International Journal of Mathematics and
Computer Science, **15**(2020), no. 1, 155–181

On the complexity of the assignment problem with ordinal data

Soufiane Drissi Oudghiri, Mohamed Hachimi

Laboratory of Engineering Sciences (LabSi)
Ibn Zohr University
BP 32/S, Agadir, Morocco

email: soufiane.drissio@gmail.com, m.hachimi@uiz.ac.ma

(Received August 10, 2019, Accepted October 4, 2019)

Abstract

We consider a variant of the assignment problem where entries of the matrix belong to an ordinal scale $\mathbb{L}$. The objective is to find the set of optimal assignments according to a particular dominance criterion which is relevant in the context of ordinal data. This paper aims is to explore the frontier between easy and hard cases depending on $|\mathbb{L}|$, the size of the ordinal scale. We give a polynomial time algorithm that solve the problem for $|\mathbb{L}| = 3$. When $|\mathbb{L}|$ is polynomial in the size of the instance we show that the decision version of the problem is NP-complete.

Key words and phrases: Assignment problem, Combinatorial optimization, Computational complexity, Polynomial algorithms, Ordinal data.
AMS (MOS) Subject Classifications: 90C27.
ISSN 1814-0432, 2020, http://ijmcs.future-in-tech.net

1 Introduction

1.1 Motivation

Ordinal scales occur in different disciplines such as social sciences and decision theory. Examples of ordinal variables and their ordinal scales (in parentheses) are opinion about government spending on the environment (too high, about right, too low) and severity of an injury in an automobile crash (uninjured, mild injury, moderate injury, severe injury, death). For ordinal scales, there is a clear ordering of the levels, but unlike numerical scales, the absolute distances among them are unknown. Pain measured with the scale (none, mild, discomforting, distressing, intense, excruciating) is ordinal, because a person who chooses the level *mild* feels more pain than if he or she chose the level *none*, but no numerical measure is given of the difference between those levels.[1]

Combinatorial optimization problems involving ordinal data arise naturally in a number of real-world settings. For instance, let us assume we have a number of newly hired teachers that need to be assigned to the same number of teaching positions. In order to choose an assignment which satisfies teachers as much as possible, each teacher is asked to evaluate the teaching positions on the following ordinal scale (high preference, medium preference, low preference). In combinatorial optimization, this situation can be modeled as an *assignment problem* where entries of the matrix are ordinal.

Let us consider a second example which illustrates a situation familiar to researchers. In order to organize a refereed conference, each submitted article must be assigned to a number of reviewers. To simplify this process, conference management software are usually used. Generally, conference management software organize this review process in two steps. First, the system asks the reviewers to declare conflicts of interests and to evaluate the papers (for which the reviewer has no conflict of interest) on some ordinal scale like the following (high interest, medium interest, low interest). Then, based on this ordinal data, the system automatically assigns a number of reviewers to each submitted article. This problem can be modeled as a *many to many allocation problem* where entries of the matrix

[1]this paragraph is largely adapted from [4], ch.1; for more insight on the theory of scales of measurement see [7].

are ordinal (for more on this problem see [1, 14]).

Providing a more precise numerical measure describing ordinal data is not always possible. In particular situations, one can show that the ordinal scale possesses some properties which allow to legitimately consider it as a numerical scale. However, sometimes, the transition from ordinal to numerical scale is purely done for the purpose of simplification. But does this simplification have any undesirable consequence? Yes, indeed, Roberts [15] shows that it may lead to meaningless conclusions (this is illustrated in example 4.1 in the appendix). Consequently, we need algorithms specifically designed for combinatorial problems with ordinal data.

1.2 Definition of the problem

This subsection is organized in the following way: first, we formalize notions of ordinal scale and ordinal data. Then, we define the instance of what we call the *ordinal assignment problem*. Next, we introduce the criterion we use to compare assignments, namely the *ordinal dominance criterion*. After that, we define the set of solutions we aim to find, namely the *minimal complete set of solutions*. Finally, we state the problem object of this paper.

As stated in the previous subsection, the only information available about the levels of an ordinal scale is their respective order. Therefore, we can formalize the notion of ordinal scale as a finite set endowed with a total order. Throughout the paper, $\mathbb{L}$ denotes an ordinal scale and $\rhd$ denotes the total order over the levels of $\mathbb{L}$. These latter are denoted, for practical reasons, as follows: $1, 2, \ldots, C$ such that $1 \rhd 2 \rhd \ldots \rhd C$. Note that $|\mathbb{L}| = C$.
For instance, the scale (high preference, medium preference, low preference) will be denoted $L = \{1, 2, 3\}$ where 1 denotes the level *high preference*, 2 denotes *medium preference* and 3 denotes *low preference*. The order $\rhd$ means here *strictly preferred to* (e.g. $1 \rhd 2$ means: the level high preference is strictly preferred to medium preference).

We study in this paper what we call the *ordinal assignment problem*. To define the instance of the ordinal assignment problem, we first recall the following definitions from graph theory.

Definition 1. *Let $G = (V, E)$ be a graph, where V is the set of vertices and E is*

the set of edges.
1. A bipartite graph $G = (V, E)$ is a graph where V is a union of two disjoint sets A and T such that every edge connects a vertex in A to one in T.
2. A bipartite graph G is said to be complete if every vertex of A is connected to every vertex of T.
3. An assignment S is a subset of E such that each vertex in V is incident to exactly one edge in S.

An instance of the ordinal assignment problem consists of a complete bipartite graph $G = (V, E)$ such that $|A| = |T| = n$ and a function $u : E \to \mathbb{L}$. We denote $A = \{a_1, \cdots, a_n\}$ and $T = \{t_1, \cdots, t_n\}$. For the sake of simplicity, we will most of the time write u_{ij} instead of $u(a_i, t_j)$.

Example 1.1. *Consider a situation where four teachers should be assigned to four positions. Each teacher expresses his preference on the following ordinal scale: (high preference, medium preference, low preference). This situation can be seen as an instance of the ordinal assignment problem where: $|A| = |T| = 4$ and $\mathbb{L} = \{1, 2, 3\}$.*
Let the entry matrix encoding the function $u : E \to \mathbb{L}$ be as follows:

$$\begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{array} \begin{array}{c} \begin{array}{cccc} t_1 & t_2 & t_3 & t_4 \end{array} \\ \begin{pmatrix} 1 & 2 & 1 & 3 \\ 2 & 2 & 2 & 1 \\ 1 & 1 & 3 & 3 \\ 2 & 3 & 1 & 3 \end{pmatrix} \end{array}.$$

Let Q and R be two assignments such that $Q = \{(a_1, t_1), (a_2, t_2), (a_3, t_3), (a_4, t_4)\}$ and $R = \{(a_1, t_4), (a_2, t_1), (a_3, t_2), (a_4, t_3)\}$. They can be visualized as follows:

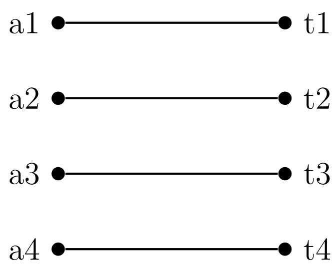

Figure 1: Assignment Q

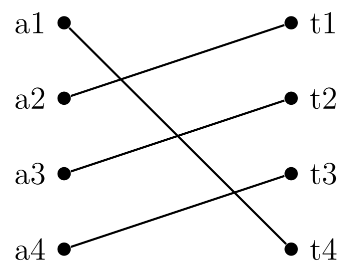

Figure 2: Assignment R

Now, in order to introduce the *ordinal dominance criterion*, the following notations are needed. Let $[i] := \{1, ..., i\}$ and $y \trianglerighteq y' \Leftrightarrow y \triangleright y'$ or $y = y'$. We associate to an assignment S the vector $x(S) := (x^1_S, \ldots, x^n_S)$, where $x^i_S = j$ if $(a_i, t_j) \in S$. In addition, we denote the value of an assignment S by the vector $v(S) := (u_{1x^1_S}, \ldots, u_{nx^n_S})$. For instance, in example 1.1 we have $x(Q) = (1, 2, 3, 4)$, $v(Q) = (1, 2, 3, 3)$ and $x(R) = (4, 1, 2, 3)$, $v(R) = (3, 2, 1, 1)$.

Definition 2. *Let S and S' be two assignments and let $y = v(S)$, $y' = v(S')$. With respect to the ordinal dominance criterion, we say that S dominates S' (resp. S strictly dominates S') denoted by $S \succeq S'$ (resp. $S \succ S'$), if there exists a permutation $\pi : [n] \to [n]$ such that $\forall i \in [n]$ $y_i \trianglerighteq y'_{\pi(i)}$ (resp. $\forall i \in [n]$ $y_i \trianglerighteq y'_{\pi(i)}$ and $\exists$ $j \in [n]$ $y_j \triangleright y'_{\pi(j)}$).*

In terms of example 1.1, the ordinal dominance criterion reads as follows: an assignment S dominates S' if there exists a permutation π from $[n]$ to $[n]$ such that every teacher a_i is at least as satisfied with S as teacher $a_{\pi(i)}$ with S'. The intuition behind the ordinal dominance criterion is that it may be desirable to hurt one teacher if that helps significantly the others. Coming back to the example 1.1, let $y = v(R)$ and $y' = v(Q)$, we detect that $R \succ Q$ by setting $\pi(1) = 3$ ($y_1 = 3 \trianglerighteq 3 = y'_3$), $\pi(2) = 2$ ($y_2 = 2 \trianglerighteq 2 = y'_2$), $\pi(3) = 1$ ($y_3 = 1 \trianglerighteq 1 = y'_1$) and $\pi(4) = 4$ ($y_4 = 1 \triangleright 3 = y'_4$).

Since the ordinal dominance criterion induces only a partial order on solutions, then we can distinguish different relations between two solutions.

Definition 3. *1. Two assignments S an S' are called equivalent if $S \succeq S'$ and $S' \succeq S$.*

2. Two assignments S and S' are called incomparable if neither $S \succeq S'$ nor $S' \succeq S$, we denote $S \sim S'$.
3. An assignment S is called optimal if there is no other assignment S' such that $S' \succ S$.

Definition 4. *([3]) 1. A complete set M is a set of solutions which verifies that for any optimal solution $S \notin M$, there is a solution $S' \in M$ such that S and S' are equivalent.*
2. A minimal set M is a set of solutions such that all elements of M are pairwise incomparable.
3. A minimal complete set is a set of solutions which is both complete and minimal.

Finally, we can clearly state the problem object of this paper.
Given a complete bipartite graph $G = (V, E)$ such that $|A| = |T| = n$ and a function $u : E \to \mathbb{L}$, the *ordinal assignment problem* designates the problem of finding a minimal complete set of solutions, such that a solution is optimal in the sense of the ordinal dominance criterion.

In the remainder of the article, the ordinal assignment problem may be denoted as OAP. If $|\mathbb{L}|$ is *fixed* (recall that $|\mathbb{L}| = C$), we will denote the problem as C-OAP.

1.3 From Ordinal dominance to Pareto dominance

Delort et al. [13] establish an original link between the ordinal dominance criterion and the Pareto dominance criterion. Before introducing their result, some definitions are needed.

For each solution S of an ordinal optimization problem and each level $l \in \mathbb{L}$, we define $S_l := \{s \in S : u(s) \trianglerighteq l\}$ (e.g. in the assignment setting: $S_l = \{(a_i, t_j) \in S : u_{ij} \trianglerighteq l\}$). Moreover, we define the *cumulative vector* of a solution S as $Cum(S) := (|S_1|, \ldots, |S_C|)$. For instance, in example 1.1, we have $Q_1 = \{(a_1, t_1)\}$, $Q_2 = \{(a_1, t_1), (a_2, t_2)\}$, $Q_3 = \{(a_1, t_1), (a_2, t_2), (a_3, t_3), (a_4, t_4)\}$ and $Cum(Q) = (1, 2, 4)$. Note that for any solution S, we necessarily have $S_C = S$. Since in the assignment problem all solutions have the same size n (recall that $|A| = |T| = n$), then for any assignment S we have $|S_C| = n$.

Definition 5. *Let $f, f' \in \mathbb{N}^k$. With respect to the weak Pareto dominance we say that:*

- *f dominates f', denoted by $f \geq f'$, if $\forall i \in [k]\ f_i \geq f'_i$.*
- *f strictly dominates f', denoted by $f > f'$, if $\forall i \in [k]\ f_i \geq f'_i$ and $\exists j \in [k]\ f_j > f'_j$.*

Interestingly enough, Delort et al. [13] show that comparing solutions according to the ordinal dominance criterion amounts to compare cumulative vectors according to the weak Pareto dominance.

Proposition 1. ([13]) *Let S and S' be two solutions of any ordinal optimization problem, the following holds:*

$$S \succeq S' \Leftrightarrow Cum(S) \geq Cum(S')$$

This result will prove useful throughout the article because we will be able to compare assignments easily. Indeed, comparing $Cum(S)$ and $Cum(S')$ according to the weak Pareto dominance criterion is much easier than comparing S and S' according to the ordinal dominance criterion. Furthermore, from definition 3 and proposition 1 we obtain:
- Two assignments S an S' are *equivalent* if $Cum(S) = Cum(S')$.
- Two assignments S and S' are *incomparable* if neither $Cum(S) \geq Cum(S')$ nor $Cum(S') \geq Cum(S)$.
- An assignment S is *optimal* if there is no other assignment S' such that $Cum(S') > Cum(S)$.

Before concluding this subsection let us discuss briefly the solution of 2-OAP. Recall that to each solution S of a 2-OAP instance corresponds a cumulative vector $Cum(S) = (|S_1|, |S_2|)$. As already noted, the last component is always equal to n ($|S_C| = n$). Thus, the minimal complete set of a 2-OAP instance contains a unique solution O which verifies $|O_1| \geq |S_1|$ for all solutions S of the instance. Therefore, O is an assignment which maximizes the number of edges of level 1. As a result, 2-OAP can be reduced to a maximum matching problem in a bipartite graph. Simply ignore all edges of level 2, then find a maximum matching in the graph. Finally, connect randomly the unmatched vertices in A to the unmatched vertices in T.

1.4 Related work

In the combinatorial optimization literature, problems involving ordinal data can be distinguished in two categories.

- *Problems where solutions are totally ordered.* These problems have generally objective functions that match any solution to some value in a totally ordered set. For instance, the *max* (resp. *min*) function consists in ranking solutions according to the best (resp. the worst) element, which corresponds in terms of example 1.1 to rank assignments according to the teacher whose satisfaction is the best (resp. the worst) among all teachers. The *k-max* (resp. *k-min*) function consists in ranking solutions according to their k^{th} largest (resp. smallest) element. Finally, the *leximax* (resp. *leximin*) relation is an enrichment of the max (resp. min) that consists in breaking ties by going down the ranking. If the best (resp. worst) elements of many solutions are equal, one compares the second bests (resp. worsts), and so on...

Finding an optimal assignment according to the mentioned objective functions (*max*, *min*, *k-max*, *k-min*, *leximax*, *leximin*) can be performed in polynomial time [12, 2, 8, 17]. Recently, the problem of computing an assignment which optimizes the *Sugeno integral* has been studied [5]. Sugeno integral is a sophisticated function which subsumes the functions: *max*, *min*, *k-max*, *k-min* and many others.

- *Problems where solutions are only partially ordered.* In these problems, we generally compare solutions according to some dominance criterion. A popular one in the context of ordinal data is the ordinal dominance criterion, which we have defined in the assignment setting (definition 2). To our knowledge, the ordinal dominance criterion was first introduced by Bartee [6]. Afterwards, it was adopted in many combinatorial optimization problems such as committee selection [13], minimal path [11], spanning tree [16] and other problems [9, 10].

2 A polynomial algorithm for 3-OAP

In this section, we will study some properties of 3-OAP and show that it is solvable in polynomial time. Let us first look at the size of the minimal complete set M when $|L| = 3$. The following lemma will be useful for establishing a tight bound on $|M|$.

Lemma 1. *Let M be a minimal complete set of a 3-OAP instance. If $S, S' \in M$ then* $|S_1| \neq |S'_1|$ and $|S_2| \neq |S'_2|$.

Proof. We first show that $S, S' \in M \Rightarrow |S_1| \neq |S'_1|$. Let us assume the contrary, that is $|S_1| = |S'_1|$. First, we recall that $|S_3| = |S'_3| = n$. Without loss of generality we assume that $|S_2| \geq |S'_2|$. Therefore, either $|S_2| = |S'_2|$ and thus $Cum(S) = Cum(S')$, which implies S and S' are equivalent. Or $|S_2| > |S'_2|$ and thus $Cum(S) > Cum(S')$, which implies that $S \succ S'$. Consequently, in both cases M is not minimal, a contradiction. By exactly the same reasoning, one shows that $S, S' \in M \Rightarrow |S_2| \neq |S'_2|$. □

Proposition 2. *If M is a minimal complete set of a 3-OAP instance, then $|M| \leq \lfloor n/2 \rfloor + 1$, and the bound is tight.*

Proof. Suppose, to the contrary, that there exists a minimal complete set of a 3-OAP instance such that $|M| > \lfloor n/2 \rfloor + 1$. We will derive a contradiction. Let P and Q be two solutions in M such that: $|P_1| = min(\{|S_1| : S \in M\})$ and $|Q_1| = max(\{|S_1| : S \in M\})$. Therefore $\forall S \in M,\ |S_1| \in [|P_1|, |Q_1|]$. From lemma 1, we have: $S, S' \in M \Rightarrow |S_1| \neq |S'_1|$, and since $|M| > \lfloor n/2 \rfloor + 1$ then the interval $[|P_1|, |Q_1|]$ contains at least $\lfloor n/2 \rfloor + 2$ distinct integers. Thus $|Q_1| - |P_1| + 1 \geq \lfloor n/2 \rfloor + 2$, equivalently $|Q_1| \geq \lfloor n/2 \rfloor + |P_1| + 1$ (i).
Now, let us prove that $|P_2| = max(\{|S_2| : S \in M\})$. Suppose the contrary, which means that $\exists\, R \in M$ such that $|R_2| > |P_2|$. Since $|P_1| = min(\{|S_1| : S \in M\})$, then $|R_1| \geq |P_1|$. It follows that $R \succ P$ and hence M is not minimal, a contradiction. By the same reasoning one proves that $|Q_2| = min(\{|S_2| : S \in M\})$. As a result, $\forall S \in M,\ |S_2| \in [|Q_2|, |P_2|]$. From lemma 1, we have: $S, S' \in M \Rightarrow |S_2| \neq |S'_2|$, and since $|M| > \lfloor n/2 \rfloor + 1$ then the interval $[|Q_2|, |P_2|]$ contains at least $\lfloor n/2 \rfloor + 2$ distinct integers. Thus we obtain: $|P_2| - \lfloor n/2 \rfloor - 1 \geq |Q_2|$ (ii). Since by definition $|Q_2| \geq |Q_1|$, then from (i) and (ii) we obtain: $|P_2| - \lfloor n/2 \rfloor - 1 \geq \lfloor n/2 \rfloor + |P_1| + 1$. Thus, $|P_2| \geq n + |P_1| + 1$ which implies: $|P_1| < 0$ or $|P_2| > n$, a contradiction.

In order to prove tightness of the given bound, consider the instance shown in figure 3 (n is assumed even). Note that if we form an assignment S by selecting the edges (a_i, t_{i+1}), (a_{i+1}, t_i) $i \in \{1, 3, \ldots, n-1\}$ exactly k times ($k \in \{1, 2, \ldots, n/2\}$) and selecting the edges (a_i, t_i), (a_{i+1}, t_{i+1}) exactly $n/2 - k$ times, we will obtain $Cum(S) = (k, n-k, n)$. Consequently, for any minimal complete solution M of the instance, we have $F = \{Cum(S) : S \in M\} = \{(0, n, n), (1, n-1, n), (2, n-2, n), \ldots, (n/2, n/2, n)\}$. □

$$\begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ \vdots \\ a_{n-1} \\ a_n \end{array} \begin{array}{c} \begin{array}{ccccccc} t_1 & t_2 & t_3 & t_4 & \dots & t_{n-1} & t_n \end{array} \\ \begin{pmatrix} 2 & 1 & & & & & \\ 3 & 2 & & & & & \\ & & 2 & 1 & & & \\ & & 3 & 2 & & & \\ & & & & \ddots & & \\ & & & & & 2 & 1 \\ & & & & & 3 & 2 \end{pmatrix} \end{array}$$

Figure 3: An instance of 3-OAP where $|M| = n/2 + 1$. All empty entries are 3.

For the remainder of this section, we will need the following definitions and notations.

- Given a 3-OAP instance, we denote by $\overline{r}$ the maximum number of edges of level 2 or less a solution may have. Formally, if M is any minimal complete set of the 3-OAP instance then $\overline{r} = max(\{|S_2| : S \in M\})$.

- A sub-assignment U of degree k, denoted U^k, is a subset of an assignment S such that $|U^k| = |U_2^k| = k$, where $U_l^k = \{(a_i, t_j) \in U^k : u_{ij} \trianglerighteq l\}$. Note that by definition no edge of level 3 belongs to U^k. Also, since $|U^k| = |U_2^k| = k$ then by definition of $\overline{r}$ we have $k \leq \overline{r}$.

We extend the notion of optimality to sub-assignments as follows:

- The cumulative vector of a sub-assignment U^k is defined as follows: $Cum(U^k) = (|U_1^k|, |U_2^k|)$.

- A sub-assignment V^k dominates U^q, denoted $V^k \succeq U^q$, if $Cum(V^k) \geq Cum(U^q)$.

- A sub-assignment V^k strictly dominates U^q, denoted $V^k \succ U^q$, if $Cum(V^k) > Cum(U^q)$.

- Two sub-assignments V^k and U^q are incomparable, denoted $V^k \sim U^q$, if neither $V^k \succeq U^q$ nor $U^q \succeq V^k$. Note that $[k > q$ and $V^k \sim U^q \Leftrightarrow |V_2^k| > |U_2^q|$ and $|U_1^q| > |V_1^k|]$.

- A sub-assignment U^k is optimal if there is no other sub-assignment of the same degree V^k such that $V^k \succ U^k$. Note that $V^k \succ U^k \Leftrightarrow |V_1^k| > |U_1^k|$.

- We define a procedure called *complete()* which takes as input a sub-assignment U^k and returns an assignment S such that $U^k \subseteq S$. The procedure can be implemented as follows. Firstly, the procedure detects all unassigned vertices (those in the graph but not assigned in U^k). Then, it randomly connects unassigned vertices in A to unassigned vertices in T. It is possible to do so because the graph is complete. Finally, the procedure returns the resulted assignment S. Note that $|S_1| \geq |U_1^k|$ and $|S_2| \geq |U_2^k|$.

Now, we will present two important structural properties of 3-OAP. They will be useful for the design of the polynomial algorithm. The first one is stated in the following theorem.

Theorem 1. *Given any optimal sub-assignment of degree $k < \overline{r}$, an optimal sub-assignment of degree $k+1$ can be constructed in polynomial time.*

Proof. Let us assume that U^k is an optimal sub-assignment. Our proof which is a constructive one will be divided into two parts. Firstly, we construct a weighted graph G_{U^k} based on both the given 3-OAP instance and U^k. Then, we show by computing a shortest path on G_{U^k} how to construct an optimal sub-assignment V^{k+1}.

- *The construction* (see Figures 4 and 5 for an example)

1. Create the bipartite graph corresponding to the given 3-OAP instance and ignore all edges of level 3.
2. All edges become oriented from A to T, except those in U^k which are oriented inversely.
3. Create two vertices: s (modeling a start vertex) and e (modeling an end vertex).
4. Create oriented arcs from s to all vertices in A which are not incident to any edge in U^k.
5. Create oriented arcs to e from all vertices in T which are not incident to any edge in U^k.
6. Arcs which connect s to A and T to e, and arcs of level 1 have a cost of 0, regardless of their orientation.
7. Arcs of level 2 have a cost of 1 if oriented from A to T, -1 otherwise.

The proof: For any path P in G_{U^k}, we define the subset of edges $P^{\rightarrow} \subseteq E$ as follows: $(x, y) \in P^{\rightarrow}$ if $x \in A,\ y \in T$ and the arc $x \rightarrow y \in P$. Similarly, $P^{\leftarrow}$ is defined as follows: $(x, y) \in P^{\leftarrow}$ if $x \in A,\ y \in T$ and the arc $y \rightarrow x \in P$. We denote by $cost(P)$ the cost of the path P (i.e. the sum of the costs of edges in P). Moreover, we say that a sub-assignment V^{k+1} is reachable from a sub-assignment U^k by the path P if $V^{k+1} = (U^k \setminus P^{\leftarrow}) \cup P^{\rightarrow}$.

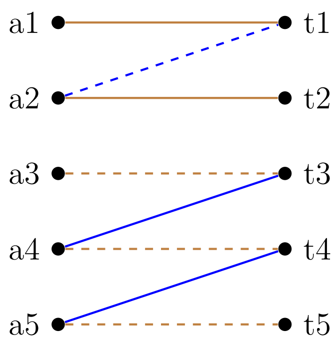

Figure 4: Blue represents level 1, while brown represents level 2. Edges of level 3 are ignored. Dashed lines represents the edges which don't belong to U^k.

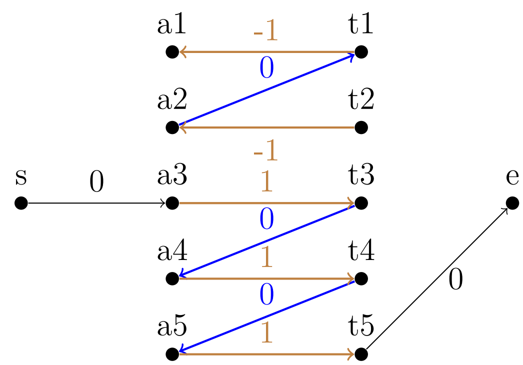

Figure 5: The graph G_{U^k} built from the instance and the sub-assignment U^k given in Figure 4.

The proof is based on the following lemmas. In order to facilitate the readability, we defer the proofs of these lemmas to the appendix.

Lemma 2. *Let P be a path in G_{U^k}. If a sub-assignment V^{k+1} is reachable from a sub-assignment U^k by P, then $|V_1^{k+1}| = |U_1^k| + 1 - cost(P)$.*

Lemma 3. *Let U^k and W^{k+1} be two sub-assignments. There exists a path Q from s to e in G_{U^k} and a sub-assignment O^k such that W^{k+1} is reachable from O^k by Q.*

Let P_s be a shortest path from s to e in the graph G_{U^k} (the existence of the path derives from lemma 3). We claim that the sub-assignment V^{k+1} reachable from U^k by P_s is optimal.
Assume to the contrary, that there is a sub-assignment W^{k+1} which strictly dominates V^{k+1}. By using lemma 3, we know that there exists a path Q in G_{U^k} and a sub-assignment O^k such that W^{k+1} is reachable from O^k by Q. Thus, From lemma 2 we obtain: $|W_1^{k+1}| = |O_1^k| + 1 - cost(Q)$. Since V^{k+1} is reachable from U^k by P_s, then by lemma 2 we have: $|V_1^{k+1}| = |U_1^k| + 1 - cost(P_s)$.
Therefore we have: $|V_1^{k+1}| - |W_1^{k+1}| = |U_1^k| - |O_1^k| + cost(Q) - cost(P_s)$.
Since U^k is an optimal sub-assignment of degree k, then $|U_1^k| \geq |O_1^k|$. P_s and Q

are two paths on the same graph G_{U^k} which both start from s and end in e, and since P_s is a shortest path, then $cost(Q) \geq cost(P_s)$. We finally conclude that $|V_1^{k+1}| - |W_1^{k+1}| \geq 0$, a contradiction.

The polynomial complexity of the method is straightforward. □

The next theorem shows how we make use of optimal sub-assignments in order to find optimal assignments.

Theorem 2. *Given one optimal sub-assignment of each degree $k \in [\overline{r}]$, a minimal complete set can be found in polynomial time.*

Proof. Let $S^1, S^2, \ldots, S^{\overline{r}}$ be optimal sub-assignments. (For ease of reading, we abuse notation and denote all the optimal sub-assignment with letter S). We will show that among these sub-assignments, those which verify the condition $S^i \sim S^{i+1}$ will serve to construct the minimal complete set.
More precisely, let $D = \{i \in [\overline{r}] : S^i \sim S^{i+1}\} \cup \{\overline{r}\}$. Let Fi, $i \in [\overline{r}]$ be the assignment returned by the procedure $complete(S^i)$. We claim that $M = \{Fi : i \in D\}$ is a minimal complete set.

- *First part:* Let us show that M is complete.
Assume that it is not the case, then there exists an optimal assignment $O \notin M$ which is not equivalent to any assignment in M. Let $p = |O_2|$.
Let us first show that $Fp \succeq O$. By definition of $\overline{r}$ we have $p \in [\overline{r}]$. It follows that S^p is optimal and consequently $|S_1^p| \geq |O_1|$ (otherwise removing edges of level 3 from O will result on a sub-assignment which strictly dominates S^p). Since $|Fp_1| \geq |S_1^p|$ therefore we obtain $|Fp_1| \geq |O_1|$. Finally, since we have $|Fp_2| \geq |O_2|$ (because $|S_2^p| = |O_2|$ and $|Fp_2| \geq |S_2^p|$) and $|Fp_3| = |O_3| = n$, we deduce that: $Fp \succeq O$.

Now we will show that M contains either Fp or an assignment which dominates Fp.
Note that if $p \in D$, then by construction of M we have: $Fp \in M$. So assume $p \notin D$ and let q be the smallest element in D such that $q > p$. In the following, we will show that $Fq \succeq Fp$. We will provide two different proofs depending on the value of q.

Case 1: $q < \overline{r}$. Assume to the contrary that $Fp \succ Fq$, then $|Fp_2| \geq |Fq_2| \geq |S_2^q|$ and thus $|Fp_2| \geq q$. Let W^{q+1} be the sub-assignment resulting from removing the least[2] ranked $n-(q+1)$ edges of Fp. Two cases are possible. Either none of the removed edges is of level 1, which implies $|W_1^{q+1}| = |Fp_1| \geq |Fq_1| \geq |S_1^q|$. Or some of the removed edges is of level 1, which implies $|W_1^{q+1}| = q+1$ and consequently $|W_1^{q+1}| > |S_1^q|$. Therefore, in both cases $|W_1^{q+1}| \geq |S_1^q|$. Since $|S_1^{q+1}| \geq |W_1^{q+1}| \geq |S_1^q|$ and $|S_2^{q+1}| > |S_2^q|$ then $S^{q+1} \succ S^q$, which contradicts the fact that $q \in D$.

Case 2: $q = \overline{r}$. Assume to the contrary that $Fp \succ Fq$. So we have $Fp \succ F\overline{r}$. Since $\overline{r} = max(\{|S_2| : S \in M\})$ and $Fp \succ F\overline{r}$ then $|Fp_2| = |F\overline{r}_2| = \overline{r}$. Therefore $Fp \succ F\overline{r}$ implies $|Fp_1| > |F\overline{r}_1|$. Let $Z^{\overline{r}}$ be the sub-assignment resulting by removing form Fp all edges of level 3. Consequently $|Z_1^{\overline{r}}| = |Fp_1| > |F\overline{r}_1|$. Since $\overline{r} = max(\{|S_2| : S \in M\})$ then all edges in $F\overline{r} \setminus S^{\overline{r}}$ are of level 3, thus $|F\overline{r}_1| = |S_1^{\overline{r}}|$. Therefore we obtain that: $|Z_1^{\overline{r}}| > |S_1^{\overline{r}}|$, which implies that $S^{\overline{r}}$ is not optimal, a contradiction.

To sum up, we have shown that in all possible cases, there is an assignment in M which dominates O. This concludes the proof of the first part.

- *Second part:* Let us now show that M is minimal.

To this end, we will first prove that $\forall k, j \in D$ we have: $S^k \sim S^j$.

Without loss of generality, assume $j > k$. Obviously, we can not have $S^k \succ S^j$ since by definition of a sub-assignment we have $|S_2^k| = k$ and $|S_2^j| = j$. So let us assume that $S^j \succ S^k$. Let B^{k+1} be the sub-assignment resulting from removing the *least ranked* $j-(k+1)$ edges of S^j. One of two possibilities holds: either none of the removed edges is of level 1 and then $|B_1^{k+1}| = |S_1^j| \geq |S_1^k|$. Or else some of the removed edges are of level 1 and then all edges of B^{k+1} are of level 1. Thus $|B_1^{k+1}| = k+1 > |S_1^k|$. Consequently we get in both cases: $|B_1^{k+1}| \geq |S_1^k|$. Since $k \in D$ and $k < \overline{r}$, then by definition of the set D we have $S^k \sim S^{k+1}$, which implies $|S_1^k| > |S_1^{k+1}|$. Therefore $|B_1^{k+1}| > |S_1^{k+1}|$, which contradicts the fact that S^{k+1} is an optimal sub-assignment.

Now we will prove that $\forall k, j \in D$: $S^k \sim S^j \Rightarrow Fk \sim Fj$.

Without loss of generality, let $j > k$. We will show that all edges in $Fk \setminus S^k$ are of level 3. Assume to the contrary that there exists an edge d in $Fk \setminus S^k$ such that d is of level 2 or 1. Let $V^{k+1} = S^k \cup d$. Therefore $|V_1^{k+1}| \geq |S_1^k|$. Since $k \in D$ and $k < \overline{r}$, then by definition of the set D we have $S^k \sim S^{k+1}$ and thus $|S_1^k| > |S_1^{k+1}|$. Consequently $|V_1^{k+1}| > |S_1^{k+1}|$, which contradicts the fact that S^{k+1}

[2] i.e. firstly removing edges of level 3 then if needed edges of level 2 and so on...

is optimal. By the same reasoning, we prove that all edges in $Fj \smallsetminus S^j$ are of level 3. Therefore, $|Fk_1| = |S_1^k|$, $|Fk_2| = |S_2^k|$ and $|Fj_1| = |S_1^j|$, $|Fj_2| = |S_2^j|$.
Since $S^k \sim S^j$ and $j > k$,
$|S_1^k| = |Fk_1| > |S_1^j| = |Fj_1|$ and $|S_2^j| = |Fj_2| > |S_2^k| = |Fk_2|$.
We conclude that $Fk \sim Fj$. □

Now, based on the proof of theorem 1 and 2, the general outline of the algorithm that solves 3-OAP may seem clear to the reader. The algorithm will gradually calculate optimal sub-assignments by successive shortest path searches. In each step (after each search), the algorithm tests if the optimal sub-assignment U calculated at the previous step is incomparable with V, the optimal sub-assignment calculated at the current step. If yes, the assignment S returned by the procedure *complete*(U) will be added to the minimal complete set.

Algorithm 1: 3-OAP

Data: An instance of 3-OAP.
Result: A minimal complete set M
Initialization: Construct the new graph G' (as described in the construction part of the proof of theorem 1)
Initialization: $U \leftarrow \emptyset$, $V \leftarrow \emptyset$, $M \leftarrow \emptyset$
while *there is a path from* s *to* e **do**
    Compute P the shortest path from s to e
    $U \leftarrow V$
    $V \leftarrow V \smallsetminus P^{\leftarrow}$
    $V \leftarrow V \cup P^{\rightarrow}$
    if *cost*$(P) > 1$ **then**
        $M \leftarrow M \cup$ *complete*(U)
    end
    Update G' by: reversing the direction of the oriented arcs which belongs to the shortest path P and changing their costs to the additive inverse
end
$M \leftarrow M \cup$ *complete*(U)
Return M.

In order to prove the correctness of the algorithm, we will first show the following loop invariant.

Lemma 4. *At the end of each loop, the current sub-assignment V is optimal.*

Proof. Let us prove the lemma by induction. In the 1st loop, the edge in the current sub-assignment V is of level 2 if and only if there is no edge of level 1 in the graph. Thus, the first sub-assignment computed by the algorithm is optimal. Now, we only need to point out that the algorithm constructs successive sub-assignments such that at the end of each loop, the sub-assignment V is reachable from U (the sub-assignment computed in the loop before) by the shortest path. Thus, by direct application of the proof of theorem 1, we deduce that all sub-assignments computed by the algorithm are optimal. □

Since the main loop in the algorithm stops if and only if no path exists from s to e, then the algorithm computes an optimal sub-assignment for each $k \in [\overline{r}]$.

Lemma 5. *$V \sim U$ if and only if $cost(P) > 1$.*

Proof. Precisely in lines 6 and 7, the algorithm constructs V such that it is reachable from U by the path P, hence from lemma 2 we obtain: $cost(P) = 1 - (|V_1| - |U_1|)$. Note that we have $cost(P) > 1$ if and only if $|U_1| > |V_1|$. Finally, recall that $|V_2| > |U_2|$. We conclude that $cost(P) > 1$ if and only if $V \sim U$. □

Lemma 5 implies that the *if condition* at line 8 guarantees that $complete(U) \in M$ if and only if $U \sim V$. Furthermore, line 13 guarantees that $\overline{r} \in D$ (see the proof of theorem 2). As a consequence, M is constructed exactly as shown in the proof of theorem 2, and therefore the algorithm returns a minimal complete set.

Before moving to the next section, we will show that C-OAP is tractable (i.e. the size of the minimal complete set of solutions is at most polynomial in the size of the instance).

Proposition 3. *C-OAP is tractable.*

Proof. To each solution of C-OAP corresponds a cumulative vector which has C components. Each component can take at most $n+1$ different values, except the last one which is always equal to n. Thus there are at most $(n+1)^{C-1}$ possible cumulative vectors. Consequently, there are at most $(n+1)^{C-1}$ optimal solutions. This is of course a rough estimation as a cumulative vector has non decreasing coordinates. □

3 Complexity of OAP

In this section, we will show that when $|\mathbb{L}|$ is polynomial in the size of the instance, the ordinal assignment problem becomes hard. Particularly, we will prove that OAP is intractable (i.e. the size of the minimal complete set can be exponential in the size of the instance) and also that the decision version of OAP is NP_complete.

Proposition 4. *OAP is intractable.*

Proof. In order to prove that OAP is intractable, we should find an instance such that the size of the complete minimal set M is exponential in n. Note that as soon as M has exponential size and it is minimal, complementing it would only increase its size. Therefore it is sufficient to exhibit an instance such that the minimal set M is exponential in n.

We claim that $|M| = 2^{n/2}$ for the instance shown in figure 6 (n is assumed even).

$$\begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \\ a_4 \\ \vdots \\ a_{n-1} \\ a_n \end{array} \begin{array}{c} \begin{array}{cccccccc} t_1 & t_2 & t_3 & t_4 & \dots & t_{n-1} & t_n \end{array} \\ \begin{pmatrix} 2 & 3 & & & & & \\ 1 & 2 & & & & & \\ & & 3 & 4 & & & \\ & & 2 & 3 & & & \\ & & & & \ddots & & \\ & & & & & n/2+1 & n/2+2 \\ & & & & & n/2 & n/2+1 \end{pmatrix} \end{array}$$

Figure 6: An instance of OAP where $|M|$ is exponential in n. All empty entries are $n/2+2$.

To prove the claim, we will show that the minimal set M is defined as follows: $S \in M \iff \forall i \in \{1, 3, \dots, n-1\}$ $[(a_i, t_i), (a_{i+1}, t_{i+1}) \in S]$ or $[(a_i, t_{i+1}), (a_{i+1}, t_i) \in S]$.
Clearly $|M| = 2^{n/2}$. Let us prove that M is minimal. To this end, let S and S' be two solutions in M, we will show that S and S' are incomparable.

Let $i \in \{2, 4, \ldots, n\}$, we define q_i as the sub-matrix: $\begin{array}{c} \\ a_{i-1} \\ a_i \end{array} \begin{array}{c} \begin{array}{cc} t_{i-1} & t_i \end{array} \\ \begin{pmatrix} i/2+1 & i/2+2 \\ i/2 & i/2+1 \end{pmatrix} \end{array}$

Consider the smallest index $k = i-1$ such a_{i-1} and a_i are assigned differently in S and S'. Exactly one of the following two cases holds: either the diagonal assignment in q_i belongs to S (i.e. $(a_{i-1}, t_{i-1}) \in S$ and $(a_i, t_i) \in S$) and the anti-diagonal belongs to S' (i.e. $(a_i, t_{i-1}) \in S'$ and $(a_{i-1}, t_i) \in S'$), or the inverse. From the instance shown in figure 6, we deduce that:
- In the first case we have: $|S_{i/2+1}| > |S'_{i/2+1}|$ and $|S'_{i/2+2}| > |S_{i/2+2}|$.
- In the second case we have: $|S_{i/2+2}| > |S'_{i/2+2}|$ and $|S'_{i/2+1}| > |S_{i/2+1}|$.
Therefore, in both cases, S and S' are incomparable. □

Now, let us consider the decision version of OAP, denoted Dec-OAP, stated as follows: Given an OAP instance and given a cumulative vector V (i.e. a vector of C components s.t. $V_C = n$ and $V_i \leq V_j, \ \forall i < j$), does there exists an assignment S such that $Cum(S) \geq V$?

Theorem 3. Dec-OAP *is Np_complete.*

Proof. Dec-OAP is obviously in NP. The proof of Np_hardness relies on a polynomial reduction from the Np_hard k-SAT problem defined as follows:
Instance: A collection of clauses $\{C^1, \ldots, C^m\}$, each clause consists of exactly k literals. A literal l_i is either a variable (e.g. $l_i = x_i$), then called positive literal, or the negation of a variable (e.g. $l_i = \overline{x_i}$), then called negative literal. The set of boolean variables is denoted by $X = \{x^1, \ldots, x^n\}$.
Question: Does there exists a truth assignment of boolean variables in X which set all clauses to true (i.e. each clause has at least one true literal) ?
Polynomial reduction: Before detailing the construction, we assume without loss of generality that: firstly, there is a total order over literals of each clause. Secondly, there is a cyclic order over occurrences of the same variable, which means that each occurrence has a successor.

-*The construction:* From an instance of k-SAT, we construct an instance of the ordinal assignment problem as follows:
- $A = \{l_1^1, l_2^1, \ldots, l_k^1, \ldots, l_1^m, l_2^m, \ldots, l_k^m\}$ such that l_q^p is the q-th literal in clause C^p.
- $T = \{C_1^1, C_2^1, \ldots, C_k^1, \ldots, C_1^m, C_2^m, \ldots, C_k^m\}$, i.e. for each clause C^i, k vertices

are created. Note that $|A| = |T| = mk$.
- $\mathbb{L} = \{1, 2, \ldots, 2m + 1\}$.
- $V = (1, k, k + 1, 2k, \ldots, (i - 1)k + 1, ik, \ldots, (m - 1)k + 1, mk, mk)$.
- Let $s(l_q^p)$ be the successor of l_q^p, and let $c(l_q^p) = p'$ if $s(l_q^p)$ belongs to $C^{p'}$. Finally, let $r(l_q^p)$ be the rank of $s(l_q^p)$ in $C^{p'}$. For each literal l_i ranked in q-th position in C^p we let $u(l_q^p, C_q^p)$ equals $2p - 1$ if l_q^p is positive, $2p$ otherwise. In addition, let $p' = c(l_q^p)$ and $q' = r(l_q^p)$, we let $u(l_q^p, C_{q'}^{p'})$ equals $2p$ if l_q^p is positive, $2p - 1$ otherwise. All the other edges of the graph are of level $2m + 1$ (recall that the graph is complete).

-*Example:* For illustration, we present in figure 7 an example where a variable x_i has four occurrences. The literal x_i is ranked in position p_4 in the clause C^d and in position p_3 in C^c, while $\overline{x_i}$ is ranked in position p_1 in C^a and in position p_2 in C^b.
The cyclic order is as follows: $x_i^d \rightarrow x_i^c \rightarrow \overline{x_i^b} \rightarrow \overline{x_i^a} \rightarrow x_i^d$ (i.e. $s(x_i^d) = x_i^c$, $s(x_i^c) = \overline{x_i^b}$ ...)

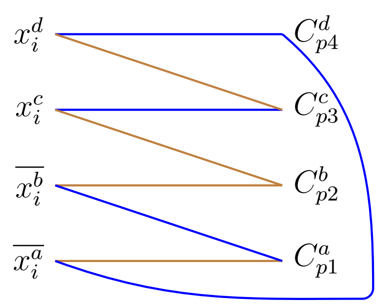

Figure 7: The edge incident to x_i^α is blue when its level is $2\alpha - 1$, and brown when its level is 2α. Edges of level $2m + 1$ are ignored.

-*Outline of the proof:* We want to prove that if there is an assignment S such that $Cum(S) \geq V$ then the answer to the k-SAT instance is yes.
We have defined the cumulative vector V such that $V_{2m} = V_{2m+1} = mk$. That implies that no edge of level $2m+1$ belongs to S. It means that for each variable, we have a cyclic graph (like the one presented in figure 7). Therefore, only two possible assignments are possible (see figures 8 and 9). Each of these two assignments encode a state of the variable, i.e. an assignment encode the state true and

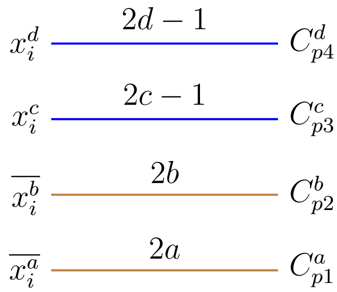

Figure 8: The first scenario, equivalent to the variable x_i true. A literal incident to a blue edge is true. A literal incident to a brown edge is false.

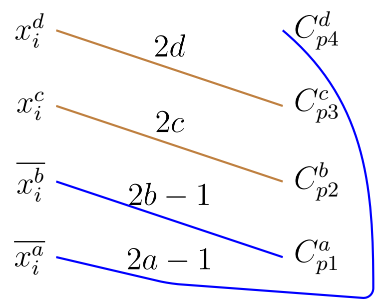

Figure 9: The second scenario, equivalent to the variable x_i false. A literal incident to a blue edge is true. A literal incident to a brown edge is false.

the other one encode the state false.
By construction, an edge of level $2p-1$ is used when the literal set the clause C_p to true, otherwise $2p$ is used (see figures 8 and 9). The cumulative vector is defined such that for each $p>1$, $V_{2p-1}-V_{2p-2}\geq 1$. That imposes that for each p, at least one edge of the assignment S is of level $2p-1$. By letting all literals incident to an edge of level $2p-1$ be true and all the others be false, we ensure that each clause is true.

-*The proof:* We claim that there is an assignment S such that $Cum(S)\geq V$ if and only if the answer to the k-SAT instance is yes.

Assume that there is an assignment S such that $Cum(S)\geq V$.
First, let us show that $\forall p\in[m]$ at least one edge of level $2p-1$ belongs to S. The case where $p=1$ is straightforward (since $|S_1|\geq|V_1|=1$). Therefore, we will show that: $p\geq 2\Rightarrow |S_{2p-1}|-|S_{2p-2}|\geq 1$. Note that no edge of level $2m+1$ belongs to S, otherwise we would have $|S_{2m}|<mk$ which contradicts the fact that $|S_{2m}|\geq|V_{2m}|=mk$. Therefore, $\forall q\in[k]$, the vertex l_q^p is connected to a vertex in T with either level $2p-1$ or $2p$. At the same time, edges of level $2p-1$ and $2p$ are, by construction, incident only to vertices l_q^p, $q\in[k]$. Consequently we deduce that: $|S_2|=k$ and $|S_{2p}|-|S_{2p-2}|=k$, which implies $|S_{2p}|=pk$,$\forall p\in[m]$. Since $|S_{2p-1}|\geq|V_{2p-1}|=(p-1)k+1$ and $|S_{2p-2}|=|S_{2(p-1)}|=(p-1)k$ then $|S_{2p-1}|-|S_{2p-2}|\geq(p-1)k+1-(p-1)k\geq 1$.

Now, we know that $\forall p \in [m]$ at least one edge of level $2p-1$ belongs to S. At the same time, we know, by construction, that edges of level $2p-1$ are incident only to vertices l_q^p, $q \in [k]$. Therefore, $\forall p \in [m]$ $\exists q \in [k]$ such that l_q^p is incident to an edge of level $2p-1$. Consequently, $\forall p \in [m]$, C^p contains at least one literal l_q^p incident to an edge of level $2p-1$. Let all literals l_q^p incident to an edge of level $2p-1$ be true and all the others be false, this truth assignment of boolean variables will therefore set all clauses to true.

To finish the proof, we need to show that this truth assignment of boolean variables is coherent, that is, we are not imposing a literal and its negation to be both true or both false at the same time. To this end, recall as noted before that no edge of level $2m+1$ belongs to S. Thus, for each variable x_i in X, only two scenarios are possible (figures 8 and 9 illustrate both possible scenarios of the example shown in figure 7). Either each occurrence of x_i is assigned to its clause ($l_q^p = x_i$ or $l_q^p = \overline{x_i} \Rightarrow (l_q^p, C_q^p) \in S$), or each occurrence of x_i is assigned to its successor's clause ($l_q^p = x_i$ or $l_q^p = \overline{x_i} \Rightarrow (l_q^p, C_{q'}^{p'}) \in S$ s.t. $p' = c(l_q^p)$ and $q' = r(l_q^p)$). Now, recall that in the first scenario, we have $u(l_q^p, C_q^p)$ equals $2p-1$ if l_q^p is positive, $2p$ otherwise. Therefore setting all literals incident to the edges of level $2p-1$ to true and all the others to false means that positive literals of the variable x_i are true and negative ones are false. Consequently, the first scenario corresponds to the case where the variable x_i is true. Likewise, in the second scenario we have: $u(l_q^p, C_{q'}^{p'})$ equals $2p$ if l_q^p is positive, $2p-1$ otherwise. Thus setting all literals incident to the edges of level $2p-1$ to true and all the others to false means that negative literals of the variable x_i are true and positive ones are false. We deduce that the second scenario corresponds to the case where the variable x_i is false. As a result, the suggested truth assignment maintains the coherence of variables, which concludes the proof of the first implication.

Conversely, assume that the answer to the k-SAT instance is yes. We construct a solution S as follows: if the variable x_i is true (resp. false), then assign occurrences of x_i as in the first scenario (resp. second scenario). By similar arguments, we prove that $Cum(S) \geq V$. □

4 Conclusion

In this paper, we have investigated the ordinal assignment problem, a variant of the assignment problem where entries of the matrix are ordinal. The objective was to find a minimal complete set of solutions according to the ordinal dominance criterion. Table 1 summarizes most of our contributions.

Neither the algorithm presented in section 2 nor the polynomial reduction in section 3 seem to extend easily to the case where $|L| > 3$. The complexity of the problem C-OAP, $C > 3$ remains an open question to investigate in a further study.

Problem	**Complexity**	**Tractability**
C-OAP, $C \leq 3$	Polynomial	Tractable
C-OAP, $C > 3$	Unknown	
OAP	Np_complete (decision version)	Intractable

Table 1: Summary of our contributions

Appendix

The following example is adapted from [5]. We assume definitions and notations of subsection 1.2 as known.

Example 4.1. *Consider the instance of ordinal assignment problem where* $|A| = |T| = 4$ *and* $\mathbb{L} = \{1, 2, 3, 4, 5\}$*. The entry matrix encoding* u *is as follows:*

$$\begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{array} \begin{array}{c} \begin{array}{cccc} t_1 & t_2 & t_3 & t_4 \end{array} \\ \begin{pmatrix} 1 & 4 & 3 & 4 \\ 4 & 2 & 3 & 2 \\ 1 & 5 & 2 & 3 \\ 2 & 5 & 4 & 4 \end{pmatrix} \end{array}$$

Let us define a monotonic (i.e. order preserving) transformation ϕ *as a function:* $\phi : \mathbb{L} \to \mathbb{Z}$*. We transform the ordinal scale into a numerical scale using two* *possible monotonic transformations ϕ and ϕ' defined by:*

u_{ij}	1	2	3	4	5
$\phi(u_{ij})$	3	2	1	-2	-3
$\phi'(u_{ij})$	2	1	0	-1	-2

Both resulting scales are compatible with the ordinal scale, i.e. $\lambda \rhd \lambda' \Rightarrow \phi(\lambda) \geq \phi(\lambda')$ *for all* $(\lambda, \lambda') \in \mathbb{L}^2$*. Let* $\mathcal{I}$ *(resp.* $\mathcal{I}'$*) be the instance of assignment problem where the entry matrix is encoded with* ϕ *(resp.*ϕ'*). Let* S *(resp.* S'*) be the assignment such that* $x(S) = (3, 2, 4, 1)$ *(resp.* $x(S') = (1, 2, 3, 4)$*).*

$$\begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{array} \begin{array}{c} \begin{array}{cccc} t_1 & t_2 & t_3 & t_4 \end{array} \\ \begin{pmatrix} 3 & -2 & 1 & -2 \\ -2 & 2 & 1 & 2 \\ 3 & -3 & 2 & 1 \\ 2 & -3 & -2 & -2 \end{pmatrix} \end{array}$$

The entry matrix of $\mathcal{I}$

$$\begin{array}{c} \\ a_1 \\ a_2 \\ a_3 \\ a_4 \end{array} \begin{array}{c} \begin{array}{cccc} t_1 & t_2 & t_3 & t_4 \end{array} \\ \begin{pmatrix} 2 & -1 & 0 & -1 \\ -1 & 1 & 0 & 1 \\ 2 & -2 & 1 & 0 \\ 1 & -2 & -1 & -1 \end{pmatrix} \end{array}$$

The entry matrix of $\mathcal{I}'$

Using a linear programming solver, we solve the classic linear assignment problem for $\mathcal{I}$ *and* $\mathcal{I}'$*. We obtain that for* $\mathcal{I}$ *(i.e. when using* ϕ *for numerical encoding), assignment* S *is optimal and* S' *is suboptimal. However, for* $\mathcal{I}'$ *(i.e. when using* ϕ' *for numerical encoding), we obtain the opposite conclusion since* S' *is optimal and* S *is suboptimal. Thus, one observes that slight changes in the numerical transformation lead to very different conclusions. This shows that the transition from ordinal to numerical scale has to be justified. Otherwise, the conclusions drawn may be meaningless.*

Proof of lemma 2. Let us define $P_l^{\leftarrow} := \{(x, y) \in P^{\leftarrow} : u_{xy} \unrhd l\}$ and $P_l^{\rightarrow} := \{(x, y) \in P^{\rightarrow} : u_{xy} \unrhd l\}$. Recall that the graph is built such that all arcs have a cost of 0, except arcs of level 2, which have a cost of 1 if they are oriented from A to T, and -1 otherwise. Thus, the cost of P equals: the number of arcs of level 2 in P which are oriented from A to T, minus the number of arcs of level 2 in P which are oriented inversely. Formally, $cost(P) = (|P_2^{\rightarrow}| - |P_1^{\rightarrow}|) - (|P_2^{\leftarrow}| - |P_1^{\leftarrow}|)$. V^{k+1} is reachable from a sub-assignment U^k by P, i.e. $V^{k+1} = (U^k \setminus P^{\leftarrow}) \cup P^{\rightarrow}$.

By the construction of the graph G_{U^k}, edges of U^k are oriented from T to A, and so $P^{\rightarrow} \cap U^k = \emptyset$. Likewise, by the construction of the graph G_{U^k}, only edges of U^k are oriented from T to A, thus $P^{\leftarrow} \subseteq U^k$.

Therefore we have the following:

$$V^{k+1} \setminus U^k = [(U^k \setminus P^{\leftarrow}) \cup P^{\rightarrow}] \setminus U^k$$
$$V^{k+1} \setminus U^k = [(U^k \setminus P^{\leftarrow}) \setminus U^k] \cup [P^{\rightarrow} \setminus U^k]$$
$$V^{k+1} \setminus U^k = P^{\rightarrow} \setminus U^k \qquad \text{since } (U^k \setminus P^{\leftarrow}) \setminus U^k = \emptyset$$
$$V^{k+1} \setminus U^k = P^{\rightarrow} \qquad \text{since } P^{\rightarrow} \cap U^k = \emptyset \tag{1}$$

Also, we have:

$$U^k \setminus V^{k+1} = U^k \setminus [(U^k \setminus P^{\leftarrow}) \cup P^{\rightarrow}]$$
$$U^k \setminus V^{k+1} = U^k \setminus (U^k \setminus P^{\leftarrow}) \qquad \text{since } P^{\rightarrow} \cap U^k = \emptyset$$
$$U^k \setminus V^{k+1} = P^{\leftarrow} \qquad \text{since } P^{\leftarrow} \subseteq U^k \tag{2}$$

Let $Z = U^k \cap V^{k+1}$ and let $Z_l = \{(x, y) \in Z : u_{xy} \trianglerighteq l\}$.
From (1) we obtain: $P^{\rightarrow} = V^{k+1} \setminus Z$, likewise from (2) we obtain: $P^{\leftarrow} = U^k \setminus Z$.
Since $Z \subseteq U^k$ and $Z \subseteq V^{k+1}$, we deduce that: $\forall l \in \mathbb{L}$, $|P_l^{\rightarrow}| = |V_l^{k+1}| - |Z_l|$ and $|P_l^{\leftarrow}| = |U_l^k| - |Z_l|$.
Now, we formulate the cost of the path P in terms of U^k and V^{k+1} as follows:
$cost(P) = [(|V_2^{k+1}| - |Z_2|) - (|V_1^{k+1}| - |Z_1|)] - [(|U_2^k| - |Z_2|) - (|U_1^k| - |Z_1|)]$.
By simplifying we obtain: $cost(P) = (|V_2^{k+1}| - |U_2^k|) - (|V_1^{k+1}| - |U_1^k|)$.
Since by definition $|V_2^{k+1}| = k + 1$ and $|U_2^k| = k$, we get finally $cost(P) = 1 - (|V_1^{k+1}| - |U_1^k|)$.
We conclude that $|V_1^{k+1}| = |U_1^k| + 1 - cost(P)$. □

Proof of lemma 3. Let $F \subseteq V$ and $H \subseteq E$, we denote by $I(F, H)$ the set of vertices in F incident to an edge in H. Let us first show that there exist a path Q in G_{U^k} which verifies the five following conditions:
(1) Q is a path from s to e (2) s is connected to a vertex in $I(A, W^{k+1} \setminus U^k)$ (3) $Q^{\rightarrow} \subseteq W^{k+1} \setminus U^k$ (4) $Q^{\leftarrow} \subseteq U^k \setminus W^{k+1}$ (5) the vertex connected to e belongs to $I(T, W^{k+1} \setminus U^k)$.
Assume that no path verifies all these five conditions.

Let us build a path R by starting from a vertex in $I(A, W^{k+1} \setminus U^k)$ and selecting, whenever possible, an arc such that $R^{\rightarrow} \subseteq W^{k+1} \setminus U^k$ and $R^{\leftarrow} \subseteq U^k \setminus W^{k+1}$. Therefore, R will end either in $I(T, W^{k+1} \setminus U^k)$ or $I(A, U^k \setminus W^{k+1})$. However, if R reaches $I(T, W^{k+1} \setminus U^k)$, then by connecting the first vertex of R to s and the last vertex of R to e, R will verify all the five conditions, a contradiction. We deduce that R ends up in $I(A, U^k \setminus W^{k+1})$.
Note that at each *arc selection* step, either we are in a vertex $v \in A$ that has at most a unique intersection with W^{k+1} or a vertex $v \in T$ that has a unique intersection with U^k. Thus, at each *arc selection*, there is at most one possible arc to select, which means that starting from a particular vertex leads to a unique possible path. Consequently, if we start from two different vertices of $I(A, W^{k+1} \setminus U^k)$ we will end in two different vertices of $I(A, U^k \setminus W^{k+1})$. That implies: $|I(A, W^{k+1} \setminus U^k)| \leq |I(A, U^k \setminus W^{k+1})|$ (i).
Since $|I(A, W^{k+1})| = |I(A, W^{k+1} \setminus U^k)| + |I(A, U^k \cap W^{k+1})|$ and $|I(A, U^k)| = |I(A, U^k \setminus W^{k+1})| + |I(A, U^k \cap W^{k+1})|$, then by adding $|I(A, U^k \cap W^{k+1})|$ to both sides of the inequality (i) we obtain: $|I(A, W^{k+1})| \leq |I(A, U^k)|$, which is a contradiction because $|I(A, W^{k+1})| = |W^{k+1}| = k+1$ and $|I(A, U^k)| = |U^k| = k$.

Now, since we have proved the existence of a path Q verifying the five conditions, let O^k to be the sub-assignment constructed as follows: $O^k = (W^{k+1} \setminus Q^{\rightarrow}) \cup Q^{\leftarrow}$.
Let us prove that W^{k+1} is reachable from O^k by Q.
We have: $O^k \setminus Q^{\leftarrow} = [(W^{k+1} \setminus Q^{\rightarrow}) \cup Q^{\leftarrow}] \setminus Q^{\leftarrow} = (W^{k+1} \setminus Q^{\rightarrow}) \setminus Q^{\leftarrow}$.
From condition (4) we deduce that $W^{k+1} \cap Q^{\leftarrow} = \emptyset$. Therefore, $O^k \setminus Q^{\leftarrow} = W^{k+1} \setminus Q^{\rightarrow}$. From condition (3) we have $Q^{\rightarrow} \subseteq W^{k+1}$. Consequently, $(O^k \setminus Q^{\leftarrow}) \cup Q^{\rightarrow} = (W^{k+1} \setminus Q^{\rightarrow}) \cup Q^{\rightarrow} = W^{k+1}$. □

References

[1] Naveen Garg, Telikepalli Kavitha, Amit Kumar, Kurt Mehlhorn, Julián Mestre, Assigning papers to referees, Algorithmica, **58**, no. 1, (2010), 119–136.

[2] J. Gorski, S. Ruzika, On k-max-optimization, Operations Research Letters, **37**, no. 1, (2009), 23–26.

[3] Pierre Hansen, Bicriterion path problems, Multiple criteria decision making theory and application, (1980), 109–127.

[4] Alan Agresti, Analysis of ordinal categorical data, **656**, John Wiley and Sons, 2010.

[5] Soufiane Drissi Oudghiri, Patrice Perny, Olivier Spanjaard, Mohamed Hachimi, Using the Sugeno Integral in Optimal Assignment Problems with Qualitative Utilities, Proceedings of the Twenty-second European Conference on Artificial Intelligence, ECAI'2016, The Hague, The Netherlands, 1406–1413.

[6] E. M. Bartee, Problem solving with ordinal measurement, Management Science, **17,** no. 10, 622–633,

[7] Stanley Smith Stevens, On the theory of scales of measurement, Bobbs-Merrill, College Division, 1946.

[8] O. Gross, The bottleneck assignment problem, P-1630, The Rand Corporation, 1959.

[9] S. Bouveret, U. Endriss, J. Lang, Fair division under ordinal preferences: Computing envy-free allocations of indivisible goods, European Conference on Artificial Intelligence (ECAI 2010), 387–392.

[10] S. Brams, P. Edelman, P. Fishburn, Fair division of indivisible items, Theory and Decision, **5**, no. 2, (2004), 147–180.

[11] U. Bossong, D. Schweigert, Minimal paths on ordered graphs, Report in Wirtschaftsmathematik, Universität Kaiserslautern, no. 24, (1996).

[12] F. Della Croce, V. Th. Paschos, A. Tsoukias, An improved general procedure for lexicographic bottleneck problems, Op. Res. Letters, **24**, (1999), 187–194.

[13] C. Delort, O. Spanjaard, P. Weng, Committee Selection with a Weight Constraint Based on a Pairwise Dominance Relation, ADT, (2011), 28–41.

[14] J. Goldsmith, R. Sloan, The AI conference paper assignment problem, Proc. AAAI Workshop on Preference Handling for Artificial Intelligence, (2007), 53–57.

[15] F. S. Roberts, Meaningfulness of conclusions from combinatorial optimization, Discrete Applied Mathematics, **29**, (1990), 221–241.

[16] D. Schweigert, Ordered graphs and minimal spanning trees, Foundations of Computing and Decision Sciences, **24**, no. 4, (1999), 219–229.

[17] P. T. Sokkalingam, Y. P. Aneja, Lexicographic bottleneck combinatorial problems, Operations Research Letters, **23**, no. 1, (1998), 27–33.