $|r| \in \mathbb{N}$ (for which $r_i = \clubsuit$ for all $i > |r|$) for the vertices of G. Alg. 1 is a sketch of the simple algorithm that is necessary to apply to this purpose. Let P be a total K-valid path on B. The first K labels are assigned to the atoms of $p_1 \in P$ (the internal order of the clique has to be preserved). Then, for all other p_j, with $j \geq 2$, the last atom of the clique p_j, in the internal order, is added to the induced order.

Proposition 2.3. *Any order r constructed by Alg. 1 from a total K-valid path P on B is a discretization order for which the consecutivity assumption is satisfied.*

Proof. By construction. □

A simple verification for the existence of a total K-valid path on B is to check its connectivity. Naturally, if B is not connected, no total paths can be constructed. But even when B is connected, a total path on B may not exist, as it is the case for the

protein backbone, even if all its $(K+1)$-cliques are considered. To overcome this issue, *auxiliary cliques* can be added in B.

Definition 2.4. *An auxiliary $(K+1)$-clique is a clique*

$$\{v_1, v_2, \ldots, v_K, v_1\}$$

where $\{v_1, v_2, \ldots, v_K\}$ is a K-clique of V having edges in E' (all distances are exact).

It is important to remark that several auxiliary cliques can be generated from one K-clique, depending on the selected internal order of its atoms. The set of vertices $\{v_1, v_2, \ldots, v_K, v_1\}$ evidently form a clique, because the distances between the duplicated v_1 and all other vertices are known. Moreover, the distance between the first and second copy of v_1 is exact and equal to 0. When deadling with protein backbones, the identification of a total K-valid path on B is only possible when auxiliary cliques are included in the pseudo de Bruijn graph B (see Section 3).

One immediate consequence in using auxiliary cliques is that atoms may be repeated one (or even several times) in the induced orders. The auxiliary clique allows for locally reordering a given subset of atoms, so that a K-overlap can become possible with other cliques. Every time an auxiliary clique is involved, an atom is repeated in the atomic sequence, exactly K places after its previous copy. This kind of orders were previously formalized in [13] and named *re-orders*. Recall that E' is the subset of E containing exact distances only.

Definition 2.5. *A repetition order (re-order) is a function $r : \mathbb{N}_+ \to V \cup \{\clubsuit\}$ with length $|r| \in \mathbb{N}_+$ (for which $r_i = \clubsuit$ for all $i > |r|$) such that:*

- *$G[\{r_1, r_2, \ldots, r_K\}]$ is a clique with edge set in E',*
- *$\forall i \in \{K+1, \ldots, |r|\}$ and $\forall j \in \{i-K+1, \ldots, i-1\}$, $(r_j, r_i) \in E'$,*
- *$\forall i \in \{K+1, \ldots, |r|\}$, either $(r_{i-K}, r_i) \in E$ or $r_{i-K} = r_i$.*

Since every re-order is a discretization order where the consecutivity assumption is satisfied, the following proposition holds.

Proposition 2.6. *Induced orders from total K-valid paths P on pseudo de Bruijn graphs B generated from G (with or without auxiliary cliques) are re-orders for the vertices of the graph G.*

3 Discretization Orders for Protein Backbones

Proteins are important molecules that perform vital functions in the bodies of living beings. They are chains of smaller molecules named *amino acids*, whose order is a priori known (in other words, every amino acid is known with its rank/position in the chain). The *protein backbone* is defined by this chain, and basically contains, in sequence for each amino acid, a nitrogen N, a carbon C_α and another carbon C, plus some additional atoms chemically bonded to them. Only 20 different amino acids can be involved in the protein synthesis. A group of atoms attached to the carbon C_α makes the 20 amino acids different from each other. Since this latter group of atoms looks like "hanging"

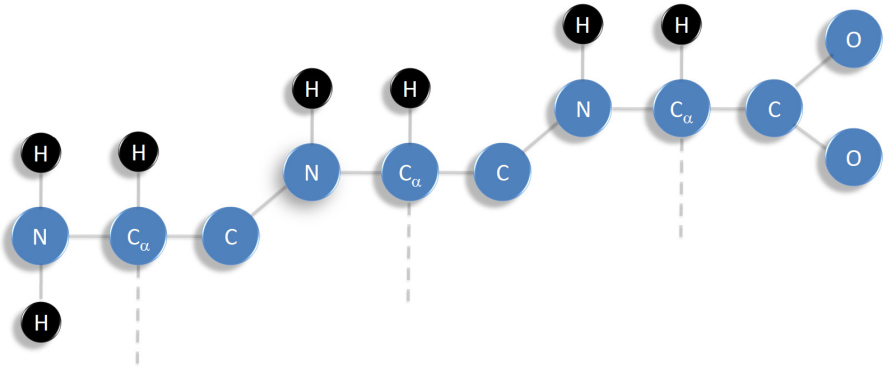

Fig. 1. The chemical structure of the considered 3-amino acid protein backbone. Some atoms are omitted because they can be positioned uniquely once the considered ones have been placed. Side chains may be attached to the atoms C_α through the bonds represented by the dashed gray lines.

from the protein backbone, it is said this is the *side chain* of the amino acid. Due to the complexity of the problem of identifying protein conformations, many proposed methods focus on protein backbones, and the information about the side chains is either approximated or negleted. In fact, once a suitable conformation for a protein backbone has been identified, there are methods that attempt the positioning of the side chains [3,10].

As in previous publications about discretization orders [13,18], a small 3-amino acid backbone will be considered in the following. Since the chemical structure of protein backbones is repetitive (no side chains $\Rightarrow$ no difference among the 20 amino acids), the identification of an order for a small molecule with 3 amino acids is sufficient, because this order can be trivially extended to protein backbones of any length.

Figure 1 shows the chemical structure of the considered 3-amino acid backbone. For every chemical bond (light gray lines in the picture), there is a known exact distance that can be considered for the discretization. Moreover, the relative distance between atoms bonded to another common atom is known, and can also be considered as exact. Finally, every quadruplet of consecutive bonded atoms form a torsion angle, from which a lower and an upper bound can be obtained for the distance between the first and the last atom of the quadruplet. Since peptide bonds, which chemically connect consecutive amino acids, give a rigid configuration to a part of the backbone structure, some of the distances derived from torsion angles can be considered as exact [17].

Table 1 shows the (non-auxiliary) cliques that can be found in the 3-amino acid backbone. Only information deduced from its chemical structure are considered in the table: the distances derived from experiments of Nuclear Magnetic Resonance (NMR) [1] are here not considered. In fact, the interest is in finding orders that are suitable for every protein backbone, so that only instance-independent distances are used for defining the 4-cliques of the pseudo de Bruijn graph B.

Table 1. 4-cliques contained in the graph representing an instance related to a 3-amino acid backbone. Auxiliary cliques are not reported.

name	*atoms*	*edge* $\{r_{i-3}, r_i\}$	*name*	*atoms*	*edge* $\{r_{i-3}, r_i\}$
c_1	N^1 C_α^1 H_α^1 C^1	exact	c_7	N^2 C_α^2 H_α^2 C^2	exact
c_2	H_α^1 C_α^1 C^1 N^2	interval	c_8	H_α^2 C_α^2 C^2 N^3	interval
c_3	C_α^1 C^1 N^2 H^2	exact	c_9	C_α^2 C^2 N^3 H^3	exact
c_4	C_α^1 C^1 N^2 C_α^2	exact	c_{10}	C_α^2 C^2 N^3 C_α^3	exact
c_5	C^1 N^2 H^2 C_α^2	exact	c_{11}	C^2 N^3 H^3 C_α^3	exact
c_6	H^2 N^2 C_α^2 H_α^2	interval	c_{12}	H^3 N^3 C_α^3 H_α^3	interval
			c_{13}	N^3 C_α^3 H_α^3 C^3	exact

In [13], a discretization order for the protein backbones was previously proposed. This order was handcrafted and satisfies the consecutivity assumption (it is a re-order, see Def. 2.5). Since then, it was generally used for discretizing MDGPs, as for example in [20], where real NMR data were considered for the first time when working with a discrete approach to distance geometry. In terms of de Bruijn graph, the handcrafted order corresponds to the following total 3-valid path in dimension 3:

$$\begin{array}{rl} \textit{(first amino acid)} & \diamondsuit \to c_1 \to c_2 \\ \textit{(second amino acid)} & \to c_4 \to c_5 \to \diamondsuit \to \diamondsuit \to c_6 \to c_7 \to \diamondsuit \to c_8 \to \diamondsuit \\ \textit{(third amino acid)} & \to c_{10} \to c_{11} \to \diamondsuit \to \diamondsuit \to c_{12} \to c_{13}\ . \end{array} \tag{1}$$

The symbol $\diamondsuit$ indicates that an auxiliary clique is used in the order. The de Bruijn graph representation of the handcrafted order starts with the auxiliary clique $(C_\alpha^1, N^1, H^1, C_\alpha^1)$. Notice that the two hydrogens bonded to the nitrogen atom N^1 of the first amino acid, as well as the two oxygens bonded to the carbon C^3 of the last amino acid, are here omitted. In fact, positions for these atoms can be calculated at the end of the computation, when a position has already been assigned to all other atoms. In the path (1), there are 7 auxiliary cliques; in general, for a protein backbone consisting of n_{aa} amino acids, $1 + 4 \cdot (n_{aa} - 2) + 2$ auxiliary cliques are necessary for constructing this path. Notice that the second amino acid can be repeated as many times as necessary in a protein backbone formed by $n_{aa} > 3$ amino acids.

The following is another possible path for the 3-amino acid backbone:

$$\begin{array}{rl} \textit{(first amino acid)} & c_1 \to c_2 \\ \textit{(second amino acid)} & \to c_3 \to c_5 \to \diamondsuit \to c_6 \to \diamondsuit \to c_7 \to c_8 \\ \textit{(third amino acid)} & \to c_9 \to c_{11} \to \diamondsuit \to c_{12} \to \diamondsuit \to c_{13}\ . \end{array} \tag{2}$$

In this case, there are two auxiliary cliques in second amino acid, and other two auxiliary cliques in the third one. As a consequence, two atoms are duplicated in each amino acid in the corresponding induced re-order. In general, for n_{aa} amino acids, $2 \cdot (n_{aa} - 1)$ repetitions are necessary. The internal order of the starting clique c_1 is: $N^1, H^1, C_\alpha^1, C^1$. Naturally, this is only one possible path that can be identified on the pseudo de Bruijn graph B. It requires fewer auxiliary cliques than the handcrafted order. However, in order to verify whether there are other possible paths for which the number of necessary auxiliary cliques is smaller (implying therefore fewer repetitions), one could attempt the

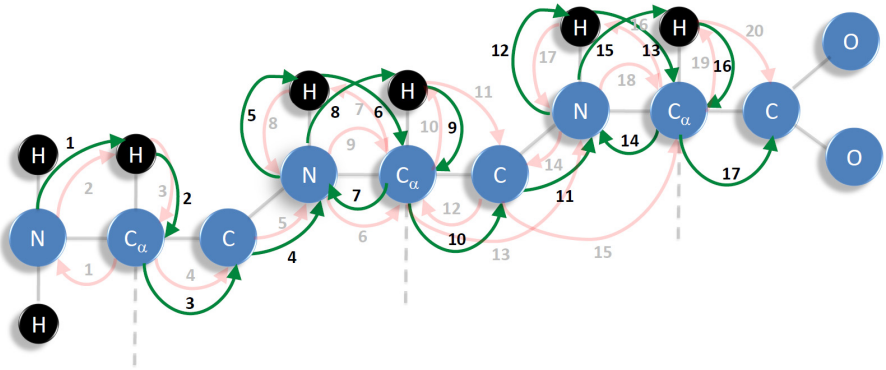

Fig. 2. An optimal (in terms of length) re-order for the protein backbone (in green). In the background, in light red, a previously proposed handcrafted order.

construction of all possible 3-valid total paths on the graph B by an exhaustive search. Naturally, even if an exhaustive search might be feasible for small instances, this is not an advisable procedure. For the considered 3-amino acid backbone, it is possible to prove that the discretization order induced from the path (2) is optimal in terms of length.

Theorem 3.1. *Let G be a graph representing an MDGP instance related to a protein backbone. For every amino acid in the protein backbone with rank greater than 2, every re-order for its atoms requires at least 2 repetitions.*

Proof. In a path starting with c_2 (see Table 1), the 4^{th} place in the induced order (refer to Alg. 1) can be either for H^1_α or for N^2, because of the constraint on the internal orders for the interval clique c_2 (refer to Def. 2.2). However, in order to construct a path to c_6 (and not to c_1), it is necessary to choose the internal order where N^2 is in position 4. At this point, the clique c_2 admits a 3-overlap with both cliques c_3 and c_4, and whichever the chosen clique is, the clique c_5 can follow either c_3 or c_4. The position of the atom H^2 in the induced order is the 5^{th} (when c_3 is chosen) or the 6^{th} (when c_4 is chosen). In order to add c_6 immediately after c_5, the atom H^2 should be instead in position 4, which is taken by N^2. However, the position 4 was fixed by c_2 at the beginning of the path. An auxiliary clique is therefore necessary for adjusting the internal order of c_5 and for making it possible to have a 3-overlap with the clique c_6. Naturally, the use of an auxiliary clique before c_6 might be avoided if a different path is rather constructed, where auxiliary cliques needs to be however involved earlier. This implies that at least one auxiliary clique is necessary for constructing a path on B from c_2 to c_6.

Similarly, it is possible to prove that at least one auxiliary clique is needed to step from the clique c_6 to the clique c_8. Because of the repetitive structure of protein backbones, the theorem is proved. □

Fig. 2 graphically shows the re-order induced from path (2), in green. Since this path is basically a sequence of 15 cliques, $4 + 14$ atoms (all atoms contained in the first clique + one atom for all others) are included in this order (repetitions are also counted). Fig. 2 also shows the order induced from path (1), in light red. In this case, there are more repetitions: there are 18 cliques in total, and therefore there are $4 + 17$ atoms in the induced order. The order induced by path (2) is optimal, as Theorem 3.1 shows.

4 Discussion and Conclusion

Given a graph G representing an instance of the MDGP, the existence of a discretization order allows to make the search space discrete and to employ ad-hoc algorithms, such as the Branch & Prune (BP) algorithm [13,14], for its solution. If the discretization order satisfies the consecutivity assumption (as it is the case for the re-orders), it is possible to verify in advance whether all atoms in the molecule admit a finite number of positions. This advantage motivated this work on the pseudo de Bruijn graph representation of discretization orders.

The problem of finding a discretization order satisfying the consecutivity assumption is NP-complete [4]. It is expected therefore that the complexity of any possible algorithm designed for the solution of this problem grows exponentially with its size. In fact, the exploration of all possible total K-valid paths on the pseudo de Bruijn graph presented in this paper can be rather expensive in general.

This exploration can, however, still be feasible when considering small molecules, such as the 3-amino acid backbone considered in this work or the 20 side chains belonging to the 20 amino acids that can form a protein. For the 3-amino acid backbone (see Section 3), this was not necessary, because it was possible to prove that path (2) is an optimal one (see Theorem 3.1 in Section 3). For the side chains, instead, an exhaustive search on all possible paths on the pseudo de Bruijn graph could be performed. Once an optimal order, in terms of length, will be identified for each of them, the discretization order for an entire protein can be constructed by combining all found orders, including the optimal backbone order induced by path (2). The final order will depend on the amino acid sequence of the considered protein.

This procedure is obviously not applicable to large molecules that cannot be separated in relevant parts, such as backbone and side chains. The benefits in using the pseudo de Bruijn graph B and exploring the total paths on B still have to be investigated for this kind of instances. As already remarked in the Introduction, a current valid alternative is the greedy algorithm proposed in [11] and extended to interval data in [18]. This algorithm is able to provide discretization orders (where the consecutivity assumption is however not ensured) in polynomial time. One possible direction for future research can be the following. Is it possible to deduce a discretization order with consecutivity assumption from a generic order provided by the greedy algorithm in [18] ?

References

1. Almeida, F.C.L., Moraes, A.H., Gomes-Neto, F.: An Overview on Protein Structure Determination by NMR. Historical and Future Perspectives of the Use of Distance Geometry Methods. In: [19], pp. 377–412 (2013)
2. de Bruijn, N.G.: A Combinatorial Problem, Koninklijke Nederlandse Akademie v. Wetenschappen 49, 758–764 (1946)
3. Canutescu, A.A., Shelenkov, A.A., Dunbrack Jr., R.L.: A Graph-Theory Algorithm for Rapid Protein Side-Chain Prediction. Protein Science 12, 2001–2014 (2003)
4. Cassioli, A., Günlük, O., Lavor, C., Liberti, L.: Discretization Vertex Orders in Distance Geometry. To appear in Discrete Applied Mathematics (2015)
5. Chikhi, R., Rizk, G.: Space-Efficient and Exact de Bruijn Graph Representation based on a Bloom Filter. Algorithms for Molecular Biology 8(22), 9 (2013)
6. Compeau, P.E.C., Pevzner, P.A., Tesler, G.: How to Apply de Bruijn Graphs to Genome Assembly. Nature Biotechnology 29, 987–991 (2011)
7. Costa, V., Mucherino, A., Lavor, C., Cassioli, A., Carvalho, L.M., Maculan, N.: Discretization Orders for Protein Side Chains. Journal of Global Optimization 60(2), 333–349 (2014)
8. Drezen, E., Rizk, G., Chikhi, R., Deltel, C., Lemaitre, C., Peterlongo, P., Lavenier, D.: GATB: Genome Assembly & Analysis Tool Box. To appear in Bioinformatics. Oxford Press (2014)
9. Ellis, T., Adie, T., Baldwin, G.S.: DNA Assembly for Synthetic Biology: from Parts to Pathways and Beyond. Integrative Biology 3, 109–118 (2011)
10. Kim, D.-S., Ryu, J.: Side-chain Prediction and Computational Protein Design Problems. Biodesign 2(1), 26–38 (2014)
11. Lavor, C., Lee, J., Lee-St. John, A., Liberti, L., Mucherino, A., Sviridenko, M.: Discretization Orders for Distance Geometry Problems. Optimization Letters 6(4), 783–796 (2012)
12. Lavor, C., Liberti, L., Maculan, N., Mucherino, A.: The Discretizable Molecular Distance Geometry Problem. Computational Optimization and Applications 52, 115–146 (2012)
13. Lavor, C., Liberti, L., Mucherino, A.: 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)
14. Liberti, L., Lavor, C., Maculan, N.: A Branch-and-Prune Algorithm for the Molecular Distance Geometry Problem. International Transactions in Operational Research 15, 1–17 (2008)
15. Liberti, L., Lavor, C., Maculan, N., Mucherino, A.: Euclidean Distance Geometry and Applications. SIAM Review 56(1), 3–69 (2014)
16. Liberti, L., Lavor, C., Mucherino, A., Maculan, N.: Molecular Distance Geometry Methods: from Continuous to Discrete. International Transactions in Operational Research 18(1), 33–51 (2011)
17. Malliavin, T.E., Mucherino, A., Nilges, M.: Distance Geometry in Structural Biology: New Perspectives. In: [19], pp. 329–350 (2013)
18. Mucherino, A.: On the Identification of Discretization Orders for Distance Geometry with Intervals. In: Nielsen, F., Barbaresco, F. (eds.) GSI 2013. LNCS, vol. 8085, pp. 231–238. Springer, Heidelberg (2013)
19. Mucherino, A., Lavor, C., Liberti, L., Maculan, N. (eds.): Distance Geometry: Theory, Methods and Applications, 410 pages. Springer (2013)
20. Mucherino, A., Lavor, C., Malliavin, T., Liberti, L., Nilges, M., Maculan, N.: Influence of Pruning Devices on the Solution of Molecular Distance Geometry Problems. In: Pardalos, P.M., Rebennack, S. (eds.) SEA 2011. LNCS, vol. 6630, pp. 206–217. Springer, Heidelberg (2011)
21. Saxe, J.: Embeddability of Weighted Graphs in k-Space is Strongly NP-hard. In: Proceedings of 17th Allerton Conference in Communications, Control and Computing, pp. 480–489 (1979)