**Solution approaches for equidistant
double- and multi-row facility
layout problems**

M.F. Anjos, A. Fischer,
P. Hungerländer

Solution approaches for equidistant double- and multi-row facility layout problems

Miguel F. Anjos[a]

Anja Fischer[b]

Philipp Hungerländer[c]

[a] *GERAD, Canada Research Chair in Discrete Nonlinear Optimization in Engineering & Polytechnique Montréal, Montréal (Québec) Canada, H3C 3A7*

[b] *TU Dortmund University, 44221 Dortmund, Germany*

[c] *Alpen-Adria Universität Klagenfurt, 9020 Klagenfurt, Austria*

anjos@stanfordalumni.org
anja.fischer@mathematik.tu-dortmund.de
philipp.hungerlaender@uni-klu.ac.at

Abstract: The facility layout problem is a well-known operations research problem that arises in multiple applications. This paper is concerned with the multi-row layout problem in which one-dimensional departments are to be placed on a given number of rows so that the sum of the weighted center-to-center distances is minimized. While the optimal solution for a single-row problem will normally have no spaces between departments, for multi-row layout problems it is necessary to allow for the presence of spaces of arbitrary lengths between departments. We consider the special case of equidistant row layout problems in which all departments have the same length, taken to be unity without loss of generality. For this class of problems we prove two theoretical results that facilitate the handling of spaces. First we show that although the lengths of the spaces are in general continuous quantities, every multi-row equidistant problem has an optimal solution on the grid. This implies that only spaces of unit length need to be used when modeling the problem, and hence that the problem can be formulated as a purely discrete optimization problem. Second we state and prove exact expressions for the minimum number of spaces that need to be added so as to preserve at least one optimal solution. One important consequence of these results is that multi-row equidistant layout problems can be modeled using only binary variables; this has a significant impact for a computational perspective. These results are used to formulate two new models for the equidistant problem, an integer linear optimization model and a semidefinite optimization model. Special attention is paid to the double-row layout case that has received much attention recently and is particularly important in practice. Our computational results with the new formulations as well as with a recent formulation by Amaral show that the semidefinite approach dominates for medium- to large-sized instances and that it is well-suited for providing high-quality lower bounds for large-scale instances in reasonable computation time. Specifically for double-row instances, we attain global optimality for some instances with up to 25 departments, and achieve optimality gaps smaller than 1% for instances with up to 50 departments.

1   Introduction

The facility layout problem is a well-known operations research problem that arises in multiple applications. The problem consists in finding an optimal location of departments inside a plant according to a given objective function. In general, the objective function may reflect transportation costs, the construction cost of a material-handling system, or simply adjacency preferences among departments. For example, the placement of machines that form a production line inside a plant is a layout problem in which one wishes to minimize the total cost of the material flow between the machines.

The variety of applications means that facility layout encompasses a broad class of optimization problems. This paper is concerned with the Multi-Row Equidistant Facility Layout Problem (MREFLP). This is one of several row-layout problems that are of interest in the design of flexible manufacturing systems (FMSs). FMSs are automated production systems that typically consist of numerically controlled machines and material handling devices under computer control with the materials handled by devices such as automated guided vehicles (AGVs). It is well-known that the layout of the machines of an FMS has a significant impact on the productivity of the facility, and furthermore that a poor layout is likely to reduce the flexibility of an FMS [27]. Among most frequently encountered layout types in practice are the single-row and multi-row layouts (Figure 1). If all the departments are to be placed in only one row, then we have an instance of the Single-Row Facility Layout Problem (SRFLP), while if more than one row can be used, then the problem is a Multi-Row Facility Layout Problem (MRFLP). We are interested here in the (SREFLP) and the (MREFLP) which are respectively the special cases of the (SRFLP) and (MRFLP) in which all the machines have the same length.

Figure 1: AGV handling materials in a single-row layout (a) and a double-row layout (b).

Single-Row Layout. Arguably the simplest layout problem is that of a single-row layout. An instance of the Single-Row Facility Layout Problem (SRFLP) consists of n one-dimensional machines, with given positive lengths $l_1, \ldots, l_n$, and pairwise weights w_{ij} often referred to as connectivities. The optimization problem can be written down as

$$\min_{\pi \in \Pi_n} \sum_{\substack{i,j \in [n] \\ i < j}} w_{ij} z_{ij}^{\pi}, \tag{1}$$

where Π_n is the set of permutations of the indices $[n] := \{1, 2, \ldots, n\}$ and z_{ij}^{π} is the center-to-center distance between machines i and j with respect to a particular permutation $\pi \in \Pi_n$.

Under the assumption that the weights w_{ij} are non-negative, the optimal solution will have no empty spaces between departments. Hence the (SRFLP) consists of finding a permutation of the departments that minimizes the total weighted sum of the center-to-center distances. Note that the assumption that $w_{ij} \geq 0$ also ensures boundedness of the objective value of the optimal layout.

Beyond the arrangement of machines in FMSs [31], practical applications of the (SRFLP) include the arrangement of rooms on a corridor in hospitals, supermarkets, or offices [48], and the assignment of airplanes to gates in an airport terminal [50]. Accordingly several heuristic algorithms have been suggested for the (SRFLP); among the best ones to date are [15, 38, 44].

Global optimization approaches for the (SRFLP) are based on relaxations of integer linear programming (ILP) and semidefinite programming (SDP) formulations. The strongest ILP approach is an LP-based cutting plane algorithm using betweenness variables that can solve instances with up to 35 departments within a few hours [1]. The strongest SDP approach to date using products of ordering variables is even stronger and can solve instances with up to 42 departments within a few hours [37].

In this context let us recall that SDP is the extension of LP from the set of non-negative vectors to the cone of symmetric positive semidefinite matrices. A (primal) SDP problem can be expressed as

$$\inf \left\{ \langle C, X \rangle \colon X \in \mathcal{P} \right\}, \text{ where } \mathcal{P} := \left\{ X \colon \langle A_i, X \rangle = b_i, \, i \in [m], \, X \succcurlyeq 0 \right\},$$

where the data matrices A_i, $i \in [m]$, and C are symmetric. For further information on SDP we refer the reader to the handbooks [5, 53]. In particular, a survey of global optimization approaches for the (SRFLP) can be found in [6].

This leads us to one of the problems addressed in this paper. The Single-Row Equidistant Facility Layout Problem (SREFLP) is the special case of the (SRFLP) in which the department lengths are all equal. The (SREFLP) arises in several applications, including sheet-metal fabrication [4], printed circuit board and disk drive assembly [13], and the optimal design of a flowline in a manufacturing system [55]. Furthermore Bhasker and Sahni [9] applied the (SREFLP) to minimize the total wire length needed when arranging circuit components on a straight line.

The (SREFLP) is also a special case of the Quadratic Assignment Problem (QAP) (see e.g. [12, 40]). While exact methods and heuristics especially designed for the (SREFLP) clearly outperform general methods for the (QAP), this is not the case for recent approaches to the (SRFLP) [34]. Indeed the most effective global optimization approaches for the (SRFLP) are also the best ones for the (SREFLP). Hence to date the (SREFLP) can also be solved to optimality for instances with up to 42 departments within a few hours.

In contexts other than manufacturing, the (SREFLP) is usually called weighted Linear Arrangement (LA). This problem was originally proposed by Harper [25, 26] to develop error-correcting codes with minimal average absolute errors. It is NP-hard [21], and remains so even if all weights are binary and the underlying graph is bipartite [20]. It follows that all the problems considered in this paper are also NP-hard, as they are extensions of (LA).

Multi-Row Layout. The Double-Row Facility Layout Problem (DRFLP) is a natural extension of the (SRFLP) in the manufacturing context when one considers that an AGV can support stations located on both sides of its linear path of travel (see Figure 1). The (DRFLP) is especially relevant for real-world applications because this is a common approach in practice for improved material handling and space usage, and thus real factory layouts most often reduce to a combination of single-row and double-row problems.

Row layout can be further generalized to the Multi-Row Facility Layout Problem (MRFLP) where the departments are arranged along several parallel rows. An instance of the (MRFLP) consists of n one-dimensional departments with given positive lengths $l_1, \ldots, l_n$, pairwise non-negative weights w_{ij} between the departments, and a set $\mathcal{R} := \{1, \ldots, m\}$ of rows available for placing the departments. The objective is to find an assignment $r \colon [n] \to \mathcal{R}$ of departments to rows, and feasible horizontal positions for the centers of the departments within the assigned rows, i.e., a function

$$p \colon [n] \to \mathbb{R} \text{ satisfying } \frac{l_i + l_j}{2} \leq |p(i) - p(j)| \text{ if } r(i) = r(j),$$

such that the total weighted sum of the center-to-center distances between all pairs of departments is minimized. Our formulation of the (MRFLP) is thus:

$$\min_{r,p} \quad \sum_{\substack{i,j \in [n] \\ i < j}} w_{ij} |p(i) - p(j)| \tag{2}$$

$$\text{s.t.} \quad \frac{l_i + l_j}{2} \leq |p(i) - p(j)|, \; i \neq j, \text{ and } \; r(i) = r(j). \tag{3}$$

The (MRFLP) has numerous applications such as computer backboard wiring [49], campus planning [17], scheduling [23], typewriter keyboard design [43], hospital layout [18], the layout of machines in an automated manufacturing system [32], balancing hydraulic turbine runners [39], numerical analysis [11], optimal digital signal processors memory layout generation [52]. Different extensions of the (MRFLP) like considering a

clearance between any two adjacent machines given as a fuzzy set [22] or the design of an FMS in one or multiple rows [19] have been proposed and tackled with genetic algorithms.

Somewhat surprisingly, the development of exact algorithms for the (DRFLP) and the (MRFLP) has received only limited attention in the literature. In the 1980s Heragu and Kusiak [31] proposed a non-linear programming model and obtained locally optimal solutions to the (SRFLP) and the (DRFLP). Recently Chung and Tanchoco [14] (see also Zhang and Murray [56]) focused exclusively on the (DRFLP) and proposed a mixed integer linear programming (MILP) formulation that was tested in conjunction with several heuristics for assigning the departments to the rows. Their approach was able to solve instances with up to 10 departments. Amaral [2] proposed an improved MILP formulation that allowed him to solve instances with up to 12 departments to optimality. Most recently Hungerländer and Anjos [36] proposed an SDP approach for the general (MRFLP) that is however only applicable to small instances with less than 12 departments.

Again our interest in this paper is on the case of (MREFLP) that has all the department lengths equal. With respect to this special case, Amaral [3] proposed an MILP formulation tailored to the Minimum Duplex Arrangement Problem, which in our terminology corresponds to the (DREFLP). His approach allows to exploit the sparsity of the instances considered and is able to solve randomly generated instances with at most 10 (very dense instances) to 20 (very sparse instances) departments. For more details on this formulation we refer to Subsection 3.1.

A Toy Example. We illustrate the (SREFLP) and the (MREFLP) with the help of a toy example. We consider 6 equidistant departments and the following given pairwise weights:

$$w_{12} = w_{13} = w_{14} = w_{23} = w_{24} = w_{34} = 2, \ w_{15} = w_{25} = w_{36} = w_{46} = 1.$$

Figure 2 illustrates the optimal layouts and corresponding total costs when using between one and four rows.
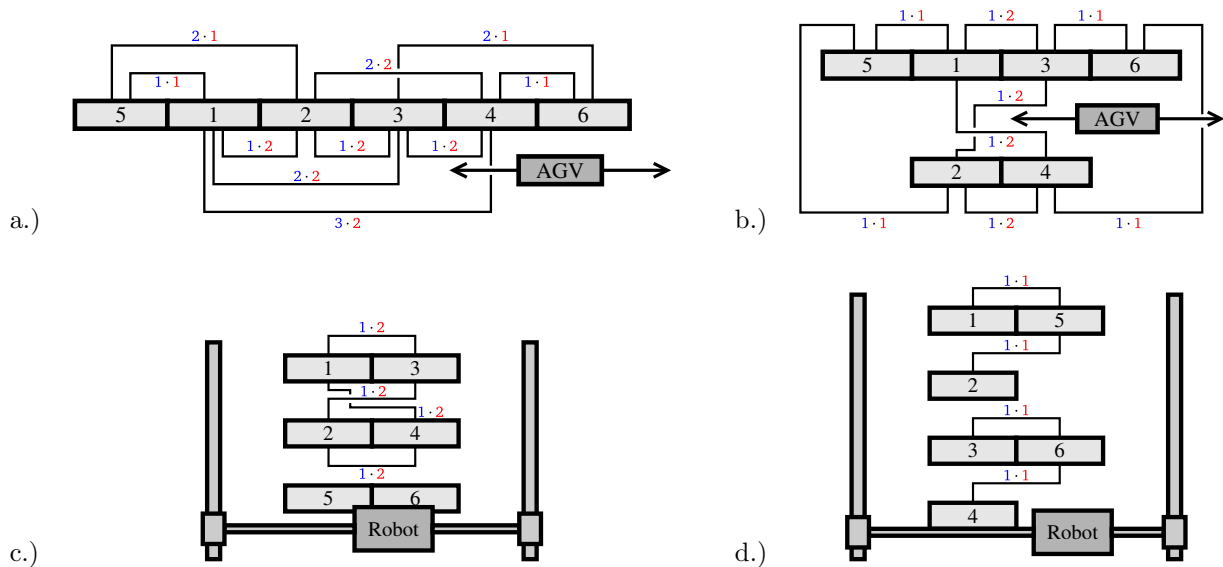
Figure 2: a.) Optimal (SREFLP) solution with total cost of 26.
b.) Optimal (DREFLP) solution with total cost of 12.
c.) Optimal solution for the (MREFLP) with 3 rows with total cost of 8.
d.) Optimal solution for the (MREFLP) with 4 rows with total cost of 4.

Outline. This paper is structured as follows. In Section 2 we state and prove our theoretical results on the structure of optimal layouts. Namely that the (MREFLP) always has an optimal solution on the grid, and that we can give exact expressions for the minimum number of spaces that need to be added to an instance of the (MREFLP) so that from an optimal solution to the resulting "space-free" problem we can recover at least one optimal solution for the (MREFLP) instance. One consequence of these results is that the (MREFLP) can

be modeled using solely binary variables. In Section 3 we focus on the double-row case, recalling an MILP formulation for the (DREFLP) by Amaral [3] and presenting two new models for that problem, one based on ILP and the other on SDP. In Section 4 we show how the three models can be extended to the general multi-row case, and in Section 5 we describe a suitable combination of optimization methods to obtain strong lower bounds and feasible layouts using the presented models. Section 6 reports the results of our computational experiments to assess the practical performance of the different approaches. Finally, Section 7 concludes the paper and summarizes some directions for future research.

2   The structure of optimal layouts

The definition of the (MREFLP) implicitly allows the spaces between two departments to be of arbitrary length. For this reason most optimization models in the literature use continuous variables to model the distances between departments.

In this section we prove two theoretical results about the structure of optimal layouts. We first show that the (MREFLP) always has an optimal solution on the grid. The key insight here is that restricting the spacing between departments to be formed using *spacing departments* preserves at least one optimal solution. Second we give exact expressions for the minimum number of such spacing departments required for each combination of numbers of departments and rows so as to preserve at least one optimal solution. One consequence of these results is that the (MREFLP) can be modeled using solely binary variables.

These results are of intrinsic theoretical interest because they reveal hitherto hidden structural properties of the (MREFLP). Moreover they are of use to improve the practical performance of the optimization models that we propose in Sections 3 and 4.

2.1   A combinatorial property of multi-row layouts

Theorem 1 is a special case of [35, Theorem 2]:

Theorem 1 *There is always an optimal solution to the (MREFLP) on the grid.*

Proof. Let an optimal solution of the (MREFLP) be given. We define an integer grid such that the centers of the departments with the leftmost centers are on a grid point. Next we divide the departments into two sets, a set S containing those with their centers already on the grid, and a set T containing the others. We assume w. l. o. g. that the indices of the departments in S are all smaller than the indices of the departments in T: $i < j, \ \forall \ i \in S, \ j \in T$.

Observe that there exists $\varepsilon > 0$ sufficiently small so that we can move all the departments in T simultaneously, either to the left or to the right, by a distance ε. This holds because all departments have (the same) integer length, and because the departments in S are arranged on the grid. The change in the objective function from any such shift of the departments in T is given by

$$\delta = \sum_{i \in T} \left(\varepsilon \sum_{j \in S, \ j \dot{<} i} w_{ij} - \varepsilon \sum_{j \in S, \ i \dot{<} j} w_{ij} \right)$$

for a shift to the left, and by $-\delta$ for a shift to the right, where $i \dot{<} j$ means that the center of j is to the right of the center of i, and ε is chosen small enough such that no department in T traverses a grid point. Due to the optimality of the given layout, δ has to be equal to zero because otherwise a shift either to the left (for $\delta < 0$) or to right (for $\delta > 0$) would improve the objective value. Hence the proposed shifting operation does not change the objective value.

Let us choose ε as the largest value such that the center of at least one department in T lies on a grid point after the shifting operation (to the left or right). If we apply this shifting to the given optimal solution, we can now move that department to the set S. Repeatedly applying this operation to the remaining departments allows us to arrange all departments on the grid in at most $n - 1$ steps without changing the objective value. □

Theorem 1 is illustrated in Figure 3. For layouts fulfilling the grid property, we say that department i lies in column j if the center of i is located at the jth grid point. For example department 5 lies in column 4 in Figure 3.

Note that the grid property is automatically fulfilled for layouts corresponding to the graph version of the (MREFLP), i.e., an extension of (LA) where two or more nodes can be assigned to the same position. Hence by Theorem 1 the Minimum Duplex Arrangement Problem considered in [3] is a special case of the (DREFLP).

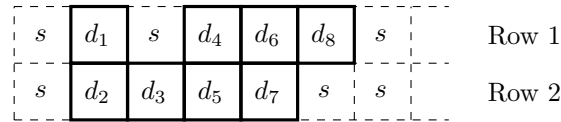| s | d_1 | s | d_4 | d_6 | d_8 | s | | Row 1 |
| s | d_2 | d_3 | d_5 | d_7 | s | s | | Row 2 |

Figure 3: Illustration of the grid property of layouts. Note that for such layouts all departments and spaces have equal size.

From now on we restrict our attention to layouts fulfilling the grid property. This restriction is clearly advantageous from both a theoretical as well as a practical point of view.

2.2   Bounds on the number of spaces

In this subsection we are interested in the minimum number of spacing departments, or simply spaces, that must be added to an instance of the (MREFLP) so that we can recover at least one optimal solution for the original (MREFLP) instance from the optimal solution to the resulting problem. Clearly this number is a function of the number of departments and the number of rows, but since we do not have a priori knowledge about the structure of optimal solutions for given cost coefficients, it does not depend on the weights w_{ij} (other than assuming their non-negativity).

In the following theorem we make three additional assumptions that allow us to reduce the number of spaces needed. Note that at least one optimal layout is preserved under these assumptions (see Lemma 3 below for a formal proof of this statement).

Assumption 1 Columns that contain only spaces can be deleted. Equivalently, if we number the columns from 1 to n there exists $k' \in [n]$ such that each column with index at most k' contains at least one department.

Assumption 2 If two non-empty neighboring columns contain altogether no more than m departments, then all corresponding departments can be assigned to the left column and the right column can be deleted. Thus with k' as in Assumption 1 we know that columns i and $i + 1$ with $i \in [k' - 1]$ contain at least $m + 1$ departments.

Assumption 3 If $d > 2m$ and the first column and the third column contain in total at most m departments, then all corresponding departments can be assigned to the third column and the first column can be deleted.

A similar argument holds for columns $k' - 2$ and k' with k' as in Assumption 1.

These assumptions are illustrated in Figure 4 where the left-hand side depicts a feasible layout and the right-hand side depicts the adaptation of that layout so that the respective assumption holds. Note that the adaptations cannot worsen the objective value of the layout.

We can now state the second theorem.

Theorem 2 *The number of columns sufficient to preserve at least one optimal layout for an instance with d departments is*

1. *equal to 1 if $d \leq m$, and equal to 2 if $m < d < \frac{3}{2}m + \frac{3}{2}$;*
2. *equal to $\left\lceil \frac{2d}{3} \right\rceil - 1$ for the (DREFLP) with $d \geq 9$;*
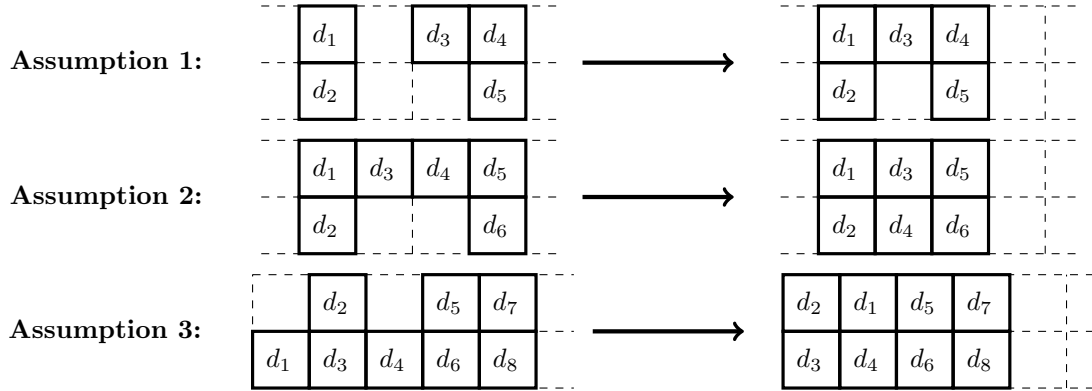
Figure 4: Illustration of Assumptions 1, 2 and 3.

3. equal to $\left\lfloor \frac{2d}{m+1} \right\rfloor$ for the (MREFLP) with an odd number of rows m; and

4. equal to $2l+1$ for the (MREFLP) with an even number of rows m and $d \in \{\frac{m}{2}+2+(m+1)(l-1), \ldots, \frac{m}{2}+1+(m+1)l\}$ for some $l \in \mathbb{N}$.

To prove Theorem 2, we begin by using the fact that Theorem 1 allows us to assume that the departments of the (MREFLP) are arranged on a grid. Hence we can represent an optimal solution of the (MREFLP) by an assignment $\alpha \colon [d] \to [d]$ of the d departments to d different columns with the interpretation

$$\alpha(i) = j, \quad \text{if department } i \in [d] \text{ lies in column } j \in [d], \quad (i, j \in [d]) \tag{4}$$

and at most m departments are assigned to each column $j \in [d]$, i.e.,

$$|\{i \in [d] \colon \alpha(i) = j\}| \le m.$$

Indeed, the modeling approach in [3] directly reflects the assignment (4) (see Subsection 3.1 for details). Furthermore, there always exists an optimal solution $\alpha^* \colon [d] \to [d]$ that fulfills additional structural properties that we already depicted in Figure 4 and now formally describe and prove in the next lemma.

Lemma 3 *Let $d, m \in \mathbb{N}$. Then there always exists an optimal solution $\alpha^* \colon [d] \to [d]$ of the (MREFLP) (fulfilling the grid structure) that assigns each department $i \in [d]$ to a column $\alpha^*(i) \in [d]$ which fulfills the following properties:*

1. *There exists a $k' \in [d]$ such that $|\{i \in [d] \colon \alpha^*(i) = l\}| \ge 1$ for all $l \in [d]$, $l \le k'$, and $|\{i \in [d] \colon \alpha^*(i) \ge k'+1\}| = 0$.*

2. *If $|\{i \in [d] \colon \alpha^*(i) = j\}| > 0$ and $|\{i \in [d] \colon \alpha^*(i) = j+1\}| > 0$ for some $j \in [d]$, $j < d$, then $|\{i \in [d] \colon \alpha^*(i) = j\}| + |\{i \in [d] \colon \alpha^*(i) = j+1\}| \ge m+1$.*

3. *Let $d > 2m$. Then $|\{i \in [d] \colon \alpha^*(i) \ge k'+1\}| = 0$ and $|\{i \in [d] \colon \alpha^*(i) = k'\}| > 0$ for some $k' \in [d]$ imply $|\{i \in [d] \colon \alpha^*(i) = k'-2\}| + |\{i \in [d] \colon \alpha^*(i) = k'\}| \ge m+1$. Furthermore $|\{i \in [d] \colon \alpha^*(i) = 1\}| + |\{i \in [d] \colon \alpha^*(i) = 3\}| \ge m+1$.*

Proof. Let $d, m \in \mathbb{N}$ and α^* be an optimal solution of the (MREFLP) fulfilling the grid structure.

1. If $|\{i \in [d] \colon \alpha^*(i) = j-1\}| = 0$ and $|\{i \in [d] \colon \alpha^*(i) = j\}| \ge 1$ for some $j \in [d]$, then the assignment α' with

$$\alpha'(l) = \begin{cases} \alpha^*(l), & \alpha^*(l) < j, \\ \alpha^*(l) - 1, & \text{otherwise,} \end{cases}$$

for $l \in [d]$ is optimal for the (MREFLP), too, because the distances between departments are not enlarged. The repeated "deletion" of empty columns proves the statement.

2. Assume that $|\{i \in [d]\colon \alpha^*(i) = j\}| + |\{i \in [d]\colon \alpha^*(i) = j+1\}| \leq m$ for some $j \in [d], j < d$. Then α' with

$$\alpha'(l) = \begin{cases} \alpha^*(l), & \alpha^*(l) \leq j, \\ \alpha^*(l) - 1, & \text{otherwise,} \end{cases}$$

for $l \in [d]$ is a feasible multi-row assignment and it is even optimal, because all distances are not enlarged (some are even shortened) and there are at most m departments in each row. Applying this approach repeatedly we get an optimal assignment $\bar{\alpha}$ such that $|\{i \in [d]\colon \bar{\alpha}(i) = j\}| > 0$ and $|\{i \in [d]\colon \bar{\alpha}(i) = j+1\}| > 0$ for some $j \in [d-1]$ imply $|\{i \in [d]\colon \bar{\alpha}(i) \in \{j, j+1\}\}| > m$.

3. Now assume, w.l.o.g., that there exists an optimal solution α^* of the (MREFLP) and $k' \in [d]$ such that $|\{i \in [d]\colon \alpha^*(i) = k'\}| > 1, |\{i \in [d]\colon \alpha^*(i) \geq k'+1\}| = 0$. By the previous statements we may assume $|\{i \in [d]\colon \alpha^*(i) = k'-1\}| > 0$ and $-\{i \in [d]\colon \alpha^*(i) \in \{k'-1, k'\}\}| > m$. If, additionally $|\{i \in [d]\colon \alpha^*(i) \in \{k'-2, k\}\}| \leq m$, the solution α' with

$$\alpha'(l) = \begin{cases} \alpha^*(l) - 2, & \alpha^*(l) = k', \\ \alpha^*(l), & \text{otherwise,} \end{cases}$$

for $l \in [d]$ is optimal, too, because all distances between departments are not enlarged.

$\square$

We are now ready to prove Theorem 2.

Proof. (of Theorem 2) We prove each of the claims of Theorem 2 in turn.

- Proof of 1: Let $d, m \in \mathbb{N}$ be given. If $d \leq m$, it is clear that arranging all departments in one column leads to costs of zero. Furthermore, as long as $m < d < \frac{3}{2}m + \frac{3}{2}$ there exists an arrangement such that only two columns are used because, w.l.o.g., we can assume that the first two columns contain $m+1$ departments and that the second column contains maximal $\lceil \frac{m}{2} \rceil$ of these departments. Then, the remaining departments could also be included in one of the first two columns, either all in the second column or also some of them in the first column.

- Proof of 2: Let $m = 2$, $d \geq 9$ and let α^* be an optimal solution of the (DREFLP) fulfilling the grid structure as well as the properties described in Lemma 3. So we might assume that there exists a $k' \in [d]$ such that $|\{i \in [d]\colon \alpha^*(i) = l\}| \geq 1$ for all $l \in [d]$, $l \leq k'$ and $|\{i \in [d]\colon \alpha^*(i) > k'\}| = 0$. (Note, $d \geq 9$ implies $k' \geq 5$.) By Lemma 3 the solution α^* fulfills $|\{i \in [d]\colon \alpha^*(i) \in \{j, j+1\}\}| \geq 3$ for all $j \in [d]$, $j < k'$, as well as $|\{i \in [d]\colon \alpha^*(i) \in \{1, 2, 3\}\}| \geq 5$ and $|\{i \in [d]\colon \alpha^*(i) \in \{k'-2, k'-1, k'\}\}| \geq 5$. We consider two cases for k'. If $(k'-6) \mod 2 \equiv 0$, then the first k' columns contain at least $10 + (k'-6)\frac{3}{2} = \frac{3}{2}k' + 1$ departments. Otherwise, if $(k'-6) \mod 2 \equiv 1$, then the first k' columns contain at least $5 + (k'-3)\frac{3}{2} = \frac{3}{2}k' + \frac{1}{2}$ departments. Now, assume for a contradiction, that $k' \geq \lceil \frac{2d}{3} \rceil$. Then the first k' columns contain at least $\lceil (\frac{3}{2} \lceil \frac{2d}{3} \rceil + \frac{1}{2}) \rceil > d$ departments, a contradiction. So the statement follows.

- Proof of 3: Let m be odd and $d > 2m$. Let α^* be an optimal solution of the (MREFLP) that fulfills all properties described in Lemma 3. We might assume by Lemma 3 that there exists $k' \in [d]$ such that $|\{i \in [d]\colon \alpha^*(i) = l\}| \geq 1$ for all $l \in [d]$, $l \leq k'$ and $|\{i \in [d]\colon \alpha^*(i) \geq k'+1\}| = 0$. Then we know by Lemma 3 that $|\{i \in [d]\colon \alpha^*(i) = j\}| + |\{i \in [d]\colon \alpha^*(i) = j+1\}| \geq m+1$ for all $j \in [d]$, $j < k'$. Assume now, for a contradiction, that $k' > \lfloor \frac{2d}{m+1} \rfloor$, then the k' columns contain at least $\frac{m+1}{2} \cdot k' \geq \frac{m+1}{2}(\lfloor \frac{2d}{m+1} \rfloor + 1) > d$ departments, a contradiction.

- Proof of 4: Let m be even and $d > 2m$. Let α^* be an optimal solution of the (MREFLP) that fulfills all properties described in Lemma 3. Assume $d \in \{\frac{m}{2} + 2 + (m+1)(l-1), \ldots, \frac{m}{2} + 1 + (m+1)l\}$ for some $l \in \mathbb{N}$. We might assume by Lemma 3 that there is a $k' \in [d]$ such that $|\{i \in [d]\colon \alpha^*(i) = m\}| \geq 1$ for all $m \in [d]$, $m \leq k'$ and $|\{i \in [d]\colon \alpha^*(i) \geq k'+1\}| = 0$. Then we know by Lemma 3 that $|\{i \in [d]\colon \alpha^*(i) =$

$j\}| + |\{i \in [d] \colon \alpha^*(i) = j + 1\}| \geq m + 1$ for all $j \in [d]$, $j < k'$. Assume now, for a contradiction, that $k' \geq 2l+2$. Then the first k' columns contain at least $\frac{2l+2}{2}(m+1) = (m+1)l+m+1 > (m+1)l+\frac{m}{2}+1 \geq d$ departments, a contradiction.

$\square$

Table 1 gives exact values for the minimum number of columns for small values of d and problems with two to four rows.

Table 1: Minimum number of columns needed for instances of (MREFLP) with $d \leq 16$ and $m = 2, 3, 4$.

d	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
2 rows	1	1	2	2	3	4	4	5	5	6	7	7	8	9	9	10
3 rows	1	1	1	2	2	3	3	4	4	5	5	6	6	7	7	8
4 rows	1	1	1	1	2	2	2	3	3	4	4	4	5	5	5	6

Let us give small toy examples for which the optimal layout contains many spaces and hence the number of columns given in Table 1 is necessary: consider problems with $m = 3$ rows, $d = 2l$ departments for some $l \in \mathbb{N}$, and with weights $w_{i(i+1)} = 1, i = 1, 3, 5, \ldots, 2l-1$, and $w_{ij} = \varepsilon$ otherwise. For ε sufficiently small, the optimal solution contains exactly one space in each column; the case with $d = 10$ is shown on the left-hand side of Figure 5. Note that in this example the objective value is not worsened if we reduce the number of rows from three to two.

Next let us point out that for the (MREFLP) with an even number of rows ¿ 2, the exact calculation of the bounds is quite involved and might be slightly improved if d cannot be written as $\frac{m}{2} + 1 + (m+1)l$ for some $l \in \mathbb{N}$. Nevertheless, although the number of spaces seems large, sometimes no improvement is possible if we want to preserve an optimal solution. To see this consider a problem with four rows and 13 departments with $w_{12} = w_{13} = w_{45} = w_{67} = w_{68} = w_{9\,10} = w_{11\,12} = w_{11\,13} = 1$ and all other weights equal to a small $\varepsilon > 0$, then all optimal solutions have a structure like the one visualized on the right-hand side of Figure 5. In this case $d = 13 = \frac{4}{2} + 1 + 5l$ with $l = 2$.
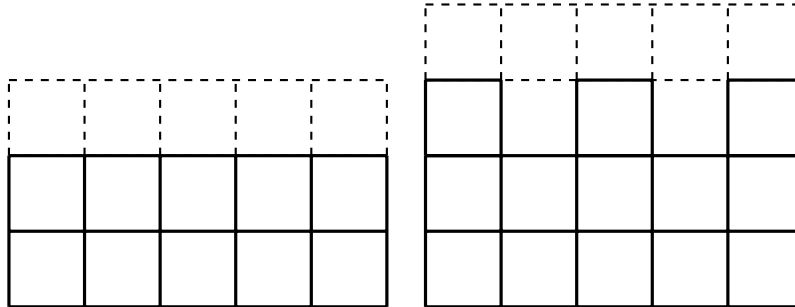
Figure 5: Worst-case examples for Theorem 2.

Theorem 2 allows us to reduce the number of spaces, and hence of variables, both in the MILP model from Amaral [3] and in the new ILP and SDP formulations proposed in Section 3. This theorem also helps to eliminate some of the symmetries in the problem, for example the position of empty columns, and hence to obtain stronger global bounds from all the relaxations. The computational results in Section 6 demonstrate the practical impact of Theorem 2.

3 Three modeling approaches for double-row layouts

In this section we focus on the double-row case. First we recall a MILP formulation for the (DREFLP) by Amaral [3]. Second, we present two new models for the (DREFLP): the first one is an ILP formulation that

uses betweenness variables together with variables modeling whether pairs of departments are assigned to the same column, and the second one is an SDP formulation based on products of ordering variables.

We note that in the approaches discussed below we do not assign the departments to a specific row (as was done for instance in recent SDP-based approaches to the (MRFLP) [36]). We instead ensure that at most m departments are assigned to each column.

3.1   A MILP formulation related to the quadratic assignment problem

To the best of our knowledge the paper by Amaral [3] contains the only approach tailored specifically for the (DREFLP). Let us briefly outline his ILP formulation. We introduce the binary variables $z_{ip} \in \{0,1\}$, $i \in [d]$, $p \in [c]$ (c the number of columns), with the interpretation

$$y_{ip} = \begin{cases} 1, & \text{department } i \text{ is assigned to column } p, \\ 0, & \text{otherwise.} \end{cases}$$

Using these variables we can rewrite the objective function (2) for the (DREFLP) as

$$\sum_{\substack{i,j \in [d], \\ i<j}} \sum_{\substack{p,q \in [c], \\ p<q}} w_{ij}(q-p)y_{ip}y_{jq}.$$

This quadratic objective function is linearized by introducing the binary variables[1]

$$z_{ipjq} = \begin{cases} 1, & \text{if department } i \text{ is assigned to column } p \text{ and department } j \text{ is assigned to column } q, \\ 0, & \text{otherwise,} \end{cases}$$

with $i,j \in [d]$, $p,q \in [c]$ and $(i \neq j, p < q) \vee (i < j, p = q)$, $w_{ij} > 0$. Hence the (DREFLP) can be formulated as the following MILP:

$$\min \sum_{\substack{i,j \in [d], \ i<j \\ w_{ij}>0}} \sum_{\substack{p,q \in [c], \\ p<q}} w_{ij}(q-p)z_{ipjq} \tag{5}$$

$$\text{s.t.} \ \sum_{p \in [c]} y_{ip} = 1, \qquad\qquad i \in [d], \tag{6}$$

$$\sum_{i \in [d]} y_{ip} \leq 2, \qquad\qquad p \in [c], \tag{7}$$

$$y_{ip} + y_{jq} - z_{ipjq} \leq 1, \qquad i,j \in [d], \ p,q \in [c], \ (i \neq j, p < q) \vee (i < j, p = q), \ w_{ij} > 0, \tag{8}$$

$$y_{ip} \in \{0,1\}, \qquad\qquad i \in [d], \ p \in [c], \tag{9}$$

$$z_{ipjq} \in [0,1], \qquad\qquad i,j \in [d], \ p,q \in [c], (i \neq j, p < q) \vee (i < j, p = q), \ w_{ij} > 0. \tag{10}$$

Note that the constraints

$$z_{ipjq} \leq y_{ip}, \ z_{ipjq} \leq y_{jq}, \ i,j \in [d], \ p,q \in [c], \ (i \neq j, p < q) \vee (i < j, p = q), \ w_{ij} > 0$$

of the standard linearization can be omitted because the weights w_{ij}, $i,j \in [d]$, $i \neq j$, are assumed to be non-negative. In order to tighten this formulation Amaral [3] also applied the techniques of Sherali-Adams [46, 47]. This results in the following tighter MILP formulation:

$$\min \sum_{\substack{i,j \in [d], \ i<j, \\ w_{ij}>0}} \sum_{\substack{p,q \in [c], \\ p<q}} w_{ij}(q-p)z_{ipjq}$$

$$\text{s.t.} \ \ (6)\text{--}(10),$$

---

[1]Note that Amaral [3] introduced more binary variables y and z than we do, as he set the number of columns c to d.

$$\sum_{\substack{j\in[d],\ j>i,\\ w_{ij}>0}} z_{ipjp} \le y_{ip}, \qquad\qquad i\in[d],\ p\in[c], \tag{11}$$

$$\sum_{\substack{j\in[d],\ j<i,\\ w_{ij}>0}} z_{jpip} \le y_{ip}, \qquad\qquad i\in[d],\ p\in[c], \tag{12}$$

$$\sum_{\substack{q\in[c],\\ q>p}} z_{ipjq} + \sum_{\substack{q\in[c],\\ q<p}} z_{jqip} + z_{ipjp} \le y_{ip}, \qquad\qquad i,j\in[d],\ i<j,\ w_{ij}>0,\ p\in[c], \tag{13}$$

$$\sum_{\substack{j\in[d],\ j\ne i,\\ w_{ij}>0}} z_{ipjq} \le 2y_{ip}, \qquad\qquad i\in[d],\ p,q\in[c],\ p<q, \tag{14}$$

$$\sum_{\substack{j\in[d],\ j\ne i,\\ w_{ij}>0}} z_{jqip} \le 2y_{ip}, \qquad\qquad i\in[d],\ p,q\in[c],\ q<p. \tag{15}$$

3.2   An ILP formulation related to the linear ordering problem

In this subsection we present a new ILP formulation for the (DREFLP). This formulation is an extension of the model proposed in [1] for the (SRFLP). We use additional variables to model that two departments can be assigned to the same column and additionally fill up the c columns with spaces (i. e., departments of length 1 and weights of zero). We collect all these spaces in a set S. To simplify notation we set the total number of departments (original ones plus spaces) to $n := 2c$ and the number of spaces is thus $s = n - d$. After the insertion of spaces we deal in fact with a space-free problem, and by Theorems 1 and 2 the optimal solution of the corresponding optimization problem is ensured to be an optimal solution of the (DREFLP).

Our model makes use of binary betweenness variables

$$b_{ijk} = b_{kji} \in \{0,1\},\ i,j,k\in[n],\ i<k,\ i\ne j\ne k,$$

and of binary column overlap variables

$$a_{ij} = a_{ji} \in \{0,1\}, i,j\in[n], i<j.$$

These two sets of variables have the following interpretations:

$$b_{ijk} = \begin{cases} 1, & \text{if department } j \text{ lies between departments } i \text{ and } k, \\ 0, & \text{otherwise,} \end{cases}$$

$$a_{ij} = \begin{cases} 1, & \text{if departments } i \text{ and } j \text{ are assigned to the same column,} \\ 0, & \text{otherwise.} \end{cases}$$

Our resulting formulation of the (DREFLP) is

$$\min \sum_{i,j\in[n],\ i<j} \frac{w_{ij}}{2}\cdot\left(\sum_{k\in[n]\setminus\{i,j\}} b_{ikj} + 2(1-a_{ij})\right) \tag{16}$$

$$\text{s.t.}\quad a_{ij} + a_{ik} + a_{jk} + b_{ijk} + b_{ikj} + b_{jik} = 1, \qquad i,j,k\in[n],\ i<j<k \tag{17}$$

$$\sum_{j\in[n]\setminus\{i\}} a_{ij} = 1, \qquad i\in[n], \tag{18}$$

$$b_{ihj} + b_{ihk} + b_{jhk} \le 2, \qquad i,j,k,h\in[n],\ i<j<k\ne h,\ i\ne h\ne j, \tag{19}$$

$$-b_{ihj} + b_{ihk} + b_{jhk} + b_{ikj} \ge 0, \qquad i,j,k,h\in[n],\ i<j<k\ne h,\ i\ne h\ne j, \tag{20}$$

$$+b_{ihj} - b_{ihk} + b_{jhk} + b_{ikj} \ge 0, \qquad i,j,k,h\in[n],\ i<j<k\ne h,\ i\ne h\ne j, \tag{21}$$

$$+b_{ihj} + b_{ihk} - b_{jhk} + b_{ikj} \ge 0, \qquad i,j,k,h\in[n],\ i<j<k\ne h,\ i\ne h\ne j, \tag{22}$$

$$-b_{ihj} + b_{ihk} + b_{jhk} + a_{hk} \geq 0, \qquad i,j,k,h \in [n], \ i < j < k \neq h, \ i \neq h \neq j, \tag{23}$$

$$+b_{ihj} - b_{ihk} + b_{jhk} + a_{hk} \geq 0, \qquad i,j,k,h \in [n], \ i < j < k \neq h, \ i \neq h \neq j, \tag{24}$$

$$+b_{ihj} + b_{ihk} - b_{jhk} + a_{hk} \geq 0, \qquad i,j,k,h \in [n], \ i < j < k \neq h, \ i \neq h \neq j, \tag{25}$$

$$b_{ijk} \in \{0,1\}, \qquad i,j,k \in [n], \ i < j, \ i \neq k \neq j, \tag{26}$$

$$a_{ij} \in \{0,1\}, \qquad i,j \in [n], \ i < j. \tag{27}$$

The objective function (16) counts all departments that lie between the departments i and j, and because of the double-row structure the corresponding sum is divided by two. We count w_{ij} towards the cost if departments i and j do not lie in the same column.

Equations (17) express that three different departments lie either in three different columns such that one of the betweenness variables equals one or that exactly two of the three departments lie in the same column such that the associated overlap variable is one. With equations (18) we ensure that each department $i \in [n]$ lies in the same column with exactly one other department. Inequalities (19) to (25) are extensions of the inequalities in [1] for the SRFLP: inequality (19) ensures that a department h cannot lie between each two of the three departments $i,j,k \in [n] \setminus \{h\}$, $i < j < k$, and inequalities (20)–(25) ensure that if department h lies between departments i and j, then h lies also between i,k or j,k or it lies in the same column as k, which also implies that k lies between i and j.

Due to the introduction of spaces our model contains some symmetries that should be broken to improve the practical performance of the model. The following constraints enforce an order of the s spaces such that space i lies left of space j or is in the same column as j iff $i < j$, $i,j \in S$:

$$a_{ij} = 0, \qquad i,j \in S, \ i+2 \leq j, \tag{28}$$

$$b_{ijk} = 1, \qquad i,j,k \in S, \ i+4 \leq j+2 \leq k, \tag{29}$$

$$b_{ijk} = 0, \qquad i,j,k \in S, \ i \neq k, \ (j > \max\{i,k\} \vee j < \min\{i,k\}). \tag{30}$$

A further way to improve the presented model is to include an adapted variant of certain inequalities for the (SRFLP) proposed by Amaral [1].

Observation 4 *Let $\beta \in \mathbb{N}, \beta \geq 4$, be even and let $T \subseteq [n]$ with $|T| = \beta$. For a partition of T in $T_1, T_2, \{k\}$ such that $T = T_1 \dot\cup T_2 \dot\cup \{k\}, (T_1 \cap T_2 = \emptyset, k \notin T_1, k \notin T_2)$ and $|T_1| = \frac{\beta}{2}$ the following inequalities are valid for the (DREFLP)*

$$\sum_{\substack{p,q \in T_1, \\ p<q}} b_{pkq} + \sum_{\substack{p,q \in T_2, \\ p<q}} b_{pkq} - \sum_{\substack{p \in T_1, \\ q \in T_2}} b_{pkq} \leq \sum_{p \in T_2} a_{kp}, \tag{31}$$

$$\sum_{\substack{p,q \in T_1, \\ p<q}} b_{pkq} + \sum_{\substack{p,q \in T_2, \\ p<q}} b_{pkq} - \sum_{\substack{p \in T_1, \\ q \in T_2}} b_{pkq} \leq \sum_{\substack{p,q \in T_1, \ o \in T_2, \\ p<q}} b_{poq}. \tag{32}$$

Proof. Let $\beta \in \mathbb{N}, \beta \geq 4$, even and $T \subseteq [n]$ with $|T| = \beta$ be given. We consider a partition of T into $T_1, T_2, \{k\}$ such that $T = T_1 \dot\cup T_2 \dot\cup \{k\}$ and $|T_1| = \frac{\beta}{2}$ (so $|T_2| = \frac{\beta}{2} - 1$). In order to prove that inequalities (31) and (32) are valid for the (DREFLP) we consider a fixed double-row assignment $\alpha \colon [n] \to [\frac{n}{2}]$ that assigns each of the n departments (original and spaces) to one of the columns. We define $\sigma_1^1 := |\{i \in T_1 \colon \alpha(i) < \alpha(k)\}|$, $\sigma_1^2 := |\{i \in T_2 \colon \alpha(i) < \alpha(k)\}|$, $\sigma_2^1 := |\{i \in T_1 \colon \alpha(i) > \alpha(k)\}|$, $\sigma_2^2 := |\{i \in T_2 \colon \alpha(i) > \alpha(k)\}|$, $\sigma_3^1 := |\{i \in T_1 \colon \alpha(i) = \alpha(k)\}|$, $\sigma_3^2 := |\{i \in T_2 \colon \alpha(i) = \alpha(k)\}|$. Then $\sigma_1^1 + \sigma_2^1 + \sigma_3^1 = \frac{\beta}{2}$, $\sigma_1^2 + \sigma_2^2 + \sigma_3^2 = \frac{\beta}{2} - 1$ and $\sigma_3^1 + \sigma_3^2 \leq 1$. The left-hand side of (31) and (32) calculates to

$$\sigma_1^1 \sigma_2^1 + \sigma_1^2 \sigma_2^2 - \sigma_1^1 \sigma_2^2 - \sigma_2^1 \sigma_1^2 = -(\sigma_1^1 - \sigma_1^2)^2 + \sigma_1^1 - \sigma_1^2 - \sigma_1^1 \sigma_3^1 + \sigma_1^1 \sigma_3^2 + \sigma_3^1 \sigma_1^2 - \sigma_1^2 \sigma_3^2 =: \gamma.$$

We consider three cases:

- $\sigma_3^1 = \sigma_3^2 = 0$: Then $\gamma = -(\sigma_1^1 - \sigma_1^2)^2 + \sigma_1^1 - \sigma_1^2 = -(\sigma_1^1 - \sigma_1^2)(\sigma_1^1 - \sigma_1^2 - 1) \leq 0$ and with $a_{ij} \geq 0$, $i,j \in [n]$, $i < j$, $b_{ijk}, i,j,k \in [n]$, $i < k$, $|\{i,j,k\}| = 3$, the validity follows in this case.

- $\sigma_3^1 = 1, \sigma_3^2 = 0$: Then $\gamma = -(\sigma_1^1 - \sigma_1^2)^2 + \sigma_1^1 - \sigma_1^2 - \sigma_1^1 + \sigma_1^2 = -(\sigma_1^1 - \sigma_1^2)^2$ and with $a_{ij} \geq 0$, $i, j \in [n]$, $i < j$, b_{ijk}, $i, j, k \in [n]$, $i < k$, $|\{i, j, k\}| = 3$, the validity follows in this case.
- $\sigma_3^1 = 0, \sigma_3^2 = 1$: Then $\gamma = -(\sigma_1^1 - \sigma_1^2)^2 + \sigma_1^1 - \sigma_1^2 + \sigma_1^1 - \sigma_1^2 = -(\sigma_1^1 - \sigma_1^2)(\sigma_1^1 - \sigma_1^2 - 2)$. This term is positive if and only if $\sigma_1^1 - \sigma_1^2 = 1$ by the integrality of the σ_i^j.

So, it suffices to show that the right-hand sides of (31) and (32) are at least one if $\sigma_3^1 = 0$, $\sigma_3^2 = 1$ and $\sigma_1^1 - \sigma_1^2 = 1$. For (31) the term $\sigma_3^2 = 1$ implies the existence of an $o \in T_2$ that lies in the same column as k. Considering (32), $\sigma_3^2 = 1$ and $\sigma_1^1 - \sigma_1^2 = 1$ imply $\sigma_1^1 > 0$, $\sigma_2^1 > 0$ and so there exist $p, q \in T_1$, $p \neq q$, and $o \in T_2$ such that o lies between p, q. □

Taking $\beta = 4$ we obtain exactly (20)–(25). In comparison to the variant for the (SRFLP) we added $\sum_{p \in T_2} a_{rp}$ or $\sum_{p,q \in T_1, o \in T_2, p < q} b_{poq}$ to the previous right-hand side with value zero, respectively.

3.3 An SDP formulation related to the linear ordering problem

We now present another new formulation for the (DREFLP). This formulation is based on a quadratic formulation using ordering variables that we rewrite using symmetric matrices. The matrix-based formulation is then relaxed into an SDP problem, and this SDP relaxation can be tightened using several classes of valid constraints.

Our quadratic formulation is based on the ordering variables x_{ij}, $i, j \in [n]$, $i \neq j$, defined as:

$$x_{ij} = \begin{cases} 1, & \text{if department } i \text{ lies left of department } j, \\ -1, & \text{otherwise.} \end{cases} \tag{33}$$

We observed in Subsection 3.2 that the center-to-center distances between departments can be encoded using betweenness variables and column overlap variables. Because we are willing to work with quadratic terms, we can express both of these variables using the ordering variables:

$$b_{ikj} = \tfrac{1}{4}(x_{ik}x_{kj} + x_{jk}x_{ki} + x_{ik} + x_{kj} + x_{jk} + x_{ki}) + \frac{1}{2}, \quad i, j, k \in [n], \ i < j,$$
$$a_{ij} = -\tfrac{1}{2}(x_{ij} + x_{ji}), \quad i, j \in [n], \ i < j. \tag{34}$$

It directly follows that we can rewrite the objective function (16) as a linear-quadratic function of the ordering variables:

$$K + \sum_{\substack{i,j \in [n] \\ i < j}} \frac{w_{ij}}{8} \left(\sum_{\substack{k \in [n] \\ k \neq i, \ k \neq j}} (x_{ik}x_{kj} + x_{jk}x_{ki}) \right) + \sum_{\substack{i,j \in [n] \\ i < j}} \frac{w_{ij}}{4}(x_{ij} + x_{ji}), \tag{35}$$

where K is a constant defined as

$$K := n \left(\sum_{\substack{i,j \in [n] \\ i < j}} \frac{w_{ij}}{4} \right). \tag{36}$$

Any feasible ordering of the departments has to satisfy the 3-cycle inequalities

$$-1 \leq x_{ij} + x_{jk} - x_{ik} \leq 1, \qquad i, j, k \in [n], \ i \neq j \neq k, \ i \neq k. \tag{37}$$

It is well known that the 3-cycle inequalities together with integrality conditions on the ordering variables suffice to describe feasible orderings, see e.g. [51, 54]. In the present context we need the following additional constraints

$$x_{ij} + x_{ji} \leq 0, \quad i, j \in [n], \ i < j, \tag{38}$$

that model the fact that:

- either department i lies to the left of department j;
- or department j lies to the left of department i;
- or both departments are assigned to the same column.

Note from the definition of the ordering variables that if two departments i and j are placed in different columns then $x_{ij} + x_{ji}$ equals zero, while if they are assigned to the same column the sum is -2. This observation is often used in models using ordering variables, such as the ones for the (SRFLP), to halve the number of variables because they require that $x_{ij} + x_{ji} = 0$, i.e., no two departments can overlap. While some overlap is allowed here, we can ensure that exactly two departments are assigned to each column using the constraints

$$\sum_{j \in [n] \setminus \{i\}} (x_{ij} + x_{ji}) = -2, \quad i \in [n]. \tag{39}$$

Next we collect the ordering variables in a vector x and reformulate the (DREFLP) as a quadratic program in ordering variables.

Theorem 5 *Minimizing the objective function* (35) *over* $x \in \{-1,1\}^{n(n-1)}$ *and* (37)–(39) *solves the (DREFLP).*

Proof. The constraints (37)–(39) together with the integrality conditions on x suffice to induce feasible double-row layouts and the definition of the objective function ensures that the distances between departments are computed correctly. □

We can rewrite the quadratic objective function (35) in matrix notation to obtain:

$$\min \{\langle C_X, X \rangle + c_x^\top x + K \colon x \in \{-1,1\}^{n(n-1)} \text{ satisfies } (37)\text{–}(39)\}, \tag{DREFLP}$$

where $X := xx^\top$ and the cost matrix C_X and the cost vector c_x are deduced from (35):

$$\langle C_X, X \rangle = \sum_{\substack{i,j \in [n] \\ i < j}} \frac{w_{ij}}{8} \sum_{\substack{k \in [n] \\ i \neq k \neq j}} (x_{ik} x_{kj} + x_{jk} x_{ki}),$$

$$c_x^\top x = \sum_{\substack{i,j \in [n] \\ i < j}} \frac{w_{ij}}{4} (x_{ij} + x_{ji}).$$

We can further rewrite the above formulation as an SDP by relaxing the nonconvex equation $X - xx^\top = 0$ to the positive semidefinite constraint

$$X - xx^\top \succcurlyeq 0.$$

Moreover, the main diagonal entries of X correspond to squared $\{-1,1\}$ variables, hence $\operatorname{diag}(X) = e$, the vector of all ones. To simplify notation let us introduce

$$Z = Z(x, X) := \begin{pmatrix} 1 & x^\top \\ x & X \end{pmatrix}, \tag{40}$$

where $\dim(Z) = n(n-1)+1$. By the Schur complement lemma [10, Appendix A.5.5], $X - xx^\top \succcurlyeq 0 \Leftrightarrow Z \succcurlyeq 0$. Hence any feasible layout is contained in the elliptope

$$\mathcal{E} := \{Z \colon \operatorname{diag}(Z) = e, \ Z \succcurlyeq 0\}.$$

In order to express constraints on x in terms of X, they have to be reformulated as quadratic conditions in x. A natural way to do this for the 3-cycle inequalities $|x_{ij} + x_{jk} - x_{ik}| = 1$ consists in squaring both sides. Additionally using $x_{ij}^2 = 1$, we obtain

$$x_{ij,jk} - x_{ij,ik} - x_{ik,jk} = -1, \qquad\qquad i,j,k \in [n], \ i \neq j \neq k, \ i \neq k. \tag{41}$$

Now we can formulate the (DREFLP) as a semidefinite optimization problem in binary variables.

Theorem 6 *The problem*

$$\min \left\{ K + \langle C_Z, Z \rangle \colon Z \text{ satisfies } (41),\ Z \in \mathcal{E},\ x \in \{-1,1\}^{n(n-1)} \text{ satisfies } (38) \text{ and } (39) \right\}$$

where Z is given by (40), K is defined in (36) and the cost matrix C_Z is given by

$$C_Z := \begin{pmatrix} 0 & \frac{1}{2}c_x \\ \frac{1}{2}c_x & C_X \end{pmatrix},$$

is equivalent to the (DREFLP).

Proof. Since $x_i^2 = 1$, $i \in \{1,\dots,n(n-1)\}$ we have $\operatorname{diag}(X - xx^\top) = 0$, which together with $X - xx^\top \succcurlyeq 0$ shows that in fact $X = xx^\top$ is integral. Hence the 3-cycle equations (41) ensure that $|x_{ij} + x_{jk} - x_{ik}| = 1$ holds. But the constraints (37)–(39) together with the integrality of x suffice to induce feasible double-row layouts due to Theorem 5. Finally the definition of K and C_Z ensures that the distances between departments are computed correctly. $\qquad\square$

Dropping the integrality condition on the first row and column of Z yields the basic semidefinite relaxation of the (DREFLP):

$$\min \left\{ K + \langle C_Z, Z \rangle : Z \text{ satisfies } (41),\ Z \in \mathcal{E},\ x \text{ satisfies } (38) \text{ and } (39) \right\}. \tag{$\text{SDP}_{\texttt{basic}}$}$$

There are several ways to tighten the above relaxation. First we will concentrate on finding further valid equality constraints. Let us start with showing that the equations (17) from our ILP model are already described via 3-cycle equations (41).

Lemma 7 *The equations (17),*

$$a_{ij} + a_{ik} + a_{jk} + b_{ijk} + b_{ikj} + b_{jik} = 1, \qquad\qquad i,j,k \in [n],\ i < j < k,$$

can be expressed as the sum of two equations of the form (41).

Proof. Applying (34) to (17) gives

$$x_{ik,kj} + x_{jk,ki} + x_{ij,jk} + x_{kj,ji} + x_{ji,ik} + x_{ki,ij} = -2,\ i,j,k \in [n],\ i < j < k,$$

which is the sum of the following two equations from (41):

$$x_{ij,jk} + x_{ki,ij} + x_{jk,ki} = -1, \qquad x_{ik,kj} + x_{kj,ji} + x_{ji,ik} = -1.$$

$$\square$$

Next we add symmetry-breaking constraints arising from the addition of spaces (as already seen in Subsection 3.2):

$$x_{21} = -1, \tag{42}$$

$$x_{ij} = 1, \qquad\qquad i,j \in S,\ i+2 \le j, \tag{43}$$

$$x_{ij} = -1, \qquad\qquad i,j \in S,\ j < i, \tag{44}$$

$$\begin{aligned} x_{i(i+1),ki} - x_{ki} - x_{i(i+1)} &= -1, \\ x_{i(i+1),k(i+1)} - x_{k(i+1)} - x_{i(i+1)} &= -1, \end{aligned} \qquad i \in S,\ i \ne n,\ k \in [d]. \tag{45}$$

Constraint (42) breaks the symmetry of the overall arrangement. Constraints (43) ensure that two spaces i and j can only be assigned to the same column if $i+1 = j$. Equations (44) guarantee that in all layouts considered the spaces have increasing labels when going from left to right. Finally constraints (45) are related to Assumption 1 in Subsection 2.2: if two spaces $i, j \in S$ lie in the same column, then each department $k \in [d]$ has to lie left to them (see also Figure 4).

Lemma 8 *The ILP symmetry-breaking equations* (28)–(30) *can be derived from* (43)–(45).

Proof. Using equations (34) that relate the variables of the ILP and SDP models, we get:

- Let $i, j \in S, i + 2 \leq 2$, then $a_{ij} = -\frac{1}{2}(x_{ij} + x_{ji}) = -\frac{1}{2}(1 - 1) = 0$.
- Let $i, j, k \in S, i + 4 \leq j + 2 \leq k$, then $b_{ijk} = \frac{1}{4}(x_{ij}x_{jk} + x_{kj}x_{ji} + x_{ij} + x_{jk} + x_{kj} + x_{ji}) + \frac{1}{2} = \frac{1}{4}(1 \cdot 1 + (-1) \cdot (-1) + 1 + 1 - 1 - 1) + \frac{1}{2} = 1$.
- Let $i, j, k \in S, i \neq k, j > \max\{i, k\}$, then $b_{ijk} = \frac{1}{4}(x_{ij}x_{jk} + x_{kj}x_{ji} + x_{ij} + x_{jk} + x_{kj} + x_{ji}) + \frac{1}{2} = \frac{1}{4}(-x_{ij} - x_{kj} + x_{ij} - 1 + x_{kj} - 1) + \frac{1}{2} = 0$.
- Let $i, j, k \in S, i \neq k, j < \min\{i, k\}$, then $b_{ijk} = \frac{1}{4}(x_{ij}x_{jk} + x_{kj}x_{ji} + x_{ij} + x_{jk} + x_{kj} + x_{ji}) + \frac{1}{2} = \frac{1}{4}(-x_{jk} - x_{ji} - 1 + x_{jk} - 1 + x_{ji}) + \frac{1}{2} = 0$.

$\square$

Equations (43) and (44) allow us to reduce the size of the semidefinite problem when it comes to the computational experiments in Section 6. However this requires all constraints containing the relevant variables to be transformed accordingly. While this is a straightforward exercise, it involves much technical detail that does not provide further insights. For this reason, we do not include the details of this transformation or of the resulting constraints. (For the same reason, we also chose not to exploit (42) though this could be done in principle.)

Again because we allow quadratic terms, we can express the inequalities (38) as equations:

$$x_{ij}x_{ji} + x_{ij} + x_{ji} = -1, \qquad\qquad\qquad i, j \in [n], \ i < j. \tag{46}$$

Equation (46) is valid because either $x_{ij} = x_{ji} = -1$ (both departments lie in the same column) or $x_{ij} + x_{ji} = 0$ and $x_{ij}x_{ji} = -1$ (they lie in different columns).

The theoretically smoothest way to deal with equations (39) would be to use them to reduce the dimension of the problem by n (for details see [33, Proposition 4.4]). Unfortunately this would make the practical implementation much more complicated. An alternative is to lift (39) into quadratic space via multiplication by an arbitrary ordering variable $x_{lm}, \ l, m \in [n], \ l \neq m$, and the addition of the resulting linear-quadratic equations to the semidefinite relaxation:

$$\sum_{\substack{j \in [n] \\ j \neq i}} (x_{ij}x_{lm} + x_{ji}x_{lm}) = -2x_{lm}, \qquad i, l, m \in [n], \ l \neq m. \tag{47}$$

Another class of valid inequalities for our model are the triangle inequalities of the max-cut polytope, see e.g. [16]: Since Z is generated as the outer product of the vector $\begin{pmatrix} 1 & x \end{pmatrix}^\top$ that has merely $\{-1, 1\}$ entries in the non-relaxed SDP formulation, any feasible layout also belongs to the metric polytope $\mathcal{M}$:

$$\mathcal{M} = \left\{ Z \colon \begin{pmatrix} -1 & -1 & -1 \\ -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{pmatrix} \begin{pmatrix} z_{ij} \\ z_{jk} \\ z_{ik} \end{pmatrix} \leq e, \ 1 \leq i < j < k \leq n(n-1) + 1 \right\}. \tag{48}$$

We note that $\mathcal{M}$ is defined through $\approx 4n^6$ facets.

In summary we get the following tractable semidefinite relaxation of the (DREFLP):

$$\min \left\{ K + \langle C_Z, Z \rangle \colon Z \in \mathcal{E} \cap \mathcal{M} \text{ satisfies } (41)\text{--}(47) \right\}. \tag{SDP$_{\text{full}}$}$$

All variables in Z with cost coefficient greater than zero appear in a 3-cycle equality (41) or in equations (47) and thus are tightly constrained in the relaxation. Such tightly constrained variables are also the reason why the various linear and semidefinite relaxations for the (SRFLP) that are based on betweenness or ordering variables produce very tight bounds in the root node relaxation even for large instances.

4 Extending the models to the multi-row case

In this section we generalize the approaches for double-row problems in the previous section to multi-row problems. Given the set $[d]$ of departments with unit length, we seek an arrangement of these in $m \in \mathbb{N}$ rows such that the weighted sum of the pairwise distances is minimized. Based on the study of optimal solutions, we present adapted versions of the MILP, ILP and SDP models from the double-row case.

We again use Theorem 2 to reduce the (MREFLP) to a space-free version by introducing enough spacing departments. Let c be the minimal number of columns needed in order to preserve at least one of the original optimal solutions. Then our transformed problem has $n := cm$ departments, where $s = n - d$ are spaces.

Extending the MILP formulation from Subsection 3.1. There is a straightforward way to extend the model by Amaral [3] to the multi-row case using the same variables. It suffices to change constraint (7) to

$$\sum_{i \in [d]} y_{ip} \leq m, \quad p \in [c], \tag{49}$$

and we obtain a formulation for the m-row case by optimizing (5) subject to (6), (49), and (8)–(10). For the tighter MILP formulation, (14) and (15) can be adjusted similarly.

Extending the ILP formulation from Subsection 3.2. A formulation of the space-free (MREFLP) is given by

$$\min \sum_{\substack{i,j \in [n], \\ i < j}} \frac{w_{ij}}{m} \cdot \left(\sum_{k \in [n] \setminus \{i,j\}} b_{ikj} + m(1 - a_{ij}) \right) \tag{50}$$

s.t. (19)–(27)

$$a_{ij} + b_{ijk} + b_{ikj} + b_{jik} \leq 1, \qquad\qquad i,j,k \in [n], \ i < j < k \tag{51}$$

$$a_{ik} + b_{ijk} + b_{ikj} + b_{jik} \leq 1, \qquad\qquad i,j,k \in [n], \ i < j < k \tag{52}$$

$$a_{jk} + b_{ijk} + b_{ikj} + b_{jik} \leq 1, \qquad\qquad i,j,k \in [n], \ i < j < k \tag{53}$$

$$a_{ij} + a_{jk} - a_{ik} \leq 1, \qquad\qquad i,j,k \in [n], \ i < k, \ i \neq j \neq k, \tag{54}$$

$$\sum_{j \in [n] \setminus \{i\}} a_{ij} = m - 1, \qquad\qquad i \in [n], \tag{55}$$

$$\sum_{\substack{i,j,k \in [n], \\ i < k, \ j \neq k, \ j \neq i}} b_{ijk} = m^3 \binom{c}{3}. \tag{56}$$

Inequalities (51)–(53) express that three departments $i,j,k \in [n]$, $i < j < k$, either lie next to each other or at least two of them are in the same column. Note, in the double-row case we could use the strengthened version (17). The inequalities (54) enforce the transitivity property that if departments i and j as well as j and k lie in the same column, then i and k also lie in the same column. Equations (55) are the generalization of (18) for the (DREFLP): each i lies in the same column as $m-1$ other departments (possibly spaces). Finally, we know exactly how many betweenness variables equal 1 in a feasible solution: let $c_1, c_2, c_3 \in \{1, \ldots, c\}$ be three different columns of a solution, then for each choice of one department from each of the three columns we count 1 towards the left-hand side of (56).

Extending the SDP formulation from Subsection 3.3. The starting point for our semidefinite relaxation for the (MREFLP) is again a quadratic problem in ordering variables. We use the x-variables defined in (33) and the 3-cycle inequalities (37) as well as (38). We change (39) to

$$\sum_{j \in [n] \setminus \{i\}} (x_{ij} + x_{ji}) = -2m + 2, \quad i \in [n], \tag{57}$$

and adjust the objective function (35)

$$K^m + \sum_{\substack{i,j \in [d], \\ i < j}} \frac{w_{ij}}{4m} \sum_{k \in [n] \setminus \{i,j\}} (x_{ik} x_{kj} + x_{jk} x_{ki}) + \sum_{\substack{i,j \in [d], \\ i < j}} \frac{(m-1) w_{ij}}{2m} (x_{ij} + x_{ji}), \tag{58}$$

where $K^m = \frac{n}{2m} \sum_{\substack{i,j \in [d], \\ i<j}} w_{ij}$.

The following result for the (MREFLP) follows directly from Theorem 5.

Corollary 9 *Minimizing* (58) *over* $x \in \{0,1\}^{n(n-1)}$ *and* (37), (38), (57) *solves the* (MREFLP).

In analogy to the double-row case, we can rewrite the (MREFLP) in matrix notation as

$$\min \{\langle C_X^m, X \rangle + c_x^m x + K^m : x \in \{-1,1\}^{n(n-1)} \text{ satisfies } (37),(38) \text{ and } (57)\}, \qquad \text{(MREFLP)}$$

where $X := xx^\top$ and the cost matrix C_X^m and cost vector c_x^m are deduced from (58). Rewriting the above formulation along the lines of the double-row case gives

$$\min \left\{ K^m + \langle C_Z^m, Z \rangle : Z \text{ satisfies } (41), \; Z \in \mathcal{E}, \; x \in \{-1,1\}^{n(n-1)} \text{ satisfies } (38) \text{ and } (57) \right\}$$

where Z is given by (40), K^m is defined in (58) and the cost matrix C_Z^m is given by

$$C_Z^m := \begin{pmatrix} 0 & \frac{1}{2} c_x^m \\ \frac{1}{2} c_x^m & C_X^m \end{pmatrix}.$$

The basic semidefinite relaxation of the (MREFLP) reads

$$\min \{K^m + \langle C_Z^m, Z \rangle : Z \text{ satisfies } (41), \; Z \in \mathcal{E}, \; x \text{ satisfies } (38) \text{ and } (57)\}. \qquad (\text{SDP}_{\text{basic}}^m)$$

In order to strengthen this relaxation we use

$$\sum_{j \in [n] \setminus \{i\}} (x_{ij} x_{kl} + x_{ji} x_{kl}) = (-2m+2) x_{kl}, \qquad i,k,l \in [n], \; k \neq l, \tag{59}$$

which can be derived by multiplying (57) for fixed i with an x-variable x_{kl}, $k,l \in [n]$, $k \neq l$. Furthermore we can use (46) instead of (38).

Finally, we add constraints to break the symmetry of the spaces S:

$$x_{ij} = 1, \qquad\qquad\qquad i,j \in S, \; i+m \le j, \tag{60}$$
$$x_{ij} = -1, \qquad\qquad\qquad i,j \in S, \; j < i, \tag{61}$$
$$x_{ij,ki} - x_{ki} - x_{ij} = -1, \qquad\qquad\qquad i,j \in S, \; j = i+m-1, \; k \in [d], \tag{62}$$
$$x_{ij,kj} - x_{kj} - x_{ij} = -1, \qquad\qquad\qquad i,j \in S, \; j = i+m-1, \; k \in [d], \tag{63}$$
$$-x_{i(i+j),i(i+k)} + x_{i(i+k)} - x_{i(i+j)} = -1, \qquad i \in S, \; j,k \in \mathbb{N}, \; k < j < m, \; i+j \le n. \tag{64}$$

The constraints (60) and (61) express that two spaces $i,j \in S$, $i < j$, can only lie in the same column if $i+m > j$. If two spaces $i,(i+m-1) \in S$ lie in the same column each of the original departments $k \in [d]$ lies left to them, see (62)–(63). Furthermore, if two spaces $i,(i+j) \in S$ lie in the same column, then all spaces $i+1,\ldots,i+j-1$ also lie in this column by equation (64). Additionally, we can use equation (42) and the triangle inequalities described in (48).

In summary we obtain the following tractable semidefinite relaxation of the (MREFLP):

$$\min \{K^m + \langle C_Z^m, Z \rangle : Z \in \mathcal{E} \cap \mathcal{M} \text{ satisfies } (41), (42), (46) \text{ and } (59)\text{–}(64)\}. \tag{65}$$

5   Implementation

In this section we give details on our implementation of exact approaches based on each of the three formulations proposed for the (DREFLP) and (MREFLP). In Subsection 5.1 we discuss how we computationally solve the respective linear and semidefinite optimization problems to obtain lower bounds on the optimal solution. In Subsection 5.2 we describe heuristics for the semidefinite approach that yield feasible layouts and hence upper bounds to the optimal solution. For the MILP and ILP approaches, the standard solvers provide upper bounds. The combination of upper and lower bounds gives for each instance both a feasible solution and a proof of how far this solution is (at most) from the true optimum.

5.1 Computing the lower bounds

For the Amaral model (6)–(15) we included all the constraints directly and used Gurobi 5.6 [24] as ILP solver. We considered two different versions. Given d departments, we tested both the model allowing d columns per row (as suggested originally by Amaral [3]) and the one with a reduced number of positions according to Theorem 2.

For our new ILP model (17)–(27), tests with Gurobi showed that one should not add all equations at once, but should separate inequalities (19)–(25). We separate (19)–(25) dynamically in a branch-and-cut approach for linear 0-1 problems. These inequalities contain the inequalities (20)–(25) that are important for the success of the approach in [1]. We decided to not additionally separate (31), (32) because only handling (19)–(25) is already computationally challenging. Indeed, we do not add all violated inequalities in each step but rather restrict the number of cutting planes to 1000 in each iteration to keep the computational effort reasonable. The same separation procedure was applied in the multi-row case.

For the SDP approach, we solve our new SDP relaxation using a spectral bundle method [29, 30] in conjunction with primal cutting plane generation [28]. In general the solution of the relaxation is not integer but nevertheless we obtain a lower bound on the layout problem (see [7, 28] for the application of a spectral bundle method in the solution of the max-cut problem and the bisection problem).

One of the main advantages of the spectral bundle method is the ability to exploit the sparsity of the semidefinite relaxation [28]. In the objective function all the entries $x_{ij,kl}$ with $|\{i, j, k, l\}| = 4$ have value zero, and the support of equations (41)–(46) is also small. However (47) as well as the triangle inequalities of the metric polytope $\mathcal{M}$ have a larger support. In order to keep the small support consisting of the first row and column and the entries $x_{ij,kl}$ with $i, j, k, l \in [n], i \neq j, k \neq l, |\{i, j, k, l\}| \leq 3$, we restrict to inequalities (47) with $i \in \{l, m\}$, i.e., we only multiply (39) for $i \in [n]$ fixed with $x_{lm}, l, m \in [n], l \neq m$, if $i \in \{l, m\}$. Additionally we do not include the triangle inequalities and instead add the odd-cycle inequalities [8] (transformed to the -1/1-setting) on the small support of the objective function, where the coefficient matrix is interpreted as the adjacency matrix of a graph. In our experiments we used a separator by C. Helmberg that is an adapted variant of the one by M. Jünger. Note that if we worked with the full support (and thus on a complete graph) and exactly separated the triangle inequalities, then there is no need for an additional odd-cycle separator because all odd-cycles with length at least five are not chordless and are so implied by the other constraints [8].

As mentioned before, (43)–(44) for (DREFLP) and (60)–(61) for (MREFLP) are used to reduce the size of the semidefinite relaxations. In our implementation we add all the equations of (SDP$_{\text{full}}$) and (65) respectively from the beginning (except the ones with large support mentioned above), and then iteratively include the odd-cycle inequalities. After 50 (null or descent) steps of the spectral bundle method we determine violated odd-cycle inequalities and restrict the separation to at most 100 additional constraints. In order to speed up the implementation we also delete constraints if they are not important anymore, see e.g. [7].

5.2 Details on the heuristics used

Gurobi provides upper bounds while solving the MILP and the ILP formulations. We describe here how we derive feasible layouts using SDP primal information. Let $\begin{pmatrix} 1 & \tilde{x} \end{pmatrix}$ denote the first row of the SDP matrix Z. Hence $\tilde{x}$ gives the values of the x-variables in the relaxation. Given a partial solution consisting of k completely filled columns, $k \in \{0, \ldots, \frac{n}{m}\}$ (we arrange departments and spaces simultaneously), we determine and position the next column in a greedy fashion. First, we determine for each subset T of the remaining departments and spaces with $|T| = m$ the sum $\tau_T = \sum_{i,j \in T, i \neq j} \tilde{x}_{ij}$. A small value of τ_T indicates that the m elements of T should be arranged in the same column.[2] Hence we choose the smallest τ_T and arrange the according departments that we denote by C in the same column. Finally we decide on the position of the new column using again the information encoded in $\tilde{x}$.

More precisely let $N \subset [n]$ denote the set of all departments and spaces that have already been assigned and set $l = \frac{|N|}{m}$. The function $\alpha_{part} \colon N \to \left[\frac{|N|}{m}\right]$ gives an assignment of the elements of N to the $\frac{|N|}{m}$ columns.

---

[2]Note that if all departments in T lie in the same column, then $\sum_{i,j \in T, i \neq j} \tilde{x}_{ij} = -m(m-1)$.

Now we calculate for the departments in C

$$\gamma_p = \sum_{\substack{i\in C,\ j\in N \\ \alpha_{part}(j)<p}} \tilde{x}_{ji} + \sum_{\substack{i\in C,\ j\in N \\ \alpha_{part}(j)\geq p}} \tilde{x}_{ij},$$

for all possible positions $p \in [l+1]$. Finally we determine $\hat{p} = \operatorname{argmax}_{p\in[l+1]} \gamma_p$, update α_{part} by

$$\alpha_{part}(i) \leftarrow \begin{cases} \hat{p}, & i\in C, \\ \alpha_{part}(i), & i\in N, \alpha_{part}(i) < \hat{p}, \\ \alpha_{part}(i)+1, & i\in N, \hat{p}\leq\alpha_{part}(i), \end{cases}$$

and set $N \leftarrow N \cup C$.

After the layout is complete, we try to improve it using a 3-OPT heuristic that searches for advantageous exchanges of two or three departments in a greedy fashion. We also test if the solution can be improved by reallocation of any column or by exchanging two or three columns.

Unfortunately, the heuristic described above is only useful in practice if the number of rows m is small, say $m \leq 3$, because of significant memory requirements for larger m. For this reason, we propose a closely related heuristic for larger m.

To reduce memory requirements we determine the departments that lie in the same row in an alternative way. We start with the pair $\{i,j\} \subset [n] \setminus N$ of the currently unassigned departments that minimizes $\tilde{x}_{ij} + \tilde{x}_{ji}$ and set $D = \{i,j\}$. Next we add the department $k \in [n]\setminus(N\cup D)$ to D that minimizes the sum $\sum_{l\in D}(\tilde{x}_{kl}+\tilde{x}_{lk})$. We iterate until $|D| = m$ and set $N \leftarrow N \cup D$. If every department has been assigned to a column, we finally determine the order of the columns in the same way as above. For $m = 2$ the two heuristics are exactly the same. Additionally we used the first heuristic for $m = 3$ and the second cheaper one for $m \in \{3,4,5\}$.

6   Computational experiments

In this section we present some computational results for `DRFLP` instances from the literature as well as instances originally studied for the `SREFLP`. All experiments were conducted on an Intel Core i7 CPU 920 with 2.67 GHz and 12 GB RAM in single processor mode using openSUSE Linux 12.2.

We test the instances used for the `SREFLP` in [34], all instances proposed for the `(DREFLP)` by Amaral [3] (denoted by A-d-{edge probability}), where the pairwise weights w_{ij} are either zero or one because of the underlying graph problem, and the small instances constructed by Hungerländer and Anjos [36] (denoted by E-d-{edge probability}). All instances together with current best upper and lower bounds are available at http://www.miguelanjos.com/flplib.

In Tables 8–11 in the appendix we state the source, the density and the best `(SREFLP)` value of each instance as well as the best upper and lower bounds for the considered single- and multi-row problems. This also shows the effect on the optimal value (or on the value of the best known solution) of a growing number of rows. Also note that most benchmark instances from the layout literature are very dense.

In the following we compare the computation times and final gaps, calculated by the solution approaches presented above. We set the time limit to one hour and extend it to five hours for some larger instances. We calculate the gap between the best layout found[3] and the best lower bound obtained by $\frac{\text{upper bound}}{\text{lower bound}} - 1$, given in percent. We denote the original approach proposed by Amaral [3] using exactly d columns by `MILP I` and the approach to solve the same model using the reduced number of columns preserving at least one optimal solution according to Theorem 2 by `MILP II`. The results for our double- and multi-row ILP models can be found in the columns `ILP` and for our semidefinite programming model in the columns `SDP`.

---

[3]We always use only the layouts found by the heuristic of the respective approach, except for one case: For the instance "A-14-20" with $m = 2$ the SDP heuristic does not find the optimal layout after one hour, although the lower bound is already tight after less then 6 minutes.

The performance of the integer linear programming models changes significantly depending on the number of rows considered. The `ILP` is the best approach for small- to medium-sized instances with $d \leq 17$ in the case $m = 2$, see Tables 2–6. But for larger m, the solution times are much higher than in the case $m = 2$ and the obtained lower bounds are rather weak, see Tables 3–5. Often the ILP solver has problems finding a good upper bound. Hence even small instances with $d = 10$ could not be solved within the time limit of one hour. One explanation for this behavior could be that equations (17) for $m = 2$ are rather strong in comparison to inequalities (51)–(53) for $m \geq 3$. Furthermore, `ILP` and `SDP` both suffer from the fact that for larger d, the number of spaces (additional departments) needed grows with m, see also Table 1 for large d. Thus we present the results of the `ILP` only for the instances where it is a possible alternative to `SDP` ($m = 2$, $d \in \{16, \ldots, 20\}$ and $d \leq 15$, m arbitrary).

Considering the results for `MILP I` and `MILP II`, we observe that reducing the number of columns in the model by Amaral [3] helps to reduce the computation times considerably. Nevertheless our computational results suggest that this improvement is not enough to create a competitive solution approach in general. For quite small instances with $d \leq 15$ one hour is sometimes not enough to solve all instances to optimality using `MILP II`. For this reason we do not present any further experiments for `MILP I` and `MILP II` with $d \geq 16$. Comparing different numbers of rows, we observe that the bounds and the solution times of `MILP II` improve if m increases. One explanation for this behavior is that for larger m the number of columns needed is reduced, and hence the number of variables is also smaller than for $m = 2$. In summary `MILP II` is in most cases the worst approach for instances with $m = 2$ and $d \geq 13$, especially if the density of the instances is high, but it is also sometimes the best algorithm if m is large and d is small ($d \leq 13$).

The lower bounds derived with the `SDP` approach are often very strong, independent of the number of rows. To demonstrate this, we tested the `SDP` approach with a time limit of five hours for all instances with $d \geq 16$. Because of the long computation times and high gaps we do not enlarge the time limit for the `ILP` and the `MILP`. Table 7 shows that especially if $d \geq 30$ then major improvements in the gaps are achieved with the increased time limit: most gaps are below three percent. Looking at the upper bound, our SDP construction heuristics yields significantly better solutions (especially for large d) than the MILP and the ILP.

In summary, we conclude that MILP II is the overall best choice for instances with $d \leq 13$ and $m \geq 3$, the ILP approach is overall the best choice for $d \leq 17$ and $m = 2$, and for all other instances the SDP is the best choice. Moreover the SDP approach is well-suited for providing high-quality lower bounds for large-scale instances in reasonable computation time. Hence it seems promising to use the SDP relaxation within a branch-and-bound scheme. Because there is at present no commercial or other software that does this automatically for SDP relaxations, this possibility is a non-trivial computational project beyond the scope of this paper, and is thus left for future research.

Table 2: Computation times (in mm:ss) for small instances with up to 9 departments and between 2 and 5 rows. All of these instances were solved to optimality within one hour of computation time by all four solution approaches.

Instance	m = 2					m = 3					m = 4					m = 5				
	opt	MILP I	MILP II	ILP	SDP	opt	MILP I	MILP II	ILP	SDP	opt	MILP I	MILP II	ILP	SDP	opt	MILP I	MILP II	ILP	SDP
A-9-10	2	00:01	00:00	00:00	00:00	0	00:00	00:00	00:01	00:00	0	00:00	00:00	00:09	00:01	0	00:00	00:00	10:58	00:04
A-9-20	9	00:02	00:00	00:00	00:00	6	00:02	00:00	00:17	00:06	3	00:00	00:00	00:00	00:01	3	00:01	00:00	00:06	00:12
A-9-30	3	00:00	00:00	00:00	00:00	2	00:00	00:00	00:03	00:03	1	00:00	00:00	00:03	00:01	1	00:00	00:00	00:21	00:06
A-9-40	11	00:03	00:00	00:01	00:01	7	00:02	00:00	00:22	00:09	5	00:01	00:00	00:03	00:03	4	00:01	00:00	00:07	00:16
A-9-50	18	00:16	00:01	00:01	00:02	11	00:07	00:00	00:07	00:06	7	00:03	00:00	00:00	00:01	5	00:02	00:00	00:02	00:05
A-9-60	23	00:19	00:01	00:01	00:02	15	00:10	00:00	00:25	00:49	9	00:03	00:00	00:00	00:01	7	00:02	00:00	00:01	00:04
A-9-70	38	00:57	00:12	00:00	00:21	23	00:20	00:02	00:34	00:03	18	00:13	00:00	00:54	02:27	13	00:07	00:00	00:07	00:31
A-9-80	51	02:10	00:20	00:01	01:08	30	00:48	00:02	02:27	00:01	24	00:23	00:01	03:25	03:58	17	00:11	00:00	00:01	00:31
A-9-90	45	01:43	00:19	00:00	00:18	27	00:18	00:02	01:45	00:01	21	00:17	00:01	01:17	00:58	15	00:05	00:00	00:01	00:18
A-10-10	2	00:01	00:00	00:01	00:01	1	00:00	00:00	03:20	00:07	0	00:00	00:00	13:09	00:04	0	00:00	00:00	00:27	00:04
A-10-20	3	00:01	00:00	00:00	00:00	3	00:01	00:00	00:21	03:03	1	00:01	00:00	00:19	00:05	1	00:01	00:00	00:11	00:05
A-10-30	7	00:04	00:01	00:00	00:01	5	00:04	00:00	00:08	00:55	3	00:01	00:00	00:06	00:19	1	00:01	00:00	00:00	00:02
A-10-50	28	00:44	00:09	00:00	00:01	19	00:37	00:05	51:29	02:10	13	00:13	00:01	06:14	00:35	9	00:04	00:00	00:02	00:13
A-10-60	25	00:47	00:06	00:02	04:10	15	00:15	00:02	32:11	00:25	11	00:09	00:01	08:53	00:38	8	00:03	00:00	00:01	00:14
E-5-50	13	00:00	00:00	00:00	00:00	6	00:00	00:00	00:00	00:00	4	00:00	00:00	00:00	00:00	0	00:00	00:00	00:00	00:00
E-5-100	46	00:00	00:00	00:00	00:00	27	00:00	00:00	00:00	00:00	17	00:00	00:00	00:00	00:00	0	00:00	00:00	00:00	00:00
E-6-50	45	00:00	00:00	00:00	00:00	29	00:00	00:00	00:00	00:09	22	00:00	00:00	00:00	00:03	12	00:00	00:00	00:00	00:03
E-6-100	99	00:01	00:00	00:00	00:00	56	00:00	00:00	00:00	00:00	49	00:00	00:00	00:00	00:19	29	00:00	00:00	00:00	00:22
E-7-50	51	00:01	00:00	00:00	00:07	31	00:01	00:00	00:00	00:23	17	00:00	00:00	00:00	00:00	9	00:00	00:00	00:00	00:01
E-7-100	126	00:06	00:00	00:00	00:47	79	00:04	00:00	00:01	00:33	50	00:01	00:00	00:00	00:01	40	00:01	00:00	00:00	00:09
E-8-50	64	00:02	00:00	00:00	00:03	37	00:01	00:00	00:02	00:09	26	00:01	00:00	00:00	00:10	25	00:01	00:00	00:00	00:36
E-8-100	191	00:23	00:04	00:00	00:11	125	00:11	00:01	01:30	00:33	74	00:06	00:00	00:00	00:01	70	00:04	00:00	00:00	01:49
E-9-50	118	00:18	00:02	00:00	02:12	70	00:09	00:00	00:15	00:49	55	00:06	00:00	00:32	04:09	40	00:02	00:00	00:03	02:21
E-9-100	306	03:09	00:20	00:00	02:50	181	00:42	00:03	00:49	00:10	140	00:28	00:01	01:30	03:03	100	00:12	00:00	00:01	00:41
O-5	70	00:00	00:00	00:00	00:00	38	00:00	00:00	00:00	00:00	32	00:00	00:00	00:00	00:05	0	00:00	00:00	00:00	00:00
O-6	136	00:01	00:00	00:00	00:00	72	00:00	00:00	00:00	00:00	64	00:00	00:00	00:00	00:02	28	00:00	00:00	00:00	00:01
O-7	236	00:05	00:00	00:00	00:13	144	00:02	00:00	00:01	00:13	102	00:01	00:00	00:00	00:03	76	00:01	00:00	00:00	00:07
O-8	366	00:20	00:03	00:00	00:01	250	00:12	00:01	02:20	01:46	148	00:04	00:00	00:00	00:03	138	00:05	00:00	00:00	02:34
O-9	508	02:33	00:19	00:00	00:28	302	00:41	00:02	00:09	00:20	238	00:20	00:01	00:46	02:44	168	00:13	00:00	00:01	01:01
Y-6	630	00:01	00:00	00:00	00:02	350	00:00	00:00	00:00	00:00	315	00:00	00:00	00:00	02:07	193	00:00	00:00	00:00	02:18
Y-7	899	00:08	00:01	00:00	01:15	577	00:04	00:00	00:03	02:01	383	00:02	00:00	00:00	00:09	311	00:01	00:00	00:00	00:41
Y-8	1095	00:39	00:08	00:00	00:27	728	00:17	00:02	05:39	01:17	430	00:05	00:00	00:00	00:01	394	00:06	00:00	00:00	00:23
Y-9	1401	04:03	00:25	00:00	01:37	848	00:59	00:05	01:19	00:04	658	00:39	00:01	02:23	10:17	476	00:13	00:00	00:01	01:16

Table 3: Computation times and gaps for instances with between 10 and 15 departments. Not all methods were able to solve these instances to optimality in the time limit of one hour.

Instance	opt	Gap (%)				Time (hh:mm:ss)				opt	Gap (%)				Time(hh:mm:ss)			
		MILP I	MILP II	ILP	SDP	MILP I	MILP II	ILP	SDP		MILP I	MILP II	ILP	SDP	MILP I	MILP II	ILP	SDP
							$m=2$									$m=3$			
E-10-50	191	0.0	0.0	0.0	0.0	00:02:02	00:00:20	00:00:05	00:05:52	114	0.0	0.0	0.0	0.0	00:00:37	00:00:04	00:01:52	00:00:32
E-10-100	427	0.0	0.0	0.0	0.0	00:19:58	00:01:38	00:00:01	00:01:39	277	0.0	0.0	17.4	0.0	00:04:52	00:00:32	01:00:00	00:01:42
E-11-100	539	23.9	0.0	0.0	0.0	01:00:00	00:31:05	00:00:02	00:04:08	351	0.0	0.0	13.6	0.0	00:26:48	00:01:55	01:00:00	00:06:48
N-15	1064	153.3	68.4	0.0	0.0	01:00:00	01:00:00	00:13:16	00:08:37	668	76.7	16.8	32.3	0.0	01:00:00	01:00:00	01:00:00	00:11:24
O-10	670	0.0	0.0	0.0	0.0	00:07:24	00:01:21	00:00:00	00:00:20	450	0.0	0.0	16.0	0.0	00:02:02	00:00:23	01:00:00	00:02:12
O-15	2556	296.9	139.3	0.0	0.0	01:00:00	01:00:00	00:02:49	00:10:06	1660	165.2	38.6	12.2	0.0	01:00:00	01:00:00	01:00:00	00:53:27
S-12	2167	84.7	17.4	0.0	0.0	01:00:00	01:00:00	00:00:02	00:01:40	1404	48.9	0.0	19.4	0.1	01:00:00	00:37:31	01:00:00	01:00:00
S-13	2940	155.9	58.1	0.0	0.0	01:00:00	01:00:00	00:00:29	00:17:55	1938	92.1	12.7	19.4	0.6	01:00:00	01:00:00	01:00:00	01:00:00
S-14	3608	187.9	74.6	0.0	0.0	01:00:00	01:00:00	00:00:56	00:55:57	2408	159.2	43.4	23.2	0.2	01:00:00	01:00:00	01:00:00	01:00:00
S-15	4466	391.9	131.2	0.0	0.3	01:00:00	01:00:00	00:02:54	01:00:00	2883	192.1	44.8	15.6	0.0	01:00:00	01:00:00	01:00:00	00:51:32
Y-10	1697	0.0	0.0	0.0	0.0	00:27:02	00:03:08	00:00:00	00:00:57	1140	0.0	0.0	25.1	0.0	00:08:37	00:00:53	01:00:00	00:14:28
Y-11	2008	41.4	3.3	0.0	0.0	01:00:00	01:00:00	00:00:07	00:03:57	1314	3.7	0.0	17.0	0.0	01:00:00	00:02:59	01:00:00	00:09:21
Y-12	2342	92.6	22.2	0.0	0.0	01:00:00	01:00:00	00:00:02	00:01:33	1510	33.5	0.0	20.3	0.0	01:00:00	00:39:51	01:00:00	00:06:34
Y-13	2730	138.2	49.5	0.0	0.0	01:00:00	01:00:00	00:00:19	00:25:06	1798	89.7	13.6	17.0	0.0	01:00:00	01:00:00	01:00:00	00:50:55
Y-14	3164	196.3	113.1	0.0	0.0	01:00:00	01:00:00	00:02:22	00:15:15	2110	157.6	48.9	22.3	0.1	01:00:00	01:00:00	01:00:00	01:00:00
Y-15	3676	357.2	114.3	0.0	0.3	01:00:00	01:00:00	00:02:33	01:00:00	2357	204.1	45.0	13.6	0.0	01:00:00	01:00:00	01:00:00	00:06:18
							$m=4$									$m=5$			
E-10-50	89	0.0	0.0	9.9	0.0	00:00:20	00:00:01	01:00:00	00:03:35	59	0.0	0.0	0.0	0.0	00:00:09	00:00:00	00:00:00	00:00:32
E-10-100	209	0.0	0.0	17.4	0.0	00:01:55	00:00:11	01:00:00	00:13:20	133	0.0	0.0	0.0	0.0	00:00:26	00:00:00	00:00:00	00:00:13
E-11-100	256	0.0	0.0	15.8	0.4	00:06:45	00:00:22	01:00:00	01:00:00	191	0.0	0.0	0.0	0.0	00:02:16	00:00:02	00:10:32	00:13:18
N-15	500	38.9	0.0	41.2	0.6	01:00:00	00:39:50	01:00:00	01:00:00	382	0.0	0.0	24.0	1.3	00:41:45	00:02:20	01:00:00	01:00:00
O-10	334	0.0	0.0	14.8	0.0	00:01:03	00:00:06	01:00:00	00:03:17	222	0.0	0.0	0.0	0.0	00:00:20	00:00:00	00:00:00	00:02:02
O-15	1250	106.3	26.3	31.4	1.4	01:00:00	01:00:00	01:00:00	01:00:00	914	62.1	0.0	17.8	0.0	01:00:00	00:09:26	01:00:00	00:11:10
S-12	995	0.0	0.0	14.9	0.0	00:45:02	00:01:06	01:00:00	00:08:40	841	0.0	0.0	19.1	0.0	00:27:09	00:00:58	01:00:00	00:33:11
S-13	1413	48.7	0.0	30.2	0.0	01:00:00	00:54:59	01:00:00	00:42:47	1132	36.4	0.0	24.0	0.3	01:00:00	00:02:58	01:00:00	01:00:00
S-14	1794	114.1	41.1	40.0	2.1	01:00:00	01:00:00	01:00:00	01:00:00	1369	59.0	0.0	22.3	0.0	01:00:00	00:06:39	01:00:00	00:37:36
S-15	2175	149.7	37.3	40.7	1.2	01:00:00	01:00:00	01:00:00	01:00:00	1612	98.0	10.9	22.0	0.3	01:00:00	01:00:00	01:00:00	01:00:00
Y-10	845	0.0	0.0	21.8	0.0	00:02:49	00:00:16	01:00:00	00:19:12	530	0.0	0.0	0.0	0.0	00:00:32	00:00:01	00:00:00	00:01:35
Y-11	947	0.0	0.0	19.4	0.0	00:12:57	00:00:33	01:00:00	00:05:45	724	0.0	0.0	0.0	0.0	00:03:37	00:00:06	00:42:39	00:18:17
Y-12	1070	0.0	0.0	15.3	0.0	00:35:45	00:01:02	01:00:00	00:01:25	908	0.0	0.0	19.3	0.0	00:28:25	00:00:53	01:00:00	00:27:49
Y-13	1314	63.2	0.0	30.1	0.0	01:00:00	00:20:12	01:00:00	00:45:32	1048	24.3	0.0	23.7	0.6	01:00:00	00:02:50	01:00:00	01:00:00
Y-14	1574	113.0	41.2	41.3	1.8	01:00:00	01:00:00	01:00:00	01:00:00	1201	76.9	0.0	23.3	0.6	01:00:00	00:10:28	01:00:00	01:00:00
Y-15	1782	180.6	41.8	41.4	1.1	01:00:00	01:00:00	01:00:00	01:00:00	1322	101.5	13.2	22.5	0.4	01:00:00	01:00:00	01:00:00	01:00:00

Table 4: Computation times and gaps for small instances from Amaral [3] with $d \in \{10, 11, 12, 13, 14\}, m \in \{2, 3\}$. Not all methods were able to solve these instances to optimality in the time limit of one hour.

		Gap (%)				Time (hh:mm:ss)					Gap (%)				Time (hh:mm:ss)			
Instance	opt	MILP I	MILP II	ILP	SDP	MILP I	MILP II	ILP	SDP	opt	MILP I	MILP II	ILP	SDP	MILP I	MILP II	ILP	SDP
						$m = 2$									$m = 3$			
A-10-40	30	0.0	0.0	0.0	0.0	00:02:21	00:00:09	00:00:01	00:00:17	20	0.0	0.0	5.3	0.0	00:00:38	00:00:06	01:00:00	00:01:43
A-10-70	49	0.0	0.0	0.0	0.0	00:03:55	00:00:31	00:00:01	00:00:02	33	0.0	0.0	10.0	0.0	00:01:43	00:00:17	01:00:00	00:01:24
A-10-80	65	0.0	0.0	0.0	0.0	00:13:43	00:01:51	00:00:00	00:00:04	44	0.0	0.0	25.7	0.0	00:03:28	00:00:42	01:00:00	00:07:09
A-10-90	65	0.0	0.0	0.0	0.0	00:11:28	00:01:52	00:00:00	00:00:05	44	0.0	0.0	25.7	0.0	00:03:32	00:00:33	01:00:00	00:04:38
A-11-10	0	-	-	-	-	00:00:00	00:00:00	00:00:15	00:00:01	0	-	-	-	-	00:00:00	00:00:00	00:01:58	00:00:07
A-11-20	17	0.0	0.0	0.0	0.0	00:00:42	00:00:12	00:00:02	00:00:31	11	0.0	0.0	0.0	0.0	00:00:28	00:00:04	00:49:57	00:06:41
A-11-30	25	0.0	0.0	0.0	0.0	00:01:45	00:00:19	00:00:02	00:00:13	16	0.0	0.0	0.0	0.0	00:00:50	00:00:06	00:01:47	00:00:23
A-11-40	30	0.0	0.0	0.0	0.0	00:01:53	00:00:26	00:00:09	00:00:26	20	0.0	0.0	05.3	0.0	00:00:36	00:00:06	01:00:00	00:11:32
A-11-50	51	0.0	0.0	0.0	0.0	00:06:05	00:01:17	00:00:07	00:00:49	34	0.0	0.0	13.3	0.0	00:01:59	00:00:27	01:00:00	00:04:60
A-11-60	37	0.0	0.0	0.0	0.0	00:01:22	00:00:29	00:00:01	00:00:07	24	0.0	0.0	0.0	0.0	00:00:44	00:00:09	00:01:50	00:00:12
A-11-70	54	0.0	0.0	0.0	0.0	00:16:12	00:02:25	00:00:03	00:01:57	35	0.0	0.0	6.1	0.0	00:02:60	00:00:25	01:00:00	00:01:30
A-11-80	74	0.0	0.0	0.0	0.0	00:35:21	00:09:54	00:00:02	00:00:55	49	0.0	0.0	16.7	0.0	00:07:56	00:01:07	01:00:00	00:03:29
A-11-90	101	36.5	16.1	0.0	0.0	01:00:00	01:00:00	00:00:04	00:07:14	66	8.2	0.0	20.0	0.0	01:00:00	00:04:04	01:00:00	00:04:16
A-12-10	1	0.0	0.0	0.0	0.0	00:00:00	00:00:00	00:00:10	00:00:02	1	0.0	0.0	0.0	0.0	00:00:00	00:00:00	00:12:04	00:01:23
A-12-20	11	0.0	0.0	0.0	0.0	00:00:15	00:00:03	00:00:03	00:00:04	7	0.0	0.0	0.0	0.0	00:00:02	00:00:02	01:00:00	00:01:11
A-12-30	13	0.0	0.0	0.0	0.0	00:00:41	00:00:07	00:00:03	00:00:16	8	0.0	0.0	0.0	0.0	00:00:22	00:00:02	01:00:00	00:01:51
A-12-40	37	0.0	0.0	0.0	0.0	00:35:05	00:00:54	00:00:03	00:00:12	24	0.0	0.0	4.3	0.0	00:03:08	00:00:29	01:00:00	00:01:43
A-12-50	43	0.0	0.0	0.0	0.0	00:07:19	00:01:08	00:00:04	00:00:38	27	0.0	0.0	0.0	0.0	00:01:41	00:00:30	01:00:00	00:00:43
A-12-60	53	0.0	0.0	0.0	0.0	00:51:40	00:02:11	00:00:02	00:00:49	33	0.0	0.0	3.1	0.0	00:04:25	00:00:46	01:00:00	00:00:25
A-12-70	77	24.2	0.0	0.0	0.0	01:00:00	00:27:12	00:00:04	00:00:57	49	0.0	0.0	11.4	0.0	00:40:15	00:01:43	01:00:00	00:00:60
A-12-80	102	61.9	9.7	0.0	0.0	01:00:00	01:00:00	00:00:01	00:00:17	65	10.2	0.0	16.1	0.0	01:00:00	00:05:31	01:00:00	00:00:12
A-12-90	108	52.1	24.1	0.0	0.0	01:00:00	01:00:00	00:00:04	00:00:26	70	29.6	0.0	22.8	0.0	01:00:00	00:13:01	01:00:00	00:01:04
A-13-10	2	0.0	0.0	0.0	0.0	00:00:01	00:00:00	00:00:17	00:00:05	1	0.0	0.0	0.0	0.0	00:00:00	00:00:00	00:50:47	00:00:38
A-13-20	24	0.0	0.0	0.0	0.0	00:05:16	00:00:29	00:00:38	00:03:22	15	0.0	0.0	0.0	0.0	00:01:01	00:00:07	01:00:00	00:03:03
A-13-30	38	0.0	0.0	0.0	0.0	00:13:42	00:01:30	00:00:53	00:01:16	25	0.0	0.0	8.7	0.0	00:04:51	00:00:33	01:00:00	00:03:02
A-13-40	42	31.2	0.0	0.0	0.0	01:00:00	00:06:38	00:00:07	00:02:03	27	0.0	0.0	3.8	0.0	00:10:59	00:00:33	01:00:00	00:02:57
A-13-50	68	44.7	0.0	0.0	0.0	01:00:00	00:18:59	00:00:42	00:04:54	44	12.8	0.0	7.3	0.0	01:00:00	00:02:48	01:00:00	00:03:52
A-13-60	70	29.6	0.0	0.0	0.0	01:00:00	00:17:23	00:00:11	00:02:05	46	0.0	0.0	4.5	0.0	00:44:06	00:02:14	01:00:00	00:06:27
A-13-70	105	72.1	12.9	0.0	0.0	01:00:00	01:00:00	00:06:56	00:04:55	69	38.0	0.0	16.9	0.0	01:00:00	00:12:26	01:00:00	00:11:16
A-13-80	138	126.2	43.8	0.0	0.0	01:00:00	01:00:00	00:00:03	00:04:08	90	76.5	0.0	15.4	0.0	01:00:00	00:51:03	01:00:00	00:03:32
A-13-90	153	146.8	56.1	0.0	0.0	01:00:00	01:00:00	00:00:09	00:10:42	101	110.4	11.0	20.2	0.0	01:00:00	01:00:00	01:00:00	00:53:40
A-14-10	4	0.0	0.0	0.0	0.0	00:00:02	00:00:01	00:00:31	00:00:18	3	0.0	0.0	0.0	0.0	00:00:02	00:00:01	01:00:00	00:06:15
A-14-20	24	0.0	0.0	0.0	0.0	00:03:45	00:00:36	00:19:20	00:05:44	16	0.0	0.0	14.3	6.7	00:01:21	00:00:10	01:00:00	01:00:00
A-14-30	36	0.0	0.0	0.0	0.0	00:40:52	00:07:46	00:02:03	00:03:39	24	0.0	0.0	33.3	0.0	00:12:18	00:02:01	01:00:00	00:19:60
A-14-40	43	26.5	0.0	0.0	0.0	01:00:00	00:15:18	00:01:41	00:02:40	28	0.0	0.0	33.3	0.0	00:21:14	00:01:30	01:00:00	00:08:31
A-14-50	94	118.6	46.9	0.0	0.0	01:00:00	01:00:00	00:19:16	00:05:02	63	70.3	0.0	18.9	1.6	01:00:00	00:21:16	01:00:00	01:00:00
A-14-60	99	86.8	39.4	0.0	0.0	01:00:00	01:00:00	00:10:18	00:01:12	65	54.8	0.0	12.1	0.0	01:00:00	00:17:05	01:00:00	00:03:19
A-14-70	138	133.9	86.5	0.0	0.0	01:00:00	01:00:00	00:08:36	00:08:57	92	114.0	22.7	22.7	0.0	01:00:00	01:00:00	01:00:00	00:35:22
A-14-80	167	153.0	74.0	0.0	0.0	01:00:00	01:00:00	00:00:21	00:04:32	111	122.0	22.0	24.7	0.0	01:00:00	01:00:00	01:00:00	00:13:02
A-14-90	187	179.1	96.8	0.0	0.0	01:00:00	01:00:00	00:14:15	00:03:17	125	160.4	42.0	31.6	0.8	01:00:00	01:00:00	01:00:00	01:00:00

Table 5: Computation times and gaps for small instances from Amaral [3] with $d \in \{10, 11, 12, 13, 14\}, m \in \{4, 5\}$. Not all methods were able to solve these instances to optimality in the time limit of one hour.

		Gap (%)				Time (hh:mm:ss)					Gap (%)				Time (hh:mm:ss)			
Instance	opt	MILP I	MILP II	ILP	SDP	MILP I	MILP II	ILP	SDP	opt	MILP I	MILP II	ILP	SDP	MILP I	MILP II	ILP	SDP
								$m = 4$									$m = 5$	
A-10-40	13	0.0	0.0	0.0	0.0	00:00:12	00:00:01	00:04:16	00:00:07	10	0.0	0.0	0.0	0.0	00:00:10	00:00:00	00:00:04	00:00:12
A-10-70	24	0.0	0.0	9.1	0.0	00:00:35	00:00:06	01:00:00	00:00:59	16	0.0	0.0	0.0	0.0	00:00:15	00:00:00	00:00:01	00:00:07
A-10-80	32	0.0	0.0	14.3	0.0	00:01:33	00:00:08	01:00:00	00:03:03	21	0.0	0.0	0.0	0.0	00:00:27	00:00:01	00:00:00	00:00:06
A-10-90	32	0.0	0.0	14.3	0.0	00:01:22	00:00:05	01:00:00	00:02:12	21	0.0	0.0	0.0	0.0	00:00:25	00:00:00	00:00:01	00:00:04
A-11-10	0	-	-	-	-	00:00:00	00:00:00	00:37:50	00:00:08	0	-	-	-	-	00:00:00	00:00:00	00:06:01	00:00:04
A-11-20	8	0.0	0.0	14.3	0.0	00:00:12	00:00:00	01:00:00	00:01:09	5	0.0	0.0	0.0	0.0	00:00:06	00:00:00	00:00:01	00:00:04
A-11-30	12	0.0	0.0	0.0	0.0	00:00:27	00:00:01	01:00:00	00:00:49	8	0.0	0.0	0.0	0.0	00:00:13	00:00:00	00:00:05	00:00:06
A-11-40	14	0.0	0.0	0.0	0.0	00:00:22	00:00:01	01:00:00	00:00:47	11	0.0	0.0	0.0	0.0	00:00:16	00:00:00	00:07:14	00:01:38
A-11-50	24	0.0	0.0	9.1	0.0	00:00:48	00:00:05	01:00:00	00:01:11	18	0.0	0.0	0.0	0.0	00:00:31	00:00:00	00:05:39	00:03:10
A-11-60	18	0.0	0.0	0.0	0.0	00:00:26	00:00:01	00:05:02	00:00:37	14	0.0	0.0	0.0	0.0	00:00:16	00:00:00	00:00:55	00:00:52
A-11-70	26	0.0	0.0	13.0	0.0	00:01:38	00:00:06	01:00:00	00:04:40	19	0.0	0.0	0.0	0.0	00:00:36	00:00:01	00:03:42	00:01:13
A-11-80	36	0.0	0.0	12.5	0.0	00:05:23	00:00:16	01:00:00	00:03:54	27	0.0	0.0	0.0	0.0	00:01:27	00:00:02	00:20:52	00:03:11
A-11-90	48	0.0	0.0	20.0	0.0	00:13:44	00:00:40	01:00:00	00:03:46	37	0.0	0.0	8.8	0.0	00:04:15	00:00:06	01:00:00	00:10:08
A-12-10	0	-	-	-	-	00:00:00	00:00:00	00:03:54	00:00:06	0	-	-	-	-	00:00:00	00:00:00	01:00:00	00:00:27
A-12-20	5	0.0	0.0	0.0	0.0	00:00:04	00:00:00	00:00:18	00:00:27	4	0.0	0.0	0.0	0.0	00:00:02	00:00:00	00:01:24	00:02:03
A-12-30	5	0.0	0.0	0.0	0.0	00:00:07	00:00:00	00:02:48	00:00:21	4	0.0	0.0	0.0	0.0	00:00:08	00:00:00	01:00:00	00:01:25
A-12-40	16	0.0	0.0	0.0	0.0	00:01:07	00:00:01	00:06:57	00:00:09	15	0.0	0.0	15.4	0.0	00:00:54	00:00:02	01:00:00	00:15:13
A-12-50	20	0.0	0.0	0.0	0.0	00:00:45	00:00:02	00:25:15	00:01:21	17	0.0	0.0	13.3	0.0	00:00:34	00:00:02	01:00:00	00:08:46
A-12-60	24	0.0	0.0	0.0	0.0	00:01:08	00:00:03	01:00:00	00:00:35	21	0.0	0.0	16.7	0.0	00:01:34	00:00:07	01:00:00	00:17:52
A-12-70	34	0.0	0.0	0.0	0.0	00:02:49	00:00:08	01:00:00	00:00:06	30	0.0	0.0	15.4	0.0	00:03:04	00:00:12	01:00:00	00:26:15
A-12-80	47	0.0	0.0	9.3	0.0	00:15:20	00:00:26	01:00:00	00:00:13	40	0.0	0.0	14.3	0.0	00:12:02	00:00:28	01:00:00	00:15:27
A-12-90	50	0.0	0.0	13.6	0.0	00:16:58	00:00:38	01:00:00	00:00:22	42	0.0	0.0	13.5	0.0	00:10:45	00:00:29	01:00:00	00:20:46
A-13-10	1	0.0	0.0	0.0	0.0	00:00:00	00:00:00	01:00:00	00:01:28	1	0.0	0.0	0.0	0.0	00:00:00	00:00:00	01:00:00	00:01:36
A-13-20	11	0.0	0.0	10.0	0.0	00:00:35	00:00:02	01:00:00	00:04:13	9	0.0	0.0	12.5	0.0	00:00:22	00:00:01	01:00:00	00:12:14
A-13-30	19	0.0	0.0	18.8	0.0	00:01:58	00:00:08	01:00:00	00:19:49	14	0.0	0.0	7.7	0.0	00:00:45	00:00:02	01:00:00	00:01:40
A-13-40	20	0.0	0.0	17.6	0.0	00:02:21	00:00:12	01:00:00	00:07:48	16	0.0	0.0	14.3	0.0	00:01:56	00:00:02	01:00:00	00:05:03
A-13-50	32	0.0	0.0	23.1	0.0	00:10:45	00:00:49	01:00:00	00:05:22	26	0.0	0.0	23.8	0.0	00:04:03	00:00:13	01:00:00	00:09:35
A-13-60	33	0.0	0.0	13.8	0.0	00:12:30	00:00:43	01:00:00	00:04:53	26	0.0	0.0	13.0	0.0	00:03:16	00:00:11	01:00:00	00:07:40
A-13-70	50	0.0	0.0	25.0	0.0	00:50:12	00:01:49	01:00:00	00:12:21	41	0.0	0.0	20.6	2.5	00:31:39	00:00:29	01:00:00	01:00:00
A-13-80	66	46.7	0.0	22.2	0.0	01:00:00	00:05:15	01:00:00	00:16:37	53	15.2	0.0	15.2	0.0	01:00:00	00:01:06	01:00:00	00:11:55
A-13-90	74	39.6	0.0	27.6	0.0	01:00:00	00:14:12	01:00:00	00:48:06	58	18.4	0.0	18.4	0.0	01:00:00	00:01:50	01:00:00	00:09:02
A-14-10	1	0.0	0.0	0.0	0.0	00:00:01	00:00:00	01:00:00	00:02:57	1	0.0	0.0	0.0	0.0	00:00:01	00:00:00	00:05:53	00:00:45
A-14-20	11	0.0	0.0	10.0	0.0	00:01:21	00:00:07	01:00:00	00:16:15	8	0.0	0.0	0.0	0.0	00:00:23	00:00:00	01:00:00	00:01:28
A-14-30	18	0.0	0.0	28.6	5.9	00:03:34	00:00:40	01:00:00	01:00:00	13	0.0	0.0	8.3	0.0	00:01:56	00:00:05	01:00:00	00:03:06
A-14-40	21	0.0	0.0	31.2	0.0	00:07:14	00:01:17	01:00:00	00:53:19	16	0.0	0.0	14.3	0.0	00:02:36	00:00:02	01:00:00	00:11:06
A-14-50	47	42.4	0.0	46.9	2.2	01:00:00	00:07:14	01:00:00	01:00:00	35	0.0	0.0	20.7	0.0	00:12:46	00:00:20	01:00:00	00:11:11
A-14-60	49	19.5	0.0	32.4	0.0	01:00:00	00:05:46	01:00:00	00:43:05	37	0.0	0.0	12.1	0.0	00:34:09	00:00:24	01:00:00	00:03:30
A-14-70	68	54.5	0.0	38.8	1.5	01:00:00	00:41:36	01:00:00	01:00:00	52	15.6	0.0	15.6	0.0	01:00:00	00:01:22	01:00:00	00:06:24
A-14-80	83	62.7	10.7	38.3	1.2	01:00:00	01:00:00	01:00:00	01:00:00	64	28.0	0.0	18.5	0.0	01:00:00	00:03:41	01:00:00	00:09:16
A-14-90	93	93.8	31.0	38.8	1.1	01:00:00	01:00:00	01:00:00	01:00:00	71	65.1	0.0	18.3	0.0	01:00:00	00:07:01	01:00:00	00:14:20

Table 6: Computation times (in hh:mm:ss) and gaps (in percent) for medium-sized instances solved with SDP and with ILP in the case $m = 2$.

| | | $m = 2$ | | | | | | $m = 3$ | | | | | $m = 4$ | | | | | $m = 5$ | | | | |
| | | ILP 1h | | SDP 1h | | SDP 5h | | | SDP 1h | | SDP 5h | | | SDP 1h | | SDP 5h | | | SDP 1h | | SDP 5h | |
instance	ub	gap	time	gap	time	gap	time	ub	gap	time	gap	time	ub	gap	time	gap	time	ub	gap	time	gap	time
A-20-10	12	20.0	01:00:00	9.1	01:00:00	9.1	05:00:00	7	16.7	01:00:00	16.7	05:00:00	4	33.3	01:00:00	0.0	03:50:42	3	0.0	00:26:33		
A-20-20	73	87.2	01:00:00	0.0	00:33:30			49	4.3	01:00:00	2.1	05:00:00	34	3.0	01:00:00	0.0	01:21:34	27	3.8	01:00:00	0.0	01:36:52
A-20-30	111	68.2	01:00:00	0.0	00:20:13			74	1.4	01:00:00	1.4	05:00:00	54	3.8	01:00:00	1.9	05:00:00	42	2.4	01:00:00	0.0	02:28:39
A-20-40	149	50.5	01:00:00	0.0	00:35:02			98	1.0	01:00:00	0.0	01:13:53	73	2.8	01:00:00	1.4	05:00:00	58	3.6	01:00:00	1.8	05:00:00
A-20-50	249	11.2	01:00:00	0.0	00:52:37			166	1.2	01:00:00	0.6	05:00:00	122	1.7	01:00:00	0.8	05:00:00	96	2.1	01:00:00	1.1	05:00:00
A-20-60	345	28.7	01:00:00	0.3	01:00:00	0.0	01:21:58	229	0.9	01:00:00	0.4	05:00:00	167	0.6	01:00:00	0.0	01:51:21	132	0.8	01:00:00	0.8	05:00:00
A-20-70	385	25.0	01:00:00	0.0	00:20:58			258	1.6	01:00:00	0.4	05:00:00	187	0.5	01:00:00	0.5	05:00:00	146	0.7	01:00:00	0.0	01:20:43
A-20-80	434	17.0	01:00:00	0.2	01:00:00	0.0	01:17:34	290	1.4	01:00:00	0.7	05:00:00	211	1.0	01:00:00	0.5	05:00:00	165	0.6	01:00:00	0.0	03:47:42
A-20-90	521	6.8	01:00:00	0.2	01:00:00	0.2	02:12:30	347	0.9	01:00:00	0.0	03:51:36	252	0.0	00:41:05			197	0.0	00:23:50		
N-16a	1496	0.0	00:21:48	0.0	00:09:33			1002	0.9	01:00:00	0.0	02:15:36	706	0.0	00:16:06			584	2.3	01:00:00	0.0	03:11:45
N-16b	1168	0.0	00:06:02	0.0	00:06:15			792	1.7	01:00:00	0.8	05:00:00	570	1.6	01:00:00	0.0	03:25:47	462	2.7	01:00:00	0.9	05:00:00
N-17	1678	0.0	00:35:31	0.0	00:27:20			1114	1.3	01:00:00	0.0	02:23:52	808	0.0	00:28:04			662	2.8	01:00:00	0.5	05:00:00
N-18	1970	0.0	00:34:44	0.0	00:45:05			1292	0.6	01:00:00	0.4	05:00:00	972	1.7	01:00:00	0.0	01:19:23	772	2.3	01:00:00	0.0	03:44:02
N-20	2782	38.8	01:00:00	0.0	00:31:52			1856	1.6	01:00:00	0.5	05:00:00	1360	2.0	01:00:00	1.0	05:00:00	1068	1.9	01:00:00	0.7	05:00:00
O-20	6414	20.9	01:00:00	0.0	00:54:27			4284	1.3	01:00:00	0.4	05:00:00	3118	0.8	01:00:00	0.1	05:00:00	2444	1.0	01:00:00	0.5	05:00:00
S-16	5446	0.0	00:20:56	0.0	00:12:41			3638	1.0	01:00:00	0.0	04:55:13	2600	0.0	00:48:46			2094	1.5	01:00:00	0.0	03:40:35
S-17	6577	0.0	00:35:02	0.3	01:00:00	0.0	01:49:13	4354	0.6	01:00:00	0.3	05:00:00	3225	2.0	01:00:00	0.9	05:00:00	2577	1.7	01:00:00	0.4	05:00:00
S-18	7788	0.2	01:00:00	0.1	01:00:00	0.0	05:00:00	5110	0.2	01:00:00	0.1	05:00:00	3892	1.9	01:00:00	1.0	05:00:00	3083	2.8	01:00:00	1.1	05:00:00
S-19	9343	1.1	01:00:00	0.5	01:00:00	0.3	05:00:00	6190	1.2	01:00:00	0.7	05:00:00	4599	2.2	01:00:00	1.2	05:00:00	3614	1.4	01:00:00	0.6	05:00:00
S-20	10841	8.5	01:00:00	0.1	01:00:00	0.0	02:58:26	7227	1.2	01:00:00	0.6	05:00:00	5260	0.4	01:00:00	0.2	05:00:00	4105	0.0	01:00:00	0.0	01:03:54
Y-20	6046	9.3	01:00:00	0.0	01:00:00	0.0	01:34:03	4033	0.9	01:00:00	0.4	05:00:00	2934	0.4	01:00:00	0.1	05:00:00	2282	0.0	00:17:29		

Table 7: Computation times (in hh:mm:ss) and gaps (in percent) for large instances solved with SDP.

| | | $m = 2$ | | | | | $m = 3$ | | | | | $m = 4$ | | | | | $m = 5$ | | | |
| | | SDP 1h | | SDP 5h | | | SDP 1h | | SDP 5h | | | SDP 1h | | SDP 5h | | | SDP 1h | | SDP 5h | |
instance	best ub	gap	time	gap	time	best ub	gap	time	gap	time	best ub	gap	time	gap	time	best ub	gap	time	gap	time
A-25-10	41	2.5	01:00:00	0.0	01:05:06	27	8.0	01:00:00	3.8	05:00:00	20	25.0	01:00:00	11.1	05:00:00	15	7.1	01:00:00	0.0	03:16:36
A-25-20	110	0.9	01:00:00	0.0	03:08:29	74	7.2	01:00:00	4.2	05:00:00	55	14.6	01:00:00	5.8	05:00:00	40	2.6	01:00:00	0.0	02:10:41
A-25-30	222	0.5	01:00:00	0.0	03:01:22	146	1.4	01:00:00	0.0	04:01:29	110	5.8	01:00:00	1.9	04:01:29	87	4.8	01:00:00	1.2	05:00:00
A-25-40	400	1.0	01:00:00	0.3	05:00:00	265	1.9	01:00:00	0.8	05:00:00	198	4.2	01:00:00	2.1	05:00:00	156	4.7	01:00:00	2.6	05:00:00
A-25-50	511	0.4	01:00:00	0.4	05:00:00	340	2.1	01:00:00	1.2	05:00:00	254	4.1	01:00:00	2.4	05:00:00	196	1.6	01:00:00	0.5	05:00:00
A-25-60	549	0.9	01:00:00	0.4	05:00:00	364	1.7	01:00:00	1.1	05:00:00	271	3.0	01:00:00	1.9	05:00:00	212	2.4	01:00:00	1.4	05:00:00
A-25-70	660	0.9	01:00:00	0.3	05:00:00	438	1.6	01:00:00	0.7	05:00:00	325	2.8	01:00:00	1.2	05:00:00	255	2.0	01:00:00	1.2	05:00:00
A-25-80	910	0.4	01:00:00	0.3	05:00:00	604	1.0	01:00:00	0.7	05:00:00	450	2.0	01:00:00	1.4	05:00:00	350	0.6	01:00:00	0.3	05:00:00
A-25-90	1084	0.4	01:00:00	0.4	05:00:00	721	1.1	01:00:00	0.7	05:00:00	537	1.7	01:00:00	1.3	05:00:00	417	0.2	01:00:00	0.0	01:54:59
N-21	2512	0.5	01:00:00	0.0	01:04:30	1664	2.0	01:00:00	1.3	05:00:00	1248	3.7	01:00:00	1.9	05:00:00	972	3.3	01:00:00	0.7	05:00:00
N-22	3064	0.9	01:00:00	0.2	05:00:00	2034	2.8	01:00:00	0.8	05:00:00	1510	2.2	01:00:00	0.9	05:00:00	1188	2.8	01:00:00	0.0	04:08:02
N-24	4120	2.4	01:00:00	0.6	05:00:00	2712	1.5	01:00:00	0.6	05:00:00	2010	3.4	01:00:00	1.3	05:00:00	1624	4.3	01:00:00	2.1	05:00:00
N-25	4604	1.9	01:00:00	0.0	04:41:49	3062	1.8	01:00:00	0.7	05:00:00	2286	4.4	01:00:00	2.0	05:00:00	1796	3.2	01:00:00	1.3	05:00:00
N-30	8230	1.6	01:00:00	0.4	05:00:00	5442	2.4	01:00:00	0.7	05:00:00	4086	4.3	01:00:00	2.1	05:00:00	3232	4.5	01:00:00	2.1	05:00:00
S-21	12431	0.6	01:00:00	0.2	05:00:00	8144	0.2	01:00:00	0.0	03:26:10	6136	2.1	01:00:00	1.6	05:00:00	4849	2.7	01:00:00	1.8	05:00:00
S-22	14208	0.1	01:00:00	0.0	05:00:00	9484	1.4	01:00:00	0.8	05:00:00	7082	2.1	01:00:00	1.2	05:00:00	5623	2.9	01:00:00	1.3	05:00:00
S-23	16521	0.8	01:00:00	0.4	05:00:00	10974	1.3	01:00:00	0.7	05:00:00	8159	1.5	01:00:00	0.9	05:00:00	6523	2.7	01:00:00	1.1	05:00:00
S-24	18658	0.3	01:00:00	0.1	05:00:00	12349	0.5	01:00:00	0.2	05:00:00	9147	0.8	01:00:00	0.3	05:00:00	7342	1.9	01:00:00	1.1	05:00:00
S-25	21172	0.8	01:00:00	0.4	05:00:00	14070	1.2	01:00:00	0.9	05:00:00	10487	2.2	01:00:00	1.5	05:00:00	8149	0.4	01:00:00	0.0	05:00:00
Y-25	10170	0.8	01:00:00	0.3	05:00:00	6761	1.3	01:00:00	0.8	05:00:00	5049	2.5	01:00:00	1.7	05:00:00	3930	1.0	01:00:00	0.5	05:00:00
Y-30	13790	0.5	01:00:00	0.1	05:00:00	9133	1.0	01:00:00	0.3	05:00:00	6889	2.8	01:00:00	1.8	05:00:00	5386	1.5	01:00:00	0.6	05:00:00
Y-35	19087	0.5	01:00:00	0.3	05:00:00	12705	1.7	01:00:00	0.5	05:00:00	9492	3.6	01:00:00	1.4	05:00:00	7504	2.1	01:00:00	0.8	05:00:00
Y-40	23749	0.9	01:00:00	0.4	05:00:00	15825	3.5	01:00:00	1.3	05:00:00	11785	4.9	01:00:00	1.5	05:00:00	9381	5.0	01:00:00	1.4	05:00:00
Y-45	31442	1.6	01:00:00	0.7	05:00:00	20896	4.5	01:00:00	1.5	05:00:00	15663	9.6	01:00:00	2.5	05:00:00	12442	7.8	01:00:00	2.1	05:00:00
Y-50	41517	3.2	01:00:00	0.9	05:00:00	27674	8.1	01:00:00	2.0	05:00:00	20809	15.7	01:00:00	4.5	05:00:00	16475	10.3	01:00:00	2.8	05:00:00
Y-60	55996	11.7	01:00:00	1.9	05:00:00	37279	18.3	01:00:00	4.1	05:00:00	27913	21.7	01:00:00	6.6	05:00:00	22370	24.8	01:00:00	6.5	05:00:00

7   Conclusions and future work

We considered the special case of equidistant row layout problems in which all departments have the same length. We showed that only spaces of unit length need to be used when modeling the problem, and we stated and proved exact expressions for the minimum number of spaces that need to be added so as to preserve at least one optimal solution. These results show that the multi-row equidistant layout can be modeled using only binary variables; this has a significant impact for a computational perspective. Using these results we proposed two new models for the equidistant problem, an ILP model and an SDP model. Our computational results show that the SDP approach dominates for medium- to large-sized instances and that it is well-suited for providing high-quality lower bounds for large-scale instances in reasonable computation time. Specifically for double-row instances, we attain global optimality for some instances with up to 25 departments, and achieve optimality gaps smaller than 1% for instances with up to 50 departments.

On the theoretical side, it remains an open question to extend the theoretical results to general double-row or multi-row problems. From the computational perspective, one direction for future research is the use of the SDP relaxation within a branch-and-bound scheme. While there is at present no commercial or other generally available software that does this automatically for SDP relaxations, this possibility is well worth exploring given the high-quality lower bounds provided by the SDP approach.

Appendix

Details on benchmark instances used

The following tables state the source, the density and the optimal solution or best bounds for our benchmark instances with $m \in [5]$ rows.

Table 8: Characteristics and optimal results for small instances with 5 to 15 departments.

Instance	Source	Size	Density	Optimal solution				
		(n)	$(\%)$	$m=1$	$m=2$	$m=3$	$m=4$	$m=5$
E-5-50		5	50	30	13	6	4	0
E-5-100		5	100	95	46	27	17	0
E-6-50		6	50	100	45	29	22	12
E-6-100		6	100	216	99	56	49	29
E-7-50		7	50	106	51	31	17	9
E-7-100	[36]	7	100	252	126	79	50	40
E-8-50		8	50	136	64	37	26	25
E-8-100		8	100	397	191	125	74	70
E-9-50		9	50	240	118	70	55	40
E-9-100		9	100	618	306	181	140	100
E-10-50		10	50	387	191	114	89	59
E-10-100		10	100	873	427	277	209	133
E-11-100		11	50	1085	539	351	256	191
N-15	[41]	15	71	2186	1064	668	500	382
O-5		5	100	150	70	38	32	0
O-6		6	100	292	136	72	64	28
O-7		7	100	472	236	144	102	76
O-8	[42]	8	100	784	366	250	148	138
O-9		9	100	1032	508	302	238	168
O-10		10	100	1402	670	450	334	222
O-15		15	100	5134	2556	1660	1250	914
S-12		12	100	4431	2167	1404	995	841
S-13	[45]	13	100	5897	2940	1938	1413	1132
S-14		14	100	7316	3608	2408	1794	1369
S-15		15	100	8942	4466	2883	2175	1612
Y-6		6	100	1372	630	350	315	193
Y-7		7	100	1801	899	577	383	311
Y-8		8	100	2302	1095	728	430	394
Y-9		9	100	2808	1401	848	658	476
Y-10	[55]	10	100	3508	1697	1140	845	530
Y-11		11	100	4022	2008	1314	947	724
Y-12		12	100	4793	2342	1510	1070	908
Y-13		13	100	5471	2730	1798	1314	1048
Y-14		14	100	6445	3164	2110	1574	1201
Y-15		15	100	7359	3676	2357	1782	1322

Table 9: Characteristics and optimal results for medium-sized to large instances with 16 to 60 departments.

Instance	Source	Size	Density	Optimal solution				
		(d)	(%)	m = 1	m = 2	m = 3	m = 4	m = 5
N-16a		16	78	3050	1494	1002	706	584
N-16b		16	70	2400	1168	[786,792]	570	462
N-17		17	74	3388	1678	1114	808	662
N-18		18	74	3986	1970	[1287,1292]	972	772
N-20	[41]	20	74	5642	2782	[1847,1856]	[1347,1360]	[1061,1068]
N-21		21	65	5084	2512	[1643,1664]	[1225,1248]	[965,972]
N-22		22	66	6184	[3059,3064]	[2018,2034]	[1497,1510]	1188
N-24		24	67	8270	[4097,4120]	[2696,2712]	[1985,2010]	[1590,1624]
N-25		25	67	9236	4604	[3040,3062]	[2242,2286]	[1773,1796]
N-30		30	67	16494	[8194,8230]	[5406,5442]	[4001,4086]	[3165,3232]
O-20	[42]	20	100	12924	6414	[4265,4284]	[3115,3118]	[2431,2444]
S-16		16	100	11019	5446	3638	2600	2094
S-17		17	100	13172	6577	[4343,4354]	[3196,3225]	[2568,2577]
S-18		18	100	15699	[7787,7788]	[5107,5110]	[3854,3892]	[3048,3083]
S-19		19	100	18700	[9311,9343]	[6149,6190]	[4545,4599]	[3593,3614]
S-20	[45]	20	100	21825	[10837,10841]	[7186,7227]	[5248,5260]	4105
S-21		21	100	24891	[12406,12431]	8144	[6042,6136]	[4762,4849]
S-22		22	100	28607	[14202,14208]	[9412,9484]	[6997,7082]	[5549,5623]
S-23		23	100	33046	[16448,16521]	[10900,10974]	[8086,8159]	[6450,6523]
S-24		24	100	37498	[18646,18658]	[12325,12349]	[9116,9147]	[7261,7342]
S-25		25	100	42349	[21091,21172]	[13951,14070]	[10332,10487]	[8148,8149]
Y-20		20	100	12185	6046	[4018,4033]	[2930,2934]	2282
Y-25		25	100	20357	[10139,10170]	[6709,6761]	[4967,5049]	[3912,3930]
Y-30		30	100	27673	[13771,13790]	[9107,9133]	[6764,6889]	[5355,5386]
Y-35		35	100	38194	[19025,19087]	[12636,12705]	[9357,9492]	[7447,7504]
Y-40	[55]	40	100	47561	[23648,23749]	[15616,15825]	[11615,11785]	[9253,9381]
Y-45		45	99	[62849,62904]	[31237,31442]	[20592,20896]	[15283,15663]	[12182,12442]
Y-50		50	99	[83086,83127]	[41156,41517]	[27129,27674]	[19915,20809]	[16032,16475]
Y-60		60	97	[111884,112126]	[54925,55996]	[35803,37279]	[26180,27913]	[21007,22370]

Table 10: Characteristics and optimal results for the instances from Amaral [3] with $d \in \{9, 10, 11, 12\}$.

Instance	Size	Density	Optimal solution				
	(d)	$(\%)$	$m = 1$	$m = 2$	$m = 3$	$m = 4$	$m = 5$
A-9-10	9	14	5	2	0	0	0
A-9-20	9	31	19	9	6	3	3
A-9-30	9	17	7	3	2	1	1
A-9-40	9	33	23	11	7	5	4
A-9-50	9	47	36	18	11	7	5
A-9-60	9	56	48	23	15	9	7
A-9-70	9	78	76	38	23	18	13
A-9-80	9	92	102	51	30	24	17
A-9-90	9	86	90	45	27	21	15
A-10-10	10	11	6	2	1	0	0
A-10-20	10	16	9	3	3	1	1
A-10-30	10	24	16	7	5	3	1
A-10-40	10	53	62	30	20	13	10
A-10-50	10	53	59	28	19	13	9
A-10-60	10	47	50	25	15	11	8
A-10-70	10	76	101	49	33	24	16
A-10-80	10	91	134	65	44	32	21
A-10-90	10	91	134	65	44	32	21
A-11-10	11	5	3	0	0	0	0
A-11-20	11	29	36	17	11	8	5
A-11-30	11	40	51	25	16	12	8
A-11-40	11	44	62	30	20	14	11
A-11-50	11	62	103	51	34	24	18
A-11-60	11	55	75	37	24	18	14
A-11-70	11	65	108	54	35	26	19
A-11-80	11	82	149	74	49	36	27
A-11-90	11	96	202	101	66	48	37
A-12-10	12	8	5	1	1	0	0
A-12-20	12	20	24	11	7	5	4
A-12-30	12	23	28	13	8	5	4
A-12-40	12	42	76	37	24	16	15
A-12-50	12	48	88	43	27	20	17
A-12-60	12	55	108	53	33	24	21
A-12-70	12	70	158	77	49	34	30
A-12-80	12	85	208	102	65	47	40
A-12-90	12	88	218	108	70	50	42

Table 11: Characteristics and optimal results for the instances from Amaral [3] with $d \in \{13, 14, 20, 25\}$.

Instance	Size	Density	Optimal solution				
	(d)	$(\%)$	$m = 1$	$m = 2$	$m = 3$	$m = 4$	$m = 5$
A-13-10	13	9	7	2	1	1	1
A-13-20	13	28	49	24	15	11	9
A-13-30	13	37	77	38	25	19	14
A-13-40	13	40	85	42	27	20	16
A-13-50	13	53	136	68	44	32	26
A-13-60	13	56	141	70	46	33	26
A-13-70	13	72	211	105	69	50	41
A-13-80	13	87	277	138	90	66	53
A-13-90	13	92	306	153	101	74	58
A-14-10	14	9	10	4	3	1	1
A-14-20	14	25	49	24	16	11	8
A-14-30	14	30	74	36	24	18	13
A-14-40	14	35	87	43	28	21	16
A-14-50	14	56	191	94	63	47	35
A-14-60	14	62	201	99	65	49	37
A-14-70	14	75	279	138	92	68	52
A-14-80	14	86	336	167	111	83	64
A-14-90	14	91	380	187	125	93	71
A-20-10	20	9	25	12	7	4	3
A-20-20	20	22	148	73	[48,49]	34	27
A-20-30	20	30	225	111	[73,74]	[53,54]	42
A-20-40	20	37	300	149	98	[72,73]	[57,58]
A-20-50	20	53	502	249	[165,166]	[121,122]	[95,96]
A-20-60	20	67	693	345	[228,229]	167	[131,132]
A-20-70	20	71	777	385	[257,258]	[186,187]	146
A-20-80	20	77	873	434	[288,290]	[210,211]	165
A-20-90	20	88	1048	[520,521]	347	252	197
A-25-10	25	11	84	41	[26,27]	20	15
A-25-20	25	18	225	110	[71,74]	[52,55]	40
A-25-30	25	30	444	222	146	[108,110]	[86,87]
A-25-40	25	44	802	[399,400]	[263,265]	[194,198]	[152,156]
A-25-50	25	53	1023	[509,511]	[336,340]	[248,254]	[195,196]
A-25-60	25	55	1098	[547,549]	[360,364]	[266,271]	[209,212]
A-25-70	25	64	1322	[658,660]	[435,438]	[321,325]	[252,255]
A-25-80	25	81	1820	[907,910]	[600,604]	[444,450]	[349,350]
A-25-90	25	91	2169	[1080,1084]	[716,721]	[530,537]	417

References

[1] A.R.S. Amaral. A new lower bound for the single row facility layout problem. Discrete Applied Mathematics, 157(1):183–190, 2009.

[2] A.R.S. Amaral. Optimal solutions for the double row layout problem. Optim Lett, 1–7, 2011.

[3] A.R.S. Amaral. On duplex arrangement of vertices. Technical report, Departamento de Informtica, Universidade Federal do Espírito Santo (UFES), Brazil, 2011.

[4] N.A.G. Aneke and A.S. Carrie. A design technique for the layout of multi-product flowlines. Int J Prod Res, 24 (3):471–481, 1986.

[5] 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, 2011.

[6] M.F. Anjos and F. Liers. Global approaches for facility layout and VLSI floorplanning. In M.F. Anjos and J.B. Lasserre, editors, Handbook on Semidefinite, Conic and Polynomial Optimization: Theory, Algorithms, Software and Applications. Springer, New York, 2011.

[7] M. Armbruster, M. Fügenschuh, C. Helmberg, and A. Martin. LP and SDP branch-and-cut algorithms for the minimum graph bisection problem: A computational comparison. Math Program Comput, 4(3):275–306, 2012.

[8] F. Barahona and A. Mahjoub. On the cut polytope. Mathematical Programming, 36:157–173, 1986.

[9] J. Bhasker and S. Sahni. Optimal linear arrangement of circuit components. Proceedings of the 20th Annual Hawaii International Conference on System Sciences, 2:99–111, 1987.

[10] S. Boyd and L. Vandenberghe. Convex Optimization. Cambridge University Press, New York, NY, 2004.

[11] M. Brusco and S. Stahl. Using quadratic assignment methods to generate initial permutations for least-squares unidimensional scaling of symmetric proximity matrices. J. Classification, 17(2):197–223, 2000.

[12] R. Burkard, M. Dell'Amico, and S. Martello. Assignment Problems. Society for Industrial and Applied Mathematics, 2009.

[13] W.-M. Chow. An analysis of automated storage and retrieval systems in manufacturing assembly lines. IIE Transactions, 18(2):204–214, 1986.

[14] J. Chung and J. Tanchoco. The double row layout problem. Int J Prod Res, 48(3):709–727, 2010.

[15] D. Datta, A.R.S. Amaral, and J.R. Figueira. Single row facility layout problem using a permutation-based genetic algorithm. Eur J Oper Res, 213(2):388–394, 2011.

[16] M. Deza and M. Laurent. Geometry of Cuts and Metrics, volume 15 of Algorithms and Combinatorics. Springer Verlag, Berlin, 1997.

[17] J. Dickey and J. Hopkins. Campus building arrangement using topaz. Transportation Research, 6(1):59–68, 1972.

[18] A.N. Elshafei. Hospital layout as a quadratic assignment problem. Operational Research Quarterly, 28(1): 167–179, 1977.

[19] M. Ficko, M. Brezocnik, and J. Balic. Designing the layout of single- and multiple-rows flexible manufacturing system by genetic algorithms. Journal of Materials Processing Technology, 157–158:150–158, 2004.

[20] M.R. Garey and D.S. Johnson. Computers and Intractability: A Guide to the Theory of NP-Completeness. W. H. Freeman & Co., New York, NY, USA, 1979.

[21] 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, 47–63, New York, 1974. doi: http://doi.acm.org/10.1145/800119.803884.

[22] M. Gen, K. Ida, and C. Cheng. Multirow machine layout problem in fuzzy environment using genetic algorithms. Computers & Industrial Engineering, 29(1–4):519–523, 1995.

[23] A. Geoffrion and G. Graves. Scheduling parallel production lines with changeover costs: practical applications of a quadratic assignment/LP approach. Operations Research, 24:595–610, 1976.

[24] I. Gurobi Optimization. Gurobi optimizer reference manual, 2014. URL http://www.gurobi.com.

[25] L.H. Harper. Optimal assignments of numbers to vertices. SIAM Journal on Applied Mathematics, 12:131–135, 1964.

[26] L.H. Harper. Optimal numberings and isoperimetric problems on graphs. Journal of Combinatorial Theory, 1: 385–393, 1966.

[27] M.M.D. Hassan. Machine layout problem in modern manufacturing facilities. Int J Prod Res, 32(11):2559–2584, 1994.

[28] C. Helmberg. A Cutting Plane Algorithm for Large Scale Semidefinite Relaxations, chapter 15, 233–256. 2004. doi: 10.1137/1.9780898718805.ch15.

[29] C. Helmberg. ConicBundle 0.3.11. Fakultät für Mathematik, Technische Universität Chemnitz, 2012. URL http://www.tu-chemnitz.de/~helmberg/ConicBundle.

[30] C. Helmberg and F. Rendl. A spectral bundle method for semidefinite programming. SIAM Journal on Optimization, 10(3):673–696, 1999.

[31] S.S. Heragu and A. Kusiak. Machine Layout Problem in Flexible Manufacturing Systems. Operations Research, 36(2):258–268, 1988.

[32] S.S. Heragu and A. Kusiak. Efficient models for the facility layout problem. Eur J Oper Res, 53(1):1–13, 1991.

[33] P. Hungerländer. Semidefinite Approaches to Ordering Problems. PhD thesis, Alpen-Adria Universität Klagenfurt, 2012.

[34] P. Hungerländer. Single-row equidistant facility layout as a special case of single-row facility layout. Int J Prod Res, 2013.

[35] P. Hungerländer and M. Anjos. A semidefinite optimization approach to space-free multi-row facility layout. Les Cahiers du GERAD G–2012–03, GERAD, Montreal, Canada, 2012.

[36] P. Hungerländer and M.F. Anjos. Semidefinite optimization approaches to multi-row facility layout. Technical report, submitted, 2014.

[37] 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.

[38] R. Kothari and D. Ghosh. Tabu search for the single row facility layout problem using exhaustive 2-opt and insertion neighborhoods. Eur J Oper Res, 224(1):93–100, 2013.

[39] G. Laporte and H. Mercure. Balancing hydraulic turbine runners: A quadratic assignment problem. Eur J Oper Res, 35(3):378–381, 1988.

[40] E.M. Loiola, N.M.M. de Abreu, P.O. Boaventura-Netto, P. Hahn, and T. Querido. A survey for the quadratic assignment problem. Eur J Oper Res, 176(2):657–690, 2007.

[41] C.E. Nugent, T.E. Vollmann, and J. Ruml. An experimental comparison of techniques for the assignment of facilities to locations. Operations Research, 16(1):150–173, 1968.

[42] T. Obata. Quadratic assignment problem: Evaluation of exact and heuristic algorithms. PhD thesis, Rensselaer Polytechnic Institute, Troy, NY, 1979.

[43] M. Pollatschek, N. Gershoni, and Y. Radday. Optimization of the typewriter keyboard by computer simulation. Angewandte Informatik, 10:438–439, 1976.

[44] H. Samarghandi and K. Eshghi. An efficient tabu algorithm for the single row facility layout problem. Eur J Oper Res, 205(1):98–105, 2010.

[45] B.R. Sarker. The amoebic matrix and one-dimensional machine location problems. PhD thesis, Department of Industrial Engineering, Texas A&M University, College Station, TX, 1989.

[46] H. Sherali and W. Adams. A hierarchy of relaxations between the continuous and convex hull representations for zero-one programming problems. SIAM Journal on Discrete Mathematics, 3(3):411–430, 1990.

[47] H.D. Sherali and W.P. Adams. Reformulation-Linearization Technique for Solving Discrete and Continuous Nonconvex Problems. Springer, 1998.

[48] D.M. Simmons. One-Dimensional Space Allocation: An Ordering Algorithm. Operations Research, 17:812–826, 1969.

[49] L. Steinberg. The backboard wiring problem: A placement algorithm. SIAM Review, 3(1):37–50, 1961.

[50] J. Suryanarayanan, B. Golden, and Q. Wang. A new heuristic for the linear placement problem. Computers & Operations Research, 18(3):255–262, 1991.

[51] A.W. Tucker. On directed graphs and integer programs. Technical report, IBM Mathematical Research Project, 1960.

[52] B. Wess and T. Zeitlhofer. On the phase coupling problem between data memory layout generation and address pointer assignment. In H. Schepers, editor, Software and Compilers for Embedded Systems, volume 3199 of Lecture Notes in Computer Science, 152–166. Springer Berlin Heidelberg, 2004.

[53] H. Wolkowicz, R. Saigal, and L. Vandenberghe, editors. Handbook of Semidefinite Programming. Kluwer Academic Publishers, Boston, MA, 2000.

[54] D.H. Younger. Minimum feedback arc sets for a directed graph. IEEE Transactions on Circuit Theory, 10(2):238–245, 1963.

[55] J. Yu and B.R. Sarker. Directional decomposition heuristic for a linear machine-cell location problem. Eur J Oper Res, 149(1):142–184, 2003.

[56] Z. Zhang and C. Murray. A corrected formulation for the double row layout problem. Int J Prod Res, 2011.