



HAL
open science

Vertex order with optimal number of adjacent predecessors

Jérémy Omer, Tangi Migot

► **To cite this version:**

Jérémy Omer, Tangi Migot. Vertex order with optimal number of adjacent predecessors. 2019. hal-02025298v1

HAL Id: hal-02025298

<https://hal.science/hal-02025298v1>

Preprint submitted on 19 Feb 2019 (v1), last revised 18 Dec 2019 (v3)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Vertex ordering with optimal number of adjacent predecessors

JÉRÉMY OMER

Univ Rennes, INSA Rennes, CNRS, IRMAR - UMR 6625, F-35000 Rennes, France
jomer@insa-rennes.fr

TANGI MIGOT

Department of Mathematics and Statistics, University of Guelph, Guelph, ON N1G 2W1, Canada
tmigot@uoguelph.ca

Abstract

In this paper, we study the complexity of the selection of a graph discretization order with a stepwise linear cost function. Finding such vertex ordering has been proved to be an essential step to solve discretizable distance geometry problems (DDGPs). DDGPs constitute a class of graph realization problems where the vertices can be ordered in such a way that the search space of possible positions becomes discrete, usually represented by a binary tree. In particular, it is useful to find discretization orders that minimize an indicator of the size of the search tree. Our stepwise linear cost function generalizes this situation and allows to discriminate the vertices into three categories depending on the number of adjacent predecessors of each vertex in the order and on two parameters K and U . We provide a complete study of NP-completeness for fixed values of K and U . Our main result is that the problem is NP-complete in general for all values of K and U such that $U \geq K + 1$ and $U \geq 2$. A consequence of this result is that the minimization of vertices with exactly K adjacent predecessors in a discretization order is also NP-complete.

Keywords vertex ordering, distance geometry problem, discretization order, complexity analysis

I. INTRODUCTION

I.1. Preliminaries

We consider an undirected graph $G := (V, E)$, where $V := \{1, \dots, |V|\}$. The edges are weighted with nonnegative integer values c_e for $e \in E$. A *vertex ordering* of G is a bijective numbering of the set of vertices $\sigma : V \rightarrow \{1, \dots, |V|\}$. Function σ defines a total order over V : for $v \in V$, $\sigma(v)$ provides the position of v in the vertex ordering and $\sigma^{-1}(i)$ is the vertex with position i in σ . For a given graph, the set of vertex orderings is Π and it holds that $|\Pi| = |V|!$.

A vertex $v \in V$ is called a *neighbor* of $u \in V$ if and only if $\{u, v\} \in E$ and we denote $\delta(v)$ the neighbors of v , while $d(v) := |\delta(v)|$ is the degree of v . The set of *predecessors* of $v \in V$, denoted as $P_\sigma(v)$, includes every vertex $u \in V$ such that $\sigma(u) < \sigma(v)$. A vertex $u \in V$ is then called a *reference* of $v \in V$ if and only if $\{u, v\} \in E$ and $\sigma(u) < \sigma(v)$. The set of references of v is denoted as $R_\sigma(v)$. In other words, a reference is an adjacent predecessor and it holds that $R_\sigma(v) = P_\sigma(v) \cap \delta(v)$.

The vertex ordering problem is the problem of finding a permutation of the vertices minimizing some objective function. The difference from one ordering problem to another relies on the nature of the objective function and on additional constraints depending on the desired applications. One of the particular applications motivating this paper is the discretization of the distance geometry problem (DGP).

I.2. The discretizable distance geometry problem

An instance of the DGP is described by a weighted graph (V, E, d) where $d : E \mapsto \mathbb{R}^+$ is a *distance function*, and a dimension $K \in \mathbb{Z}^+$. The problem consists in finding an embedding $x : V \mapsto \mathbb{R}^K$ such that $\|x(u) - x(v)\|_2 = d(u, v), \forall \{u, v\} \in E$. The DGP naturally appears for instance when searching for the 3D-conformation of a molecule when all we know is a sparse set of pairwise distances between its atoms [6]. The DGP is NP-hard in general [13], and it has received a vivid attention recently (see, e.g., [9, 11, 12], or [10] for a recent introduction).

In [1], the authors show that the DGP can be solved by enumeration if we can find a *discretization order* of the graph, which they formally define as follows.

Definition 1. Let $G = (V, E)$ be an undirected graph and $K \in \mathbb{Z}^+$ such that $K \leq |V|$. A *discretization order* of G is a vertex ordering σ , such that:

1. the subgraph induced by $\{\sigma^{-1}(1), \dots, \sigma^{-1}(K)\}$ is complete, and
2. for all $v \in V$ such that $\sigma(v) > K$, $|R_v(\sigma)| \geq K$.

The problem of finding a discretization order of a graph G is called *Discretization vertex order problem*¹ (DVOP) in the literature [8]. When there exists a discretization order of G , the set of solutions is discrete (and finite) and can be enumerated efficiently using a branch-and-prune (BP) algorithm [9, 11]. In this case, the level k of the BP tree is associated with the vertex $v \in V$ such that $\sigma(v) = k$: the nodes of level k enumerate the potential positions of v in \mathbb{R}^K . It has been shown that under reasonable assumptions on d , a vertex with K references whose positions in \mathbb{R}^K are already known can be located in at most two different positions, whereas a vertex with $K + 1$ or more references has at most one possible position in \mathbb{R}^K .

The difficulty is that the potential realizations of the vertices are not computed during the search for a discretization order. Therefore, the exact number of nodes in the BP tree cannot be known before executing the BP.

As a compromise, the authors of [12] define an indicator of the size of the BP tree for a given discretization order σ . For this, they define a *double vertex* as a vertex that has exactly K references in σ (because the vertex may be assigned to two different positions). In contrast, a vertex with more than K references is a *single vertex*. Since double vertices are responsible for the growth of the BP tree, the first approach is to minimize their number. The decision problem associated with the minimization of double vertices has been called *Minimum double order problem* (MDOP) in [12].

MINIMUM DOUBLE ORDER PROBLEM (MDOP)

Input: A simple undirected graph $G = (V, E)$, two integers $K \leq |V|$ and N .

Question: Is there a discretization order of V such that the number of double vertices is smaller or equal to N ?

I.3. Contributions and Outline

All in all, finding a discretization order, σ , that minimizes the number of double vertices is an ordering problem over a simple undirected graph, which discriminates in some sense three classes of vertices

¹The problem is also sometimes called *Trilateration ordering problem* (TOP), see e.g.[1].

1. indispensable vertices: the initial clique, $\{\sigma^{-1}(1), \dots, \sigma^{-1}(K)\}$,
2. desirable vertices: single vertices, $\{v : |R_v(\sigma)| \geq K + 1\}$, and
3. undesirable vertices: double vertices, $\{v : |R_v(\sigma)| = K\}$.

This problem has already been treated numerically in [12], where the authors developed cutting plane algorithms to solve an integer programming formulation of the problem. In particular, they observed that although several different methods have been tested, none could find optimal solutions of instances with more than 100 vertices in less than one hour. Despite these experiments, they did not establish any result about the theoretical complexity of MDOP.

Lavor et al. [8] argue that DVOP is trivially NP-complete, because the first K vertices in the discretization order must form a clique. Since DVOP is a particular case of MDOP, it is straightforward that the latter is NP-complete in general. The limit of this result is that K is not a parameter that is expected to take large values in DGP. Since it stands for the dimension of a molecule conformation, it will in general be equal to 2 or 3. As a consequence, we should be more interested in complexity results where the value of K is fixed. For K fixed, Lavor et al. [8] show that DVOP is in P by exhibiting a greedy algorithm that solves the problem in polynomial time. In contrast, close variants of DVOP are NP-complete even for K fixed. For instance, Cassioli et al. [1] study the variant of DVOP where every vertex with order $\geq K + 1$ is adjacent to its K predecessors. They named this variant *Contiguous trilateration order problem*, and showed that it is NP-complete for any positive fixed value of K .

The main contribution of this article is in the study of the complexity of MDOP for any positive and fixed value of K . For this, we consider a generalization of MDOP that emphasizes the specificity of the problem. We extend the problem by introducing one new parameter $U \geq K$ that will allow for a hierarchy in the undesirable vertices. The set of feasible orders remains the same, but the objective function will not only penalize the vertices with exactly K references but also those with $K + 1$ to $U - 1$ references whenever $U \geq K + 2$. More precisely, let σ be a feasible order, i.e., $\{\sigma^{-1}(1), \dots, \sigma^{-1}(K)\}$ forms a clique and $|R_v(\sigma)| \geq K$ for all v such that $\sigma(v) \geq K + 1$, and for all $v \in V$, let

$$f_\sigma(v) := \max\{0, U - |R_v(\sigma)|\} \quad (1)$$

be the number of references of v below U . We then wish to minimize the objective

$$F_{K,U}(\sigma) := \sum_{\sigma(v) \geq K} f_\sigma(v), \quad (2)$$

which is a stepwise-linear function of the numbers of references in σ . Observe that for $U = K + 1$, we fall back to MDOP. We name the associated decision problem *Stepwise linear minimum vertex ordering* (SLVO). In the rest of the article, the parameters K and U are respectively called *minimum* and *critical* numbers of references. The restriction of SLVO where the parameters are fixed to values K and U is denoted as SLVO(κ, U). For simplicity, we will use the same notations for the optimization problems associated with MDOP, SLVO and SLVO(κ, U) as long as it is not ambiguous.

STEPWISE LINEAR MINIMUM VERTEX ORDERING (SLVO)

Input: A simple undirected graph $G = (V, E)$, three integers $K < |V|$, $K \leq U \leq |V|$ and N .

Question: Is there a discretization order of G , σ , such that $F_{K,U}(\sigma) \leq N$?

In our view, SLVO emphasizes better that the difficulty of the problem lies in the breakpoints defined by U in the objective function. It also offers some perspectives of new applications. For instance, a web social network that wishes to create some new community or service will be interested in the optimization of their advertisement campaign. Given that people are more likely to join a community already joined by several friends of them, the order in which emails or notifications are sent is of importance. In this context the initial clique may stand for influential personalities who support the community, and the minimum number of references represents a threshold under which the community would lose credit. The cost for not reaching the critical number of references can be associated with incentives such as special offers for users who need to be convinced. Although, this application is still fictional, our opinion is that the framework is wide enough to welcome others.

The rest of the paper is organized as follows. The main contributions of this paper are the complexity results presented in Section II and Section III. In particular, our main result is that, even with fixed K and U , SLVO is NP-complete whenever $U \geq \max\{K + 1, 2\}$. On the other hand, the problem is polynomial if $K = U$ or $U \leq 1$. We conclude the article with a discussion about perspectives in the development of solution algorithms and in the study of approximation algorithms in Section IV.

II. POLYNOMIAL VERSIONS OF SLVO(K, U)

Some specific instances of SLVO(K, U) can be solved in polynomial time. As already mentioned, DVOP is in P when K is fixed. The extension of this result to SLVO is in the study of the problem with fixed K and U such that $U = K$ (i.e., SLVO(K, K)). In this case, the objective function $F_{K,U}(\sigma)$ is vanishing for any vertex ordering σ , so SLVO(K, K) is tantamount to finding a discretization order. As shown in [8], this can be done in polynomial time using Algorithm 1, whose execution time is in $\mathcal{O}(|V|^K \times (|E| |V|^2))$. If $U = K$ step 9 of the algorithm is not useful. Actually, the algorithm can stop as soon as a discretization order is found.

Theorem 1. SLVO(K, K) is in P for all $K \in \mathbb{Z}^+$.

<pre> 1 for all K-cliques, C, of G do 2 Set the rank of the vertices of C to $1, \dots, K$; 3 $O := C, k := K$; 4 while a vertex has not been ordered do 5 Let i be a vertex of $V \setminus O$ with maximum number of adjacent vertices in O; 6 If i has less than K adjacent vertices in O, treat the next clique; 7 Assign rank $k + 1$ to i; 8 $k := k + 1$; 9 Update the best discretization order found so far.; 10 If $O < V$ for all initial cliques, then no discretization order exists. Otherwise, return the best discretization order found so far.</pre>
--

Algorithm 1: Greedy Algorithm.

This result is of particular importance for the discretizable DGP, since it states that once the initial clique is given, it can be known in polynomial time whether the problem has a solution or not. The greedy algorithm suggested in [8] has also been used in practice in [12]. Its use as an approximation algorithm is discussed in Section IV.

Theorem 2. $\text{slvo}(0,1)$ is in P .

Proof. Setting $K = 0$ and $U = 1$ yields any permutation of the vertices, σ , is a discretization order, while $f_v(\sigma) = 1$ if and only if vertex v has no reference in σ . For any $k \geq 1$, if a vertex is not in the same connected component as $\{\sigma(1), \dots, \sigma(k)\}$, then it does not have any reference among them. Consequently, the minimum cost of a discretization order is at least equal to the number of connected components in G . Reciprocally, one can readily build a vertex with total cost exactly equal to the number of connected components by ordering those components one after the other. Hence, minimizing the objective function is equivalent to counting the number of connected components of G , which can be achieved in polynomial time. \square

III. NP-COMPLETE CASES

The following results state that it is sufficient to search for the smallest values of K and U such that $\text{slvo}(K,U)$ is NP-complete. The proof is divided into two lemmata respectively for increasing K and U .

Lemma 1. Let $K, U \in \mathbb{Z}^+$ such that $U \geq K + 1$. If $\text{slvo}(K,U)$ is NP-complete, then $\text{slvo}(K+P, U+P)$ is also NP-complete for all $P \in \mathbb{Z}^+$.

Proof. Let $K, U \in \mathbb{Z}^+, U \geq K + 1$, and let the graph $G = (V, E)$ and the positive integer N constitute an arbitrary instance of $\text{slvo}(K,U)$. For $P \in \mathbb{Z}^+$, we build an instance of $\text{slvo}(K+P, U+P)$ defined by $G_P = (V \cup V_P, E \cup E_P)$ and N , where:

- the subgraph of G_P induced by V_P is a P -clique;
- there is one edge between each vertex of V and each vertex of V_P .

More formally,

$$V_P = \{|V| + 1, \dots, |V| + P\}, \text{ and } E_P = \{\{u, v\} : u, v \in V_P, u \neq v\} \cup \{\{u, v\} : u \in V, v \in V_P\}.$$

Assume that G admits a discretization order, σ . Then we can build a vertex order, σ_P , of G_P , by positioning the vertices of V_P first followed by those of V in the order given by σ , i.e.:

$$\sigma_P(V_P) = \{1, \dots, P\}, \text{ and } \sigma_P(v) = \sigma(v) + P, \forall v \in V.$$

One can verify that the subgraph of G_P induced by $\{\sigma_P^{-1}(1), \dots, \sigma_P^{-1}(K + P)\}$ is a clique, because σ is discretization order, and that

$$|R_{\sigma_P}(v)| = |R_{\sigma}(v)| + P, \forall v \in V,$$

which means that $F_{K,U}(\sigma) = F_{K+P, U+P}(\sigma_P)$.

Reciprocally, for any vertex order σ_P of G_P , we can build a vertex order, σ , of G by removing the elements of V_P from σ_P . The number of references of each vertex of V in this new order is at most reduced by P . It follows that if σ_P is a discretization order, then σ is a discretization order of G such that $f_{\sigma}(v) \leq f_{\sigma_P}(v), \forall v \in V$.

Finally, the above shows that (G, N) is a YES instance of $\text{slvo}(K,U)$ if and only if (G_P, N) is a YES instance of $\text{slvo}(K+P, U+P)$, which concludes the proof. \square

The above result does not specify the impact of an arbitrary increase in the value of U (in particular, one that is larger than the increase in the value of K). A closer look at the proof of

Lemma 1 indicates that its last part would not generalize in this case. To illustrate this, consider $\text{slvo}(K,U)$ and $\text{slvo}(K,U+P)$ (instead of $\text{slvo}(K+P,U+P)$) and the two instances (G,N) and (G_P,N) considered in the above proof. The difficulty is that a discretization order in G_P, σ_P , would not necessarily yield a discretization order, σ , in G by simply removing the vertices that are not in G . Indeed, this operation decreases by up to P the number of references, which does not need to be greater than K in σ_P . Hence some vertices may have less than K references in σ .

Nevertheless, if $K = 0$, every argument used in the proof of Lemma 1 remains valid if we wish to reduce $\text{slvo}(0,U+P)$ from $\text{slvo}(0,U)$. This justifies the following result.

Lemma 2. *Let $U \in \mathbb{Z}^+$. If $\text{slvo}(0,U)$ is NP-complete, then $\text{slvo}(0,U+P)$ is also NP-complete for all $P \in \mathbb{Z}^+$.*

In the previous section, we have proved that $\text{slvo}(K,U)$ is in **P** for $K = U$ and for $K = 0, U = 1$. We are thus left with the question of the complexity of $\text{slvo}(0,2)$ and $\text{slvo}(1,2)$. Indeed, if those two problems are NP-complete, Lemmata 1 and 2 show that $\text{slvo}(K,U)$ is NP-complete for all $K, U \in \mathbb{Z}^+$ such that $U \geq 2$ and $U \geq K + 1$. We start with the study of $\text{slvo}(1,2)$.

Theorem 3. *$\text{slvo}(1,2)$ is NP-complete.*

We will show the theorem by polynomial reduction from 3-SAT, which is one of the 21 NP-complete problems of [7]. We consider an instance, (c, x) of 3-SAT defined by the set of clauses $c = \{c_1, \dots, c_m\}$ defined over boolean variables $x = \{x_1, \dots, x_n, x_{n+1}, \dots, x_{2n}\}$, where x_{n+i} stands for the negation of x_i for all $i \in \{1, \dots, n\}$. For $j = 1, \dots, m$, we denote, j_1, j_2 and j_3 the indices of the three terms of clause c_j , i.e., $c_j = x_{j_1} \vee x_{j_2} \vee x_{j_3}$. For $i \in \{1, \dots, n\}$, we also denote as $C(i)$ the set of clauses that involve x_i or x_{i+n} .

Remark 1. *We assume without loss of generality that there is no clause with a variable and its opposite and that all the variables appear in at least one clause.*

We then proceed as follows to transform (c, x) into an instance (G, n) of $\text{slvo}(1,2)$. The set of vertices of $G = (V, E)$ is the union of six different sets $V = X \cup X' \cup C \cup Y \cup \{O\} \cup B$, where X and X' correspond to the variables, C and Y correspond to the clauses and their terms, and $\{O\}$ and B are artificial vertices required for the validity of the reduction. An illustration of the part of G related to some variable $x_i, i \in \{1, \dots, n\}$ is given in Figure 1. The exact rules that lead to the construction of G are as follows.

- $X \cup X'$: for each variable $x_i, i = 1, \dots, 2n$, one pair of vertices $(X_i, X'_i) \in X \times X'$, connected with one edge.
- $C \cup Y$: for each clause c_j one vertex $C_j \in C$ and three vertices $Y_{j,1}, Y_{j,2}, Y_{j,3} \in Y$ that stand for the three terms of the clause: three edges connect $Y_{j,1}, Y_{j,2},$ and $Y_{j,3}$ to C_j , and two edges connect $Y_{j,k}$ to X_{j_k} and X'_{j_k} for $k = 1, \dots, 3$.
- B : for all $i \in \{1, \dots, n\}$, one gadget $\{B_i^0\} \cup B_i$, such that B_i induces a binary tree rooted at B_i^1 and whose leaves are connected to at most two vertices of $C(i)$ each such that two leaves do not connect to a same clause. Vertex B_i^0 is connected only to B_i^1, X_i and X_{i+n} .
- $\{O\}$: one vertex, which will be used as the initial clique. Vertex O is connected to every vertex of X, X' and C . For all $i \in \{1, \dots, n\}$, O is also connected to B_i^1 and to the vertices of the B_i that are connected to exactly one vertex of $C(i)$.

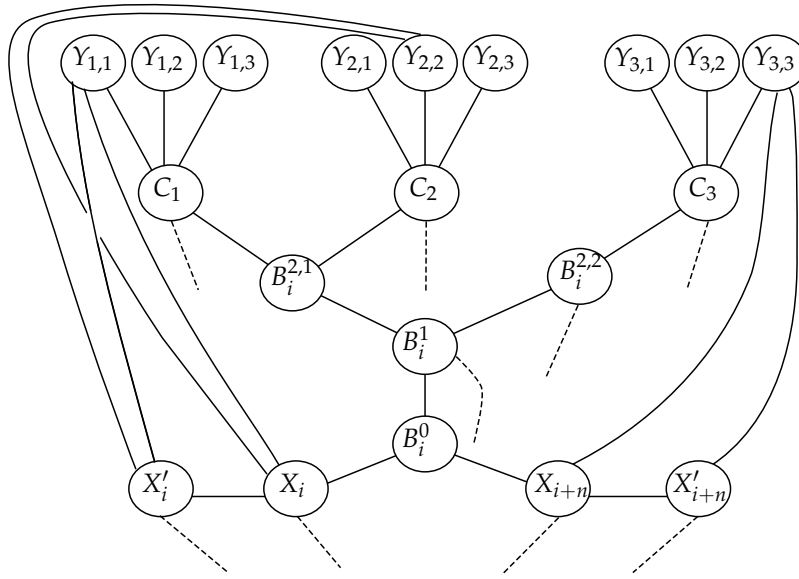


Figure 1: Part of the graph corresponding to x_i and x_{i+n} for $i \in \{1, \dots, n\}$. The dotted lines stand for edges that connect a vertex to O . In this case, $C(i) = \{c_1, c_2, c_3\}$, x_i is in first position in c_1 and in second position in c_2 , and the negation of x_i appears in third position in c_3 .

Observe that the number of vertices in gadget B_i is at most twice larger than the number of clauses in $C(i)$, because it is a binary tree whose number of leaves is less than the number of clauses in $C(i)$. It is then straightforward to verify that the transformation from (c, x) to (G, n) is polynomial.

In the proofs and discussions below, it is more convenient to focus once and for all on discretization orders of G started with O . We thus extend G with another gadget connected only to O . The gadget is composed of $n + 1$ levels including $n + 1$ vertices each, $O_{p,q}$, $1 \leq p \leq n + 1$, $1 \leq q \leq n + 1$, and one last level containing two vertices $O_{n+2,1}$ and $O_{n+2,2}$. The first level is totally connected to O and the last one is totally connected to $O_{n+2,1}$ and $O_{n+2,2}$. The other levels are connected only to those directly above and below so that each vertex has two neighbors above and two neighbors below. The gadget is illustrated in Figure 2. The graph obtained as the union of G and this gadget is denoted as G^O .

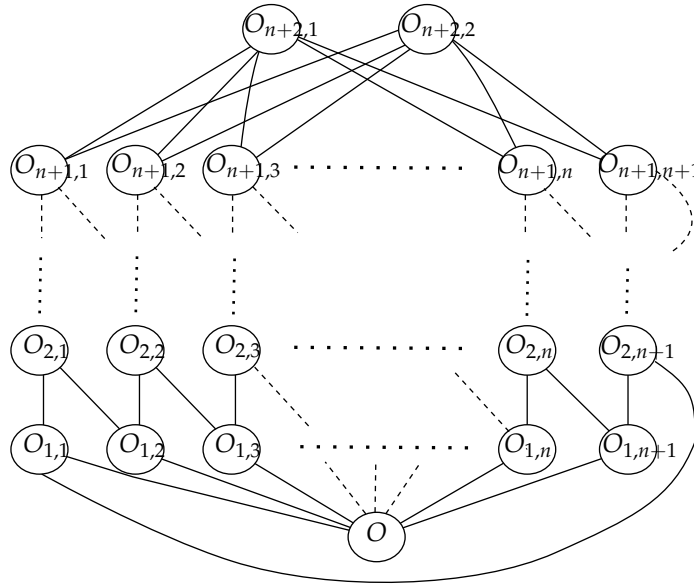


Figure 2: Illustration of the gadget rooted at O .

Proposition 1. *There is a discretization order of G^O with cost at most $n + 1$ if and only if there is a discretization order of G with cost at most n starting with O .*

Proof. Let σ be a discretization order of G such that $\sigma(1) = O$ and $F_{1,2}(\sigma) \leq n$. We build a discretization order, σ^O , of G^O by setting $\sigma^O(O_{n+2,1}) = 1$, $\sigma^O(O_{n+2,2}) = 2$ and by inserting the levels one by one in the order from $n + 1$ to 1. We then set all the vertices of G in σ^O in the same order as that given by σ . In σ^O : $O_{n+2,2}$ is the only vertex of the gadget with a non-vanishing cost; O has more than two references; the other vertices of G keep as many references as in σ . As a consequence, $F_{1,2}(\sigma^O) = F_{1,2}(\sigma) + 1 \leq n + 1$.

Reciprocally, let σ^O be a discretization order of G^O such that $F_{1,2}(\sigma^O) \leq n + 1$. A recurrence on the levels of the gadget shows that if $(\sigma^O)^{-1}(1)$ does not belong to the gadget rooted at O , then the constraints on the number of references of $\sigma^O(O_{n+2,1})$ and $\sigma^O(O_{n+2,2})$ can only be satisfied if at least $n + 1$ vertices of the gadget have exactly one reference. Since the second vertex of σ^O also has a non-vanishing cost, this is in contradiction with $F_{1,2}(\sigma^O) \leq n + 1$. We deduce that $(\sigma^O)^{-1}(1)$ belongs to the gadget, hence we can simply remove the gadget from σ^O to get a discretization order of G with cost at most n . \square

Since we will only be interested in discretization orders of G^O with cost $n + 1$, this result indicates that we can simply drop the gadget rooted at O and consider discretization orders of G with cost n and whose first vertex is O . In the remainder, we thus focus on G and set $\sigma(O) = 1$ for every discretization order σ of G . This allows to push the analysis of discretization orders of G further.

1. Let σ be a discretization order of G that starts with O (i.e., $\sigma(O) = 1$). For $i = 1, \dots, 2n$, X_i and X'_i are neighbors and they are both adjacent to O . Since $\sigma(O) = 1$, X_i and X'_i have at least one reference, and the one with higher position in σ has at least two references. Since there is no possible benefit in having more than two references, this means that X_i and X'_i can always take contiguous positions in a minimum cost discretization order. The relative position of the two vertices will not make any difference in the number of references of their neighbors, but it

might impact their own costs. Indeed, the second among X_i and X'_i in the order will always have a zero cost, because it has two references, but the first one may have only O as reference. Notice now that every neighbor of X'_i is also adjacent to X_i , but X_i has one extra neighbor, B_i^0 . This means that X_i can only have more references than X'_i (other than one another). We get that it can only be beneficial to set X_i first among the two in the order, i.e., $\sigma(X'_i) = \sigma(X_i) + 1$.

2. Now, considering any vertex order where $\sigma(O) = 1$ and $\sigma(X'_i) = \sigma(X_i) + 1$, we can propagate the deductions to any vertex $Y_{j,k}$ such that $j_k = i$. This vertex is adjacent to X_i and X'_i , so their contiguity involves that either they are both references of $Y_{j,k}$ or $Y_{j,k}$ is a reference of both. Given that $Y_{j,k}$ has only three neighbors (C_j is the third), the latter would involve that $f_\sigma(Y_{j,k}) \geq 1$. Even if X_i has no other reference than O and $Y_{j,k}$, it would still not increase the total cost if $Y_{j,k}$ was set after X_i and X'_i instead. Indeed, the cost of X_i would increase to 1 but that of $Y_{j,k}$ would decrease to 0.

The preliminary analysis shows that we can focus the search for a solution of (G, n) to the discretization orders σ such that $\sigma(O) = 1$, $\sigma(X'_i) = \sigma(X_i) + 1, \forall i$, and $\sigma(Y_{j,k}) > \sigma(X'_i) = \sigma(X_i) + 1$ for all i, j, k such that $j_k = i$. Given that X_i has no other neighbor than O , X'_i , $\{Y_{j,k}\}_{j_k=i}$ and B_i^0 , it will have a zero cost in σ if and only if $\sigma(B_i^0) < \sigma(X_i)$.

Lemma 3. *Let σ be a discretization order of G such that $\sigma(1) = O$ and $F_{1,2}(\sigma) = n$. Then, there exists a discretization order of G , $\bar{\sigma}$, such that $\bar{\sigma}(1) = O$ and*

1. $F_{1,2}(\bar{\sigma}) = n$;
2. $\bar{\sigma}(X'_i) = \bar{\sigma}(X_i) + 1$ for all $i \in \{1, \dots, 2n\}$;
3. $\bar{\sigma}(Y_{j,k}) > \bar{\sigma}(X_i)$ for all i, j, k such that $j_k = i$;
4. for all $i \in \{1, \dots, n\}$, $f_{\bar{\sigma}}(X_i) = 1$ or $f_{\bar{\sigma}}(X_{i+n}) = 1$.

Proof. From the discussion preceding the lemma, we have seen that if σ is a discretization order of G such that $\sigma(1) = O$ and $F_{1,2}(\sigma) = n$, there is another discretization order, $\bar{\sigma}$, with cost $F_{1,2}(\bar{\sigma}) \leq F_{1,2}(\sigma)$ such that $\bar{\sigma}(O) = 1$ and

- $\bar{\sigma}(X'_i) = \bar{\sigma}(X_i) + 1, \forall i = 1, \dots, 2n$;
- $\bar{\sigma}(Y_{j,k}) > \bar{\sigma}(X_i)$ for all i, j, k such that $j_k = i$.

Now, assume that there is some $i \in \{1, \dots, n\}$ such that X_i and X_{i+n} both have two references. Given that X_i has no other neighbor than O , X'_i , $\{Y_{j,k}\}_{j_k=i}$ and B_i^0 , the properties of $\bar{\sigma}$ imply that $\bar{\sigma}(B_i^0) < \bar{\sigma}(X_i)$. The same argument applied to X_{i+n} yields $\bar{\sigma}(B_i^0) < \bar{\sigma}(X_{i+n})$. It follows that B_i^0 can have only one reference in $\bar{\sigma}$ (i.e., B_i^1). All in all, we get that for all $i \in \{1, \dots, n\}$, $f_{\bar{\sigma}}(X_i) = 1$ or $f_{\bar{\sigma}}(X_{i+n}) = 1$ or $f_{\bar{\sigma}}(B_i^0) = 1$. Observing that $F_{1,2}(\bar{\sigma}) = n$, we can even further state that

$$f_{\bar{\sigma}}(X_i) + f_{\bar{\sigma}}(X_{i+n}) + f_{\bar{\sigma}}(B_i^0) = 1, \forall i \in \{1, \dots, n\},$$

and that every other vertex has at least two references.

Assume that $f_{\bar{\sigma}}(B_i^0) = 1$: we just discussed that in this case $\bar{\sigma}(B_i^0) < \bar{\sigma}(X_i)$ and $\bar{\sigma}(B_i^0) < \bar{\sigma}(X_{i+n})$. We can prove by induction on the binary tree B_i that we necessarily have $\bar{\sigma}(B_i^0) > \bar{\sigma}(C_j), \forall C_j \in C(i)$. Combined with the property that $\bar{\sigma}(Y_{j,k}) > \bar{\sigma}(X_i)$ for all i, j, k such that $j_k = i$, we get that $\bar{\sigma}(C_j) < \bar{\sigma}(Y_{j,k})$ for all $C_j \in C(i)$ and $k = 1, 2, 3$. This leads to C_j having only O as reference, a contradiction. As a consequence, we know that for all $i \in \{1, \dots, n\}$ either $f_{\bar{\sigma}}(X_i) = 1$ or $f_{\bar{\sigma}}(X_{i+n}) = 1$.

□

Proof of Theorem 3. We consider an instance (c, x) of 3-SAT and the corresponding instance (G, n) of SLVO(1,2), as described above. We prove the theorem by showing that (c, x) is satisfiable if and only if there is a discretization order of G, σ , such that $\sigma(O) = 1$ and $F_{1,2}(\sigma) = n$.

Assume first that (c, x) is satisfiable and let \bar{x} be a feasible solution. From this solution, we construct a vertex order of G, σ , where $\sigma(O) = 1$ and for $i \in \{1, \dots, n\}$, $\sigma(X_i) = 2i, \sigma(X'_i) = 2i + 1$ if $\bar{x}_i = \text{TRUE}$, and $\sigma(X_{i+n}) = 2i, \sigma(X'_{i+n}) = 2i + 1$ if $\bar{x}_i = \text{FALSE}$. We then insert in σ the vertices of Y that correspond to the variables of x set to TRUE in \bar{x} . Every vertex of C is then inserted in the vertex order, followed by all those of B . One can verify that up to this stage, the only ordered vertices with exactly one reference are those of X , whose common reference is O . Indeed, the ordered vertices of X' also have O as reference and another in X , and the ordered vertices of Y have one reference in X and another one in X' . Moreover, every vertex of C has O as reference and at least one reference in Y , because \bar{x} is a feasible solution of (c, x) . Finally, one can verify that the vertices of B can be ordered to have exactly two references as long as they come after the vertices of C in the order.

The following vertices in σ are the vertices of X that do not appear at the beginning of the order, i.e., X_i if $\bar{x}_i = \text{FALSE}$, or X_{i+n} if $\bar{x}_i = \text{TRUE}$. At this stage, each one of these variables has O and B_i^0 as references. The remaining vertices of X' and Y can then be inserted last in σ without increase in the objective value. As a consequence, the objective value of σ is exactly n .

Assume then that (G, n) is a YES instance of SLVO(1,2), such that there is a discretization order of G, σ , that satisfies $F_{1,2}(\sigma) = n$, and $\sigma(O) = 1$. Lemma 3 states that there is another discretization, $\bar{\sigma}$, such that $\bar{\sigma}(1) = O$ and

- $F_{1,2}(\bar{\sigma}) = n$;
- $\bar{\sigma}(X'_i) = \bar{\sigma}(X_i) + 1$ for all $i \in \{1, \dots, 2n\}$;
- $\bar{\sigma}(Y_{j,k}) > \bar{\sigma}(X_i)$ for all i, j, k such that $j_k = i$;
- for all $i \in \{1, \dots, n\}$, $f_{\bar{\sigma}}(X_i) = 1$ or $f_{\bar{\sigma}}(X_{i+n}) = 1$.

This also means that $f_{\bar{\sigma}}(v) = 0, \forall v \notin X$. Now, denote as \bar{x} the vector of boolean values such that for $i = 1, \dots, n$, $\bar{x}_i = \text{TRUE}$ if $f_{\bar{\sigma}}(X_i) = 1$, and $\bar{x}_i = \text{FALSE}$ if $f_{\bar{\sigma}}(X_{i+n}) = 1$. We show that \bar{x} is a feasible solution of the instance of 3-SAT.

We assume by contradiction that there is a clause c_j that is not satisfied by \bar{x} , i.e., $\bar{x}_{j_k} = \text{FALSE}$ for $k = 1, 2, 3$. For $k = 1, 2, 3$, we then have $f_{\bar{\sigma}}(X_{j_k}) = 0$ by definition of \bar{x} . Arguments similar to those used in the proof of Lemma 3 yield

$$\bar{\sigma}(Y_{j,k}) > \bar{\sigma}(X_{j_k}) > \bar{\sigma}(B_{j_k}^0) > \dots > \bar{\sigma}(C_j).$$

This means in particular that C_j has only one reference, which is in contradiction with the definition of $\bar{\sigma}$. We conclude that (c, x) is a YES instance of 3-SAT. \square

One can observe that the constraints $|R_\sigma(v)| \geq 1$ for all $v \neq O$ did not intervene anywhere in the proofs of Theorem 3 and Lemma 3. Actually, it is automatically satisfied for all neighbors of O , and $|R_\sigma(v)| \geq 2$ for every other vertex v if $F_{1,2}(\sigma) = n$. Moreover, if σ is feasible for SLVO(1,2) it is of course feasible for SLVO(0,2) and $F_{0,2}(\sigma) = F_{1,2}(\sigma) + 2$ (because $f_\sigma(O) = 2$ if $K = 0$). This means that the proof of Theorem 3 could be immediately adapted to show that SLVO(0,2) is NP-complete by showing that (c, x) is a YES instance if and only if there is a vertex order of G, σ , such that $\sigma(O) = 1$ and $F_{0,2}(\sigma) = n + 2$.

Theorem 4. SLVO(0,2) is NP-complete.

We conclude this complexity study by summing up the results in Table 1.

$K \backslash U-K$	0	1	2	≥ 3
0	P	P	NP	NP
1	P	NP	NP	NP
2	P	NP	NP	NP
≥ 3	P	NP	NP	NP

Table 1: Complexity of $\text{slvo}(K,U)$.

Since MDOP for fixed K is equivalent to $\text{slvo}(K,K+1)$, we immediately deduce the following.

Corollary 1. *If K is fixed, MDOP is in P if $K = 0$ and it is NP-complete for all $K \geq 1$.*

IV. PERSPECTIVES

Now that we have established that $\text{slvo}(K,U)$ is NP-complete for every interesting value of K and U , the next step should be in a thorough study of its approximability. In this perspective, it is interesting that in both [3] and [12], the authors have noticed that the greedy method described in Algorithm 1 performed very well on MDOP with K fixed to 3. Although this might be a lead for the types of instances they used, we can show that, in general, it does not even approximate the optimal solution of $\text{slvo}(1,2)$ (i.e., MDOP with fixed $K = 1$) within a constant factor. This is shown by the following example.

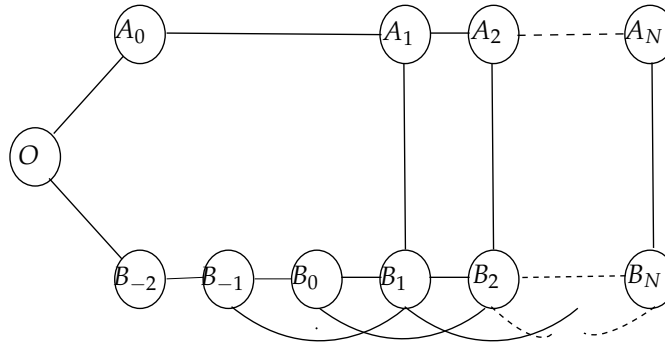


Figure 3: Counter-example of approximation within a constant factor for the greedy algorithm.

Example 1. Consider the instance illustrated in Figure 3, which is composed of $2N + 5$ vertices: one root O , and two branches $\{A_0, A_1, \dots, A_N\}$ and $\{B_{-2}, B_{-1}, \dots, B_N\}$. As in the previous section, we could add a gadget rooted in O to make sure that any solution where O is not ordered before the two branches will cost more than N , so we set $\sigma(O) = 1$.

The optimal value of this instance is 4, and it can be obtained with the discretization order

$$(O, B_{-2}, B_{-1}, B_0, \dots, B_N, A_0, \dots, A_N).$$

Indeed, in this order, B_{-2}, B_{-1}, B_0 , and A_0 have one reference, and the other vertices (except O) have two. After setting $\sigma(O) = 1$, Algorithm 1 picks one vertex among those with most references, i.e., either A_0 or

B_{-2} . Assuming that it repeatedly picks a vertex in the A branch, the algorithm builds a discretization order where the first $N + 1$ vertices (A_0, \dots, A_N) have only one reference. Finally, the approximation ratio of the algorithm is $(N + 1)/4$ for this instance.

Another finding is that different values of K could lead to different approximability results for $\text{slvo}(K, U)$. Indeed, we provide in Appendix A a proof that $\text{slvo}(K, U)$ is **NP**-complete for all $K \geq 3$, $U \geq K + 1$, by reduction from minimum vertex cover (**vc**) in bounded degree graphs. From an instance, G_{VC} , of **vc**, the proof constructs an instance G of $\text{slvo}(K, K+1)$, where K is the maximum edge degree in G_{VC} . We then show that there is a vertex cover of G_{VC} with size N if and only if there is a discretization order of G with value N . In particular, this shows that the best approximation ratio that can be achieved for $\text{slvo}(K, K+1)$ is at best that achieved for **vc** in bounded degree graphs (see e.g., [5] for such results). However, this reduction is not valid for $U \leq 3$. Another similar reduction from the **TRIANGLE PACKING** problem [2] is still valid for $K = 2$ and $U = 3$, but not for smaller values of U . This leaves the possibility that better approximation can be found for $U = 2$.

Finally, there is still much to be done in the practical solution of $\text{slvo}(K, U)$, since recent studies still fail in the search for optimal solutions of $\text{slvo}(K, U)$ for graphs with as few as 100 vertices [1, 12]. We also hope that we will soon see some other real applications of $\text{slvo}(K, U)$ than the discretization of DGPs.

REFERENCES

- [1] Andrea Cassioli, Oktay Günlük, Carlile Lavor, and Leo Liberti. “Discretization vertex orders in distance geometry”. In: *Discrete Applied Mathematics* 197 (2015), pp. 27–41.
- [2] Michael R. Garey, David S. Johnson, and Larry J. Stockmeyer. “Some simplified NP-complete graph problems”. In: *Theoretical Computer Science* 1.3 (1976), pp. 237–267.
- [3] Douglas S. Gonçalves and Antonio Mucherino. “Optimal partial discretization orders for discretizable distance geometry”. In: *International Transactions in Operational Research* 23.5 (2016), pp. 947–967.
- [4] Venkatesan Guruswami, C. Pandu Rangan, M. S. Chang, G. J. Chang, and C. K. Wong. “The Vertex-Disjoint Triangles Problem”. In: *Graph-Theoretic Concepts in Computer Science*. Ed. by Juraj Hromkovič and Ondrej Šýkora. Springer Berlin Heidelberg, 1998, pp. 26–37.
- [5] Eran Halperin. “Improved Approximation Algorithms for the Vertex Cover Problem in Graphs and Hypergraphs”. In: *SIAM Journal on Computing* 31.5 (Jan. 2002), pp. 1608–1623.
- [6] Bruce Hendrickson. “The Molecule Problem: Exploiting Structure in Global Optimization”. In: *SIAM Journal on Optimization* 5.4 (Nov. 1995), pp. 835–857.
- [7] Richard M. Karp. “Reducibility among Combinatorial Problems”. In: *Complexity of Computer Computations: Proceedings of a symposium on the Complexity of Computer Computations*. Ed. by Raymond E. Miller, James W. Thatcher, and Jean D. Bohlinger. Boston, MA: Springer US, 1972, pp. 85–103.
- [8] Carlile Lavor, Jon Lee, Audrey Lee-St. John, Leo Liberti, Antonio Mucherino, and Maxim Sviridenko. “Discretization orders for distance geometry problems”. In: *Optimization Letters* 6.4 (2012), pp. 783–796.
- [9] Carlile Lavor, Leo Liberti, Nelson Maculan, and Antonio Mucherino. “The discretizable molecular distance geometry problem”. In: *Computational Optimization and Applications* 52.1 (2012), pp. 115–146.

- [10] Leo Liberti and Carlile Lavor. *Euclidean distance geometry: an introduction*. Springer, 2017.
- [11] Leo Liberti, Carlile Lavor, and Nelson Maculan. “A Branch-and-Prune algorithm for the Molecular Distance Geometry Problem”. In: *International Transactions in Operational Research* 15.1 (2008), pp. 1–17.
- [12] Jérémy Omer and Douglas S. Gonçalves. “An integer programming approach for the search of discretization orders in distance geometry problems”. In: *Optimization Letters* (2017).
- [13] James B. Saxe. “Embeddability of weighted graphs in K -space is strongly NP-hard”. In: *Proceedings of 17th Allerton Conference in Communications, Control and Computing*. Monticello, IL, 1979, pp. 480–489.

A. ALTERNATIVE REDUCTION FOR $K \geq 3$

In this section, we prove that $\text{slvo}(k,u)$ is **NP**-complete for all $K \geq 3, U = K + 1$ by reduction from the minimum vertex cover problem (vc) in graphs with vertex degrees bounded by K .

Alternative proof of NP-completeness of $\text{slvo}(k,u)$ for $K \geq 3$ and $U = K + 1$. We show the result by polynomial reduction from vc in graphs with bounded degrees, which is **NP**-complete for any maximum degree $\Delta \geq 3$ [2].

Let $G_{VC} = (V, E)$ and $N \leq |V|$ define an arbitrary instance of vc with vertex degrees bounded by $\Delta \geq 3$. We then set $K = \Delta, U = K + 1$ and construct the corresponding instance (G, N) of $\text{slvo}(k,u)$. To avoid confusion in the remainder, we index with VC the quantities that refer to G_{VC} (e.g. $d_{VC}(i)$ or $\delta_{VC}(i)$), and we do not index those referring to G . The vertices of G are given by the union of four sets of vertices $C \cup V_V \cup V_E \cup V^i$, where

1. $C = \{c_1, \dots, c_K\}$ is a clique of G ;
2. $V_V = \{v_i : i \in V\}$ and for all $i \in V, d_{VC}(i)$ edges connect v_i to the $K + 1 - d_{VC}(i)$ first vertices of C ;
3. $V_E = \{v_{i,j} : \{i, j\} \in E\}$ and for all $\{i, j\} \in E, K$ edges connect $v_{i,j}$ to the vertices of C , and two edges connect $v_{i,j}$ to v_i and v_j ($\in V_V$) as illustrated in Figure 4;

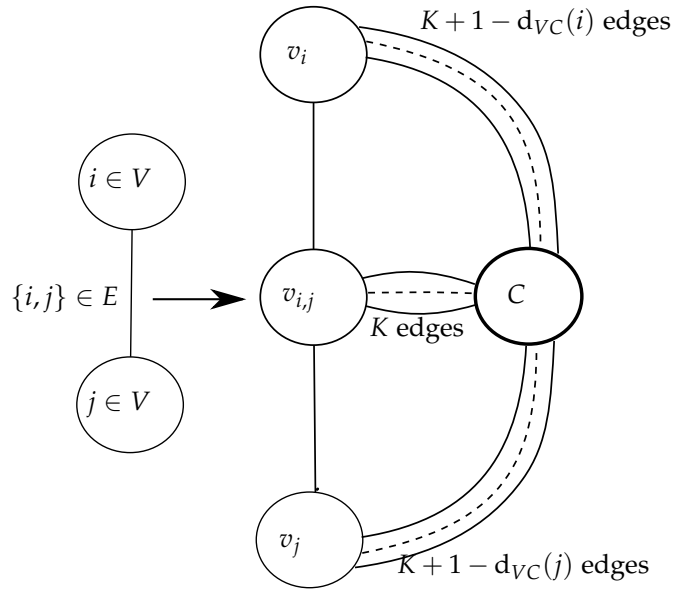


Figure 4: Construction of V_V and V_E from V and E .

4. for all $i \in V$, the gadget $V^i = \{g_1^i, \dots, g_{d_{VC}(i)}^i\}$ has $d_{VC}(i)$ vertices. Each vertex $g_k^i \in V^i$ is connected to $v_i \in V_V$, to the K vertices of C , to g_{k-1}^i if $k \neq 1$ and to g_{k+1}^i if $k \neq d_{VC}(i)$. The gadget is illustrated in Figure 5.

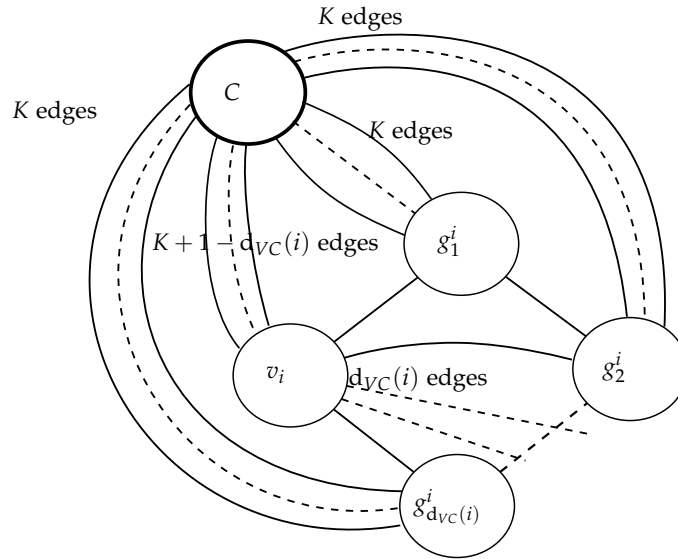


Figure 5: Illustration of the gadget V_i for $i \in V$.

Stated otherwise, the vertices of C induce the K -clique which will come first in the discretization order. The vertices of V_V and V_E correspond to the vertices and edges of G_{VC} , and the gadget V^i will guarantee the validity of the reduction. It is straightforward that the transformation from G_{VC} to G is polynomial. We will prove that a vertex cover of G_{VC} with cardinality N exists if and

only if a discretization order of G with cost N exists, but we start with preliminary remarks about G .

First, observe that C is a clique of G so that it can be set at the beginning of a discretization order of G . Moreover, the first $K + 1$ vertices of a discretization order must form a clique. If there is no vertex of V_V in this clique, it only includes vertices that are neighbors to every vertex of C . This means that the vertices of C can be set at the beginning of the order without modifying its cost. Otherwise, the $K + 1$ -clique contains at most one vertex $v_i \in V_V$, because there is no edge with both ends in V_V . This vertex can always be set $K + 1$ -th in the order without modifying its cost, so the vertices of C can be set at the beginning of the order if there is no other vertex in the clique. By construction of V_E and V^i there can be either one vertex $v_{i,j} \in V_E$ or at most two vertices g_k^i and g_{k+1}^i of V^i in the $K + 1$ -clique. In both cases, we can modify the vertex order to obtain

$$\begin{aligned}\sigma(c_k) &= k, \forall k \in \{1, \dots, K\}, \\ \sigma(g_k^i) &= K + k, \forall k \in \{1, \dots, d_{VC}(i)\}, \\ \sigma(v_i) &= K + d_{VC}(i) + 1.\end{aligned}$$

If there is also some $v_{i,j} \in V_E$ in the $K + 1$ -clique, we also set $\sigma(v_{i,j}) = K + d_{VC}(i) + 2$. The first $K + 1$ vertices in σ still form a clique and the $d_{VC}(i) + 1$ following vertices all have $K + 1$ references. If we keep the relative order of the other vertices unchanged, this modification can only increase their numbers of references, so the total cost of the order does not increase. As a consequence, we can focus on discretization orders starting with C without loss of generality.

Secondly, a vertex $v_{i,j} \in V_E$ is connected to the K vertices of C and to the two vertices $v_i, v_j \in V_V$. As a consequence, its cost will vanish if and only if it comes after v_i or v_j in the order. Similarly, the costs of every vertex of V^i will vanish if and only if they all come after v_i in the order. And reciprocally, the cost of $v_i \in V_V$ vanishes if it comes after every vertex $v_{i,j} \in V_E$ or after every vertex of V^i .

Using the above preliminary remarks, we show that a solution, σ , of $\text{slvo}(K,U)$ with cost $F_{K,K+1}(\sigma) = N$ can be built from a vertex cover, $I \subset V$, of G_{VC} with cardinality N . For this we set the orders of the vertices from the beginning to the end as follows.

1. $\sigma(c_k) = k, \forall k \in \{1, \dots, K\}$, then
2. $\forall i \in I : \sigma(g_1^i) \leq \dots \leq \sigma(g_{d_{VC}(i)}^i) \leq \sigma(v_i)$, then
3. $\forall \{i, j\} \in E : \sigma(v_{i,j}) \geq \sigma(v_k), \forall k \in I$, then
4. $\forall i \in V \setminus I : \sigma(v_i) \geq \sigma(v_{i,j}), \forall \{i, j\} \in E$, then
5. $\forall i \in V \setminus I, \forall j \in V^i : \sigma(g_{d_{VC}(i)}^i) \geq \dots \geq \sigma(g_1^i) \geq \sigma(v_i)$.

By Item 2., for all $i \in I$, g_1^i has K references, but v_i and every other vertex of V^i have $K + 1$ references. Item 3. sets the position of the vertices of V_E only after those of $\{v_i : i \in I\}$. Since I is a vertex cover of G_{VC} this guarantees that every vertex of V_E has at least $K + 1$ references. According to Items 4. and 5., the remaining vertices of V_V then come after all the vertices of V_E and each vertex of V^i comes after v_i for all $i \in V \setminus I$. As a consequence, these vertices also have a vanishing cost. Finally, this means that the only vertices with non-vanishing costs are $\{g_1^i : i \in I\}$, hence $F_{K,K+1}(\sigma) = |I| = N$.

Reciprocally, let $\tilde{\sigma}$ be a discretization order of G such that $\tilde{\sigma}(V_1) = \{1, \dots, K\}$ and $F_{K,K+1}(\tilde{\sigma}) = N$. We have already seen that we can consider that $\tilde{\sigma}(C) = \{1, \dots, K\}$ without loss of generality.

We then modify $\tilde{\sigma}$ to build another discretization order of G , σ , such that the only vertices with non-vanishing costs are $\{g_1^i : i \in I\}$ for some set I of size at most N .

Let $v \notin \{g_1^i : i \in V\}$ such that $f_{\tilde{\sigma}}(v) = 1$. Then either, $v \in V_V$, $v \in V_E$ or $v \in V^i \setminus \{g_1^i\}$ for $i \in V$.

- If $v = v_i \in V_V$, we can move vertices of V^i down in the order so that $\sigma(g_1^i) \leq \dots \leq \sigma(g_{d_{VC}^i}^i) \leq \sigma(v_i)$. We get $f_{\sigma}(g_1^i) = 1$, $f_{\sigma}(g_k^i) = 0, \forall k \geq 2$ and $f_{\sigma}(v_i) = 0$.
- If there exists $i \in V$ such that $v = g_k^i \in V^i, k \neq 1$, we can perform similar changes so that $\sigma(g_1^i) \leq \dots \leq \sigma(g_{d_{VC}^i}^i) \leq \sigma(v_i)$.
- If $v = v_{i,j} \in E$, we can also set v_i and some vertices of V^i to lower position in the order so that $\sigma(g_1^i) \leq \dots \leq \sigma(g_{d_{VC}^i}^i) \leq \sigma(v_i) \leq \sigma(v_{i,j})$. Once again, g_1^i becomes the only vertex with non-vanishing cost among them.

In all the above modifications, we only set vertices to lower positions in the order while leaving the relative orders of the remaining vertices unchanged. This means that only the number of references of the vertices that have been moved can decrease. But, in every case there is at least one vertex with non-vanishing cost among them before the modifications and exactly one after. As a consequence, we get $F_{K,K+1}(\sigma) = |I| \leq N$, where $I = \{i \in V : f_{\sigma}(g_1^i) = 1\}$.

Finally, $\forall i \in V \setminus I$ and $\forall v \in \{v_i\} \cup V^i$, $f_{\sigma}(v) = 0$, so $\sigma(v_i) \geq \sigma(v_{i,j}), \forall \{i,j\} \in E$. As a consequence, every vertex of V_E has at least one reference among the vertices of $\{v_i : i \in I\}$, which means that I is a vertex cover of G_{VC} of size at most N . \square

Remark 2. *In the same vein as the reduction from vc , we can also prove that $slvo(2,3)$ is **NP**-complete by a reduction from **TRIANGLE PACKING**. This problem has been proved to be **NP**-complete for graphs with a 3-clique in [2] and later studied for several classes of graphs in [4].*