

# New Semidefinite Programming Relaxations for the Linear Ordering and the Traveling Salesman Problem

Philipp Hungerländer

Department of Mathematics, Alpen-Adria Universität Klagenfurt, Austria,  
philipp.hungerlaender@aau.at

---

## Abstract

In 2004 Newman [43] suggested a semidefinite programming relaxation for the Linear Ordering Problem (LOP) that is related to the semidefinite program used in the Goemans-Williamson algorithm to approximate the Max Cut problem [22]. Her model is based on the observation that linear orderings can be fully described by a series of cuts. Newman [43] shows that her relaxation seems better suited for designing polynomial-time approximation algorithms for the (LOP) than the widely-studied standard polyhedral linear relaxations.

In this paper we strengthen the relaxation proposed by Newman [43] and conduct a polyhedral study of the corresponding polytope. Furthermore we relate the relaxation to other linear and semidefinite relaxations for the (LOP) and for the Traveling Salesman Problem and elaborate on its connection to the Max Cut problem.

*Keywords:* Linear Ordering Problem; Max Cut Problem; Traveling Salesman Problem; Target Visitation Problem; Vertex Ordering Problems; Semidefinite Programming; Approximation Algorithms; Global Optimization

---

## 1. Introduction

For the past decades, combinatorial methods and linear programming (LP) techniques have partly failed to yield improved approximation guarantees for many well-studied vertex ordering problems such as the Linear Ordering Problem (LOP) or the Traveling Salesman Problem (TSP). Semidefinite programming (SDP) has proved to be a powerful tool for obtaining strong approximation results for a variety of cut problems, starting with the Max Cut problem [22].

SDP is the extension of LP to linear optimization over the cone of symmetric positive semidefinite matrices. This includes LP problems as a special case, namely when all the matrices involved are diagonal. A (primal) SDP can be expressed as the following optimization problem

$$\begin{aligned} \inf_X \{ \langle C, X \rangle : X \in \mathcal{P} \}, \\ \mathcal{P} := \{ X \mid \langle A_i, X \rangle = b_i, i \in \{1, \dots, m\}, X \succeq 0 \}, \end{aligned} \tag{SDP}$$

where the data matrices  $A_i$ ,  $i \in \{1, \dots, m\}$  and  $C$  are symmetric. For further information on SDP we refer to the handbooks [2, 53] for a thorough coverage of the theory, algorithms and software in this area, as well as a discussion of many application areas where semidefinite programming has had a major impact.

SDP has been successfully applied to many other problems besides the Max Cut problem that can be considered as cut problems such as the dicut problem [16], coloring  $k$ -colorable graphs [36], maximum  $k$ -cut [19], maximum bisection and maximum uncut [27], to name a few. In contrast, there is no such comparably general approach for approximating vertex ordering problems.

In 2004 Newman [43] suggested an SDP relaxation for the (LOP) that is related to the semidefinite program used in the Goemans-Williamson algorithm to approximate the Max Cut problem [22]. She observes that linear orderings can be fully described by a series of cuts. Accordingly she suggests an SDP relaxation using cut variables<sup>1</sup>

---

<sup>1</sup>For a formal definition of the term “cut variables” we refer to the beginning of Subsection 2.3.

to approximate the (LOP). Newman [43] shows that her relaxation seems well-suited for designing polynomial-time approximation algorithms for the (LOP). In detail she proves that for sufficiently large  $n$  her SDP relaxation has an integrality gap of no more than 1.64 for a class of random graphs on  $n$  vertices. These random graphs include the graphs used in [45] to demonstrate integrality gaps of 2 for widely-studied polyhedral linear relaxations,

Due to the interesting connection between orderings and cuts and the promising theoretical results, we decided to study the SDP relaxation suggested by Newman [43] in more detail. In this paper we propose a formulation of the (LOP) using cut variables and strengthen the SDP relaxation of Newman [43] by studying the corresponding polytope. Furthermore we extend the relaxation to other vertex ordering problems and elaborate on its connection to the Max Cut problem. As our relaxations provide new polynomial-time convex approximations of the (LOP) and (TSP) with a rich mathematical structure, we hope that they may be helpful to improve approximation results for vertex ordering problems.

We refer to our companion paper [32] for an application of the theoretical results in this paper, i.e. we use the new semidefinite relaxations suggested here to design an exact SDP approach for the Target Visitation Problem (TVP) [25, 31] that is a combination of the (LOP) and the (TSP).<sup>2</sup>

The paper is structured as follows. In Section 2 we recall some basic facts about the (LOP) and consider different ways of modelling it. Section 3 is mainly devoted to the polyhedral study of the (LOP) model using cut variables. In Section 4 we show the exact relation of our model using cut variables to the Max Cut problem and show how to extend it to other vertex ordering problems. In Section 5 we use small but hard (LOP) and (TSP) instances to compare the strength of our relaxations proposed with other linear and semidefinite relaxations from the literature. Section 6 concludes the paper.

## 2. Linear and Quadratic Models for the Linear Ordering Problem

In this section first we briefly review the basic properties, state-of-the-art exact and heuristic approaches and main areas of application of the (LOP). Then we discuss linear and quadratic formulations of the (LOP) using ordering variables in Subsection 2.2. Finally in Subsection 2.3 we consider a quadratically constrained quadratic program using cut variables proposed by Newman [43] that gives an upper bound for the (LOP). We show how to adapt this program to obtain a quadratic formulation in cut variables for the (LOP) that forms the basis for the polyhedral study in Section 3.2.

### 2.1. A Brief Review on the Linear Ordering Problem

Ordering problems associate to each ordering (or permutation) of the set  $[n] := \{1, 2, \dots, n\}$  a profit and the goal is to find an ordering of maximum profit. In the simplest case of the (LOP), this profit is determined by those pairs  $(u, v) \in [n] \times [n]$ , where  $u$  comes before  $v$  in the ordering. Thus in its matrix version the (LOP) can be defined as follows. Given an  $n \times n$  matrix  $W = (w_{ij})$  of integers, find a simultaneous permutation  $\pi$  of the rows and columns of  $W$  such that

$$\sum_{\substack{i, j \in [n] \\ i < j}} w_{\pi(i), \pi(j)},$$

is maximized. Equivalently, we can interpret  $w_{ij}$  as weights of a complete directed graph  $G$  with vertex set  $V = [n]$ . A tournament consists of a subset of the arcs of  $G$  containing for every pair of vertices  $i$  and  $j$  either arc  $(i, j)$  or arc  $(j, i)$ , but not both. Then the (LOP) consists of finding an acyclic tournament, i.e. a tournament without directed cycles, of  $G$  of maximum total edge weight. Let us further clarify this definition with the help of a toy example. We consider 4 vertices and the pairwise weights  $w_{12} = w_{41} = w_{34} = 1$ ,  $w_{31} = w_{24} = 2$ . Figure 1 illustrates the optimal ordering of the vertices and the corresponding benefit.

The (LOP) is well known to be NP-hard [20] and it is even NP-hard to approximate the (LOP) within the factor  $\frac{65}{66}$  [45]. If all entries of  $W$  are nonnegative, a  $\frac{1}{2}$ -approximation is trivial: In any ordering of the vertices,

---

<sup>2</sup>We note that the content of the companion paper is as disjoint from this paper as possible: it designs an exact SDP approach to the (TVP) and its associated (military) applications. In the companion paper we omitted all proofs concerning the polyhedral properties and referred to this paper for details.

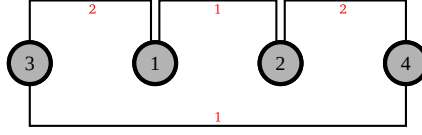


Figure 1: We are given 4 vertices and the pairwise weights  $w_{12} = w_{41} = w_{34} = 1$ ,  $w_{31} = w_{24} = 2$ . We display the optimal linear ordering  $(3, 1, 2, 4)$  with the corresponding benefit of  $1 + 1 + 2 + 2 = 6$ .

either the set of forward edges or the set of backward edges accounts for at least half of the total edge weight. If a better approximation factor than half could be obtained using a polynomial-time algorithm, this would disprove the famous Unique Games Conjecture by Khot [38]. Furthermore Newman and Vempala [45] showed that widely-studied polyhedral linear relaxations for the (LOP) cannot be used to narrow the quite large gap  $[\frac{1}{2}, \frac{65}{66}]$  (for more details see Subsection 2.2 below).

The (LOP) arises in a large number of applications in such diverse fields as economics (ranking and voting problems [50] and input-output analysis [40]), sociology (determination of ancestry relationships [21]), graph drawing (one sided crossing minimization [35]), archaeology, scheduling (with precedences [4]), assessment of corruption perception [1] and ranking in sports tournaments. Additionally problems in the context of mathematical psychology and the theory of social choice can be formulated as linear ordering problems, see [18] for a survey. For a recent survey and comparison of exact and heuristic methods for solving the (LOP) we refer to [42].

## 2.2. Formulations Using Ordering Variables

The (LOP) has a natural formulation as an integer linear program (ILP) in 0-1 variables. Let us introduce binary ordering variables  $x_{ij}$  with  $x_{ij} = 1$  if vertex  $i$  comes before vertex  $j$  and  $x_{ij} = 0$  otherwise. It is well-known [52, 55] that the following constraints describe linear orderings of  $n$  vertices:

$$x_{ij} + x_{ji} = 1, \quad i, j \in [n], \quad i \neq j, \quad (1)$$

$$x_{ij} + x_{jk} + x_{ki} \in \{1, 2\}, \quad i, j, k \in [n], \quad (2)$$

$$x_{ij} \in \{0, 1\}, \quad i, j \in [n], \quad i \neq j. \quad (3)$$

The first condition models the fact that either vertex  $i$  is before vertex  $j$  or vertex  $j$  is before vertex  $i$ . The second condition rules out the existence of directed 3-cycles and is sufficient to insure that there is no directed cycle. Hence the feasible solutions for these constraints describe acyclic tournaments of the complete directed graph  $G$  with vertex set  $[n]$ . Maximizing the objective function

$$\sum_{\substack{i, j \in [n], \\ i \neq j}} w_{ij} x_{ij} \quad (4)$$

over the constraints (1) – (3) therefore solves the (LOP). The equations (1) are used to eliminate  $x_{ji}$  for  $j > i$ . This leads to the following formulation of the (LOP) as an LP in binary variables (cf. [24])

$$\max \left\{ \sum_{1 \leq i < j \leq n} (w_{ij} - w_{ji}) x_{ij} + w_{ji} : x \in \mathcal{P}_{LOP} \right\}, \quad (LOP)$$

where the linear ordering polytope is defined as

$$\mathcal{P}_{LOP} = \text{conv} \left\{ x : x \in \{0, 1\}^{\binom{[n]}{2}}, \quad 0 \leq x_{ij} + x_{jk} - x_{ik} \leq 1, \quad i, j, k \in [n], \quad i < j < k \right\}.$$

The linear relaxation ( $\text{LP}_{LOP}$ ) is obtained by replacing the integrality conditions on the variables with the bound constraints  $0 \leq x_{ij} \leq 1$ ,  $i, j \in [n]$ ,  $i < j$ .

Newman and Vempala [45] proved that ( $\text{LP}_{LOP}$ ) has an integrality gap of 2. The graphs used to demonstrate these integrality gaps are random graphs with uniform edge probability of approximately  $\frac{2^{\log n}}{n}$ , where  $n$  is the number of

vertices. Newman and Vempala [45] further show that the integrality gap for these graphs remains 2 even if  $(\text{LP}_{\text{LOP}})$  is augmented with  $k$ -fence constraints for any  $k$ , and with  $k$ -Möbius ladder constraints for  $k$  up to 7. If  $(\text{LP}_{\text{LOP}})$  is augmented with  $k$ -Möbius ladder constraints for general  $k$ , the gap is at least  $\frac{33}{17} \approx 1.94$ .

The current state-of-the-art exact algorithm for the  $(\text{LOP})$  is an ILP branch-and-cut approach that was developed by the working group of Reinelt in Heidelberg and is based on sophisticated cut generation procedures (for details see [42]). It can solve large instances from specific instance classes with up to 150 vertices, while it fails on other much smaller instances with only 50 vertices. Hungerländer and Rendl [33] proposed an SDP approach that proved to be a valuable alternative to the ILP approach for larger and/or notoriously difficult instances. In the following we briefly recall the underlying formulation of this approach.

For this purpose it is convenient to reformulate the  $(\text{LOP})$  in variables taking the values  $-1$  and  $+1$ . The variable transformation

$$y_{ij} = 2x_{ij} - 1, \quad i, j \in [n], i < j, \quad (5)$$

leads to the following equivalent formulation of the  $(\text{LOP})$ :

$$\begin{aligned} \max \quad & \sum_{\substack{i, j \in [n] \\ i < j}} (w_{ij} - w_{ji}) \frac{y_{ij} + 1}{2} + w_{ji}, \\ \text{subject to:} \quad & -1 \leq y_{ij} + y_{jk} - y_{ik} \leq 1, \quad i, j, k \in [n], i < j < k, \\ & y_{ij} \in \{-1, 1\}^{\binom{n}{2}}, \quad i, j \in [n], i < j. \end{aligned} \quad (6)$$

In [29] it is shown that one can easily switch between the  $\{0, 1\}$  and  $\{-1, 1\}$  formulations of bivalent problems so that the resulting bounds remain the same and structural properties (like semidefiniteness constraints in SDPs) are preserved. Additionally we want to reformulate the 3-cycle inequalities (6) as quadratic conditions. A natural way to do this consists in squaring both sides of  $|y_{ij} + y_{jk} - y_{ik}| = 1$  and using  $y_{ij}^2 = 1$  to simplify the resulting expression. This finally leads to the simple 3-cycle equations

$$y_{ij}y_{jk} - y_{ij}y_{ik} - y_{ik}y_{jk} = -1, \quad i, j, k \in [n], i < j < k. \quad (7)$$

In summary we obtain the following alternative formulation of the  $(\text{LOP})$ :

$$\begin{aligned} \max \quad & \sum_{\substack{i, j \in [n] \\ i < j}} (w_{ij} - w_{ji}) \frac{y_{ij} + 1}{2} + w_{ji}, \\ \text{subject to:} \quad & y_{ij}y_{jk} - y_{ij}y_{ik} - y_{ik}y_{jk} = -1, \quad i, j, k \in [n], i < j < k, \\ & y_{ij} \in \{-1, 1\}^{\binom{n}{2}}, \quad i, j \in [n], i < j. \end{aligned} \quad (8)$$

### 2.3. A Formulation Using Cut Variables

Newman [43] presented another integer quadratically constrained quadratic program for the  $(\text{LOP})$  that can be seen as a generalization of the semidefinite programming relaxation of the Max Cut problem [22] (see also Subsection 4.1 for a more details on this connection).

The model by Newman [43] is based on the following observation that relates cuts and orderings: A linear ordering of  $n$  vertices can be fully described by a series of  $n$  cuts. Let a vector  $v$  of size  $n(n-1)$  that contains bivalent cut variables be given. We can write the components  $v_{(i-1)n+j}$ ,  $i \in [n]$ ,  $j \in [n-1]$ , of  $v$  more compactly as

$$v_i^j \in \{-1, 1\}, \quad i \in [n], j \in [n-1]. \quad (9)$$

Additionally we will use the parameters

$$v_i^0 := -1, \quad v_i^n := 1, \quad i \in [n]. \quad (10)$$

Now we can describe a linear ordering of  $n$  vertices using the variable vector  $v$ , where vertex  $i \in [n]$  is associated to

the  $n - 1$  variables  $v_i^1, \dots, v_i^{n-1}$ , as follows: We define the mapping  $\varphi$  between linear orderings  $\pi$  of the vertices  $i \in [n]$  on the one hand and the cut variables  $v_i^j$ ,  $i \in [n]$ ,  $j \in [n - 1]$  on the other hand:

$$\varphi(\pi) \rightarrow \begin{cases} v_i^j = -1, & \text{if vertex } i \text{ comes after position } j \text{ in the ordering,} \\ v_i^j = +1, & \text{if vertex } i \text{ is at position } j \text{ or before in the ordering.} \end{cases} \quad (11)$$

To further clarify this definition, we encode the optimal ordering (3,1,2,4) of the toy example depicted in Figure 1 in  $v$ , where we separate variables associated to different vertices by a vertical dash  $|$ :

$$v_{\text{toy}} = (-1 \quad 1 \quad 1 \quad | \quad -1 \quad -1 \quad 1 \quad | \quad 1 \quad 1 \quad 1 \quad | \quad -1 \quad -1 \quad -1)^\top. \quad (12)$$

Now let us express the binary ordering variables (3) as linear-quadratic terms in the cut variables  $v$ , where we additionally use the parameters (10):

$$x_{ij} = \frac{1}{4} \sum_{\substack{k,l \in [n], \\ k < l}} (v_i^k - v_i^{k-1})(v_j^l - v_j^{l-1}) = \frac{1}{4} \sum_{\substack{k,l \in [n], \\ k < l}} (v_i^k v_j^l + v_i^{k-1} v_j^{l-1} - v_i^k v_j^{l-1} - v_i^{k-1} v_j^l), \quad i, j \in [n], i \neq j.$$

Newman [43] presents the following quadratically constrained quadratic program that gives an upper bound to the optimal solution of the (LOP), where she assumes w.l.o.g.  $n$  odd to simplify constraint (13f):

$$\max \frac{1}{4} \sum_{\substack{i,j \in [n], \\ i \neq j}} \sum_{\substack{k,l \in [n], \\ k < l}} w_{ij} (v_i^k v_j^l + v_i^{k-1} v_j^{l-1} - v_i^k v_j^{l-1} - v_i^{k-1} v_j^l) \quad (13a)$$

$$\text{subject to: } v_i^k v_j^l + v_i^{k-1} v_j^{l-1} - v_i^k v_j^{l-1} - v_i^{k-1} v_j^l \geq 0, \quad i, j \in [n], k, l \in [n], \quad (13b)$$

$$v_i^k v_i^k = 1 \quad i, k \in [n], \quad (13c)$$

$$v_i^0 v_0 = -1, \quad i \in [n], \quad (13d)$$

$$v_i^n v_0 = 1, \quad i \in [n], \quad (13e)$$

$$\sum_{i,j \in [n]} v_i^{\frac{n}{2}} v_j^{\frac{n}{2}} = 0, \quad (13f)$$

$$v_i^k \in \{1, -1\}, \quad i, k \in [n]. \quad (13g)$$

Next Newman [43] obtains a semidefinite programming relaxation for the (LOP) by removing the integrality conditions (13g). She shows that for sufficiently large  $n$  this semidefinite programming relaxation has (with high probability) an integrality gap of no more than 1.64 (hence smaller than 2) on random graphs with uniform edge probability  $p = \frac{d}{n}$  (i.e. every edge in the complete directed graph on  $n$  vertices is chosen with probability  $p$ ). Note that in particular the graphs used in [45] to demonstrate integrality gaps of 2 for the widely-studied polyhedral linear relaxations fall into this category of random graphs. Also note that Newman [44, Section 3.2.2] proposes a related quadratically constrained quadratic program in 0-1 variables that is a formulation for the (LOP).

Next we want to extend (and simplify) the quadratically constrained quadratic program (13) to obtain a formulation of the (LOP) in cut variables. First we can rewrite the objective function (13a) as

$$\sum_{\substack{i,j \in [n], \\ i \neq j}} \frac{1}{4} w_{ij} \left( 2 + \sum_{k \in [n]} (v_i^{k-1} (v_j^k - v_j^{k-1})) \right) = \sum_{\substack{i,j \in [n], \\ i \neq j}} \frac{w_{ij}}{4} \left( 2 + \sum_{k \in [n]} (v_i^{k-1} v_j^k - v_i^{k-1} v_j^{k-1}) \right). \quad (14)$$

Next consider the quadratic constraint (13b) ensuring that for each vertex  $i \in [n]$  two consecutive variables  $v_i^{h-1}$  and  $v_i^h$  differ for only one  $h \in [n]$  in an integral solution. We can replace (13b) by the following simpler linear constraint

that is equivalent for integral solutions:

$$v_i^j \leq v_i^k, \quad i \in [n], j, k \in [n-1], j < k. \quad (15)$$

Note that also for non-integral solutions (15) enforces monotonicity of variables belonging to the same vertex and ensures together with (10):  $-1 \leq v_i^j \leq 1$ ,  $i \in [n]$ ,  $j \in [n-1]$ .

Furthermore we replace the quadratic equation (13f) by the following set of linear equations:

$$\sum_{i \in [n]} v_i^j = 2j - n, \quad j \in [n-1]. \quad (16)$$

Assuming w.l.o.g.  $n$  odd, equality (13f) only ensures that half of the variables  $v_i^{\frac{n}{2}}$ ,  $i \in [n]$ , are  $-1$  in an integral solution. Hence it is equivalent to  $\sum_{i \in [n]} v_i^{\frac{n}{2}} = 0$ .

The following lemma indicates that the linear constraints proposed form a minimal constraint system for the (LOP).

**Lemma 1.** *The constraints (15) and (16) form, together with the integrality conditions (9), a minimal constraint system for modeling the (LOP) with the help of  $n(n-1)$  cut variables.*

*Proof.* Applying the mapping  $\varphi$  defined in (11) to a linear ordering of  $n$  vertices yields cut variables that fulfill the constraints (15), (16) and the integrality conditions (9). The inverse operation to (11) is given by

$$v_i^j - v_i^{j-1} = \begin{cases} 2, & \text{if vertex } i \text{ is at position } j \text{ in the linear ordering,} \\ 0, & \text{if vertex } i \text{ is not at position } j \text{ in the linear ordering.} \end{cases} \quad (17)$$

Hence applying (17) to cut variables that fulfill constraints (15), (16) and the integrality conditions (9) yields a linear ordering of  $n$  vertices. Omitting any of the constraints would allow for feasible cut variable configurations that cannot be associated through (17) to a valid linear ordering.  $\square$

Finally we are able to formulate the (LOP) as an integer quadratic program in  $v$ :

**Theorem 2.** *The integer quadratic program*

$$\begin{aligned} \max \quad & \frac{1}{4} \sum_{\substack{i, j \in [n], \\ i \neq j}} w_{ij} \left( 2 + \sum_{k \in [n]} (v_i^{k-1} v_j^k - v_i^{k-1} v_j^{k-1}) \right), \\ \text{subject to: } \quad & v_i^j \leq v_i^k, \quad i \in [n], j, k \in [n-1], j < k, \\ & \sum_{i \in [n]} v_i^j = 2j - n, \quad j \in [n-1]. \\ & v_i^j \in \{-1, 1\}, v_i^0 = -1, v_i^n = 1, \quad i \in [n], j \in [n-1]. \end{aligned} \quad (18)$$

is a formulation of the (LOP).

*Proof.* The objective function (14) gives the correct objective value for any feasible linear ordering. Furthermore the constraints (15) model monotonicity on  $v \in \{-1, 1\}^{n(n-1)}$  and suffice together with the integrality conditions on  $v$  and (16) to induce all feasible linear orderings of  $n$  vertices (see also Lemma 1).  $\square$

The integer quadratic program (18) forms the basis for semidefinite relaxations using cut variables that we analyze in the following section.

### 3. Semidefinite Relaxations for the Linear Ordering Problem

In Subsection 3.1 we briefly recall the standard semidefinite relaxation for the (LOP) using ordering variables, for more details see [33]. In Subsection 3.2 we conduct a polyhedral study to obtain semidefinite relaxations for the (LOP) based on cut variables.

### 3.1. Relaxations Using Ordering Variables

In this subsection we briefly review how to obtain standard semidefinite relaxations from (8). We are interested in lifting the (LOP) into quadratic space and hence we take the vector  $y$  collecting the bivalent ordering variables and consider the matrix  $Y = yy^\top$ . The linear-quadratic ordering polytope can be defined as:

$$\mathcal{P}_{LQO} := \text{conv} \left\{ \begin{pmatrix} 1 \\ y \end{pmatrix} \begin{pmatrix} 1 \\ y \end{pmatrix}^\top : y \in \{-1, 1\}^{\binom{n}{2}}, y \text{ satisfies (6)} \right\}.$$

Let us first slightly rewrite (7) using the entries of matrix  $Y$ :

$$y_{i,j,k} - y_{i,j,k} - y_{i,k,k} = -1, \quad i, j, k \in [n], i < j < k. \quad (19)$$

In [6] it is shown that these equations formulated in the  $\{0, 1\}$ -model describe the smallest linear subspace that contains  $\mathcal{P}_{LQO}$ . Now we are able to give a matrix-based formulation of the (LOP).

**Theorem 3.** [33] *The (LOP) is equivalent to the following optimization problem*

$$\max \left\{ \langle C_y, Z_y \rangle : Z_y \text{ satisfies (7)}, y \in \{-1, 1\}^{\binom{n}{2}} \right\}, \quad (20)$$

where all ordering variables and their products are contained in the variable matrix  $Z_y := \begin{pmatrix} 1 & y^\top \\ y & Y \end{pmatrix} \in \mathbb{R}^{\binom{n}{2}+1}$  with  $Y = yy^\top$  and the cost matrix  $C_y$  is given by  $C_y := \begin{pmatrix} K_y & c^\top \\ c & 0 \end{pmatrix}$ ,  $K_y = \sum_{\substack{i,j \in [n], \\ i < j}} \frac{w_{ij} + w_{ji}}{2}$ ,  $c_{ij} = \frac{w_{ij} - w_{ji}}{2}$ ,  $i, j \in [n]$ ,  $i < j$ .

Finally we can further rewrite the above matrix-based formulation as an SDP, where we denote by  $e$  the vector of all ones and by  $\mathcal{E}$  the elliptope  $\mathcal{E} := \{ Z_y : \text{diag}(Z_y) = e, Z_y \succeq 0 \}$ .

**Theorem 4.** [33] *The problem*

$$\max \left\{ \langle C_y, Z_y \rangle : Z_y \text{ satisfies (7)}, Z_y \in \mathcal{E}, y \in \{-1, 1\}^{\binom{n}{2}} \right\},$$

is equivalent to the (LOP).

We are now dropping the integrality condition on  $y$  and obtain the following basic semidefinite relaxation:

$$\max \left\{ \langle C_y, Z_y \rangle : Z_y \text{ satisfies (7)}, Z_y \in \mathcal{E} \right\}. \quad (\text{SDP}_1^{\text{ord}})$$

It is easy to show that  $(\text{SDP}_1^{\text{ord}})$  is at least as strong as the linear relaxation  $(\text{LP}_{\text{LOP}})$  [33].

There are some obvious ways to tighten  $(\text{SDP}_1^{\text{ord}})$ . First of all we observe that  $Z_y$  is actually a matrix with  $\{-1, 1\}$  entries in (20). Hence it satisfies the triangle inequalities defining the metric polytope  $\mathcal{M}$ , see e.g. [15]:

$$\mathcal{M} = \left\{ Z_y : \begin{pmatrix} -1 & -1 & -1 \\ -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{pmatrix} \begin{pmatrix} Z_{i,j} \\ Z_{j,k} \\ Z_{i,k} \end{pmatrix} \leq e, \quad 1 \leq i < j < k \leq \binom{n}{2} + 1 \right\}. \quad (21)$$

By additionally asking for  $Z_y \in \mathcal{M}$  we can improve the basic relaxation  $(\text{SDP}_1^{\text{ord}})$ :

$$\max \left\{ \langle C_y, Z_y \rangle : Z_y \text{ satisfies (7)}, Z_y \in (\mathcal{E} \cap \mathcal{M}) \right\}. \quad (\text{SDP}_2^{\text{ord}})$$

Another generic improvement was suggested by Lovász and Schrijver in [41]. Applied to this model, their approach suggests to multiply the 3-cycle inequalities (6) by the nonnegative expressions  $(1 - y_{lm})$  and  $(1 + y_{lm})$ . This results in the following inequalities

$$\begin{aligned} -1 - y_{lm} &\leq y_{ij} + y_{jk} - y_{ik} + y_{ij,lm} + y_{jk,lm} - y_{ik,lm} \leq 1 + y_{lm}, \quad i, j, k, l, m \in [n], i < j < k, l < m, \\ -1 + y_{lm} &\leq y_{ij} + y_{jk} - y_{ik} - y_{ij,lm} - y_{jk,lm} + y_{ik,lm} \leq 1 - y_{lm}, \quad i, j, k, l, m \in [n], i < j < k, l < m, \end{aligned} \quad (22)$$

defining the polytope  $\mathcal{LS}$

$$\mathcal{LS} := \{ Z_y : Z_y \text{ satisfies (22)} \}. \quad (23)$$

Now  $(\text{SDP}_1^{\text{ord}})$  can also be improved by asking in addition that  $Z \in \mathcal{LS}$ :

$$\max \left\{ \langle C_y, Z_y \rangle : Z_y \text{ satisfies (7), } Z_y \in (\mathcal{E} \cap \mathcal{LS}) \right\}. \quad (\text{SDP}_3^{\text{ord}})$$

Combining  $(\text{SDP}_2^{\text{ord}})$  and  $(\text{SDP}_3^{\text{ord}})$  we obtain the following relaxation of  $\mathcal{P}_{LQO}$ :

$$\max \left\{ \langle C_y, Z_y \rangle : Z_y \text{ satisfies (7), } Z_y \in (\mathcal{E} \cap \mathcal{M} \cap \mathcal{LS}) \right\}. \quad (\text{SDP}_4^{\text{ord}})$$

Note that  $(\text{SDP}_4^{\text{ord}})$  forms the basis for the currently strongest exact approach to both the Single-Row Facility Layout Problem [34] and Multi-Level Verticality Optimization [7].

### 3.2. Relaxations Using Cut Variables

In this subsection we suggest several new semidefinite relaxations using cut variables by studying the polytope corresponding to the quadratic programming formulation (18). We start with rewriting (18) in terms of matrices to obtain another matrix-based formulation of the (LOP):

$$\max \left\{ \langle C_a, Z_v \rangle : v \in \{-1, 1\}^{n(n-1)}, v \text{ satisfies (15) and (16)} \right\}, \quad (24)$$

where all cut variables and their products are contained in the  $(n^2 - n + 1) \times (n^2 - n + 1)$  variable matrix  $Z_v := \begin{pmatrix} 1 & v^\top \\ v & V \end{pmatrix}$

with  $V = vv^\top$  and the cost matrix  $C_a$  is given by  $C_a := \begin{pmatrix} K_a & a^\top \\ a & A \end{pmatrix}$ ,  $K_a = \sum_{\substack{i,j \in [n], \\ i \neq j}} \frac{w_{ij}}{4}$ , and

$$a_i^k = \begin{cases} -\frac{1}{8} \sum_{\substack{j \in [n], \\ j \neq i}} w_{ij}, & \text{if } k = 1, \\ \frac{1}{8} \sum_{\substack{j \in [n], \\ j \neq i}} w_{ij}, & \text{if } k = n - 1, \\ 0, & \text{otherwise,} \end{cases} \quad i \in [n], k \in [n - 1],$$

$$A_{i,j}^{k,l} = \begin{cases} -\frac{w_{ij}}{4}, & \text{if } k = l, \\ \frac{w_{ij}}{4}, & \text{if } k = l - 1, \\ 0, & \text{otherwise,} \end{cases} \quad i, j \in [n], i \neq j, k, l \in [n - 1], k \leq l.$$

To further clarify this definition, let us again recall the toy example depicted in Figure 1 with the optimal ordering (3,1,2,4). The input data translates as follows into our matrix-based formulation (24), where  $v_{\text{toy}}$  has already been stated in (12):

$$K_a = 1.75, \quad a = \begin{pmatrix} -\frac{3}{8} \\ 0 \\ \frac{1}{8} \\ -\frac{1}{8} \\ 0 \\ \frac{1}{4} \\ 0 \\ 0 \\ \frac{3}{8} \\ -\frac{3}{8} \\ 0 \\ \frac{1}{8} \end{pmatrix}, \quad A = \begin{pmatrix} 0 & 0 & 0 & -\frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -\frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2} \\ -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{4} & \frac{1}{4} & 0 \\ 0 & -\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{4} & \frac{1}{4} \\ 0 & 0 & -\frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{4} \\ -\frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -\frac{1}{4} & \frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix},$$



$$v_{\text{toy}} = \begin{pmatrix} -1 \\ +1 \\ +1 \\ -1 \\ -1 \\ +1 \\ +1 \\ +1 \\ +1 \\ -1 \\ -1 \\ -1 \end{pmatrix}, \quad Z_{v_{\text{toy}}} = \begin{pmatrix} +1 & -1 & +1 & +1 & -1 & -1 & +1 & +1 & +1 & +1 & -1 & -1 & -1 \\ -1 & +1 & -1 & -1 & +1 & +1 & -1 & -1 & -1 & -1 & +1 & +1 & +1 \\ +1 & -1 & +1 & +1 & -1 & -1 & +1 & +1 & +1 & +1 & -1 & -1 & -1 \\ +1 & -1 & +1 & +1 & -1 & -1 & +1 & +1 & +1 & +1 & -1 & -1 & -1 \\ -1 & +1 & -1 & -1 & +1 & +1 & -1 & -1 & -1 & -1 & +1 & +1 & +1 \\ -1 & +1 & -1 & -1 & +1 & +1 & -1 & -1 & -1 & -1 & +1 & +1 & +1 \\ +1 & -1 & +1 & +1 & -1 & -1 & +1 & +1 & +1 & +1 & -1 & -1 & -1 \\ +1 & -1 & +1 & +1 & -1 & -1 & +1 & +1 & +1 & +1 & -1 & -1 & -1 \\ +1 & -1 & +1 & +1 & -1 & -1 & +1 & +1 & +1 & +1 & -1 & -1 & -1 \\ -1 & +1 & -1 & -1 & +1 & +1 & -1 & -1 & -1 & -1 & +1 & +1 & +1 \\ -1 & +1 & -1 & -1 & +1 & +1 & -1 & -1 & -1 & -1 & +1 & +1 & +1 \\ -1 & +1 & -1 & -1 & +1 & +1 & -1 & -1 & -1 & -1 & +1 & +1 & +1 \end{pmatrix}, \quad \langle C_a, Z_{v_{\text{toy}}} \rangle = 6.$$

Furthermore we denote the entries of matrix  $V$  that are related to products of the cut variables  $v_i^k$  and  $v_j^l$  as  $v_{i,j}^{k,l}$ ,  $i, j \in [n]$ ,  $k, l \in [n-1]$ . Finally we can further rewrite the (LOP) as a semidefinite program in cut variables:

**Theorem 5.** *The problem*

$$\min \left\{ \langle C_a, Z_v \rangle : Z_v \text{ satisfies (15) and (16)}, Z_v \in \mathcal{E}, v \in \{-1, 1\}^{n(n-1)} \right\}, \quad (25)$$

is equivalent to the (LOP).

*Proof.* The Schur complement lemma [5, Appendix A.5.5] implies  $V - vv^\top \succcurlyeq 0 \Leftrightarrow Z_v \succcurlyeq 0$ . Since  $v_{i,i}^{j,j} = 1$ ,  $i \in [n]$ ,  $j \in [n-1]$ , and  $\text{diag}(Z_v) = e$ , we have  $\text{diag}(V - vv^\top) = 0$ , which together with  $V - vv^\top \succcurlyeq 0$  shows that in fact  $V = vv^\top$  is integral. By Theorem 2, integrality on  $V$  and  $v$  together with (15) and (16) suffice to induce all feasible linear orderings of  $n$  vertices and finally the objective function  $\langle C_a, Z_v \rangle$  gives the correct benefit for all linear orderings.  $\square$

The above formulation of the (LOP) contains the  $n-1$  equalities stated in (16). We use the following theorem to eliminate  $n-1$  variables through (16) and hence to reduce the number of variables to  $(n-1)^2$ .

**Theorem 6.** [33] *Let  $m$  linear equality constraints  $Dy = d$  be given. If there exists some invertible  $m \times m$  matrix  $F$ , we can partition the linear system in the following way*

$$Dy = \begin{bmatrix} F & G \end{bmatrix} \begin{bmatrix} f \\ g \end{bmatrix} = d. \quad (26)$$

*Then we do not weaken the relaxation by first moving into the subspace given by the equations, and then lifting the problem to matrix space.*

In other words, it is equivalent in terms of tightness of the relaxation to eliminate  $m$  variables or to lift the  $m$  equality constraints in all possible ways to quadratic space, i.e. squaring them and multiplying them with all other linear equality and inequality constraints. Hence working with the smaller SDP reduces the computational effort for solving the (LOP) and also simplifies the model as we do not have to lift (16) in all possible ways and hence are able to avoid additional constraint classes. But note that due to the variable elimination we have to examine two versions for several constraint types, namely the cases where vertex  $n$  is considered and not considered.

Dropping the integrality condition on  $v$  in (25) and reducing the problem dimension with the help of (16), we obtain the following basic semidefinite relaxation of the (LOP) in cut variables:

$$\max \{ \langle C_s, Z_s \rangle : Z_s \text{ satisfies (15)}, Z_s \in \mathcal{E} \}. \quad (\text{SDP}_0^{\text{cut}})$$

The cost and variable matrices with index  $s$  consist of the first  $(n-1)^2 + 1$  rows and columns of their larger counterparts  $C_a$  and  $Z_v$  from above, where additionally the weights involving vertex  $n$  are added to  $C_s$  (by exploiting (16)).<sup>3</sup>

<sup>3</sup>We refrain from writing down  $C_s$  explicitly as the technical expressions are not relevant for the following analysis.

Note that the equality constraints (16) are implicitly assured in the above semidefinite relaxation and hence the SDP relaxation analyzed by Newman [43] is rudimentary version of  $(\text{SDP}_0^{\text{cut}})$ . In general  $(\text{SDP}_0^{\text{cut}})$  gives quite weak upper bounds to the optimal solution value of the (LOP) (for details see the numerical experiments in Subsection 5.1). Hence we will suggest several ways to improve the tightness of  $(\text{SDP}_0^{\text{cut}})$ .

First we propose  $n(n-1)(n-2)$  valid equalities for the linear ordering polytope in cut variables

$$\mathcal{P}_{\text{LOPCUT}} := \text{conv} \left\{ \begin{pmatrix} 1 \\ v \end{pmatrix} \begin{pmatrix} 1 \\ v \end{pmatrix}^\top : v \in \{-1, 1\}^{(n-1)^2}, v \text{ satisfies (15)} \right\},$$

and show that their rank is  $n(n-1)(n-2) - 1$ , i.e. their number minus 1. The following equalities, consisting of 5 different types, can be deduced by exploiting the structure of  $v$  induced by (15) and (16):

$$v_i^j - v_i^k - v_{i,i}^{j,k} = -1, \quad i, j, k \in [n-1], j < k, \quad (27)$$

$$m_k \sum_{i=1}^{n-1} v_i^j + m_j \sum_{i=1}^{n-1} v_i^k - \sum_{h=1}^{n-1} \sum_{i=1}^{n-1} v_{h,i}^{j,k} = m_k m_j, \quad m_k = 2k - n - 1, m_j = 2j - n + 1, j, k \in [n-1], j < k, \quad (28)$$

$$v_i^1 + v_j^1 + v_{i,j}^{1,1} = -1, \quad i, j \in [n-1], i < j, \quad (29)$$

$$v_i^{n-1} + v_j^{n-1} - v_{i,j}^{n-1,n-1} = 1, \quad i, j \in [n-1], i < j, \quad (30)$$

$$v_{i,j}^{k,k} + v_{i,j}^{k-1,k-1} - v_{i,j}^{k,k-1} - v_{i,j}^{k-1,k} = 0, \quad i, j \in [n-1], i < j, k \in [n-1], k \neq 1. \quad (31)$$

Now let us prove that the above equalities are valid for  $\mathcal{P}_{\text{LOPCUT}}$ :

**Lemma 7.** *The  $n(n-1)(n-2)$  equalities (27) – (31) are valid for  $\mathcal{P}_{\text{LOPCUT}}$ .*

*Proof.* Using  $V = vv^\top$  and additionally (16) for (28), we can rewrite (27) – (31) as

$$v_i^j - v_i^k - v_i^j v_i^k + 1 = (1 - v_i^k)(1 + v_i^j) = 0, \quad i, j, k \in [n-1], j < k, \quad (32)$$

$$m_k \sum_{i=1}^{n-1} v_i^j + m_j \sum_{i=1}^{n-1} v_i^k - \sum_{h=1}^{n-1} \sum_{i=1}^{n-1} v_h^j v_i^k - m_j m_k = v_n^j - v_n^k - v_n^j v_n^k + 1 = (1 - v_n^k)(1 + v_n^j) = 0, \quad j, k \in [n-1], j < k, \quad (33)$$

$$v_i^1 + v_j^1 + v_i^1 v_j^1 + 1 = (1 + v_i^1)(1 + v_j^1) = 0, \quad i, j \in [n-1], i < j, \quad (34)$$

$$v_i^{n-1} + v_j^{n-1} - v_i^{n-1} v_j^{n-1} - 1 = (-1 + v_i^{n-1})(-1 + v_j^{n-1}) = 0, \quad i, j \in [n-1], i < j, \quad (35)$$

$$v_i^k v_j^k + v_i^{k-1} v_j^{k-1} - v_i^k v_j^{k-1} - v_i^{k-1} v_j^k = (v_i^k - v_i^{k-1})(v_j^k - v_j^{k-1}) = 0, \quad i, j \in [n-1], i < j, k \in [n-1], k \neq 1. \quad (36)$$

The equalities (32) are valid for  $\mathcal{P}_{\text{LOPCUT}}$  because the combination  $v_i^k = -1$  and  $v_i^j = 1$  is not feasible for  $j < k < n$  due to monotonicity of the variables belonging to the same vertex, see (15). The same argument also holds true for vertex  $n$  in (33), where we additionally use (16) to rewrite the equation. The equalities (34) are valid for  $\mathcal{P}_{\text{LOPCUT}}$  because there can only be one vertex at position 1 in the linear ordering and hence  $v_i^1 = v_j^1 = 1$  is not allowed. The same argument holds true for (35): There can only be one vertex at position  $n$  in the linear ordering and hence  $v_i^{k-1} = v_j^{k-1} = -1$  is not allowed. Finally the equalities (36) are valid for  $\mathcal{P}_{\text{LOPCUT}}$  as there can be only one vertex at position  $k \in [n]$  in the linear ordering:  $v_i^k - v_i^{k-1} \geq 0$  holds for  $i \in [n-1]$  because of (15) and  $v_i^k - v_i^{k-1} > 0$  is only possible if vertex  $i$  is at position  $k$  in the linear ordering. □

Next we examine the rank of the equalities proposed.

**Lemma 8.** *The equalities (27) – (31) have rank  $n(n-1)(n-2) - 1$ .*

*Proof.* First we show that the  $\frac{(n+1)n(n-2)}{2}$  equalities  $(27)_{k-j=1}$ ,  $(28)_{k-j=1}$ , (29), (30) and (31) are linear dependent and have rank  $\frac{(n+1)n(n-2)}{2} - 1$ . Note that in these equalities all products of cut variables  $v_{i,j}^{k,l}$ ,  $i, j, k, l \in [n-1]$ ,  $i < j$ ,  $|k-l| \leq 1$ , occur twice, each time with multiplicity  $\pm 1$ . Hence in a linear dependent linear combination of the equations

considered, the terms  $v_{i,j}^{k,k}$ ,  $i, j \in [n-1]$ ,  $i < j$ ,  $k \in [n-1]$ , in (29) – (31) have to cancel out. But this is only possible in one unique way:

$$\sum_{\substack{i,j \in [n-1] \\ i < j}} \left[ -\left(v_i^1 + v_j^1 + v_{i,j}^{1,1}\right) + (-1)^{n-1} \left(v_i^{n-1} + v_j^{n-1} - v_{i,j}^{n-1,n-1}\right) + \sum_{k=2}^{n-1} (-1)^k \left(v_{i,j}^{k,k} + v_{i,j}^{k-1,k-1} - v_{i,j}^{k,k-1} - v_{i,j}^{k-1,k}\right) \right] = \\ \sum_{\substack{i,j \in [n-1] \\ i < j}} \left[ -\left(v_i^1 + v_j^1\right) + (-1)^{n-1} \left(v_i^{n-1} + v_j^{n-1}\right) + \sum_{k=2}^{n-1} (-1)^{k+1} \left(v_{i,j}^{k,k-1} + v_{i,j}^{k-1,k}\right) \right].$$

Next canceling out the remaining quadratic terms in the above equation is again only possible in a unique way by adding the equalities  $(27)_{k-j=1}$  and  $(28)_{k-j=1}$  with appropriate signs. The remaining linear terms also cancel out and hence we have found a linear dependent linear combination of the equalities  $(27)_{k-j=1}$ ,  $(28)_{k-j=1}$ , (29), (30) and (31):

$$\sum_{\substack{i,j \in [n-1] \\ i < j}} \left[ -\left(v_i^1 + v_j^1\right) + (-1)^{n-1} \left(v_i^{n-1} + v_j^{n-1}\right) + \sum_{k=2}^{n-1} (-1)^{k+1} \left(v_{i,j}^{k,k-1} + v_{i,j}^{k-1,k}\right) \right] + \\ \sum_{k=2}^{n-1} \left[ \sum_{i \in [n-1]} (-1)^k \left(v_i^{k-1} - v_i^k - v_{i,i}^{k-1,k}\right) + (-1)^{k+1} \left( (2k-n-1) \left( \sum_{i \in [n-1]} \left(v_i^{k-1} + v_i^k\right) \right) - \sum_{i,j \in [n-1]} v_{i,j}^{k-1,k} \right) \right] = \\ \sum_{k=2}^n \left( (-1)^{k+1} \sum_{i,j \in [n-1]} \left(v_{i,j}^{k-1,k} - v_{i,j}^{k-1,k}\right) + (-1)^{k+1} \sum_{i \in [n-1]} \left( (2k-n)v_i^k - (2k-n)v_i^k \right) \right) + \\ (n-2) \sum_{i \in [n-1]} \left(v_i^1 - v_i^1\right) + (-1)^{n-1} (n-2) \sum_{i \in [n-1]} \left(v_i^{n-1} - v_i^{n-1}\right) = 0.$$

In summary the  $\frac{(n+1)n(n-2)}{2}$  equalities  $(27)_{k-j=1}$ ,  $(28)_{k-j=1}$ , (29), (30) and (31) are linear dependent and have rank  $\frac{(n+1)n(n-2)}{2} - 1$  because all choices for obtaining a linear dependent linear combination were uniquely determined. The remaining  $\frac{n(n-2)(n-3)}{2}$  equalities  $(27)_{k-j \neq 1}$  and  $(28)_{k-j \neq 1}$  are linear independent: The products of the cut variables  $v_{i,i}^{j,k}$ ,  $i, j, k \in [n-1]$ ,  $k \geq j+2$  occur once in  $(27)_{k-j \neq 1}$  and once in  $(28)_{k-j \neq 1}$ , each time with multiplicity  $\pm 1$ . To cancel them out the two particular equalities have to be combined in a unique way, resulting in a linear term in cut variables unequal to zero. Summing up, we have shown that the equalities (27) – (31) have rank  $\frac{(n+1)n(n-2)}{2} - 1 + \frac{n(n-2)(n-3)}{2} = n(n-1)(n-2) - 1$ .  $\square$

We can also show that the the equalities (27) suffice together with the implicitly assured linear constraints (16) to ensure monotonicity in the variable vector for all vertices.

**Lemma 9.** *The monotonicity constraints (15) are assured by the equalities (16) and (27) together with  $Z_s \in \mathcal{E}$ .*

*Proof.*  $Z_s \in \mathcal{E}$  guarantees  $-1 \leq v_{i,i}^{j,k} \leq 1$ ,  $i, j, k \in [n-1]$ ,  $j < k$ . Now applying the equalities (27) yields

$$v_i^k - v_i^j = 1 - v_{i,i}^{j,k} \geq 0, \quad i, j, k \in [n-1], \quad j < k.$$

Finally (15) for all vertices  $i \in [n-1]$  together with (16) also ensures (15) for vertex  $n$ .  $\square$

In summary  $(\text{SDP}_0^{\text{cut}})$  can be tightened by adding the equalities analyzed above that implicitly ensure (15):

$$\max \{ \langle C_s, Z_s \rangle : Z_s \text{ satisfies (27) – (31), } Z_s \in \mathcal{E} \}, \quad (\text{SDP}_1^{\text{cut}})$$

Now we can further improve the relaxation strength of  $(\text{SDP}_1^{\text{cut}})$  by adding several types of inequalities valid for  $\mathcal{P}_{\text{LOPCUT}}$ . First we discuss inequalities obtained by exploiting the structure of  $v$  induced by (15) and (16). Secondly we suggest valid inequalities associated to the integrality conditions  $v \in \{-1, 1\}^{(n-1)^2}$ .

**Lemma 10.** *The following inequality constraints hold for  $Z_s \in \mathcal{P}_{\text{LOPCUT}}$ :*

$$-v_{i,j}^{h,m} - v_{i,j}^{g,l} + v_{i,j}^{g,m} + v_{i,j}^{h,l} \leq 0, \quad i, j, g, h, l, m \in [n-1], \quad g < h, l < m, \quad (37)$$

$$2(m-l)(v_i^g - v_i^h) + \sum_{j \in [n-1]} (v_{i,j}^{h,m} + v_{i,j}^{g,l} - v_{i,j}^{g,m} - v_{i,j}^{h,l}) \leq 0, \quad i, g, h, l, m \in [n-1], \quad g < h, l < m. \quad (38)$$

*Proof.* Using  $V = vv^\top$  and (16) we can rewrite (37) and (38) as

$$-v_{i,j}^{h,m} - v_{i,j}^{g,l} + v_{i,j}^{g,m} + v_{i,j}^{h,l} = (v_i^h - v_i^g)(v_j^l - v_j^m) \leq 0, \quad i, j, g, h, l, m \in [n-1], \quad g < h, l < m, \quad (39)$$

$$2(l-m)v_i^h + 2(m-l)v_i^g + \sum_{j \in [n-1]} (v_{i,j}^{h,m} + v_{i,j}^{g,l} - v_{i,j}^{g,m} - v_{i,j}^{h,l}) = (v_i^h - v_i^g) \left( 2(l-m) + \sum_{j \in [n-1]} v_j^l - \sum_{j \in [n-1]} v_j^m \right) = (v_i^h - v_i^g)(v_n^l - v_n^m) \leq 0, \quad i, g, h, l, m \in [n-1], \quad g < h, l < m. \quad (40)$$

Now (39) holds as (15) ensures  $v_i^h - v_i^g \geq 0$  and  $v_j^l - v_j^m \leq 0$ . Furthermore the inequalities (40) are valid for  $\mathcal{P}_{\text{LOPCUT}}$  because (16) yields  $2(l-m) + \sum_{j \in [n-1]} v_j^l - \sum_{j \in [n-1]} v_j^m = v_n^l - v_n^m$  and (15) guarantees  $v_i^h - v_i^g \geq 0$ ,  $v_n^l - v_n^m \leq 0$ .  $\square$

Next we show that the inequalities (37) and (38) are linear combinations of the following smaller set of constraints.

**Lemma 11.** *Inequalities (37) and (38) are assured by*

$$-v_{i,j}^{h,l} - v_{i,j}^{h-1,l-1} + v_{i,j}^{h-1,l} + v_{i,j}^{h,l-1} \leq 0, \quad i, j \in [n-1], \quad i < j, \quad h, l \in [n-1], \quad h \neq 1, \quad l \neq 1. \quad (41)$$

$$-2v_i^h + 2v_i^{h-1} + \sum_{j \in [n-1]} (v_{i,j}^{h,l} + v_{i,j}^{h-1,l-1} - v_{i,j}^{h-1,l} - v_{i,j}^{h,l-1}) \leq 0, \quad i, h, l \in [n-1], \quad h \neq 1, \quad l \neq 1. \quad (42)$$

*Proof.* Adding (41) <sub>$h=s, l=t$</sub>  and (41) <sub>$h=s-1, l=t$</sub>  yields

$$-v_{i,j}^{s,t} - v_{i,j}^{s-2,t-1} + v_{i,j}^{s-2,t} + v_{i,j}^{s,t-1} \leq 0, \quad i, j \in [n-1], \quad i < j.$$

Additionally adding (41) <sub>$h=s, l=t-1$</sub>  and (41) <sub>$h=s-1, l=t-1$</sub>  to the above inequalities gives

$$-v_{i,j}^{s,t} - v_{i,j}^{s-2,t-2} + v_{i,j}^{s-2,t} + v_{i,j}^{s,t-2} \leq 0, \quad i, j \in [n-1], \quad i < j.$$

Thus any inequality of type (37) can be written as a (telescoping) sum of particular inequalities from (41). Analogically (38) is assured by (42).  $\square$

Adding the inequalities discussed above to  $(\text{SDP}_1^{\text{cut}})$  yields the following stronger semidefinite relaxation:

$$\max \{ \langle C_s, Z_s \rangle : Z_s \text{ satisfies (27) – (31), (41) and (42), } Z_s \in \mathcal{E} \}, \quad (\text{SDP}_2^{\text{cut}})$$

As  $Z_s \in \mathcal{P}_{\text{LOPCUT}}$  is actually a matrix with  $\{-1, 1\}$  entries, we can also tighten our semidefinite relaxations by asking for  $Z_s \in \mathcal{M}$ , see also (21). Furthermore also vertex  $n$  has to fulfill the triangle inequalities. The corresponding

inequalities are obtained with the help of (16) and read as follows:

$$\begin{aligned}
& -v_{i,j}^{k,l} + \sum_{h \in [n-1]} (v_{i,h}^{k,m} + v_{j,h}^{l,m}) - (2m-n)(v_i^k + v_j^l) \leq 1, \quad i, j, k, l, m \in [n-1], \\
& -v_{i,j}^{k,l} - \sum_{h \in [n-1]} (v_{i,h}^{k,m} + v_{j,h}^{l,m}) + (2m-n)(v_i^k + v_j^l) \leq 1, \quad i, j, k, l, m \in [n-1], \\
& v_{i,j}^{k,l} + \sum_{h \in [n-1]} (v_{i,h}^{k,m} - v_{j,h}^{l,m}) - (2m-n)(v_i^k - v_j^l) \leq 1, \quad i, j, k, l, m \in [n-1], \\
& v_{i,j}^{k,l} + \sum_{h \in [n-1]} (-v_{i,h}^{k,m} + v_{j,h}^{l,m}) - (2m-n)(-v_i^k + v_j^l) \leq 1, \quad i, j, k, l, m \in [n-1],
\end{aligned} \tag{43}$$

$$\begin{aligned}
& \sum_{j \in [n-1]} (v_{i,j}^{k,l} + v_{i,j}^{k,m} + (2l-n)v_j^m + (2m-n)v_j^l) - \sum_{j,h \in [n-1]} v_{j,h}^{l,m} \\
& \quad - (2m+2l-2n)v_i^k \leq 1 + (2l-n)(2m-n), \quad i, k, l, m \in [n-1], \\
& \sum_{j \in [n-1]} (v_{i,j}^{k,l} - v_{i,j}^{k,m} - (2l-n)v_j^m - (2m-n)v_j^l) + \sum_{j,h \in [n-1]} v_{j,h}^{l,m} \\
& \quad + (2m-2l)v_i^k \leq 1 - (2l-n)(2m-n), \quad i, k, l, m \in [n-1], \\
& \sum_{j \in [n-1]} (-v_{i,j}^{k,l} + v_{i,j}^{k,m} - (2l-n)v_j^m - (2m-n)v_j^l) + \sum_{j,h \in [n-1]} v_{j,h}^{l,m} \\
& \quad - (2m-2l)v_i^k \leq 1 - (2l-n)(2m-n), \quad i, k, l, m \in [n-1], \\
& \sum_{j \in [n-1]} (-v_{i,j}^{k,l} - v_{i,j}^{k,m} + (2l-n)v_j^m + (2m-n)v_j^l) - \sum_{j,h \in [n-1]} v_{j,h}^{l,m} \\
& \quad + (2m+2l-2n)v_i^k \leq 1 + (2l-n)(2m-n), \quad i, k, l, m \in [n-1],
\end{aligned} \tag{44}$$

$$\begin{aligned}
& \sum_{j \in [n-1]} ((2k+2m-2n)v_j^l + (2k+2l-2n)v_j^m + (2m+2l-2n)v_j^k) - \sum_{j,h \in [n-1]} (v_{j,h}^{k,l} + v_{j,h}^{k,m} + v_{j,h}^{l,m}) \\
& \quad \leq 1 + (2l-n)(2m-n) + (2k-n)(2l-n) + (2k-n)(2m-n), \quad k, l, m \in [n-1], \\
& \sum_{j \in [n-1]} ((2k-2m)v_j^l - (2k+2l-2n)v_j^m + (-2m+2l)v_j^k) - \sum_{j,h \in [n-1]} (v_{j,h}^{k,l} - v_{j,h}^{k,m} - v_{j,h}^{l,m}) \\
& \quad \leq 1 - (2l-n)(2m-n) + (2k-n)(2l-n) - (2k-n)(2m-n), \quad k, l, m \in [n-1], \\
& \sum_{j \in [n-1]} (-(2k+2m-2n)v_j^l + (2k-2l)v_j^m + (2m-2l)v_j^k) - \sum_{j,h \in [n-1]} (-v_{j,h}^{k,l} + v_{j,h}^{k,m} - v_{j,h}^{l,m}) \\
& \quad \leq 1 - (2l-n)(2m-n) - (2k-n)(2l-n) + (2k-n)(2m-n), \quad k, l, m \in [n-1], \\
& \sum_{j \in [n-1]} ((-2k+2m)v_j^l + (-2k+2l)v_j^m - (2m+2l-2n)v_j^k) - \sum_{j,h \in [n-1]} (-v_{j,h}^{k,l} - v_{j,h}^{k,m} + v_{j,h}^{l,m}) \\
& \quad \leq 1 + (2l-n)(2m-n) - (2k-n)(2l-n) - (2k-n)(2m-n), \quad k, l, m \in [n-1],
\end{aligned} \tag{45}$$

Adding  $Z_s \in \mathcal{M}$  and (43)–(45) to  $(\text{SDP}_1^{\text{cut}})$  and  $(\text{SDP}_2^{\text{cut}})$  respectively yields the following two SDP relaxations:

$$\max \{ \langle C_s, Z_s \rangle : Z_s \text{ satisfies (27) – (31) and (43) – (45), } Z_s \in (\mathcal{E} \cap \mathcal{M}) \}, \tag{SDP}_3^{\text{cut}}$$

$$\max \{ \langle C_s, Z_s \rangle : Z_s \text{ satisfies (27) – (31), (41), (42) and (43) – (45), } Z_s \in (\mathcal{E} \cap \mathcal{M}) \}. \tag{SDP}_4^{\text{cut}}$$

We can also reformulate the monotonicity constraints on the cut variables as linear-quadratic inequalities (as we

have done it above for the 3-cycle inequalities (6), see constraints (22)) with the help of the approach suggested by Lovász and Schrijver [41]. Hence multiplying the monotonicity constraints (15) by the nonnegative expressions  $(1 - v_h^l)$  and  $(1 + v_h^l)$  gives

$$v_i^j - v_{i,h}^{j,l} - v_i^k + v_{i,h}^{k,l} \leq 0, \quad v_i^j + v_{i,h}^{j,l} - v_i^k - v_{i,h}^{k,l} \leq 0, \quad i, j, k, h, l \in [n-1], j < k, \quad (46)$$

and correspondent inequalities if vertex  $n$  is involved:

$$2(k-j)v_h^l + \sum_{i \in [n-1]} (v_i^k - v_i^j + v_{i,h}^{j,l} - v_{i,h}^{k,l}) \leq 2(k-j), \quad j, k, h, l \in [n-1], j < k, \quad (47)$$

$$2(j-k)v_h^l + \sum_{i \in [n-1]} (v_i^k - v_i^j - v_{i,h}^{j,l} + v_{i,h}^{k,l}) \leq 2(k-j), \quad j, k, h, l \in [n-1], j < k, \quad (48)$$

$$(2l-n)(v_i^k - v_i^j) + v_i^j + \sum_{h \in [n-1]} v_{i,h}^{j,l} - v_i^k - \sum_{h \in [n-1]} v_{i,h}^{k,l} \leq 0, \quad i, j, k, l \in [n-1], j < k, \quad (49)$$

$$(2l-n)(v_i^j - v_i^k) + v_i^j - \sum_{h \in [n-1]} v_{i,h}^{j,l} - v_i^k + \sum_{h \in [n-1]} v_{i,h}^{k,l} \leq 0, \quad i, j, k, l \in [n-1], j < k. \quad (50)$$

But the inequalities above are not helpful to further strengthen our relaxations as they are already implicitly contained in  $(\text{SDP}_3^{\text{cut}})$ .

**Lemma 12.** *The lifted monotonicity constraints (46) are assured by the equalities (27) together with  $Z_s \in \mathcal{M}$ .*

*Proof.* Applying the equalities (27) to the left hand side of (46) yields

$$-1 + v_{i,i}^{j,k} - v_{i,h}^{j,l} + v_{i,h}^{k,l}, \quad -1 + v_{i,i}^{j,k} + v_{i,h}^{j,l} - v_{i,h}^{k,l}, \quad i, j, k, h, l \in [n-1], j < k.$$

But the two expressions above are  $\leq 0$  due to  $Z_s \in \mathcal{M}$ .  $\square$

Analogical (but more cumbersome) calculations show that also (47) – (50) are already contained in  $(\text{SDP}_3^{\text{cut}})$ .

Finally we state explicitly how to model the binary ordering variables via the entries of matrix  $Z_s$  and then use this transformations to add the constraints (1) – (3) to  $(\text{SDP}_4^{\text{cut}})$ .

**Lemma 13.** *We can express the linear ordering variables as:*

$$x_{ij} = \frac{1}{4} \left[ 1 - v_j^1 + \sum_{k=2}^{n-1} (v_{i,j}^{k-1,k} - v_{i,j}^{k-1,k-1}) + v_i^{n-1} - v_{i,j}^{n-1,n-1} \right], \quad i, j \in [n-1], i \neq j, \quad (51)$$

$$x_{in} = \frac{1}{4} \left[ n-1 + \sum_{j \in [n-1]} v_j^1 + \sum_{k=2}^{n-1} \left( 2v_i^{k-1} + \sum_{j \in [n-1]} (v_{i,j}^{k-1,k-1} - v_{i,j}^{k-1,k}) \right) - (n-3)v_i^{n-1} + \sum_{j \in [n-1]} v_{i,j}^{n-1,n-1} \right], \quad i \in [n-1]. \quad (52)$$

$$x_{n,j} = \frac{1}{4} \left[ n-1 + (n-3)v_j^1 + \sum_{k=2}^{n-1} \left( -2v_j^k + \sum_{i \in [n-1]} (v_{i,j}^{k-1,k-1} - v_{i,j}^{k-1,k}) \right) - \sum_{i \in [n-1]} (v_i^{n-1} - v_{i,j}^{n-1,n-1}) \right], \quad j \in [n-1]. \quad (53)$$

*Proof.* Applying  $V = vv^\top$ , (10) and (16) to (51) and (52) yields

$$x_{ij} = \frac{1}{4} \left[ 2 + \sum_{k \in [n]} v_i^{k-1} (v_j^k - v_j^{k-1}) \right], \quad i, j \in [n-1], i \neq j$$

$$x_{in} = \frac{1}{4} \left[ 2 + \sum_{k \in [n]} v_i^{k-1} (v_n^k - v_n^{k-1}) \right], \quad i \in [n-1], \quad x_{n,j} = \frac{1}{4} \left[ 2 + \sum_{k \in [n]} v_n^{k-1} (v_j^k - v_j^{k-1}) \right], \quad j \in [n-1].$$

Due to (15)  $v_j^k - v_j^{k-1} \geq 0$ ,  $j, k \in [n]$ , holds and  $v_j^k - v_j^{k-1} = 2$ , if and only if vertex  $j$  is at position  $k$  in the linear ordering. Hence the term  $v_i^{k-1} (v_j^k - v_j^{k-1})$ ,  $i, j, k \in [n]$ ,  $i \neq j$ , is equal to 2, if and only if vertex  $j$  is at position  $k$  and

additionally vertex  $i$  comes before vertex  $j$  in the linear ordering. In summary  $\sum_{k \in [n]} v_i^{k-1} (v_j^k - v_j^{k-1})$ ,  $i, j \in [n]$ ,  $i < j$ , is equal to 2, if vertex  $i$  comes before vertex  $j$  in the linear ordering, and equal to 0 otherwise.  $\square$

Note that equation (51) has already been implicitly used by the deduction of the cost matrix  $C_a$  in (24).

**Corollary 14.** *The (LOP) objective function (4) and the (LOP) constraints (1) – (3) can be reformulated as linear-quadratic expressions in cut variables.*

In particular the equality constraints (1) for the (LOP) are already assured by (SDP<sub>1</sub><sup>cut</sup>).

**Lemma 15.** *The equality constraints (1) are linear combinations of the equalities (29) – (31).*

*Proof.* For two fixed vertices  $i$  and  $j$  with  $i, j \in [n-1]$ ,  $i < j$ , we combine (29) – (31) to obtain

$$2 - v_i^1 - v_j^1 - v_{i,j}^{1,1} + v_i^{n-1} + v_j^{n-1} - v_{i,j}^{n-1,n-1} + \sum_{k=2}^{n-1} (v_{i,j}^{k,k} + v_{i,j}^{k-1,k-1} - v_{i,j}^{k,k-1} - v_{i,j}^{k-1,k}) = 4(x_{ij} + x_{ji}).$$

Hence the constraints (1) not considering vertex  $n$  are linear combinations of the constraints (29) – (31).

Next we analyze the equalities (1) considering vertex  $n$ , i.e. our fixed vertices are  $i$ ,  $i \in [n-1]$ , and  $n$ . Adding up 4 times (52) and 4 times (53) yields

$$\begin{aligned} 4(x_{in} + x_{ni}) &= 2(n-1) + (n-1)v_i^1 + \sum_{j \in [n-1]} v_j^1 + \sum_{k=2}^{n-1} \sum_{j \in [n-1]} (v_{i,j}^{k,k} - v_{i,j}^{k-1,k} - v_{i,j}^{k,k-1} + v_{i,j}^{k-1,k-1}) \\ &\quad - (n-1)v_i^{n-1} - \sum_{j \in [n-1]} v_j^{n-1} + \sum_{j=1}^{n-1} v_{i,j}^{n-1,n-1}, \quad i \in [n-1]. \end{aligned} \tag{54}$$

But the above linear-quadratic expression is again a linear combination of the equalities (29) – (31): Summing up (29) for  $j \in [n-1]$ ,  $j \neq i$ , (30) for  $j \in [n-1]$ ,  $j \neq i$  and (31) for  $j, k \in [n-1]$ ,  $j \neq i$ ,  $k \neq 1$  gives (54).  $\square$

Finally adding the (LOP) inequality constraints (2) and (3), reformulated via the entries of matrix  $Z_s$ , to (SDP<sub>4</sub><sup>cut</sup>) gives the last relaxation of this section:

$$\max \{ \langle C_s, Z_s \rangle : Z_s \text{ satisfies (2), (3), (27) – (31), (41) and (42), } Z_s \in (\mathcal{E} \cap \mathcal{M}) \}. \tag{SDP}_5^{\text{cut}}$$

**Corollary 16.** *The semidefinite programming relaxation (SDP<sub>5</sub><sup>cut</sup>) is as least as strong as the linear programming relaxation (LP<sub>LOP</sub>).*

*Proof.* As (SDP<sub>5</sub><sup>cut</sup>) ensures all constraints of (LP<sub>LOP</sub>), the claim follows immediately.  $\square$

## 4. Extensions to Other Combinatorial Optimization Problems

In this section we first elaborate on the connection of our model using cut variables to the Max Cut problem in Subsection 4.1. In Subsections 4.2 and 4.3 we show that our model can be extended to two further vertex ordering problems, namely the Traveling Salesman and the Target Visitation Problem.

### 4.1. Max Cut With Constraints on the Partition

Let us start with reconsidering the formulation of the (LOP) from (24), where we now allow for an arbitrary cost matrix  $C_v : \begin{pmatrix} K_v & b^\top \\ b & B \end{pmatrix}$ :

$$\max \{ \langle C_v, Z_v \rangle : v \in \{-1, 1\}^{n(n-1)}, v \text{ satisfies (15) and (16)} \}. \tag{55}$$

Problem (55) is most naturally interpreted as the following combinatorial optimization problem that we introduced as Quadratic Position Problem (QPP) in our companion paper [32]. The input of the (QPP) consists of  $n$  vertices and individual and pairwise benefits  $b_i^k$ ,  $i \in [n]$ ,  $k \in [n-1]$ , and  $b_{i,j}^{k,l}$ ,  $i, j \in [n]$ ,  $i < j$ ,  $k, l \in [n-1]$ ,  $k \neq l$ . The optimization problem can be written down as

$$\max_{\pi \in \Pi_n} \sum_{\substack{i,j \in [n], i < j, \\ k,l \in [n-1], k \neq l}} (b_i^k w_i^k(\pi) + b_{i,j}^{k,l} w_i^k(\pi) w_j^l(\pi)), \quad (56)$$

where  $\Pi_n$  is the set of permutations of the vertices  $[n]$ , and  $w_i^k(\pi)$ ,  $i \in [n]$ ,  $k \in [n-1]$ , is 1, if vertex  $i$  is at position  $h$  with  $h \leq k$  in the particular ordering  $\pi \in \Pi_n$ . Otherwise  $w_i^k(\pi)$  is set to -1. Hence the individual benefits  $b_i^k$  are obtained, if vertex  $i$  is at location  $k$  or before and the pairwise benefits  $b_{i,j}^{k,l}$  are collected, if

- vertex  $i$  is at position  $k$  or before and vertex  $j$  is at position  $l$  or before,
- vertex  $i$  comes after position  $k$  and vertex  $j$  comes after position  $l$ .

Next we suggest another possible interpretations of (55) as a special Max Cut problem on  $n(n+1)$  vertices. We consider a weighted graph  $G = (V, E)$  with  $|V| = n(n+1)$  and weights  $w_e$  for  $e \in E$  that we summarize in the weighted adjacency matrix  $W$ . Now let us define the following optimization problem that asks for a maximum cut in  $G$ , where the partition  $(S, T)$  has to fulfill some additional properties:

$$\max \sum_{i \in S, j \in T, ij \in E} w_{ij}, \quad (57a)$$

$$\text{subject to: } S = \bigcup_{i \in [n]} S_i, \quad T = \bigcup_{i \in [n]} T_i, \quad S_i, T_i \subseteq [(i-1)(n+1) + 1, i(n+1)], \quad i \in [n], \quad (57b)$$

$$\max S_i < \min T_i, \quad i \in [n], \quad (57c)$$

$$|S_i| \neq |S_j|, \quad i, j \in [n], i < j. \quad (57d)$$

Condition (57b) models that the partition  $(S, T)$  consists of  $n$  smaller partitions on  $n+1$  consecutive vertices. Condition (57c) enforces that all disjoint sets  $S_i$  and  $T_i$ ,  $i \in [n]$ , contain vertices with consecutive numbers. Finally (57d) ensures that all disjoint sets  $S_i$ ,  $i \in [n]$ , (and hence also  $T_i$ ) have different cardinality.

Now we consider the Laplacian matrix  $L_G = \text{Diag}(We) - W$  of  $G$ , where  $e$  is the vector of all ones. We partition the vertex set  $V$  into two disjoint subsets  $A$  and  $B$  with

$$A := A_1 \cup A_2, \quad A_1 = \bigcup_{i \in [n]} (i-1)(n+1) + 1, \quad A_2 = \bigcup_{i \in [n]} i(n+1), \quad B := [n(n+1)] \setminus A. \quad (58)$$

Next we set

$$K_v = \sum_{\substack{i \in A_1, j \in A_2, \\ ij \in E}} w_{ij} + \frac{1}{2} \sum_{\substack{i \in A, j \in B, \\ ij \in E}} w_{ij}, \quad b = \frac{1}{4} \left( \sum_{i \in A_2} L_{B,i} - \sum_{i \in A_1} L_{B,i} \right), \quad B = \frac{1}{4} L_{B,B}. \quad (59)$$

Now we can establish a connection between the combinatorial optimization problems (55) and (57):

**Theorem 17.** *Problem (55) with  $C_v$  defined in (59) is equivalent to problem (57).*

*Proof.* Let us interpret the cut variables (9) together with the parameters (10) as follows:

$$v_i^{j-1} = \begin{cases} -1, & \text{if vertex } (i-1)(n+1) + j \text{ belongs to set } S, \\ +1, & \text{if vertex } (i-1)(n+1) + j \text{ belongs to set } T, \end{cases} \quad i \in [n], j = [n+1]. \quad (60)$$

Hence (60) connects the cut variables  $v$  to the partition  $(S, T)$  of the  $n(n+1)$  vertices of  $G$ . Clearly the constraints (15), (16) and (60) impose the same restrictions on  $(S, T)$  as the constraints (57b) – (57d). Finally the objective function



$\langle C_v, Z_v \rangle$  with  $C_v$  defined in (59) gives the sum of the weights of the edges in the cut  $(S, T)$ , where the constant term  $K_v$  and the linear terms in  $b$  account for the weights of edges incident to one or two vertices from the set  $A$  that are fixed by (10).  $\square$

#### 4.2. Traveling Salesman Problem

Let us now consider the probably most famous of all (combinatorial) optimization problems. The Traveling Salesman Problem (TSP) asks the following question: Given a list of cities and the distances between each pair of cities, what is the shortest possible tour that visits each city exactly once and returns to the origin city? Even though the (TSP) is NP-hard [37], a large number of heuristics and exact methods are known, so that some instances with tens of thousands of cities can be solved completely<sup>4</sup> and even problems with millions of cities can be approximated within a small fraction of 1%. Nonetheless the (TSP) continues to pose grand challenges. The Christofides's algorithm approximates the cost of an optimal symmetric (TSP) tour within the factor 1.5 [9]. Lampis [39] proved that no  $\frac{185}{184}$ -approximation algorithm exists for the symmetric (TSP) unless  $P = NP$ . Is it possible to further narrow this quite large gap? In the asymmetric (TSP) paths may not exist in both directions or the distances might be different, forming a directed graph. This may be e.g. due to traffic collisions, one-way streets and motorways. Papadimitriou and Vempala [47] proved that no  $\frac{118}{117}$ -approximation algorithm exists for the asymmetric (TSP) unless  $P = NP$ . It is an open question if a constant factor approximation exists. Results on the even more difficult non Euclidean (TSP) are e.g. discussed in [46]. We will see below that our semidefinite relaxations can be used to model all these variants of the (TSP). Hence they provide new polynomial-time convex approximations of (TSP) with a rich mathematical structure. Our hope is that they may help to improve approximation results for the (TSP) and maybe even can be used to answer one of the open question mentioned above.

We refer to the books [10, 26, 48] and the references therein for extensive material on the (TSP), its variants and various applications, details on many heuristic and exact methods and relevant theoretical results.

Now let us state one of the best-known integer programming formulation of the (TSP). First we introduce the traveling salesman variables

$$s_{ij} := \begin{cases} 1, & \text{if city } j \text{ is visited immediately after city } i, \\ 0, & \text{otherwise.} \end{cases} \quad (61)$$

Now for a given distance matrix  $D = (d_{ij})$  the (TSP) can be formulated as:

$$\min \sum_{\substack{i, j \in [n] \\ i \neq j}} d_{ij} s_{ij} \quad (62)$$

$$\text{subject to: } \sum_{\substack{j \in [n], \\ j \neq i}} s_{ij} = 1, \quad \sum_{\substack{j \in [n], \\ j \neq i}} s_{ji} = 1, \quad i \in [n], \quad (63)$$

$$\sum_{\substack{i, j \in S \\ i \neq j}} s_{ij} \leq |S| - 1, \quad \forall S \subset [n], \quad 2 \leq |S| \leq n, \quad (64)$$

$$s_{ij} \in \{0, 1\}, \quad i, j \in [n], \quad i \neq j.$$

Constraints (63) and (64) are the standard constraints for the asymmetric (TSP). If we replace the integrality conditions by the bound constraints

$$0 \leq s_{ij} \leq 1, \quad i, j \in [n], \quad i \neq j, \quad (65)$$

we obtain the standard relaxation  $(\text{LP}_{\text{TSP}})$  that was suggested by Dantzig et al. [12] in 1954. Its optimal value coincides with the LP bound of Held and Karp [28]. The integrality gap of  $(\text{LP}_{\text{TSP}})$  is conjectured to be  $\frac{4}{3}$  for the metric symmetric (TSP).

<sup>4</sup>The branch-and-cut algorithm by Applegate et al. [3] holds the current record, solving an instance with 85,900 cities.

Wolsey [54] showed that Christofides' algorithm [9] computes a tour of length at most  $\frac{3}{2}$  times the optimal value of  $(\text{LP}_{\text{TSP}})$  (see also [49]). An interesting question is whether a similar result may be proved for one of our semidefinite relaxations.

In the following we explain how to model the  $(\text{TSP})$  as a special  $(\text{QPP})$ : First we show that the  $(\text{TSP})$  variables can be expressed via the entries of matrix  $Z_s$  and then we use this property to reformulate the  $(\text{TSP})$  as a semidefinite optimization problem that yields provably stronger lower bounds than  $(\text{LP}_{\text{TSP}})$ .

**Lemma 18.** *We can express the traveling salesman variables (61) via the entries of matrix  $Z_s$ :*

$$s_{ij} = \frac{1}{4} \left[ 1 - v_{i,j}^{1,1} + v_{i,j}^{1,2} + v_j^2 + \sum_{k=2}^{n-2} (v_{i,j}^{k,k+1} - v_{i,j}^{k-1,k+1} - v_{i,j}^{k,k} + v_{i,j}^{k-1,k}) - v_i^{n-2} + v_{i,j}^{n-2,n-1} - v_{i,j}^{n-1,n-1} - v_{i,j}^{n-1,1} \right], \quad i, j \in [n-1], i \neq j, \quad (66)$$

$$s_{in} = \frac{1}{4} \left[ 5 - n + \sum_{j \in [n-1]} (v_{i,j}^{1,1} - v_{i,j}^{1,2} - v_j^2) + \sum_{k=2}^{n-2} \sum_{j \in [n-1]} (v_{i,j}^{k-1,k+1} - v_{i,j}^{k-1,k} - v_{i,j}^{k,k+1} + v_{i,j}^{k,k}) + (n-1)v_i^{n-2} + \sum_{j \in [n-1]} (-v_{i,j}^{n-2,n-1} + v_{i,j}^{n-1,n-1} + v_{i,j}^{n-1,1}) \right], \quad i \in [n-1], \quad (67)$$

$$s_{nj} = \frac{1}{4} \left[ 5 - n + \sum_{i \in [n-1]} (v_{i,j}^{1,1} - v_{i,j}^{1,2}) - (n-1)v_j^2 + \sum_{k=2}^{n-2} \sum_{i \in [n-1]} (v_{i,j}^{k-1,k+1} - v_{i,j}^{k,k+1} - v_{i,j}^{k-1,k} + v_{i,j}^{k,k}) + \sum_{i \in [n-1]} (v_i^{n-2} - v_{i,j}^{n-2,n-1} + v_{i,j}^{n-1,n-1} + v_{i,j}^{n-1,1}) \right], \quad j \in [n-1], \quad (68)$$

*Proof.* Using  $V = vv^\top$  and (10) in (66) yields

$$s_{ij} = \frac{1}{4} \left[ \sum_{k \in [n-1]} (v_i^k - v_i^{k-1})(v_j^{k+1} - v_j^k) + (1 - v_i^{n-1})(v_j^1 + 1) \right], \quad i, j \in [n-1], i \neq j.$$

Due to (15),  $v_i^k - v_i^{k-1} \geq 0$ ,  $i \in [n-1]$ , and  $v_j^{k+1} - v_j^k \geq 0$ ,  $j \in [n-1]$ , hold and  $(v_i^k - v_i^{k-1})(v_j^{k+1} - v_j^k) = 1$ , if city  $i$  is visited directly before city  $j$  or if city  $i$  is the last city of the tour (i.e. the  $n^{\text{th}}$  city) and city  $j$  is the first city of the tour.

Using  $V = vv^\top$ , (10) and (16) in (67) and (68) respectively gives

$$s_{in} = \frac{1}{4} \left[ \sum_{k \in [n-1]} (v_i^k - v_i^{k-1})(v_n^{k+1} - v_n^k) + (1 - v_i^{n-1})(v_n^1 + 1) \right], \quad i \in [n-1],$$

$$s_{ni} = \frac{1}{4} \left[ \sum_{k \in [n-1]} (v_n^k - v_n^{k-1})(v_i^{k+1} - v_i^k) + (1 - v_n^{n-1})(v_i^1 + 1) \right], \quad i \in [n-1].$$

Analogically to the reasoning above, (15) ensures that both  $s_{in}$  and  $s_{ni}$  are equal to 1, if city  $n$  is visited directly after (first type) or directly before (second type) city  $i$ .  $\square$

**Corollary 19.** *The  $(\text{TSP})$  objective function (62) and the  $(\text{TSP})$  constraints (63) – (65) can be reformulated via the entries of matrix  $Z_s$ .*

In particular the equality constraints (63) for the  $(\text{TSP})$  are already assured by  $(\text{SDP}_1^{\text{cut}})$ .

**Lemma 20.** *The equality constraints (63) for the  $(\text{TSP})$  are assured by (27) – (31).*

*Proof.* Rewriting (66) for  $i = j$  gives:

$$\begin{aligned} s_{ii} &= \frac{1}{4} \left[ v_{i,i}^{1,2} + v_i^2 + \sum_{k=2}^{n-2} (v_{i,i}^{k,k+1} - v_{i,i}^{k-1,k+1} - 1 + v_{i,i}^{k-1,k}) - v_i^{n-2} + v_{i,i}^{n-2,n-1} - 1 - v_{i,i}^{n-1,1} \right] \\ &= \frac{1}{4} \left[ (-v_i^1 + v_i^2 + v_{i,i}^{1,2} - 1) + \sum_{k=2}^{n-2} [(-v_i^k + v_i^{k+1} + v_{i,i}^{k,k+1} - 1) + (v_i^{k-1} - v_i^{k+1} - v_{i,i}^{k-1,k+1} + 1) + (-v_i^{k-1} + v_i^k + v_{i,i}^{k-1,k} - 1)] \right. \\ &\quad \left. + (-v_i^{n-2} + v_i^{n-1} + v_{i,i}^{n-2,n-1} - 1) + (v_i^1 - v_i^{n-1} - v_{i,i}^{n-1,1} + 1) \right], \quad i \in [n-1]. \end{aligned}$$

Now applying (27) yields  $s_{ii} = 0$ ,  $i \in [n-1]$ . In the same way we can show that the equalities (28)–(31) ensure  $s_{nn} = 0$ .

Now fixing a city  $i$ ,  $i \in [n-1]$ , and summing up (66) for  $j \in [n-1]$ , gives

$$\begin{aligned} \frac{1}{4} \left[ n-1 - \sum_{j \in [n-1]} v_{i,j}^{1,1} + \sum_{j \in [n-1]} v_{i,j}^{1,2} + \sum_{j \in [n-1]} v_j^2 + \sum_{k=2}^{n-2} \sum_{j \in [n-1]} (v_{i,j}^{k,k+1} - v_{i,j}^{k-1,k+1} - v_{i,j}^{k,k} + v_{i,j}^{k-1,k}) \right. \\ \left. - (n-1)v_i^{n-2} + \sum_{j \in [n-1]} v_{i,j}^{n-2,n-1} - \sum_{j \in [n-1]} v_{i,j}^{n-1,n-1} - \sum_{j \in [n-1]} v_{i,j}^{n-1,1} \right], \end{aligned}$$

Finally adding (67) yields 1 and hence we proved  $\sum_{j \in [n], j \neq i} s_{ij} = 1$ ,  $i \in [n-1]$ . The remaining (TSP) equalities, i.e.  $\sum_{\substack{j \in [n], \\ j \neq i}} s_{ji} = 1$ ,  $i \in [n-1]$ , and  $\sum_{\substack{j \in [n], \\ j \neq n}} s_{nj} = \sum_{\substack{j \in [n], \\ j \neq n}} s_{jn} = 1$ , can be shown in an analogous fashion.  $\square$

Finally adding the (TSP) inequalities (64) and (65), reformulated as linear-quadratic terms in cut variables, to  $(\text{SDP}_4^{\text{cut}})$  gives the following relaxation:

$$\max \{ \langle C_s, Z_s \rangle : Z_s \text{ satisfies (27) – (31), (41), (42), (64) and (65), } Z_s \in (\mathcal{E} \cap \mathcal{M}) \}. \quad (\text{SDP}_6^{\text{cut}})$$

**Corollary 21.** *The semidefinite programming relaxation  $(\text{SDP}_6^{\text{cut}})$  is as least as strong as the linear programming relaxation  $(\text{LP}_{\text{TSP}})$ .*

*Proof.* As  $(\text{SDP}_6^{\text{cut}})$  ensures all constraints of  $(\text{LP}_{\text{TSP}})$ , the claim follows immediately.  $\square$

Note that the formulation from Theorem 5 is in fact (with an appropriately defined cost matrix) also a formulation of the (TSP) with only polynomially many constraints.

Let us now propose another formulation of the (TSP) with polynomially many constraints. To do so we first suggest an additional constraint type that connects the (LOP) with the (TSP) variables:

$$s_{ij} - x_{ij} - \frac{1}{n-1} \sum_{\substack{k \in [n], \\ k \neq i}} x_{ki} \leq 0, \quad i, j \in [n], i \neq j. \quad (69)$$

The term  $\frac{1}{n-1} \sum_{\substack{k \in [n], \\ k \neq i}} x_{ki}$  ensures that we are allowed to go from the last city visited back to the starting city and hence to close the tour. Now we exploit this link between (LOP) and (TSP) variables in the following lemma:

**Lemma 22.** *For  $x_{ij} \in \{0, 1\}$ ,  $i, j \in [n]$ ,  $i \neq j$ , and  $s_{ij} \in \{0, 1\}$ ,  $i, j \in [n]$ ,  $i \neq j$ , the subtour elimination constraints (64) are implicitly satisfied by the (LOP) constraints (1) and (2) together with the (TSP) equalities (63) and the linking constraints (69).*

*Proof.* The (LOP) constraints ensure that the ordering variables  $x_{ij}$ ,  $i, j \in [n]$ ,  $i \neq j$ , describe a feasible ordering of  $n$  vertices. Now the linking constraints (69) together with the (TSP) equalities further guarantee that the (TSP) variables describe the same ordering as a tour.  $\square$

But let us point out that the subtour elimination constraints (64) may still help to tighten relaxations where the linking constraints and all other (LOP) and (TSP) constraints are satisfied. Finally we discuss how the linking constraints (69) can be reformulated via the entries of matrix  $Z_s$ : For  $i, j \in [n-1]$ ,  $i \neq j$ , we just have to subtract the term  $\frac{1}{4} \left(1 + v_j^1 - v_i^{n-1} - v_{i,j}^{n-1,1}\right)$  from (66) to rule out the edges from the last city visited to the starting city. Hence in this way we account for the term  $-\frac{1}{n-1} \sum_{\substack{k \in [n], \\ k \neq i}} x_{ki}$  in (69). Now subtracting (51) from the adapted version of (66) gives:

$$v_j^2 + \sum_{k=2}^{n-2} (v_{i,j}^{k-1,k} - v_{i,j}^{k-1,k+1}) + v_{i,j}^{n-2,n-1} - v_i^{n-1} \leq 1, \quad i, j \in [n-1], i \neq j.$$

Note that these differences of (TSP) and (LOP) variables result in easier expressions in cut variables than the (TSP) variables. Similar simplifications also occur when rewriting (69) for  $i = n$  and  $j = n$ :

$$\begin{aligned} - \sum_{j \in [n-1]} v_j^2 - 2 \sum_{k=2}^{n-1} v_i^{k-1} + \sum_{k=2}^{n-2} \sum_{j \in [n-1]} (v_{i,j}^{k-1,k+1} - v_{i,j}^{k-1,k}) + (n-1)v_i^{n-2} - \sum_{j \in [n-1]} v_{i,j}^{n-2,n-1} &\leq n-3, \quad i \in [n-1], \\ -(n-1)v_j^2 + 2 \sum_{k=2}^{n-1} v_j^k + \sum_{k=2}^{n-2} \sum_{i=1}^{n-1} (v_{i,j}^{k-1,k+1} - v_{i,j}^{k-1,k}) + \sum_{i=1}^{n-1} (v_i^{n-2} - v_{i,j}^{n-2,n-1}) &\leq n-3, \quad j \in [n-1]. \end{aligned}$$

As all constraints valid for the (TSP) also hold for the (LOP) and vice versa, we combine all constraint types discussed in this paper in our strongest relaxation ( $\text{SDP}_7^{\text{cut}}$ ):

$$\max \{ \langle C_s, Z_s \rangle : Z_s \text{ satisfies (2), (3), (27) – (31), (41), (42), (64), (65) and (69), } Z_s \in (\mathcal{E} \cap \mathcal{M}) \}. \quad (\text{SDP}_7^{\text{cut}})$$

Finally note that all constraint types in ( $\text{SDP}_7^{\text{cut}}$ ) may help to improve the bounds obtained.

**Lemma 23.** *No constraint types in ( $\text{SDP}_7^{\text{cut}}$ ) are dominated.*

*Proof.* We can give an instance for each constraint type, where this particular constraint type helps to further improve the value of the relaxation, even if all other constraints are already satisfied. Most conveniently the respective instances are generated by replicating the particular constraints in the cost function.  $\square$

### 4.3. Target Visitation Problem

Another vertex ordering problem is the Target Visitation Problem (TVP) that has been suggested in [25]. It is a composition of the (LOP) and the (TSP), i.e. we are looking for a permutation  $(p_1, p_2, \dots, p_n)$  of  $n$  targets with given pairwise weights  $w_{ij}$ ,  $i, j \in [n]$ , and pairwise distances  $d_{ij}$ ,  $i, j \in [n]$ , maximizing the objective function

$$\sum_{\substack{i, j \in [n] \\ i < j}} w_{p_i p_j} - \left( \sum_{i \in [n-1]} d_{p_i p_{i+1}} + d_{p_n p_1} \right).$$

As the (LOP) and the (TSP) are special cases of the (TVP), the (TVP) is also NP-hard.

The formulation of the (TVP) was inspired by the use of single unmanned aerial vehicles that have been used increasingly over the last decades. Applications of the (TVP) include environmental assessment, combat search and rescue and disaster relief [25].

Clearly our relaxations ( $\text{SDP}_1^{\text{cut}}$ ) – ( $\text{SDP}_7^{\text{cut}}$ ) can also be used to obtain upper bounds for the (TVP), if we define the  $C_s$  appropriately. For more information on the (TVP) we refer to our companion paper [32], where we apply the semidefinite relaxations using cut variables from this paper to design an exact semidefinite optimization approach that is able to obtain reasonable upper and lower bounds on a variety of benchmark instances with up to 50 targets. Also note that Hildenbrandt et al. [31] recently conducted the first polyhedral study of the (TVP) polytope. They present several possible IP-models for the (TVP) and compare them to their usability for branch-and-cut approaches. Based on their findings Hildenbrandt et al. [30] are currently developing an exact IP approach for the (TVP) that has strong potential to solve large-scale (TVP) instances to optimality.

## 5. Numerical Examples

In this section we aim to compare the relaxation strength of our new relaxations  $(SDP_1^{\text{cut}}) - (SDP_7^{\text{cut}})$  with other relaxations from the literature for the (LOP) (Subsection 5.1) and the (TSP) (Subsection 5.2) respectively. It is not difficult to create small but hard instances by just considering the facets of the (LOP) and the (TSP) polytopes for small dimensions. We will see below that our new relaxations are worse than the best relaxations for the respective problem type. However notice that  $(SDP_1^{\text{cut}}) - (SDP_7^{\text{cut}})$  are the only relaxations considered that can be applied to both the (LOP) and the (TSP). We exploited exactly this property in our companion paper [32].

### 5.1. Comparison of Relaxations for the Linear Ordering Problem on Small Facets

In this subsection we compare the  $(LP_{\text{LOP}})$  to the semidefinite relaxations from Section 3. We consider the full description of the linear ordering polytope in small dimensions, and try to recover the correct right hand side of the classes of facets for  $n \in \{6, 7\}$ . The members of each class are equal modulo a permutation of the vertices, and we need therefore only consider one representative per class. The facets are collected under <http://comopt.ifi.uni-heidelberg.de/software/SMAPO/lop/lop.html>. As usual,  $n$  denotes the number of vertices and  $opt$  gives the optimal solution. All relaxations are solved to optimality using the standard settings of SEDUMI [51]. Table 1 summarizes our results.

We observe that the semidefinite relaxations provide a substantial improvement over  $(LP_{\text{LOP}})$  in the approximation of the linear ordering polytope of small dimensions. Furthermore the relaxations on ordering variables seem in general tighter than the relaxations on cut variables. Nonetheless the strongest relaxation using ordering variables  $(SDP_4^{\text{ord}})$  does not dominate relaxations  $(SDP_5^{\text{cut}}) - (SDP_7^{\text{cut}})$  (on these instances), see facet number 14.

### 5.2. Comparison of Relations for the Traveling Salesman Problem on Small Facets

For the (TSP) we consider instances corresponding to classes of facets of the symmetric (TSP) on 8 vertices [8]. The facets are collected under <http://comopt.ifi.uni-heidelberg.de/software/SMAPO/tsp/tsp.html>. These benchmark instances were also used in [11, 13, 14, 17]. Cvetković et al. [11] suggested a semidefinite relaxation that is however dominated by the Held-Karp bound [23]. De Klerk et al. [14] and de Klerk and Sotirov [13] proposed SDP relaxations for the symmetric (TSP) that can be obtained via an SDP relaxation of the more general quadratic assignment problem and are motivated by the theory of association schemes. Finally Fischer [17] tested her polyhedral linear relaxation for the quadratic (TSP) on the facets of the symmetric (TSP). In Table 2 we compare  $(SDP_1^{\text{cut}}) - (SDP_7^{\text{cut}})$  with the Held-Karp bound [28] and the other relaxations described above.

We already know from Corollary 21 that  $(SDP_6^{\text{cut}})$  dominates the Held-Karp bound [28] and hence also the SDP relaxation from [11]. Additionally we observe that on the one hand  $(SDP_4^{\text{cut}})$  dominates the semidefinite relaxation from [14] on all instances considered and on the other hand our strongest relaxation  $(SDP_7^{\text{cut}})$  is dominated by the relaxations from [13] and [17]. It would be interesting to investigate, if general results between the strength of the different relaxations could be established. Finally note that  $(SDP_1^{\text{cut}}) - (SDP_7^{\text{cut}})$  are the only relaxations from this study that can also be applied to the asymmetric and even to the non-Euclidean (TSP)

## 6. Conclusion

In this paper we suggested a formulation of the Linear Ordering Problem with the help of cut variables and conducted a polyhedral study of the corresponding polytope. In this way we improved the semidefinite relaxation proposed by Newman [43] that proved well-suited for obtaining strong approximation results for the Linear Ordering Problem. We elaborated on the connection of this model using cut variables to the Max Cut problem and related our new semidefinite relaxations to other linear and semidefinite relaxations for the Linear Ordering Problem and for the Traveling Salesman Problem.

In the companion paper [32] we show that semidefinite relaxations using cut variables can be applied to obtain reasonable bounds for difficult, large-scale combinatorial optimization problems. As our relaxations provide new polynomial-time convex approximations of the (LOP) and (TSP) with a rich mathematical structure, we hope that they may be helpful to improve approximation results for vertex ordering problems. Their close relation to the Max Cut problem could be a useful property in that direction and therefore a possible subject of future research.

Linear Ordering Problem Instances Constructed from Facet Defining Inequalities													
Facet	$n$	Opt	(LP)	SDP <sub>1</sub> <sup>ord</sup>	SDP <sub>2</sub> <sup>ord</sup>	SDP <sub>3</sub> <sup>ord</sup>	SDP <sub>4</sub> <sup>ord</sup>	SDP <sub>1</sub> <sup>cut</sup>	SDP <sub>2</sub> <sup>cut</sup>	SDP <sub>3</sub> <sup>cut</sup>	SDP <sub>4</sub> <sup>cut</sup>	SDP <sub>5</sub> <sup>cut</sup>	SDP <sub>7</sub> <sup>cut</sup>
FC3	6	7	7.5	7.35	7	7	7	7.32	7	7	7	7	7
FC 4	6	8	8.5	8.35	8	8	8	8.42	8.02	8	8	8	8
FC5	6	8	8.5	8.35	8	8	8	8.42	8.02	8	8	8	8
FC3	7	7	7.5	7.35	7	7	7	7.57	7.35	7	7	7	7
FC4, 20	7	8	8.5	8.35	8	8	8	8.76	8.35	8.20	8.03	8.03	8.03
FC5	7	9	9.5	9.37	9	9.09	9	9.88	9.37	9.11	9.01	9.01	9.01
FC6	7	9	9.5	9.37	9	9	9	9.84	9.30	9.09	9.02	9.02	9.02
FC7	7	10	10.5	10.37	10.11	10	10	10.99	10.46	10.23	10.07	10.07	10.07
FC8	7	10	10.5	10.37	10.19	10	10	10.92	10.42	10.28	10.08	10.08	10.08
FC9	7	10	10.5	10.37	10.11	10	10	10.93	10.43	10.26	10.06	10.06	10.06
FC10, 25	7	9	9.5	9.37	9.06	9	9	9.59	9.05	9.13	9	9	9
FC11	7	10	10.5	10.37	10	10	10	11.18	10.36	10.24	10.03	10.02	10.02
FC12	7	10	10.5	10.37	10	10.03	10	10.96	10.42	10.28	10.08	10.08	10.08
FC13	7	10	10.5	10.37	10.19	10	10	10.99	10.43	10.29	10.08	10.08	10.08
FC14	7	10	10.5	10.35	10.35	10.24	10.22	10.96	10.50	10.51	10.21	10.21	10.21
FC15	7	11	11.5	11.37	11.22	11	11	12.11	11.52	11.38	11.16	11.15	11.15
FC16	7	11	11.5	11.37	11.22	11	11	12.50	11.58	11.46	11.16	11.15	11.15
FC17	7	13	13.5	13.40	13	13	13	13.71	13.26	13.05	13	13	13
FC18	7	14	14.5	14.40	14.17	14.04	14	14.88	14.35	14.10	14	14	14
FC19	7	14	14.5	14.40	14.10	14.01	14	15.00	14.33	14.19	14.01	14.01	14.01
FC21	7	9	9.5	9.37	9	9.01	9	9.88	9.08	9.11	9.01	9.01	9.01
FC22	7	10	10.5	10.37	10.11	10	10	10.99	10.13	10.23	10.07	10.07	10.07
FC23	7	10	10.5	10.37	10.19	10	10	10.92	10.19	10.28	10.08	10.08	10.08
FC24	7	10	10.5	10.37	10.11	10	10	10.93	10.15	10.26	10.06	10.06	10.06
FC26	7	11	11.5	11.37	11.23	11	11	12.11	11.24	11.40	11.16	11.15	11.15
FC27	7	13	13.5	13.40	13	13	13	13.71	13.03	13.05	13	13	13

Table 1: Marginal improvement of various semidefinite relaxations as compared to the linear relaxation on facets of the linear ordering polytope for  $n = 6$  and  $n = 7$ .

Travelling Salesman Problem Instances Constructed from Facet Defining Inequalities for the TSP Polytope on 8 Vertices												
Facet ( $n = 8$ )	Opt	Held-Karp [28]	SDP [11]	SDP [14]	SDP [13]	LP [17]	SDP <sub>1</sub> <sup>cut</sup>	SDP <sub>2</sub> <sup>cut</sup>	SDP <sub>3</sub> <sup>cut</sup>	SDP <sub>4</sub> <sup>cut</sup>	SDP <sub>6</sub> <sup>cut</sup>	SDP <sub>7</sub> <sup>cut</sup>
FC3	2	2	2	2	2	2	1.586	1.634	2	2	2	2
FC4	2	2	1.098	1.628	2	2	1.098	1.300	1.540	1.629	2	2
FC5	2	2	1.172	1.172	1.893	2	1.172	1.172	1.172	1.172	2	2
FC6	10	9	8.507	8.671	10	10	7.324	8.090	8.241	8.671	9	9
FC7	10	9	9	9	10	10	7.890	8.419	8.623	9	9	9
FC8	10	9	8.566	8.926	10	10	7.529	8.227	8.443	8.926	9	9
FC9	10	9	8.586	8.586	10	9	6.929	7.741	8.053	8.586	9	9
FC10	10	9	8.570	8.926	10	9.500	7.440	8.174	8.485	8.926	9	9
FC11	10	9	9	9	10	9.600	7.883	8.511	8.683	9	9	9
FC12	10	9	8.411	8.902	10	10	6.970	7.960	8.331	8.902	9	9
FC13	10	9	8.422	8.899	10	9	6.968	7.926	8.359	8.899	9	9
FC14	0	0	0	0	0	0	-0.707	-0.290	-0.282	0	0	0
FC15	12	11	10.586	10.667	11.777	11	9.757	10.206	10.351	10.667	11	11
FC16	13	12	12	12	12.777	12	10.962	11.244	11.505	12	12	12
FC17	14	$12\frac{2}{3}$	12.408	12.444	13.663	13	11.063	11.574	11.892	12.444	12.667	12.667
FC18	16	14	14	14.078	15.651	16	12.096	13.258	13.434	14.115	14.126	14.126
FC19	18	16	16	16	17.824	18	13.573	14.780	15.219	16	16	16
FC20	18	16	16	16	17.698	18	13.852	14.966	15.332	16	16	16
FC21	18	16	16	16	18	17.500	14.273	15.194	15.513	16	16	16
FC22	18	16	15.185	15.926	18	17	13.207	14.607	15.211	15.937	16.023	16.023
FC23	20	18	18	18.025	19.568	19	16.356	17.124	17.561	18.034	18.044	18.044
FC24	22	20	20	20	21.287	20	19.515	19.715	20	20	20	20
FC25	26	23	23	23.033	25.460	25	20.249	21.785	22.355	23.042	23.055	23.055
FC26	38	35	34.586	34.739	37.141	35	32.082	33.052	33.759	34.739	35	35

Table 2: Comparison of the relaxation strength of various linear and semidefinite relaxations on facets of the traveling salesman polytope for  $n = 8$ .

- [1] H. Achatz, P. Kleinschmidt, and J. Lambsdorff. Der corruption perceptions index und das linear ordering problem. *ORNews*, 26:10–12, 2006.
- [2] M. F. Anjos and J. B. Lasserre, editors. *Handbook on Semidefinite, Conic and Polynomial Optimization Theory, Algorithms, Software and Applications*. International Series in Operations Research & Management Science. Springer-Verlag, New York, 2012.
- [3] D. L. Applegate, R. E. Bixby, V. Chvátal, and W. J. Cook. *The Traveling Salesman Problem: A Computational Study*. Princeton University Press, 2006.
- [4] K. Boenchenndorf. *Reihenfolgenprobleme/Mean-flow-time sequencing*. Mathematical Systems in Economics 74. Verlagsgruppe Athenäum, Hain, Scriptor, 1982.
- [5] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.
- [6] C. Buchheim, A. Wiegele, and L. Zheng. Exact Algorithms for the Quadratic Linear Ordering Problem. *INFORMS Journal on Computing*, 22(1):168–177, 2010.
- [7] M. Chimani and P. Hungerländer. Exact approaches to multilevel vertical orderings. *INFORMS Journal on Computing*, 25(4):611–624, 2013.
- [8] T. Christof, M. Jünger, and G. Reinelt. A complete description of the traveling salesman polytope on 8 nodes. *Operation Research Letters*, 10(9):497–500, 1991.
- [9] N. Christofides. Worst-case analysis of a new heuristic for the traveling salesman problem. Technical report, GSIA, Cranegie-Mellon University, 1976.
- [10] W. J. Cook. *In Pursuit of the Traveling Salesman: Mathematics at the Limits of Computation*. Princeton University Press, 2011.
- [11] D. Cvetković, M. Cangalović, and V. Kovačević-Vujčić. Semidefinite programming methods for the symmetric traveling salesman problem. In *Proceedings of the 7th International IPCO Conference on Integer Programming and Combinatorial Optimization*, pages 126–136. Springer, 1999.
- [12] G. Dantzig, D. Fulkerson, and S. Johnson. Solution of a large scale traveling salesman problem. *Journal of the Operations Research Society of America*, 2:393–410, 1954.
- [13] E. de Klerk and R. Sotirov. Improved semidefinite programming bounds for quadratic assignment problems with suitable symmetry. *Mathematical Programming*, 133(1-2):75–91, 2012.
- [14] E. de Klerk, D. V. Pasechnik, and R. Sotirov. On semidefinite programming relaxations of the traveling salesman problem. *SIAM Journal on Optimization*, 19(4):1559–1573, 2008.
- [15] M. Deza and M. Laurent. *Geometry of Cuts and Metrics*, volume 15 of *Algorithms and Combinatorics*. Springer Verlag, Berlin, 1997.
- [16] U. Feige and M. X. Goemans. Approximating the value of two power proof systems, with applications to MAX 2SAT and MAX DICUT. In *Proceedings of the Third Israel Symposium on Theory of Computing and Systems*, pages 182–189, 1995.
- [17] A. Fischer. *A Polyhedral Study of Quadratic Traveling Salesman Problems*. PhD thesis, Chemnitz University of Technology, Germany, 2013.
- [18] P. C. Fishburn. Induced binary probabilities and the linear ordering polytope: a status report. *Mathematical Social Sciences*, 23(1):67–80, 1992.
- [19] A. Frieze and M. Jerrum. Improved approximation algorithms for MAXk-CUT and MAX BISECTION. *Algorithmica*, 18(1):67–81, 1997.
- [20] M. R. Garey, D. S. Johnson, and L. Stockmeyer. Some simplified np-complete problems. In *STOC '74: Proceedings of the sixth annual ACM symposium on Theory of computing*, pages 47–63. ACM, 1974.
- [21] F. Glover, T. Klasterin, and D. Klingman. Optimal weighted ancestry relationships. *Management Science*, 20:1190–1193, 1974.
- [22] M. Goemans and D. Williamson. Improved approximation algorithms for maximum cut and satisfiability problems using semidefinite programming. *Journal of the ACM*, 42:1115–1145, 1995.
- [23] M. X. Goemans and F. Rendl. Combinatorial optimization. In H. Wolkowicz, R. Saigal, and L. Vandenberghe, editors, *Handbook of Semidefinite Programming*. Kluwer Academic Publishers, 2000.
- [24] M. Grötschel, M. Jünger, and G. Reinelt. A cutting plane algorithm for the linear ordering problem. *Operations Research*, 32(6):1195–1220, 1984.
- [25] D. Grundel and D. Jeffcoat. Formulation and solution of the target visitation problem. *Proceedings of the AIAA 1st Intelligent Systems Technical Conference*, 2004.
- [26] G. Gutin and A. Punnen. *The Traveling Salesman Problem and Its Variations*. Springer, 2002.
- [27] E. Halperin and U. Zwick. A unified framework for obtaining improved approximation algorithms for maximum graph bisection problems. *Random Structures and Algorithms*, 20(3):382–402, 2002.
- [28] M. Held and R. Karp. The traveling salesman problem and minimum spanning trees. *Operations Research*, 18:1138–1162, 1970.
- [29] C. Helmsberg. Fixing variables in semidefinite relaxations. *SIAM Journal on Matrix Analysis and Applications*, 21(3):952–969, 2000.
- [30] A. Hildenbrandt, O. Heismann, and G. Reinelt. The target visitation problem. Presentation held at the 18th Combinatorial Optimization Workshop in Aussois on January 09, 2014.
- [31] A. Hildenbrandt, G. Reinelt, and O. Heismann. Integer programming models for the target visitation problem. In *Proceedings of the 16th International Multiconference of the Information Society*, volume A, pages 569–572, 2013.
- [32] P. Hungerländer. A semidefinite optimization approach to the target visitation problem. *Optimization Letters*, 2014. doi: 10.1007/s11590-014-0824-9.
- [33] P. Hungerländer and F. Rendl. Semidefinite relaxations of ordering problems. *Mathematical Programming*, 140(1):77–97, 2013.
- [34] P. Hungerländer and F. Rendl. A computational study and survey of methods for the single-row facility layout problem. *Computational Optimization and Applications*, 55(1):1–20, 2013.
- [35] M. Jünger and P. Mutzel. 2-layer straightline crossing minimization: performance of exact and heuristic algorithms. *Journal of Graph Algorithms and Applications*, 1:1–25, 1997.
- [36] D. Karger, R. Motwani, and M. Sudan. Approximate graph coloring by semidefinite programming. *Journal of the ACM*, 45(2):246–265, 1998.
- [37] R. Karp. Reducibility among combinatorial problems. In R. Miller and J.W.Thather, editors, *Complexity of Computer Computation*, pages 85–103. Plenum Press, 1972.
- [38] S. Khot. On the power of unique 2-prover 1-round games. In *Proceedings of the Thirty-fourth Annual ACM Symposium on Theory of Computing*, STOC '02, pages 767–775, 2002.



- [39] M. Lampis. Improved inapproximability for tsp. In A. Gupta, K. Jansen, J. Rolim, and R. Servedio, editors, *Approximation, Randomization, and Combinatorial Optimization. Algorithms and Techniques*, volume 7408 of *Lecture Notes in Computer Science*, pages 243–253. Springer Berlin Heidelberg, 2012.
- [40] W. Leontief. Quantitative input-output relations in the economic system of the united states. *The Review of Economics and Statistics*, 18(3): 105–125, 1936.
- [41] L. Lovász and A. Schrijver. Cones of matrices and set-functions and 0-1 optimization. *SIAM Journal on Optimization*, 1:166–190, 1991.
- [42] R. Martí and G. Reinelt. *The Linear Ordering Problem: Exact and Heuristic Methods in Combinatorial Optimization*. Applied Mathematical Sciences. Springer, 2011.
- [43] A. Newman. Cuts and orderings: On semidefinite relaxations for the linear ordering problem. In K. Jansen, S. Khanna, J. Rolim, and D. Ron, editors, *Lecture Notes in Computer Science*, volume 3122, pages 195–206. Springer, 2004.
- [44] A. Newman. *Algorithms for String and Graph Layout*. PhD thesis, Massachusetts Institute of Technology, USA, 2004.
- [45] A. Newman and S. Vempala. Fences are futile: On relaxations for the linear ordering problem. In *Proceedings of the 8th International IPCO Conference on Integer Programming and Combinatorial Optimization*, pages 333–347. Springer, 2001.
- [46] M. Padberg and G. Rinaldi. A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems. *SIAM Review*, 33(1):60–100, 1991.
- [47] C. H. Papadimitriou and S. Vempala. On the approximability of the traveling salesman problem. *Combinatorica*, 26(1):101–120, 2006.
- [48] G. Reinelt. *The traveling salesman: computational solutions for TSP applications*. Springer, 1994.
- [49] D. B. Shmoys and D. P. Williamson. Analyzing the held-karp tsp bound: A monotonicity property with application. *Information Processing Letters*, 35(6):281–285, 1990.
- [50] P. Slater. Inconsistencies in a schedule of paired comparisons. *Biometrika*, 48(3-4):303–312, 1961.
- [51] J. Sturm. Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11–12: 625–653, 1999.
- [52] A. W. Tucker. On directed graphs and integer programs. Technical report, IBM Mathematical Research Project, 1960.
- [53] H. Wolkowicz, R. Saigal, and L. Vandenbergh, editors. *Handbook of Semidefinite Programming*. Kluwer Academic Publishers, Boston, MA, 2000.
- [54] L. A. Wolsey. Heuristic analysis, linear programming and branch and bound. *Mathematical Programming Studies*, 13:121–134, 1980.
- [55] D. H. Younger. Minimum feedback arc sets for a directed graph. *IEEE Transactions on Circuit Theory*, 10(2):238–245, 1963.