European Journal of Operational Research 214 (2011) 199–215

Contents lists available at ScienceDirect

European Journal of Operational Research

journal homepage: www.elsevier.com/locate/ejor

Production, Manufacturing and Logistics

A convex optimisation framework for the unequal-areas facility layout problem

Ibolya Jankovits [b,1], Chaomin Luo [a,2], Miguel F. Anjos [b,*,3], Anthony Vannelli [c,4]

[a] Department of Electrical & Computer Engineering, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1
[b] Department of Management Sciences, University of Waterloo, Waterloo, Ontario, Canada N2L 3G1
[c] School of Engineering, University of Guelph, Guelph, Ontario, Canada N1G 2W1

ARTICLE INFO

Article history:
Received 15 April 2007
Accepted 9 April 2011
Available online 23 April 2011

Keywords:
Facility layout
Semidefinite programming
Convex programming
Global optimisation

ABSTRACT

The unequal-areas facility layout problem is concerned with finding the optimal arrangement of a given number of non-overlapping indivisible departments with unequal area requirements within a facility. We present a convex-optimisation-based framework for efficiently finding competitive solutions for this problem. The framework is based on the combination of two mathematical programming models. The first model is a convex relaxation of the layout problem that establishes the relative position of the departments within the facility, and the second model uses semidefinite optimisation to determine the final layout. Aspect ratio constraints, frequently used in facility layout methods to restrict the occurrence of overly long and narrow departments in the computed layouts, are taken into account by both models. We present computational results showing that the proposed framework consistently produces competitive, and often improved, layouts for well-known large instances when compared with other approaches in the literature.

1. Introduction

The unequal-areas facility layout problem (FLP) is concerned with finding the optimal arrangement of a given number of non-overlapping indivisible departments with unequal area requirements within a facility. The objective of the FLP is to minimize the total expected cost of flows inside the facility, where the cost incurred for each pair of departments is taken as the rectilinear distance between the centroids of the departments times the projected flow between them. The projected flow may reflect transportation costs, the construction of a material-handling system, the costs of laying communication wiring, or even adjacency preferences among departments. The problem contains two sets of constraints: department area requirements and department location requirements (such as departments not overlapping, lying within the facility, and in some cases being fixed to a location, or being forbidden from specific regions). We assume that the facility and the departments are all rectangular. Since the height and width of the departments can vary, finding their optimal rectangular shapes is also part of the problem. The ratios height/width and width/height, called aspect ratios, also pose a challenge since departments with low aspect ratios are most practical in real-world applications, but this makes the problem harder. A solution to the FLP is a block layout that specifies the relative location and the dimensions of each department. Once a block layout has been achieved, a detailed layout can be designed which specifies department locations, aisle structures and input/output point locations [8,24,27,43].

A thorough survey of the facility-layout problem is given in [18], where the papers on facility layout are divided into three broad categories. The first is concerned with algorithms for tackling the FLP as defined above. The second category is concerned with extensions that take into account additional issues that arise in real-world applications, such as designing dynamic layouts by taking time-dependency issues into account, designing layouts under uncertainty conditions, and computing layouts that optimize two or more objectives simultaneously. The third category is concerned with specially structured instances of the problem, such as the layout of machines along a production line. In this paper, we shall focus exclusively on the block layout FLP.

The FLP as described above is a hard optimisation problem. In fact, even the restricted version where the shapes of the

* Corresponding author. Current address: Département de Mathématiques et génie industriel & GERAD, École Polytechnique de Montréal, Montréal, QC, Canada H3T 1J4.

E-mail addresses: ibi.jankovits@gmail.com (I. Jankovits), c2luo@engmail.uwaterloo.ca (C. Luo), anjos@stanfordalumni.org (M.F. Anjos), vannelli@uoguelph.ca (A. Vannelli).

1 Research supported by a student internship from OCE-CITO and Bell Canada Laboratories, and by an Ontario Graduate Scholarship in Science and Technology (OGSST).

2 Research supported by a Postgraduate Award from the Natural Sciences and Engineering Research Council of Canada, and by an Ontario Graduate Scholarship.

3 Research partially supported by grants #312125 and #314668 from the Natural Sciences and Engineering Research Council of Canada.

4 Research partially supported by grant #15296 from the Natural Sciences and Engineering Research Council of Canada.

0377-2217/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2011.04.013

departments are all equal and fixed, and the optimisation is taken over a fixed finite set of possible department locations, is NP-hard. This restricted version is known as the quadratic assignment problem (QAP), see for example [37]. The largest QAP instances of the well-known Nugent set, those with 27, 28, and 30 departments, were solved to proven optimality using vast amounts of computational power and important improvements in mathematical programming algorithms [4].

Two types of approaches for finding provably optimal solutions for the FLP have been proposed in the literature. The first type are graph-theoretic approaches that assume that the desirability of locating each pair of facilities adjacent to each other is known. Initially, the area and shape of the departments are ignored, and each department is simply represented by a node in a graph. Adjacency relationships between departments can now be represented by arcs connecting the corresponding nodes in the graph. The objective is then to construct a graph that maximizes the weight on the adjacencies between nodes. We refer the reader to [17] for more details. The second type are mathematical programming formulations with objective functions based on an appropriately weighted sum of centroid-to-centroid distances between departments. Exact mixed integer programming formulations were proposed in [33,35], and nonlinear programming formulations are presented in some detail in Section 2 below. More recently, FLPs with up to eleven departments were solved to global optimality [39,12,11,32]. Thus, most of the approaches in the literature that tackle realistically sized problems are based on heuristics with no guarantee of optimality. These include genetic algorithms, tabu search, simulated annealing, fuzzy logic, and many others, see, e.g. [25,18,30,40].

The contribution of this paper is a two-stage convex-optimisation-based framework for efficiently finding competitive solutions for this problem. (Two-stage approaches for this problem using techniques different from ours are presented in [34,13].) The framework is based on the combination of two mathematical programming models. The first model is a convex relaxation of the layout problem that establishes the relative position of the departments within the facility, while the second model uses semidefinite optimisation to determine the final layout. Both models account for aspect ratio constraints, which are frequently used in facility layout methods to restrict the occurrence of overly long and narrow departments in the computed layouts. We present computational results showing that the proposed methodology consistently produces competitive, and often improved, layouts for well-known large instances when compared with other approaches in the literature.

This paper is structured as follows. In Section 2, the most recent nonlinear programming methods for the FLP are summarized. In Section 3, the proposed framework is motivated and derived. Computational results demonstrating the strength and potential of this framework are presented in Section 4. Finally, possible directions for future research are discussed in Section 5.

2. Previous nonlinear-programming-based methods

Throughout this paper we label the departments $i = 1,\ldots,N$, where N is the total number of departments. The position of each department i is expressed by the coordinates of its centre and is denoted by (x_i,y_i). It is assumed that the nonnegative costs c_{ij} per unit distance between departments i and j are given and are symmetric, i.e. $c_{ij} = c_{ji}$. We will approximate each department by a circle of radius r_i. The idea of using circular departments, or of approximating departments using circles, has been considered in several contexts (see for example [10,15,48] and the references therein).

We begin by describing the target distance methodology employed in [1,2]. Let each module i be represented by a circle of radius r_i, where r_i is proportional to $\sqrt{a_i}$, the square root of the area of module i. Following [1], we define the target distance for each pair of circles i, j as

$$t_{ij} = \alpha(r_i + r_j)^2,$$

where $\alpha > 0$ is a parameter. To prevent circles from overlapping, the target distance is enforced via the objective function by introducing a penalty term which acts as a repeller:

$$f\left(\frac{D_{ij}}{t_{ij}}\right),$$

where $f(z) = \frac{1}{z} - 1$ for $z > 0$, and $D_{ij} = (x_i - x_j)^2 + (y_i - y_j)^2$. The objective function is thus given by

$$\sum_{1\leqslant i<j\leqslant N} c_{ij}D_{ij} + f\left(\frac{D_{ij}}{t_{ij}}\right).$$

The interpretation here is that the first term is an attractor that makes the two circles move closer together and pulls them towards a layout where $D_{ij} = 0$, while the second term is a repeller that prevents the circles from overlapping. Indeed, if $D_{ij} \geqslant t_{ij}$ then there is no overlap between circles and the repeller term is zero or slightly negative, while the attractor in the objective function applies an attractive force to the two circles. On the other hand, if $D_{ij} < t_{ij}$ then the repeller term is positive, and it approaches positive infinity as D_{ij} tends to zero, preventing the circles from overlapping completely.

In summary, the model aims to ensure that $\frac{D_{ij}}{t_{ij}} = 1$ at optimality, so choosing $\alpha < 1$ sets a target value t_{ij} that allows some overlap of the areas of the respective circles, which means that a relaxed version of the non-overlap requirement of the circles is enforced. In practice, by properly adjusting the value of α, we achieve a reasonable separation between all pairs of circles. The complete attractor-repeller (AR) model as given in [1] is:

$$\min_{(x_i,y_j),w_F,h_F} \quad \sum_{1\leqslant i<j\leqslant N} c_{ij}D_{ij} + f\left(\frac{D_{ij}}{t_{ij}}\right), \tag{1}$$

$$\text{s.t.} \quad x_i + r_i \leqslant \frac{1}{2}w_F \quad \text{and} \quad r_i - x_i \leqslant \frac{1}{2}w_F, \quad \text{for } i = 1,\ldots,N, \tag{2}$$

$$y_i + r_i \leqslant \frac{1}{2}h_F, \quad \text{and} \quad r_i - y_i \leqslant \frac{1}{2}h_F, \quad \text{for } i = 1,\ldots,N, \tag{3}$$

$$w_F^{low} \leqslant w_F \leqslant w_F^{up}, \tag{4}$$

$$h_F^{low} \leqslant h_F \leqslant h_F^{up}, \tag{5}$$

where (x_i,y_i) are the coordinates of the centre of circle i as previously defined; w_F, h_F are the width and height of the facility; and w_F^{low}, w_F^{up}, h_F^{low}, and h_F^{up} are the lower and upper bounds of the width and the height of the facility, respectively. The first two sets of constraints require that all the circles be entirely contained within the facility, and the remaining two pairs of inequalities bound the width and height of the facility. (Note that the geometric centre of the facility outline is at the origin of the $x - y$ plane.)

An important drawback of model (1)–(5) is that the objective function is not convex, and hence the overall model is not convex. By modifying it so as to obtain a convex problem, we expect to obtain a relaxation that captures better global information about the problem. Also, note that there is no force between i and j if $D_{ij}^2 = t_{ij}/c_{ij}$. For these reasons, the analysis in [1,2] motivates the definition of the following generalized target distance T_{ij}:

$$T_{ij} := \sqrt{\frac{t_{ij}}{c_{ij}+\varepsilon}},$$

where $\varepsilon > 0$ is a sufficiently small number such that if $D_{ij} \approx T_{ij}$ then $D_{ij} \approx \sqrt{t_{ij}/c_{ij}}$. This target distance takes both the relative size of the departments and the connection cost between them into account. Furthermore, it is defined even when $c_{ij} = 0$.

Using T_{ij}, a convexified version of model (1)–(5) is:

$$\begin{aligned} \min_{(x_i,y_j),w_F,h_F} \quad & \sum_{1\leqslant i<j\leqslant N} F_{ij}(x_i,x_j,y_i,y_j), \\ \text{s.t.} \quad & (2)-(5), \end{aligned} \tag{6}$$

where

$$F_{ij}(x_i,x_j,y_i,y_j) := \begin{cases} c_{ij}D_{ij} + t_{ij}/D_{ij} - 1 & \text{if } D_{ij} > T_{ij}, \\ 2\sqrt{c_{ij}t_{ij}} - 1 & \text{if } 0 \leqslant D_{ij} \leqslant T_{ij}. \end{cases}$$

It was shown in [1] that model (6) is convex, and that by construction, F_{ij} attains its minimum value whenever the positions of circles i and j satisfy $D_{ij} \leqslant T_{ij}$. This includes the case where $D_{ij} = 0$, i.e. both circles completely overlap. Of course, we do not want such a placement, therefore what we seek are arrangements of the circles where $D_{ij} \approx T_{ij}$, since for such arrangements, the minimum value of F_{ij} is still attained but the resulting overlap is minimized.

In practice, the approach in [2] sacrifices the convexity of the model for the sake of computational practicality, by adding the term $-\ln(D_{ij}/T_{ij})$ to the objective function so that an appropriate algorithm will stop at a solution that is on or near the flat portion of the objective function but is farthest from the origin, i.e., where $D_{ij} \approx T_{ij}$. Hence the more practical model is:

$$\begin{aligned} \min_{(x_i,y_j),w_F,h_F} \quad & \sum_{1\leqslant i<j\leqslant N} F_{ij}(x_i,x_j,y_i,y_j) - K\ln\left(\frac{D_{ij}}{T_{ij}}\right), \\ \text{s.t.} \quad & (2)-(5), \end{aligned} \tag{7}$$

where the constant K is a penalty factor.

A closely related approach, the spring embedding (SE) method, was proposed in [10]. The SE method is based on relating the arrangement of circles within a facility to a dynamically balanced spring system. The strength of the spring between each pair of circles is represented by c_{ij}, and the force created in the spring is represented by the Euclidean distance between the centres of the circles. The SE method is based on model (1)–(5) with a different objective function:

$$\begin{aligned} \min_{(x_i,y_j),w_F,h_F} \quad & \sum\nolimits_{1\leqslant i<j\leqslant N} c_{ij}D_{ij} + \max\{0, K_{ij}(r_i + r_j - \sqrt{D_{ij}})\}, \\ \text{s.t.} \quad & (2)-(5), \end{aligned} \tag{8}$$

where the $K_{ij} > 0$, $1 \leqslant i < j \leqslant N$, are penalty factors. As in previously described methods, the $\sum_{1\leqslant i<j\leqslant N} c_{ij}D_{ij}$ component seeks to make the distances between departments as small as possible by attracting all pairs of circles to each other. On the other hand, the penalty term representing the total energy of the springs is introduced to enforce non-overlapping. The penalty term assumes a non-negative value proportional to the magnitude of the area overlap of the circles and results in a repulsive force. If there is no overlap, the total energy function is not penalized, and only an attractive force remains. In summary, the objective function of model (8) represents the total energy function and minimizes the degree of imbalance inside a facility.

Because model (8) ensures that there is no overlap within the resulting final layout of circles, it requires more computational effort to solve than model (7). Since our first stage is only meant to compute the relative positions of the departments, and small overlap among the circles is acceptable, we base our first stage on model (7).

Finally, we mention how the nonlinear programming approach in [2] computed the final block layouts. One major challenge are the non-overlap constraints that can be expressed as:

$$|x_i - x_j| \geqslant \frac{1}{2}(w_i + w_j) \text{or} |y_i - y_j| \geqslant \frac{1}{2}(h_i + h_j), \quad \text{for all } 1 \leqslant i < j \leqslant N. \tag{9}$$

The final stage of the method in [2] reformulates the disjunctive nonoverlap constraints by introducing two new complementary variables, X_{ij} and Y_{ij}, that satisfy $X_{ij} \geqslant \frac{1}{2}(w_i + w_j) - |x_i - x_j|$, $X_{ij} \geqslant 0$ and $Y_{ij} \geqslant \frac{1}{2}(h_i + h_j) - |y_i - y_j|$, $Y_{ij} \geqslant 0$, the nonoverlap constraints are equivalent to the bilinear constraints $X_{ij}Y_{ij} = 0$. Because of these constraints, this model is a mathematical program with complementarity constraints (MPCC). To enforce the restriction that at any feasible point $X_{ij} = 0$ or $Y_{ij} = 0$ must hold, the complementarity constraints are penalized in the objective to obtain the bilinear penalty layout (BPL) model [2]:

$$\begin{aligned} \min_{(x_i,y_i),h_i,w_i,h_F,w_F} \quad & \sum\nolimits_{1\leqslant i<j\leqslant N} c_{ij}(|x_i - x_j| + |y_i - y_j|) + KX_{ij}Y_{ij}, \\ \text{s.t.} \quad & X_{ij} \geqslant \frac{1}{2}(w_i + w_j) - |x_i - x_j| \quad \text{for all } 1 \leqslant i < j \leqslant N, \\ & Y_{ij} \geqslant \frac{1}{2}(h_i + h_j) - |y_i - y_j| \\ & \quad \text{for all } 1 \leqslant i < j \leqslant N, \\ & X_{ij} \geqslant 0 \quad \text{and } Y_{ij} \geqslant 0 \quad \text{for all } 1 \leqslant i < j \leqslant N, \\ & \frac{1}{2}w_F - \left(x_i + \frac{1}{2}w_i\right) \geqslant 0 \quad \text{and} \\ & \quad \left(x_i - \frac{1}{2}w_i\right) + \frac{1}{2}w_F \geqslant 0 \quad \text{for } i = 1,\dots,N, \\ & \frac{1}{2}h_F - \left(y_i + \frac{1}{2}h_i\right) \geqslant 0 \quad \text{and} \\ & \quad \left(y_i - \frac{1}{2}h_i\right) + \frac{1}{2}h_F \geqslant 0 \\ & \quad \text{for } i = 1,\dots,N, \\ & w_ih_i = a_i, w_i^{\max} \geqslant w_i \geqslant w_i^{\min} \quad \text{and} \\ & \quad h_i^{\max} \geqslant h_i \geqslant h_i^{\min} \\ & \quad \text{for } i = 1,\dots,N, \\ & w_F^{\max} \geqslant w_F \geqslant w_F^{\min} \quad \text{and} \\ & h_F^{\max} \geqslant h_F \geqslant h_F^{\min}, \end{aligned} \tag{10}$$

where K is a penalty factor.

3. Proposed convex optimisation framework

3.1. The first stage model

The first contribution of this paper is an improved first stage model that is based on model (7). The first improvement is that in the new model the objective function does not improve as the circles start overlapping and the distance between the circle centres becomes less than $r_i + r_j$. A second improvement is the inclusion of some information about aspect ratios. Thirdly, a systematic approach to making parameter choices is introduced.

The second contribution is a new semidefinite-optimisation-based second stage used to obtain the final layouts. The net result is a methodology that consistently produces competitive, and often better layouts for large FLPs, when compared with other approaches in the literature.

3.1.1. Improved objective function

The approach for not rewarding the objective function as the circles start overlapping is based on $r_i + r_j$ (the actual sum of radii), $\sqrt{\frac{t_{ij}}{c_{ij}+\epsilon}}$ (the generalized target distance from [2]) and τ_{ij} (a new target distance). In the improved first stage model, t_{ij} is defined slightly differently than in the previously mentioned models. The α is taken out of the definition of t_{ij} (but kept in the objective function, see F_{ij} in (11)), thus $t_{ij} = (r_i + r_j)^2$ is the actual sum of radii squared. Furthermore, we set a new parameter υ_{ij} and the new target distance τ_{ij} as follows:

$$\text{if } t_{ij} > \sqrt{\frac{t_{ij}}{c_{ij}+\epsilon}} \text{ then } v_{ij} = c_{ij}t_{ij} \text{ and } \tau_{ij} = t_{ij}$$

$$\text{else } \upsilon_{ij} = 2\sqrt{t_{ij}c_{ij}+\epsilon} - 1 \text{ and } \tau_{ij} = \sqrt{\frac{t_{ij}}{c_{ij}+\epsilon}}.$$

The idea is that if the generalized target distance is less than the actual sum of radii squared, then the objective function is truncated at a higher level than $2\sqrt{t_{ij}c_{ij}+\epsilon}-1$ (so that overlapping is not rewarded) by setting the new target distance, τ_{ij}, to t_{ij} and setting the cost to $c_{ij}t_{ij}$. On the other hand, if the generalized target distance is greater than or equal to the actual sum of radii squared, then this indicates that t_{ij} is in the flat part of the convexified function and therefore the cost can be set to the value at the flat part of the function, $\upsilon_{ij} = 2\sqrt{t_{ij}c_{ij}+\epsilon}-1$, and the new target distance can be set to the generalized target distance. This concept leads to the first stage model below:

$$\begin{aligned}
&\min_{(x_i,y_j),w_F,h_F} && \sum_{1\leqslant i<j\leqslant N} F_{ij}(x_i,x_j,y_i,y_j) - K\ln\left(\frac{D_{ij}}{t_{ij}}\right)\\
&\text{s.t.} && x_i + r_i \leqslant \frac{1}{2}w_F \quad \text{and } r_i - x_i \leqslant \frac{1}{2}w_F, \quad \text{for } i=1,\dots,N,\\
& && y_i + r_i \leqslant \frac{1}{2}h_F \quad \text{and } r_i - y_i \leqslant \frac{1}{2}h_F, \quad \text{for } i=1,\dots,N,\\
& && w_F^{low} \leqslant w_F \leqslant w_F^{up},\\
& && h_F^{low} \leqslant h_F \leqslant h_F^{up},
\end{aligned} \tag{11}$$

where

$$\tau_{ij} = \begin{cases} t_{ij} & \text{if } t_{ij} \geqslant \sqrt{\frac{t_{ij}}{c_{ij}+\epsilon}},\\ \sqrt{\frac{t_{ij}}{c_{ij}+\epsilon}} & \text{otherwise,}\end{cases}$$

$$\upsilon_{ij} = \begin{cases} c_{ij}t_{ij} & \text{if } t_{ij} \geqslant \sqrt{\frac{t_{ij}}{c_{ij}+\epsilon}},\\ 2\sqrt{t_{ij}c_{ij}+\epsilon}-1 & \text{otherwise,}\end{cases}$$

$$F_{ij}(x_i,x_j,y_i,y_j) = \begin{cases} c_{ij}D_{ij} + \frac{\alpha t_{ij}}{D_{ij}} - 1 & \text{if } \tau_{ij} \leqslant D_{ij},\\ \upsilon_{ij} & \text{if } \tau_{ij} > D_{ij},\end{cases}$$

where D_{ij} and ϵ are as defined previously.

3.1.2. Aspect ratios

One way to obtain very low costs in the FLP is by aligning the departments in a stack of long, narrow departments where the centroids are very close to one another. Therefore, another goal of the FLP is to try to give the rectangular-shaped departments dimensions that are not too far off from being square. The concept of aspect ratio measures how far off a department's shape is from being square. The aspect ratio of department i is defined as $\beta_i := \max\{\frac{h_i}{w_i}, \frac{w_i}{h_i}\}$, where h_i is the height and w_i is the width of department i. As the aspect ratio becomes smaller (approaching 1), the problem becomes more constrained, the total cost increases, and feasible solutions become harder to find. With the exception of (10), none of the above models have any control over the aspect ratios.

Our first stage model tries to include some information about the desired width and height for each department. The aspect ratios can be thought of as a bound on a department's width and height. For example, if department i has area a_i and a restriction that its minimum length should be larger than μ_i, so that $h_i \geqslant \mu_i$, $w_i \geqslant \mu_i$ must hold (as in [48] for example), then (provided $\mu_i > 0$) one can show that

$$\frac{h_i}{w_i} = \frac{h_i^2}{w_i h_i} \geqslant \frac{\mu_i^2}{a_i} \quad \text{and} \quad \frac{w_i}{h_i} = \frac{w_i^2}{h_i w_i} \geqslant \frac{\mu_i^2}{a_i}, \quad \text{as well as} \quad \frac{h_i}{w_i} = \frac{h_i w_i}{w_i^2} \leqslant \frac{a_i}{\mu_i^2} \quad \text{and} \quad \frac{w_i}{h_i} = \frac{w_i h_i}{h_i^2} \leqslant \frac{a_i}{\mu_i^2}.$$

Given that the aspect ratio is $\beta_i = \max\left\{\frac{h_i}{w_i}, \frac{w_i}{h_i}\right\}$, the implied bound on the aspect ratio is $\beta_i \leqslant \frac{a_i}{\mu_i^2}$. Therefore, by altering the minimum side length of departments, one can control the upper bound of the aspect ratio. It is important to note that this upper bound will be different for each department i as the area a_i for each department differs.

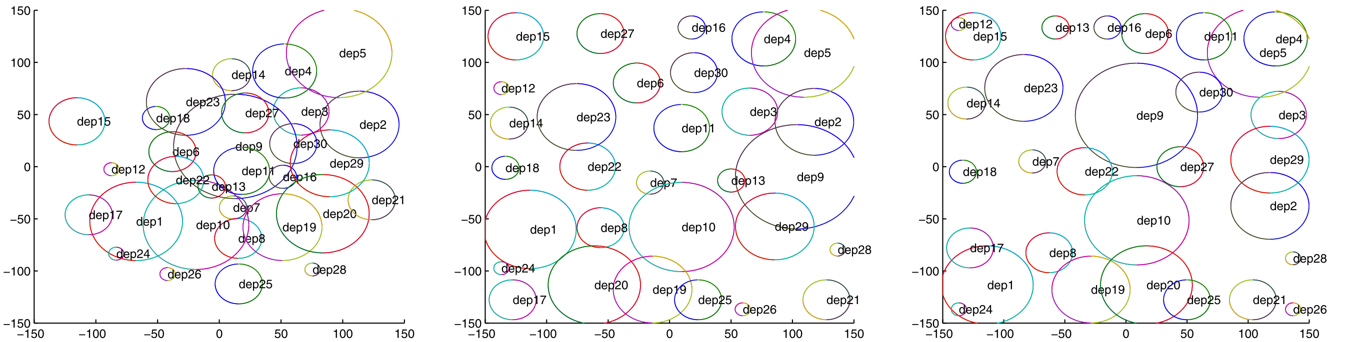
Since the aspect ratio is closely related to the minimum side length of the department, the idea of controlling the aspect ratio via the minimum side length of departments provides a link back to the notion of having the departments represented by circles. In the circle-based models above, the radii of circles that represent departments are considered to be a given parameter, even though one only knows the desired area of the departments. If one simply takes the radius to be $r_i = \sqrt{a_i/\pi}$, the results tend to yield costs that are relatively low but also departments with relatively large aspect ratios, i.e., with a large difference between the length and width. Intuitively, one reason for the large aspect ratios is that it is much harder for the larger departments than for smaller departments to shape themselves into square-like rectangles when the circle sizes only depend on the desired department areas. Also, large departments usually have many more neighbours than smaller departments, and rely more heavily on these neighbours to move/jump around to allow the large department the flexibility it needs to form itself into a square-like shape. Therefore if larger departments have relatively larger circles to represent their areas, the circles may reserve enough area inside the facility for the second stage model to be able to form departments with lower aspect ratios. For this reason in our framework, the radii are set to $r_i = \sqrt{\frac{a_i}{\pi}}\log_2\left(1 + \frac{a_i}{\varphi^2}\right)$ where φ is a parameter for controlling the desired smallest length or width for each department in the layout. By construction, the log scaling factor is greater than or equal to 1, and as φ decreases, the log factor increases and the optimisation model aims for lower aspect ratios, while if φ increases then the log factor decreases and allows the model to aim for higher aspect ratios.

Since the areas of the circles that represent the departments are increased, the floor dimensions also have to be adjusted in order to allow the circles to fit with a nice spread (and reduced overlap). Therefore, the facility dimensions are also adjusted by a factor $\chi = \max_i\left\{\log_2\left(1 + \frac{a_i}{\varphi^2}\right)\right\}$. This way the circles will fit nicely and have the potential of leading to small aspect ratios for the department in the layout obtained using the second stage.

It is shown in Section 4 that, together with the second stage described in Section 3.2, this approach provides a variety of layouts with relatively low aspect ratios and costs. Since real-world problems are customarily simplified before they are modeled using the FLP, even if the model does solve the problem to optimality, the original problem being solved is much too simplified compared to its real-world setting. With our proposed framework, it is possible to get a wide variety of layouts with competitive costs. In this way, a layout that most closely meets all the problems requirements (even the ones that were not accounted for in the FLP model) can be selected from the range of layouts obtained.

3.1.3. Parameter selection

The model (11) has two main parameters that need to be adjusted experimentally: α and K. The parameter α controls the amount of overlap amongst circles, by penalizing overlaps in the objective function. K acts as a dispersion parameter, so if K is too small, all the circles will be placed on top of each other around



(a) K is too small
Max Aspect Ratio*: 10.27
Cost*: 24303

(b) K is good
Max Aspect Ratio*: 3.75
Cost*: 25000

(c) K is too big
Max Aspect Ratio*: 16.77
Cost*: 27257

Fig. 1. Layouts for a 30-department instance from [45] with different values of K in model (11) $*$Calculated after the second stage has computed the layout.

the centre of the layout, as illustrated in Fig. 1(a). If K is too large, then all the circles will be pushed to the edges of the layout as illustrated in Fig. 1(c). Finding a good balance between these two extremes will result in the most promising layouts after the second stage, as in Fig. 1(b).

Since the ranges of α and K within which good layouts can be found vary from one instance to another, one must first experiment with extreme values for each parameter to get a good sense of which values will yield the most promising results. We do this by first fixing α to a value around 3 (since this was found to be always within α's range) and varying K, to find a value of K for which all the circles are approaching the centre of the layout and are practically on top of each other, and another value of K that starts pushing all the circles to the layout edges. Afterwards K is fixed to a value that provided one of the smallest costs in the previous experiments (when α was fixed and K was varied) and α is varied to extreme values that provide solutions where all circles start overlapping or are pushed to the edges of the layout. Once the ranges of the parameters have been determined, one can either solve the first stage model for all possible combinations of α and K on a grid within these extreme valued ranges (to find the best possible layout) or randomly sample within the same ranges. We have used the latter option. Some evidence supporting the use of random sampling is given in [22]. Future research will consider more systematic ways to set these parameters.

3.1.4. First stage model variations

Facility layout problems sometimes require departments not to be narrower than a certain value. For example, a 10-department problem in [48] has the restriction that the departments cannot be narrower than 5 units in either dimension. This restriction can be accounted for in the model (11) by changing the φ parameter. As explained in Section 3.1.1, the log function helps in adjusting the respective circle sizes for each department to allow them to have enough space in order to be able to shape themselves closer to a square shape.

Often facility layout problems do not allow the entire floor plan to be used for placing departments. There may be locations on the floor plan that are already occupied by existing facilities such as elevator shafts, utilities and columns, etc. For example, [44,45] consider a 30-department problem with 3 occupied spaces at the corner, as shown in Fig. 2. With model (11), two different approaches can be used to solve this 30-department problem. The first approach considers the 3 occupied spaces as additional departments, hence solving the problem at the first stage as a 33-department problem. These 3 additional departments have associated flows of zero, and fixed positions at the spots where they are required to be. The second approach initially ignores these 3 occupied spaces and solves the first stage as a 30-department problem, and then adds these 3 extra departments into the problem at the second stage. The computational results in Section 4 suggest that the latter approach yields layouts with lower costs.

3.2. The second stage model

Given the fixed-outline of the facility and the locations of circles (from the first stage), the second stage model uses semidefinite programming to provide the precise location and rectangular dimensions of the departments while minimizing the layout costs.

3.2.1. Semidefinite programming

Semidefinite programming (SDP) refers to the class of optimisation problems where a linear function of a symmetric matrix variable X is optimized subject to linear constraints on the elements of X and the additional constraint that X must be positive semidefinite. The standard SDP problem has the form:

$$\begin{aligned} \min \quad & C \bullet X, \\ \text{s.t.} \quad & A_i \bullet X = b_i, \quad i = 1, 2, \ldots m, \\ & X \succeq 0, \end{aligned}$$

where $A \bullet X = \sum_{i,j} A_{i,j} B_{i,j}$ is the scalar matrix product, and $X \succeq 0$ denotes that X is a symmetric positive semidefinite matrix. This includes linear programming (LP) problems as a special case, namely when all the matrices involved are diagonal.

The fact that SDP problems can be solved in polynomial-time to within a given accuracy follows from the complexity analysis of the ellipsoid algorithm (see [19]). A variety of polynomial-time interior-point algorithms for solving SDPs have been proposed in the literature, and several excellent solvers for SDP are now available. We refer the reader to the semidefinite programming webpage [20] as well as the books [14,49] for a thorough coverage of the theory and algorithms in this area, as well as a discussion of several application areas where semidefinite programming researchers have made significant contributions. In particular, SDP has been successfully applied in the development of approximation algorithms for several classes of hard combinatorial optimisation problems. A mixed integer SDP model was recently proposed in [42] to find global lower bounds for the floorplanning problem in physical circuit design, a problem closely related to the FLP. The survey articles [3,26] provide an excellent overview of the results in this area.

Based on the relative position of departments obtained from the first stage, we formulate the FLP as a convex optimisation problem using SDP. In this section, the second stage model is formulated by applying semidefinite optimisation modelling techniques to the area and aspect ratio constraints.

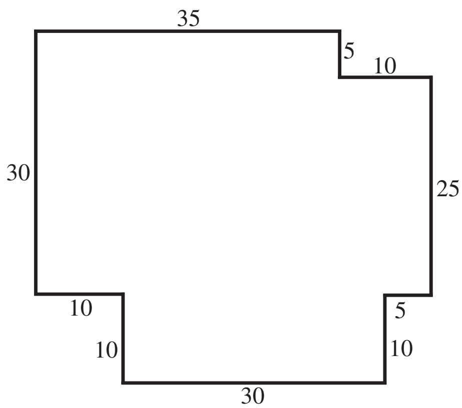


Fig. 2. Floor plan for the 30-department layout.

3.2.2. Formulation of the area and aspect ratio constraints

One property of positive semidefinite matrices is that all their principal minors are non-negative. Therefore for $a_i > 0$, if we relax the area constraint $w_i h_i = a_i$ to $w_i h_i \geqslant a_i$, we can express the relaxed area constraint as a positive semidefiniteness constraint:

$$\begin{pmatrix} w_i & \sqrt{a_i} \\ \sqrt{a_i} & h_i \end{pmatrix} \succeq 0.$$

Assuming that the aspect ratio of department i must be bounded above by a given value $\beta_i^* > 0$, then letting $w_i^{low} = h_i^{low} = \sqrt{a_i/\beta_i^*}$ where $a_i = w_i h_i$, we have:

$$w_i \geqslant w_i^{low} > 0 \Rightarrow w_i^2 \geqslant a_i/\beta_i^* \iff \beta_i^* w_i^2 \geqslant a_i \iff \beta_i^* \geqslant h_i/w_i.$$

Similarly, $h_i \geqslant h_i^{low} > 0$ implies $\beta_i^* h_i^2 \geqslant a_i \iff \beta_i^* \geqslant w_i/h_i$. Thus, $\beta_i^* \geqslant h_i/w_i$ is equivalent to the following positive semidefiniteness constraint:

$$\begin{pmatrix} \beta_i^* & w_i \\ w_i & a_i \end{pmatrix} \succeq 0,$$

and $\beta_i^* \geqslant w_i/h_i$ is equivalent to

$$\begin{pmatrix} \beta_i^* & h_i \\ h_i & a_i \end{pmatrix} \succeq 0.$$

Combining all these constraints yields the following semidefinite programming model:

$$\begin{aligned}
\min_{(x_i,y_j),w_i,h_i,w_F,h_F} \quad & \sum_{1 \leqslant i<j \leqslant N} c_{ij}(u_{ij} + v_{ij}), \\
\text{s.t.} \quad & u_{ij} \geqslant x_i - x_j \quad \text{for all } 1 \leqslant i<j \leqslant N, \\
& u_{ij} \geqslant x_j - x_i \quad \text{for all } 1 \leqslant i<j \leqslant N, \\
& v_{ij} \geqslant y_i - y_j \quad \text{for all } 1 \leqslant i<j \leqslant N, \\
& v_{ij} \geqslant y_j - y_i \quad \text{for all } 1 \leqslant i<j \leqslant N, \\
& x_i + r_i \leqslant \frac{1}{2} w_F \quad \text{and } r_i - x_i \leqslant \frac{1}{2} w_F, \quad \text{for } i = 1, \dots, N, \\
& y_i + r_i \leqslant \frac{1}{2} h_F \quad \text{and } r_i - y_i \leqslant \frac{1}{2} h_F, \quad \text{for } i = 1, \dots, N, \\
& w_i^{low} \leqslant w_i \leqslant w_i^{up}, \quad \text{for } i = 1, \dots, N, \\
& h_i^{low} \leqslant h_i \leqslant h_i^{up}, \quad \text{for } i = 1, \dots, N, \\
& w_F^{low} \leqslant w_F \leqslant w_F^{up}, \quad \text{for } i = 1, \dots, N, \\
& h_F^{low} \leqslant h_F \leqslant h_F^{up}, \quad \text{for } i = 1, \dots, N, \\
& \begin{pmatrix} w_i & \sqrt{a_i} \\ \sqrt{a_i} & h_i \end{pmatrix} \succeq 0, \quad \text{for } i = 1, \dots, N, \\
& \begin{pmatrix} \beta_i^* & h_i \\ h_i & a_i \end{pmatrix} \succeq 0, \quad \text{for } i = 1, \dots, N, \\
& \begin{pmatrix} \beta_i^* & w_i \\ w_i & a_i \end{pmatrix} \succeq 0, \quad \text{for } i = 1, \dots, N,
\end{aligned} \tag{12}$$

This is the basis for our second-stage model, but it is not yet complete because the non-overlap constraints (9) are not accounted for in this model. They are enforced using additional linear inequality constraints as described in the next section.

3.2.3. Determining relative positions and enforcing non-overlap

With the relative positions of the departments determined in the first stage, one can determine for each pair of departments which of the two conditions in (9) applies, and eliminate the absolute values from these constraints. The result is a single linear non-overlap constraint for each pair of departments.

We convert the solution of model (11) into a planar graph using the Delaunay Triangulation (DT) and its dual construction, the Voronoi diagram (VD). The DT for a set of points P in the plane is the triangulation $DT(P)$ such that no point in P is inside the circumcircle of any triangle in $DT(P)$. Delaunay triangulations maximize the minimum angle of all the angles of the triangles in the triangulation, and hence avoid sliver-like triangles.

As illustrated in Fig. 5, we use the positions of the centres of the circles obtained from solving model (11) to generate the VD. Each cell in the VD contains exactly one department centre and every point in a given cell is closer to its generating department centre than to any other. In Fig. 5(c), the dashed circles represent the solution of the first stage, the fine lines represent the corresponding DT and the black bold lines represent the cells of the corresponding VD. The edges of the DT represent the relative positions of the 9 departments.

These relative positions are then encoded in a Relative Position Matrix (RPM). A RPM is an $N \times N$ non-negative symmetric matrix with zeros on the diagonal. (Since the matrix is symmetric, the information in the upper triangular matrix is sufficient.) The entries of the RPM matrix have the following meaning:

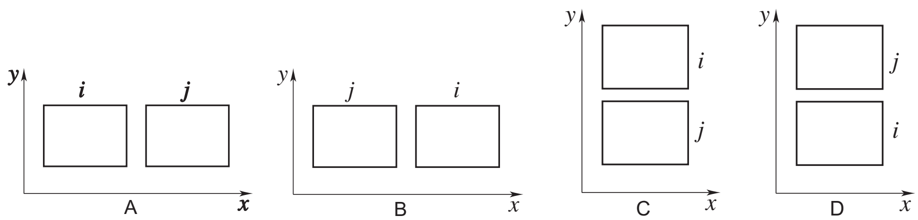
- "1-" is used to represent that department i is horizontally separated from department j. Furthermore, "11" means that department i is to the left of department j (Fig. 3(a)), and "12" means that department i is to the right of department j (Fig. 3(b)).
- "2-" is used to represent that department i is vertically separated from department j. Furthermore, "21" means that department i is above department j (Fig. 3(c)), and "22" means that department i is below department j (Fig. 3(d)).

If two departments are separated in both directions (Fig. 4), the following rule is applied to determine in which direction the separation is enforced. If $\Delta y \geqslant \Delta x$, then only the vertical separation is enforced, and if $\Delta x > \Delta y$, then only the horizontal separation is enforced.

Having determined the RPM, we can set one inequality constraint to ensure non-overlap for each pair (i,j) of departments. For instance, if the entry (i,j) of the RPM equals "11", then for this pair we add the constraint $x_j - x_i \geqslant \frac{1}{2}(w_i + w_j)$ to (12); and if the entry equals "21", then the constraint added to (12) is $y_i - y_j \geqslant \frac{1}{2}(h_i + h_j)$. (Similar reasonings apply for the other two cases).

4. Computational results

The proposed framework was tested using the non-linear optimisation solver KNITRO 5.0 accessed via the modeling language AMPL for the first stage model (11), and the SDP solver SeDuMi [41] for the second stage model (12). All the tests were performed on an IBM eserver XSeries 460 running Microsoft Windows Server 2003 and equipped with 8 Intel Xeon 2.84 GHz CPUs and 32 Gb of RAM. (Each test ran on a single CPU using 3 Gb of RAM.) The radii



(a) “11” (b) “12” (c) “21” (d) “22”

Fig. 3. Horizontal and vertical relative positioning amongst modules.

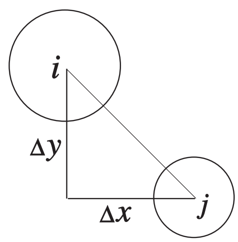


Fig. 4. Diagonal relative positioning amongst modules.

of the approximating circles was set to $r_i = \sqrt{\frac{a_i}{\pi}}\log_2\left(1+\frac{a_i}{\varphi^2}\right)$ where $\varphi = 2$ or the desired minimum length for the instance (if any is specified) and the generalized target distance was set with $\epsilon = 0.1$. To solve model (11), KNITRO requires initial starting points for the centres of the circles. Since it is not clear a priori what the best starting configuration is, following [2], the centres of the N circles were initially placed at regular intervals around a large circle of radius $r = \chi(w_F + h_F)$. The value χ is the adjusting ratio that corresponds to the ratios that the circle sizes were enlarged by, and hence $\chi = \max_i\left(\log_2\left(1+\frac{a_i}{\varphi^2}\right)\right)$. Therefore, the initial centres (x_i, y_i) of the departmental circles can be set to $x_i = r\cos\theta_i$ and $y_i = r\sin\theta_i$, where $\theta_i = 2\pi(i-1)$.

4.1. Results on Benchmarks from the literature with up to 14 departments

First, we report results on some of the classical benchmarks, and some recently proposed benchmarks from [32], with up to 14 departments. For each instance of size up to 11, we compare our best layout with the known optimal layout [39,32]. For larger instances, we compare with the best known layout reported in [29]. These results provide some sense of how the computational time for our method increases with the number of departments, and also of the quality of the layouts obtained.

With respect to the quality of the results, based on the five instances for which the global optimal solution is known, we see that the distance from optimality of our best layouts varies significantly. While for FO11, it is close to global optimality, for other instances it is rather far from optimality.

The more important point that these results illustrate is the computational efficiency of our framework: all these instances with between 9 and 14 departments are solved in less than 2 second, while the recent approach in [29] requires several minutes. The efficiency of our framework is especially important when applying it to larger instances with 20 departments or more, which is the main objective of this research.

4.2. Armour and buffa 20-department benchmark

Arguably the best known large benchmark instance in FLP research is the Armour and Buffa 20-department problem [5]. This instance uses a symmetrical flow matrix and rectilinear distances. As proposed originally in [5], it does not have any requirements on the minimum side length or the maximum allowable aspect ratio. In the original paper [5], the problem was approached by requiring all departments to be made up of contiguous rectangular building blocks, and then departmental adjacent pairwise exchanges were performed. The paper [47] approached this problem by assuming rectangular departmental shapes placed in bays. A genetic optimisation algorithm with adaptive penalty functions to improve the solution was applied in [46], and a simulated annealing approach with probabilistically based aspect ratios was used in [23]. A genetic algorithm that employed concepts of evolutionary hybrid algorithms to obtain a better local optima was used in [16]. Most recently, the models (7) and (10) were used together in [2] to obtain the lowest-cost layouts for this instance in the literature to date. (see Table 1)

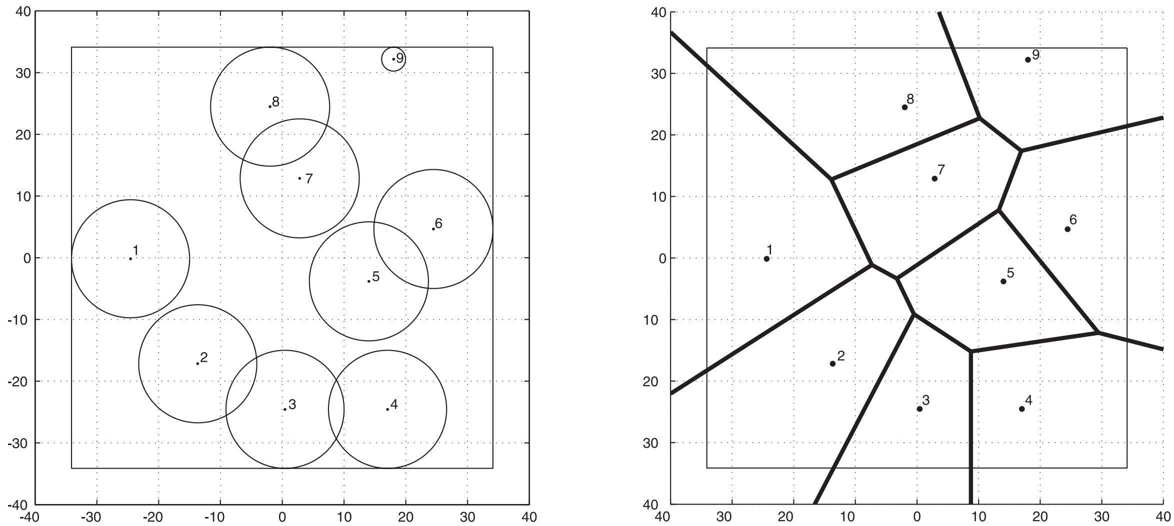
4.2.1. Computational results

Table 2 presents the results obtained by applying our framework to the Armour and Buffa 20-department problem. (The corrected cost matrix from [38,21] was used.)

As explained in Section 3.1.3, the α and K parameters were varied randomly and also systematically to obtain different layouts. Our framework found the layout in Fig. 6, with a cost of 2708. The layout in Fig. 6 shows department 7 as having the largest aspect ratio (equal to 6). The parameter combination $\alpha = 1.04$ and $K = 1690$ was used to solve the first stage model in 0.769 seconds and the second stage model in 16.6 seconds. In comparison, the genetic algorithm in [46] with a (lower) bound of 7 on the aspect ratio found a best layout with a cost of 5255.0 (over 10 runs of the algorithm) and the best layout reported in [2] was 4591.3 (with an aspect ratio of 5 however). Thus the new layout obtained using our approach found improved layouts with aspect ratios as low as 4 and improved by 41% on the previous best cost found for an aspect ratio bound of 7.

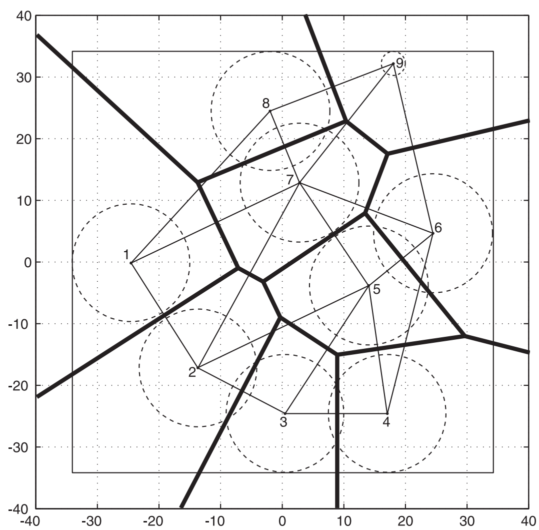
4.3. Nugent 30-department benchmark

Another large problem that is well known in the literature is the Nugent 30-department QAP problem [36]. In [44,45] this problem was transformed into a FLP with three additional pre-positioned departments, as illustrated in Fig. 2. The three pre-positioned departments represent areas that are occupied by existing facilities such as elevator shafts or columns. A genetic algorithm was used in [44], while in [45] the author used a simulated annealing algorithm with a slicing structure/tree method containing the information on partitioning the floor. Both [44,45] aim for relatively low aspect ratios (all below 2.5) and minimal deadspace. Deadspace refers to overlap, both between two departments and between a department and the occupied areas. Their algorithm uses a penalizing component in the objective function to compensate for larger aspect ratios and existing deadspace (the latter gauges the degree of shape distortion of departments due to overlap of occupied



(a) Solution of the first stage model

(b) Corresponding Voronoi diagram



(c) Superposition of Delaunay triangulation

$$\begin{pmatrix} 0 & 21 & 11 & 11 & 11 & 11 & 11 & 22 & 11 \\ 0 & 0 & 11 & 11 & 11 & 11 & 22 & 22 & 22 \\ 0 & 0 & 0 & 11 & 22 & 22 & 22 & 22 & 22 \\ 0 & 0 & 0 & 0 & 22 & 22 & 22 & 22 & 22 \\ 0 & 0 & 0 & 0 & 0 & 11 & 22 & 22 & 22 \\ 0 & 0 & 0 & 0 & 0 & 0 & 12 & 12 & 22 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 22 & 22 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 11 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

(d) Resulting relative position matrix

Fig. 5. Construction of the RPM for a 9-department example.

areas). Hence, the layouts in [44,45] provide some awkwardly shaped facilities that are not always practical in the real world (see departments 15 and 19 in Fig. 7(a) and department 29 in Fig. 7(b)).

Other papers have attempted to solve this hard problem, but due to its size, the aspect ratio requirements, and the three pre-positioned departments, each one of these approaches made its unique simplifications. The unfortunate result is that it is not possible to compare our costs directly with theirs. In [23], this 30-department problem was tackled using a simulated annealing algorithm in which a solution is encoded as a matrix that has information about relative locations of the facilities on the floor. To compare their results with [44,45], they converted Tam's costs to minimum total transportation distances by subtracting the penalties from the objective function to obtain an estimate that their model beat. The more recent paper [28] developed a shape-based block layout approach that uses a hybrid genetic algorithm (which adds the strength of simulated annealing and tabu search algorithms to the genetic algorithm). Even though they claim to be basing their test data on [36,44,45], the areas and aspect ratios in [28] are different from those of the original problem, and the pre-positioned departments are ignored. Most recently, the FACOPT software package [6] uses simulated annealing and a genetic algorithm to solve the 30-department problem. However, FACOPT only beats [44,45] results with its genetic algorithm-based model and there is no indication in [6] about whether the pre-positioned departments were considered and what aspect ratios were obtained in these layouts.

4.3.1. Computational results

Table 3 illustrates the lowest costs that were obtained by [44,45,28,6] for the 30-department problem with occupied spaces. Also, all of these models contain a penalty component in their objective function, hence these models minimize a penalty-added objective function value for feasible solutions (including penalties incurred due to unsatisfied shape constraints). The papers [44,45] have aspect ratio requirements that vary from 1 to 2.5, [6] uses an upper bound of 3 as the aspect ratio requirement, and [28] does

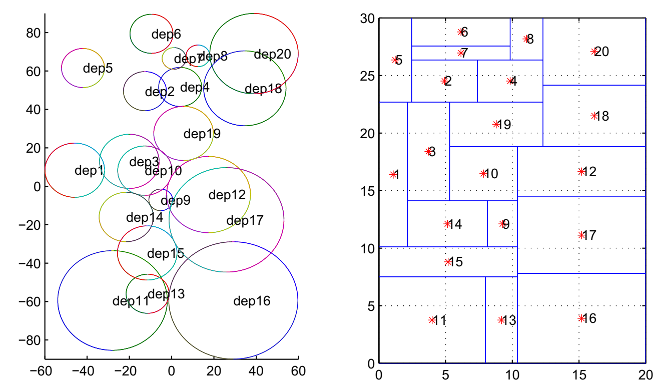
Table 1
Performance of our framework on some literature benchmarks.

Instance	Cost of best layout by our framework	CPU time required by our framework	Cost of optimal layout (*or best known solution*)	Gap (%)
9-department in [9]	298.3	0.61s	235.95	26.4
10-department in [48]	29,193.3	1.00s	20,396.19	43.1
O10 in [32]	320.07	0.95s	238.27	34.3
FO10 in [32]	35.71	0.67s	29.41	21.4
FO11 in [32]	35.33	0.73s	33.93	4.1
12-department in [9]	185.65	0.42s	*142*	30.7
12-department in [7]	12,222.3	0.89s	*8702*	40.5
14-department in [7]	7416.0	1.36s	*5004*	48.2

Table 2
Comparison of the algorithms for the Armour and Buffa problem.

β_i^*	Cost of best layout in [46]	Cost of best layout in [16]	Cost of best layout in [2]	Cost of best layout by our framework
8	5255.0[a]	4793.5[a]	4591.3[a]	3014.2
7	5255.0	4793.5	4591.3[a]	2979.3
6	5524.7[a]	5397.6[a]	4591.3[a]	**2708.0**
5	5524.7	5397.6	4591.3	3009
4	5743.1	5370.9	4786.4	2960.5

[a] Cost of layout for that specific aspect ratio is the best for a lower aspect ratio.



(a) Layout of department circles from first stage model

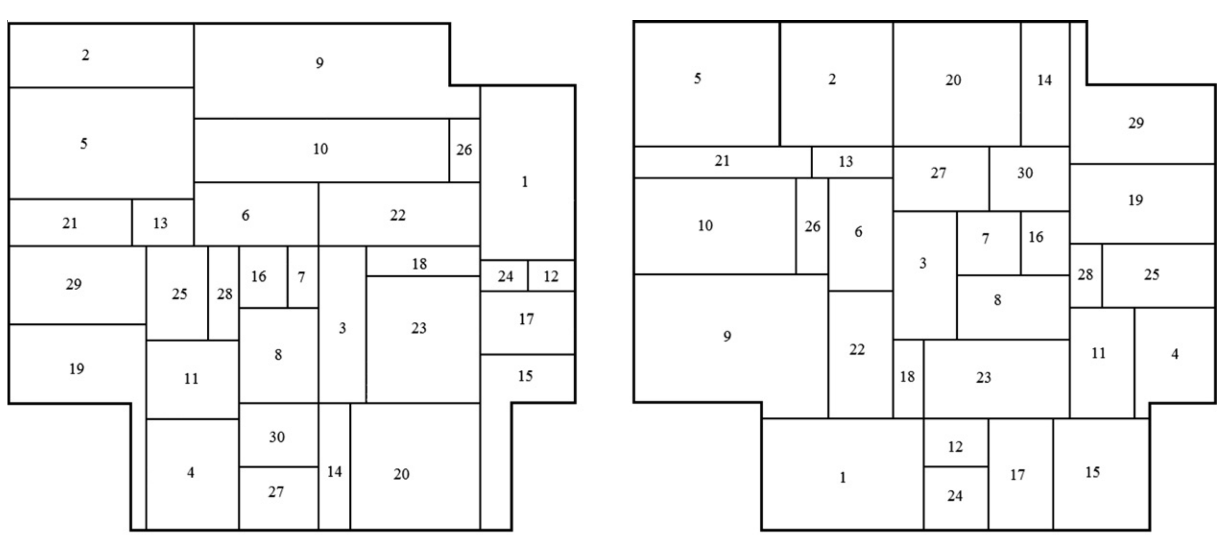
(b) Final layout

Fig. 6. Best layout achieved for the 20-department problem.

not mention what aspect ratio bound was used. The authors of [23] were the first to use a model that did not contain a penalty component in its objective function.

However, the authors of [18] did not consider the occupied areas and set up their objective function to find the best feasible solution with a maximum aspect ratio of 2. We compare with the unpenalized layout from [45] by subtracting the penalites from the objective function value as was done in [31]. Table 4 compares the unpenalized results in [45], the results in [23], and our framework's cost for the problem without the pre-occupied areas. As shown, our model provides very consistent costs as the aspect ratios increase or decrease. For this specific problem, our model was not able to achieve better results than [23], however we obtained costs that were only 8.6% higher. Fig. 8 illustrates the facility layout obtained by our framework has a cost of 23420 and a maximum aspect ratio of 7.67 (which took 2.91 seconds for the first stage and 255.2 seconds for the second stage to solve).

Table 5 on the other hand shows results that were obtained with our approach for the 30-department problem with preoccupied areas. Fig. 9 illustrates the solution obtained for the 30-department problem that uses 3 extra circles to estimate the preoccupied areas. This layout has a cost of 23994 and a largest aspect ratio of 7.41 (which took 4.1 seconds for the first stage and 236 seconds for the second stage to solve). Once again, it can be noted that the layout costs are very consistent for varying aspect ratios, which is an important feature of our framework.



(a) Minimum cost layout in [44]

(b) Minimum cost layout in [45]

Fig. 7. Tam's minimum cost layouts for 30-department problem (Figures taken from [44,45]).

Table 3
Comparison of the best results in the literature for Tam's 30-department problem with penalty costs.

	Cost of best layout in [44]	Cost of best layout in [45]	Cost of best layout in [28][a]	Cost of best layout in [6]
Layout Cost	47422.3	47483.7	51373	42638.6

[a] Layout dimension is not considered and aspect ratios are not mentioned in publication

Table 4
Comparison of the algorithms for Tam's 30-department problem without occupied areas.

β_i^*	Cost of best layout in [45][a]	Cost of best layout in [23][b]	Cost of best layout found by our framework
10	23416.5	21560.6	24098
9	23416.5	21560.6	23924
8	23416.5	21560.6	**23420**
7	23416.5	21560.6	23974
6	23416.5	21560.6	23770
5	23416.5	21560.6	24916
4	23416.5	21560.6	25000

[a] Minimum cost reported in [31] was estimated by subtracting the penalties from the objective value given in [45].
[b] Facilities have an upper limit of 2.0 for their aspect ratios.

4.4. New 30-department instances

Finally, to further demonstrate the potential of our methodology, we present results on five new instances of the FLP. These instances have 30 departments and a randomly generated symmetrical flow matrix. The objective function uses the rectilinear distances. For each instance, we consider maximum allowable aspect ratios of β^* = 4, 5, 6, 7, 8, 9, 10. The specific data for each instance is included as an Appendix A.

4.4.1. Computational results

Table 6 presents the results obtained by applying our framework to the new instances.

As before, the layout costs are consistent for each instance over all the values of the aspect ratios. The running times also remain consistent, varying between 2 to 3 seconds for the first stage, and 3 to 5 minutes for the second stage. Thus, these examples again demonstrate the main features of our framework, namely the fast solution times and the consistency of the solution quality.

Table 5
Results of our framework for varying aspect ratio constraints on the 30-department problem with occupied areas and no penalty costs.

β_i^*	Cost of best layout found by our framework
10	24017
9	24477
8	23994
7	24200

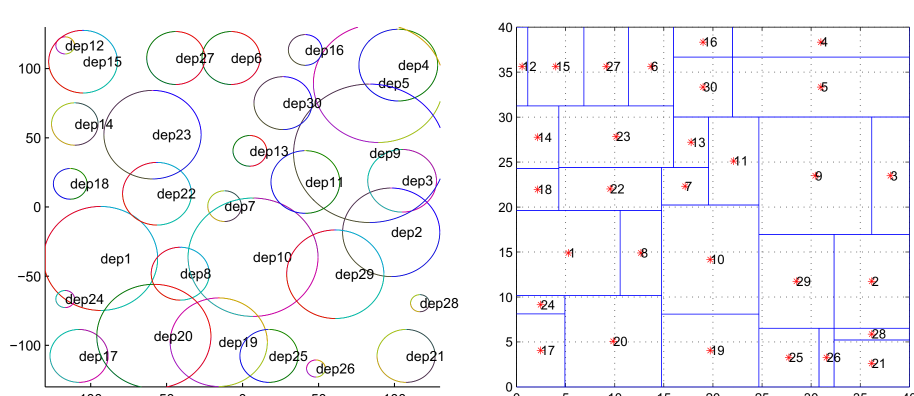
5. Conclusions and future research

This paper proposes a two-stage convex optimisation framework for efficiently finding competitive solutions for this problem. The first stage is a convex relaxation of the layout problem that establishes the relative position of the departments within the facility, while the second stage uses semidefinite optimisation to determine the final layout. Aspect ratio constraints are taken into account at both stages. Our computational results show that while it is not suitable for small instances, the proposed methodology consistently produces competitive, and often improved, layouts for well-known large instances when compared with other approaches in the literature.

Future research will investigate how to better control the aspect ratios within this framework. For instance, using ellipsoids instead of circles to approximate the initial positions of departments could provide better results (this has not been attempted in the literature to date). Ellipsoids would likely provide more realistic estimations of department positions, since departments in real-world applications are not square-shaped.

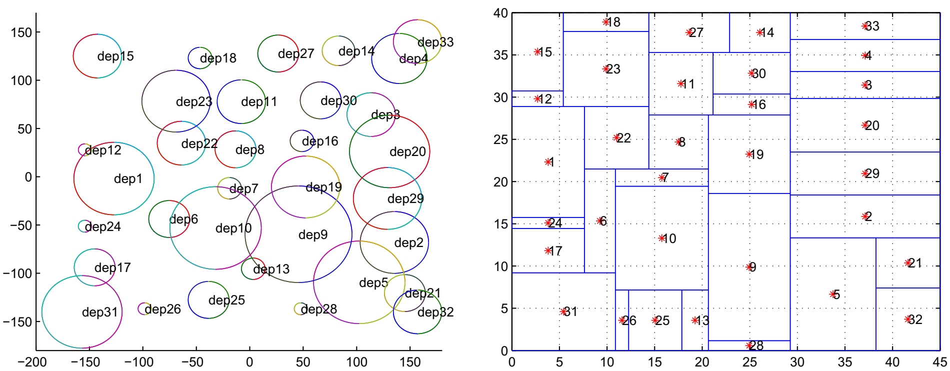
Further work also includes adjusting the φ (the parameter that can control what the desired smallest length or width should be in each department's layout) and potentially using a different value φ_i for each department.

Finally, different combinations of first stage and second stage models from past papers should be tested, to see which combination provide the best overall layouts.



(a) Solution of the first-stage model (b) Final layout from the second-stage model

Fig. 8. Best floor plan for 30-department problem (with no occupied area).



(a) Solution of the first-stage model (b) Final layout from the second-stage model

Fig. 9. Best floor plan for 30-department problem (with circles 31–33 representing occupied areas).

Table 6
Best layout obtained using our framework for each of the new instances.

β_i^*	Cost of best layout found for instance A	Cost of best layout found for instance B	Cost of best layout found for instance C	Cost of best layout found for instance D	Cost of best layout found for instance E
10	9445	10511	16079	11358	12444
9	9591	10532	16142	11228	13118
8	9312	10506	15904	10937	13160
7	9320	10414	15961	10903	12871
6	9504	10604	15748	11054	12642
5	9544	10424	15759	11014	12815
4	9509	10199	15438	11055	12465

Appendix A. New instance A

Facility dimensions: $h_F = 14$ and $w_F = 13$
Department areas:

Dept	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Area	2	4	9	9	4	3	2	6	7	10	2	7	5	10	6

Dept	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Area	6	9	5	3	8	7	7	9	5	10	2	9	5	2	9

Pairwise unit distance costs:

c_{ij}	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	6.0	3.5	1.0	4.0	0.0	4.0	3.0	7.0	0.0	2.5	1.0	4.0	3.5	0.0	5.0
2	—	4.5	0.0	1.0	0.0	9.0	0.0	1.5	9.0	4.0	5.5	0.0	1.0	0.0	5.0
3	–	–	2.0	0.5	2.0	1.5	0.0	2.0	0.0	1.5	0.0	1.5	3.5	7.5	3.5
4	–	–	–	5.5	9.0	2.5	9.5	1.0	3.0	3.5	1.5	0.0	0.0	0.0	4.0
5	–	–	–	–	1.5	0.0	0.5	4.0	5.0	0.0	4.5	1.5	0.0	3.5	3.0
6	–	–	–	–	–	7.5	0.0	1.0	7.5	7.0	4.0	2.5	3.5	5.0	6.0
7	–	–	–	–	–	–	3.0	0.0	5.0	4.0	5.5	5.0	6.5	1.5	0.0
8	–	–	–	–	–	–	–	2.0	3.0	3.5	4.0	2.5	0.0	4.0	6.0
9	–	–	–	–	–	–	–	–	2.0	1.0	5.0	5.0	1.5	2.0	2.5
10	–	–	–	–	–	–	–	–	–	1.0	4.5	0.5	2.0	4.0	0.0
11	–	–	–	–	–	–	–	–	–	–	0.0	5.0	3.0	0.5	1.0
12	–	–	–	–	–	–	–	–	–	–	–	0.0	7.0	0.0	0.5
13	–	–	–	–	–	–	–	–	–	–	–	–	5.0	4.5	3.5
14	–	–	–	–	–	–	–	–	–	–	–	–	–	6.0	0.0
15	–	–	–	–	–	–	–	–	–	–	–	–	–	–	0.0

Appendix A (*continued*)

c_{ij}	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	1.0	3.5	0.0	3.5	0.0	0.0	0.0	2.0	5.5	4.5	3.5	4.5	6.0	4.5
2	3.5	6.5	6.5	0.0	4.5	1.0	0.0	8.5	0.0	1.5	2.0	3.5	5.0	4.0
3	0.0	4.0	2.0	3.0	1.0	0.0	5.0	0.0	3.5	1.0	4.5	1.5	5.0	3.0
4	1.0	0.5	1.5	3.5	1.0	7.0	4.5	0.0	5.5	5.5	6.5	4.5	4.0	5.0
5	5.5	5.0	0.0	0.0	1.0	6.0	6.0	3.5	1.0	5.0	0.0	2.5	0.0	0.0
6	0.0	10.0	0.0	1.0	4.5	0.0	0.0	0.0	2.0	4.0	0.5	0.0	3.0	0.0
7	5.0	5.0	5.5	3.5	5.0	0.0	1.0	5.5	0.0	2.5	2.0	0.0	0.0	4.0
8	3.5	2.5	0.5	0.0	0.0	3.5	2.5	5.0	1.0	4.0	1.0	1.5	8.0	6.5
9	0.0	0.0	3.0	7.0	0.5	0.0	3.5	3.5	0.0	5.0	3.0	0.0	7.0	6.0
10	4.0	0.0	4.5	2.0	0.5	0.0	2.5	0.0	2.0	5.0	4.0	5.0	1.5	0.0
11	2.0	0.0	2.5	3.5	1.0	10.0	6.0	1.5	0.0	3.0	5.0	3.0	4.0	0.5
12	0.0	0.0	1.0	5.0	0.0	0.5	6.5	2.0	5.0	6.0	1.5	0.5	4.0	2.0
13	1.5	0.0	2.0	0.5	1.5	4.5	1.0	2.5	4.0	0.0	3.5	0.0	3.0	0.5
14	2.5	5.5	8.0	3.0	2.5	0.5	0.0	1.0	3.0	2.5	5.0	3.5	3.5	4.5
15	0.0	0.5	3.5	4.5	6.0	1.5	0.0	0.0	5.0	0.0	6.0	4.5	3.5	3.0
16	2.5	5.0	5.5	2.0	4.5	5.5	0.0	0.0	7.0	4.5	2.5	2.0	4.0	4.5
17	–	0.0	5.0	0.0	2.0	3.0	3.5	5.0	5.0	1.0	4.5	1.0	4.5	0.0
18	–	–	3.5	4.5	2.5	1.5	0.0	5.5	3.0	0.0	2.0	2.0	4.5	6.0
19	–	–	–	1.5	0.0	9.0	8.0	1.5	3.5	1.5	6.0	2.5	0.0	3.5
20	–	–	–	–	5.5	0.0	3.0	0.0	0.5	5.0	1.0	1.5	0.0	0.0
21	–	–	–	–	–	7.0	0.0	0.0	3.5	0.5	2.5	0.0	4.0	0.0
22	–	–	–	–	–	–	2.0	4.5	4.0	2.0	0.5	5.0	4.0	0.0
23	–	–	–	–	–	–	–	6.0	2.5	3.5	3.5	8.5	2.5	5.0
24	–	–	–	–	–	–	–	–	0.5	0.0	4.0	3.5	0.0	0.0
25	–	–	–	–	–	–	–	–	–	5.5	0.0	2.5	4.0	0.0
26	–	–	–	–	–	–	–	–	–	–	0.0	8.5	4.0	5.0
27	–	–	–	–	–	–	–	–	–	–	–	1.0	0.0	2.5
28	–	–	–	–	–	–	–	–	–	–	–	–	1.0	0.0
29	–	–	–	–	–	–	–	–	–	–	–	–	–	0.0

Appendix B. New instance B

Facility dimensions: $h_F = 10$ and $w_F = 20$
Department areas:

Dept	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Area	7	6	9	7	3	4	10	4	9	9	4	4	9	7	4

Dept	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Area	4	9	8	9	8	7	3	8	9	3	9	10	3	9	5

Pairwise unit distance costs:

c_{ij}	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1.0	0.0	0.5	0.5	0.0	5.0	0.5	5.5	4.5	4.5	1.0	4.0	0.0	0.0	0.0
2	–	0.0	3.0	0.0	2.0	0.0	8.0	5.0	2.0	8.0	4.0	2.5	4.5	0.0	5.5
3	–	–	0.5	4.0	0.0	5.5	8.5	4.0	5.0	1.5	2.5	1.0	8.5	0.5	5.5
4	–	–	–	0.0	2.5	1.5	2.5	1.0	1.0	4.5	3.5	0.0	0.0	0.0	1.0
5	–	–	–	–	3.5	4.5	4.5	2.5	0.0	5.0	6.5	0.0	6.0	9.0	0.0
6	–	–	–	–	–	1.5	4.0	0.5	8.0	0.0	5.0	0.0	0.0	5.0	1.5
7	–	–	–	–	–	–	5.0	0.0	3.0	2.5	0.0	0.0	5.0	1.0	4.5
8	–	–	–	–	–	–	–	2.0	1.0	0.5	0.0	5.5	4.0	0.0	5.0
9	–	–	–	–	–	–	–	–	4.5	0.0	0.0	4.0	7.5	5.0	0.0
10	–	–	–	–	–	–	–	–	–	6.5	0.0	2.5	1.0	3.5	3.0
11	–	–	–	–	–	–	–	–	–	–	0.0	4.0	0.0	6.0	0.0

Appendix B (*continued*)

c_{ij}	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
12	–	–	–	–	–	–	–	–	–	–	–	5.0	5.5	0.0	5.0
13	–	–	–	–	–	–	–	–	–	–	–	–	3.5	2.5	4.5
14	–	–	–	–	–	–	–	–	–	–	–	–	–	3.5	0.0
15	–	–	–	–	–	–	–	–	–	–	–	–	–	–	1.5

c_{ij}	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	6.0	0.0	0.0	4.0	1.0	1.5	4.5	6.5	0.0	0.0	0.0	0.0	5.0	4.5
2	4.5	0.0	7.0	5.0	0.0	6.0	2.0	4.5	5.5	0.0	3.0	0.0	0.0	0.0
3	0.5	3.0	4.5	0.0	7.0	5.0	0.0	0.5	1.5	0.0	0.0	7.5	4.0	0.0
4	1.5	1.0	0.0	0.0	4.5	3.5	0.0	4.0	5.0	0.0	0.0	6.5	0.5	2.0
5	4.0	1.0	4.0	1.5	6.0	2.5	5.0	0.0	5.0	0.0	4.0	4.0	4.5	5.0
6	2.5	0.0	3.5	5.0	2.0	0.0	0.5	0.0	4.0	2.5	5.0	0.0	5.0	1.0
7	8.0	5.0	3.5	2.0	6.0	3.0	0.5	1.0	3.5	0.0	4.5	0.0	1.0	1.5
8	2.0	0.0	5.5	2.5	2.5	6.0	0.0	4.0	3.5	2.5	2.5	6.0	5.0	1.0
9	2.0	3.0	0.0	0.0	5.0	7.5	0.0	3.0	1.5	5.0	1.0	0.0	2.5	1.0
10	1.0	0.0	4.0	0.5	2.0	0.0	0.5	0.0	5.0	0.0	0.0	1.5	0.0	8.5
11	4.0	0.0	10.0	2.5	4.0	4.5	2.0	1.0	2.0	5.5	0.0	3.0	1.5	3.0
12	0.0	2.5	1.0	0.0	4.0	3.0	3.5	0.0	2.0	0.0	5.0	2.0	4.5	2.0
13	3.0	5.0	2.5	0.0	5.0	0.0	1.5	0.0	4.0	4.5	5.0	0.0	7.5	2.5
14	0.0	2.0	8.0	1.5	4.5	1.5	0.5	3.0	5.0	0.0	4.0	4.0	4.5	5.5
15	0.5	2.5	8.5	0.0	0.0	0.0	2.5	1.0	5.0	4.5	0.0	0.0	3.5	4.0
16	5.0	0.0	0.0	6.0	0.0	1.5	3.0	5.0	2.5	1.0	2.0	0.0	1.0	2.5
17	–	5.5	0.0	1.0	0.0	0.0	4.0	4.5	0.0	2.5	0.0	4.5	4.5	8.0
18	–	–	3.5	3.0	2.0	5.0	4.5	1.0	0.5	4.5	7.5	3.5	0.0	4.0
19	–	–	–	5.0	0.0	1.0	4.5	0.5	6.0	0.5	3.0	2.5	2.5	0.0
20	–	–	–	–	4.5	4.0	0.0	0.0	0.0	5.0	6.0	4.0	2.5	0.5
21	–	–	–	–	–	1.0	2.5	0.0	1.5	5.0	4.5	2.5	0.0	0.0
22	–	–	–	–	–	–	0.0	1.0	3.0	4.0	5.5	0.0	3.0	3.5
23	–	–	–	–	–	–	–	4.0	0.0	1.0	4.0	4.5	2.5	0.0
24	–	–	–	–	–	–	–	–	6.0	4.0	6.5	3.0	0.0	2.0
25	–	–	–	–	–	–	–	–	–	7.5	0.0	0.0	1.0	6.5
26	–	–	–	–	–	–	–	–	–	–	2.5	2.0	8.0	10.0
27	–	–	–	–	–	–	–	–	–	–	–	0.0	10.0	0.5
28	–	–	–	–	–	–	–	–	–	–	–	–	5.0	0.0
29	–	–	–	–	–	–	–	–	–	–	–	–	–	1.0

Appendix C. New instance C

Facility dimensions: $h_F = 25$ and $w_F = 14$
Department areas:

Dept	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Area	15	14	13	14	13	13	7	9	15	14	12	14	12	14	13

Dept	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Area	11	9	14	15	13	8	10	8	13	7	7	10	9	15	9

Pairwise unit distance costs:

c_{ij}	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	2.5	5.0	0.0	0.5	0.0	0.0	0.0	1.5	4.0	1.0	8.0	4.0	8.5	8.0	6.5
2	–	0.0	1.0	0.0	1.0	5.0	6.0	5.5	3.0	8.0	0.0	2.5	4.0	3.0	2.5

(*continued on next page*)

Appendix C (*continued*)

c_{ij}	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
3	–	–	3.0	4.5	4.0	4.0	3.0	2.0	2.5	0.0	8.0	3.0	4.5	5.0	4.5
4	–	–	–	6.5	4.5	3.0	4.5	0.0	0.0	1.0	4.5	0.5	3.5	0.0	7.0
5	–	–	–	–	1.0	3.5	4.0	2.5	1.5	3.0	4.5	4.5	9.0	2.5	5.0
6	–	–	–	–	–	1.5	5.0	6.5	1.5	0.5	2.0	7.0	0.0	0.0	7.0
7	–	–	–	–	–	–	6.5	3.5	5.5	4.5	4.5	0.0	5.0	3.5	3.0
8	–	–	–	–	–	–	–	0.5	0.0	0.5	0.0	5.0	6.0	4.5	2.0
9	–	–	–	–	–	–	–	–	0.0	1.0	0.5	3.0	6.5	1.0	7.5
10	–	–	–	–	–	–	–	–	–	1.5	4.0	0.5	0.5	0.0	0.0
11	–	–	–	–	–	–	–	–	–	–	4.0	1.0	3.5	0.0	0.0
12	–	–	–	–	–	–	–	–	–	–	–	0.0	3.0	3.0	5.0
13	–	–	–	–	–	–	–	–	–	–	–	–	1.5	0.0	8.0
14	–	–	–	–	–	–	–	–	–	–	–	–	–	6.0	3.0
15	–	–	–	–	–	–	–	–	–	–	–	–	–	–	5.0

c_{ij}	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	2.0	2.0	5.0	4.0	3.0	7.0	4.5	2.5	6.0	5.0	1.5	0.0	6.0	5.5
2	0.0	4.5	4.5	0.0	2.5	3.0	3.5	2.5	5.0	2.5	1.5	5.5	3.5	5.0
3	5.5	0.0	4.0	4.5	0.5	6.0	0.0	0.5	1.0	2.0	3.0	1.0	3.5	2.0
4	7.5	0.0	5.5	0.0	5.0	1.5	1.0	0.0	6.5	0.0	5.0	1.5	9.0	0.0
5	5.0	5.5	7.5	0.0	0.5	0.0	7.0	0.0	0.0	1.0	8.0	4.5	0.0	0.0
6	0.0	1.0	7.0	2.0	5.5	4.0	4.5	5.5	2.5	4.0	3.5	0.5	0.0	3.5
7	0.0	2.0	3.0	3.0	2.0	2.0	0.0	0.0	2.0	6.0	3.5	1.5	2.5	4.0
8	7.0	3.5	0.5	4.5	6.0	3.0	0.0	8.0	1.0	5.0	0.0	0.0	7.5	0.0
9	6.0	6.5	0.0	0.0	6.0	2.5	2.5	4.5	8.0	0.0	0.5	0.0	5.5	0.0
10	4.0	2.0	0.0	0.0	0.0	7.0	2.0	4.0	1.5	0.0	5.0	5.0	3.5	1.5
11	1.5	3.0	2.5	2.0	0.0	3.0	5.5	0.0	0.0	3.5	5.0	1.0	4.5	0.0
12	0.5	1.5	3.5	0.0	1.0	4.5	4.0	3.0	4.0	1.5	3.0	0.0	3.0	0.5
13	10.0	0.0	4.5	2.0	0.0	0.0	7.0	1.0	2.5	0.0	0.0	0.0	0.0	0.0
14	0.0	8.0	0.0	0.0	0.0	5.5	6.5	3.0	2.0	1.5	0.5	0.0	7.0	2.5
15	5.5	6.5	5.5	8.5	5.5	0.5	0.0	2.0	7.0	0.0	0.0	4.0	3.0	2.5
16	9.5	0.0	1.5	0.5	4.0	0.0	3.5	2.5	4.5	4.5	5.0	3.0	0.0	1.5
17	–	3.0	5.5	2.5	0.5	1.0	3.0	4.0	7.5	6.0	4.0	0.0	7.0	4.5
18	–	–	0.0	1.0	1.5	5.5	3.0	1.0	3.0	2.0	5.5	0.0	4.5	4.5
19	–	–	–	3.0	0.0	5.5	6.5	2.0	1.0	0.5	7.5	1.5	4.5	0.0
20	–	–	–	–	0.5	3.5	5.0	4.5	0.0	8.0	5.5	1.0	5.0	4.0
21	–	–	–	–	–	3.0	5.0	3.0	3.0	0.0	3.0	3.0	5.0	5.0
22	–	–	–	–	–	–	0.0	3.0	1.0	1.5	4.0	5.0	2.5	0.5
23	–	–	–	–	–	–	–	0.0	2.0	0.0	5.0	0.0	4.5	1.5
24	–	–	–	–	–	–	–	–	1.5	4.5	1.5	4.0	4.0	5.0
25	–	–	–	–	–	–	–	–	–	3.0	0.0	5.0	0.0	6.5
26	–	–	–	–	–	–	–	–	–	–	1.0	4.5	0.0	1.5
27	–	–	–	–	–	–	–	–	–	–	–	0.0	8.5	0.0
28	–	–	–	–	–	–	–	–	–	–	–	–	2.0	9.5
29	–	–	–	–	–	–	–	–	–	–	–	–	–	0.0

Appendix D. New instance D

Facility dimensions: $h_F = 16$ and $w_F = 16$
Department areas:

Dept	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Area	7	10	8	10	10	7	6	10	5	9	10	10	9	9	9

Dept	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Area	9	5	10	6	6	5	5	9	10	5	9	7	7	9	9

Pairwise unit distance costs:

c_{ij}	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	1.0	1.0	4.0	2.0	1.0	0.5	1.0	5.0	0.0	4.5	4.0	1.0	1.5	0.0	2.5
2	–	0.0	3.5	3.0	3.0	0.0	3.0	5.0	1.0	3.5	0.0	5.0	5.5	2.0	7.5
3	–	–	2.0	0.0	1.5	3.5	3.5	2.0	4.5	5.0	4.0	4.0	2.0	0.0	9.0
4	–	–	–	5.0	2.0	2.0	1.0	5.5	1.5	0.0	4.5	0.0	3.5	4.5	0.0
5	–	–	–	–	5.0	3.0	3.0	3.5	0.0	3.5	9.0	0.0	1.0	10.0	5.0
6	–	–	–	–	–	3.5	2.0	5.0	1.5	7.0	1.0	0.0	3.0	5.0	0.0
7	–	–	–	–	–	–	4.5	0.0	2.5	1.0	1.0	0.0	3.0	0.0	3.0
8	–	–	–	–	–	–	–	0.0	4.0	0.5	1.5	1.0	0.0	5.5	1.0
9	–	–	–	–	–	–	–	–	4.0	4.5	2.0	0.0	0.0	9.5	1.5
10	–	–	–	–	–	–	–	–	–	0.0	4.5	3.0	1.5	0.0	0.5
11	–	–	–	–	–	–	–	–	–	–	9.5	3.0	1.0	0.0	5.0
12	–	–	–	–	–	–	–	–	–	–	–	0.0	0.0	4.0	4.0
13	–	–	–	–	–	–	–	–	–	–	–	–	0.5	5.0	0.0
14	–	–	–	–	–	–	–	–	–	–	–	–	–	2.5	2.0
15	–	–	–	–	–	–	–	–	–	–	–	–	–	–	2.5

c_{ij}	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	0.0	4.5	4.5	0.0	3.0	3.0	5.0	6.5	3.0	2.5	0.0	0.5	2.5	3.0
2	1.5	2.5	0.5	4.0	5.0	3.5	3.5	7.5	0.0	5.0	0.0	3.0	0.0	0.0
3	4.5	3.5	0.0	3.0	6.5	1.5	1.0	0.0	1.0	4.0	7.0	3.5	3.0	2.0
4	4.0	4.5	0.0	0.0	2.5	0.0	8.0	3.5	1.0	3.0	7.5	4.0	4.5	7.5
5	1.0	2.5	2.5	1.5	0.0	1.5	1.0	5.0	1.0	9.0	0.5	4.5	4.5	0.0
6	0.0	2.5	0.0	0.0	0.0	3.0	0.5	0.0	0.0	2.0	5.5	3.5	0.5	5.5
7	0.0	4.0	2.0	0.0	0.0	9.5	0.0	0.0	2.5	1.5	0.0	0.5	1.0	6.5
8	5.5	4.0	0.5	0.0	0.0	0.0	0.0	0.0	2.5	0.0	3.5	6.5	0.0	7.0
9	0.5	4.5	0.0	2.5	3.5	0.5	0.0	0.0	4.5	4.0	4.0	3.0	3.0	0.0
10	5.0	5.5	2.0	0.5	5.0	2.5	4.0	2.0	0.0	3.5	0.0	4.5	0.0	0.0
11	5.0	0.5	0.0	5.0	4.5	0.0	2.0	6.5	0.0	0.0	1.5	3.0	8.5	4.5
12	0.0	0.0	2.5	0.0	6.5	1.0	7.0	0.0	5.5	3.0	0.0	1.0	5.0	1.0
13	6.0	4.0	5.0	5.0	6.5	0.0	3.0	3.5	0.0	1.5	1.0	0.0	2.0	0.0
14	0.0	4.0	5.0	0.0	6.0	1.0	3.0	0.5	0.0	6.5	4.0	2.0	4.0	3.5
15	4.0	5.0	4.5	2.0	3.0	6.0	4.0	0.0	0.0	0.0	0.0	0.0	0.0	2.5
16	0.0	0.0	0.0	0.0	3.0	3.5	0.5	1.5	4.0	5.0	0.0	4.0	2.0	1.5
17	–	5.0	0.0	4.0	4.0	5.0	0.0	3.5	0.0	8.5	5.0	6.5	9.0	6.5
18	–	–	3.0	5.5	6.0	5.0	4.0	3.0	5.5	6.5	1.0	7.5	1.0	0.0
19	–	–	–	0.0	3.0	4.5	5.5	0.0	3.0	0.0	5.0	0.5	8.5	4.5
20	–	–	–	–	0.0	0.0	4.5	5.0	3.5	3.0	1.5	1.5	3.0	3.5
21	–	–	–	–	–	1.5	2.5	3.5	5.0	2.5	3.5	5.5	5.5	6.0
22	–	–	–	–	–	–	0.0	4.5	0.0	3.5	4.0	3.0	4.5	8.5
23	–	–	–	–	–	–	–	0.5	0.5	0.0	0.0	2.0	4.5	0.0
24	–	–	–	–	–	–	–	–	8.0	1.5	8.0	2.0	6.0	4.0
25	–	–	–	–	–	–	–	–	–	2.5	0.0	3.5	0.5	1.5
26	–	–	–	–	–	–	–	–	–	–	5.5	4.5	0.5	0.0
27	–	–	–	–	–	–	–	–	–	–	–	5.0	6.0	5.5
28	–	–	–	–	–	–	–	–	–	–	–	–	3.5	8.0
29	–	–	–	–	–	–	–	–	–	–	–	–	–	0.0

Appendix E. New instance E

Facility dimensions: $h_F = 12$ and $w_F = 20$
Department areas:

Dept	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Area	6	7	8	7	9	5	7	8	6	10	7	6	9	5	5

Dept	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
Area	6	6	10	7	8	7	10	10	7	10	8	10	10	5	9

Pairwise unit distance costs:

c_{ij}	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1	4.5	2.0	7.0	1.0	5.0	0.0	4.5	9.0	5.0	0.0	7.0	1.5	3.5	0.0	0.0
2	–	2.0	0.0	8.5	3.0	5.0	1.0	5.0	0.0	2.0	4.5	6.5	0.5	7.5	6.0
3	–	–	3.5	2.0	1.5	0.0	2.0	0.0	5.5	1.5	3.5	2.5	0.5	0.0	0.0
4	–	–	–	0.0	7.0	6.5	4.0	3.5	1.5	6.0	3.0	2.0	0.0	0.0	4.5
5	–	–	–	–	0.0	4.0	4.5	5.0	5.0	0.0	5.0	4.0	1.0	0.0	0.5
6	–	–	–	–	–	2.5	4.0	4.0	1.5	5.0	0.5	4.0	2.0	0.0	8.0
7	–	–	–	–	–	–	5.0	2.0	2.0	3.5	4.0	0.0	0.0	2.0	1.0
8	–	–	–	–	–	–	–	4.0	2.0	5.0	4.5	1.0	7.0	0.0	4.5
9	–	–	–	–	–	–	–	–	4.5	1.0	4.5	0.0	0.5	5.0	4.0
10	–	–	–	–	–	–	–	–	–	3.5	2.0	4.5	5.0	2.5	2.5
11	–	–	–	–	–	–	–	–	–	–	3.5	0.0	2.0	0.0	0.0
12	–	–	–	–	–	–	–	–	–	–	–	3.5	0.0	0.5	0.0
13	–	–	–	–	–	–	–	–	–	–	–	–	4.0	1.5	5.0
14	–	–	–	–	–	–	–	–	–	–	–	–	–	4.0	0.0
15	–	–	–	–	–	–	–	–	–	–	–	–	–	–	0.5

c_{ij}	17	18	19	20	21	22	23	24	25	26	27	28	29	30
1	4.5	0.5	4.5	3.0	2.0	0.0	8.0	0.0	0.0	7.0	4.0	2.0	5.5	5.0
2	2.5	5.0	0.5	4.0	1.5	5.0	7.5	5.0	4.5	8.0	5.0	5.5	2.5	2.5
3	5.5	4.0	6.0	4.5	0.0	4.5	0.0	0.5	3.5	0.0	0.0	2.5	3.0	0.0
4	0.0	2.0	0.5	6.5	4.5	3.0	0.0	0.0	0.0	0.0	3.0	2.5	4.0	0.0
5	7.0	5.0	0.0	1.5	0.0	1.0	1.5	9.5	5.5	4.5	0.0	8.0	0.0	1.0
6	3.0	4.0	7.0	5.0	5.5	1.0	9.0	3.5	4.5	9.0	2.5	0.0	5.0	2.5
7	1.5	0.0	1.5	0.5	1.5	4.0	5.0	0.0	3.5	6.5	5.0	2.0	3.5	3.5
8	9.0	0.0	3.5	5.0	1.5	2.0	0.5	1.5	0.0	3.0	6.0	3.0	2.0	4.5
9	0.0	4.0	0.0	4.0	0.0	3.5	4.5	8.0	0.0	0.0	4.5	0.0	3.0	1.0
10	3.0	7.5	6.0	5.0	3.0	1.5	10.0	2.0	0.0	2.5	0.0	1.5	7.0	2.5
11	0.5	3.0	4.5	3.0	0.0	5.5	1.0	4.5	6.0	1.0	3.0	1.0	5.0	0.5
12	5.0	0.0	0.0	0.0	0.0	3.0	3.0	4.0	9.0	5.0	3.5	2.0	5.0	2.0
13	3.0	9.0	0.0	0.0	0.0	4.0	0.0	3.0	2.0	6.5	5.0	6.5	4.5	0.5
14	4.5	6.5	1.0	5.0	1.5	1.0	3.0	8.0	0.5	3.0	2.5	1.5	0.0	9.5
15	9.0	9.0	4.0	3.0	0.0	2.5	7.0	4.5	0.0	3.5	5.0	3.0	3.5	1.5
16	2.0	3.0	1.5	1.0	3.5	0.0	7.5	2.0	0.0	3.0	0.0	0.0	0.0	0.0
17	–	4.5	3.5	2.0	0.0	0.0	0.0	4.5	0.0	1.0	3.5	2.5	4.0	8.0
18	–	–	2.5	6.5	8.0	4.0	2.5	6.0	0.5	7.0	4.0	4.5	6.0	3.5
19	–	–	–	4.5	8.0	0.0	0.0	5.0	4.5	2.5	5.5	7.5	0.0	1.0
20	–	–	–	–	1.5	6.0	1.5	0.0	5.0	3.0	1.0	3.5	0.5	1.5
21	–	–	–	–	–	4.0	3.5	8.5	4.5	4.5	2.0	7.0	1.5	9.5
22	–	–	–	–	–	–	2.0	2.5	4.5	0.0	0.0	4.5	4.0	2.5
23	–	–	–	–	–	–	–	4.0	2.5	0.5	0.0	0.0	3.5	4.5
24	–	–	–	–	–	–	–	–	1.5	3.0	1.0	5.0	0.0	0.5
25	–	–	–	–	–	–	–	–	–	1.0	4.5	0.0	5.5	3.0
26	–	–	–	–	–	–	–	–	–	–	2.5	0.0	3.0	0.0
27	–	–	–	–	–	–	–	–	–	–	–	5.5	3.5	0.0
28	–	–	–	–	–	–	–	–	–	–	–	–	0.0	6.5
29	–	–	–	–	–	–	–	–	–	–	–	–	–	0.0

References

[1] M.F. Anjos, A. Vannelli, An attractor–repeller approach to floorplanning, Mathematical Methods of Operational Research 56 (1) (2002) 3–27.
[2] M.F. Anjos, A. Vannelli, A new mathematical-programming framework for facility-layout design, INFORMS Journal Computing 18 (1) (2006) 111–118.
[3] M.F. Anjos, H. Wolkowicz, Semidefinite programming for discrete optimization and matrix completion problems, Discrete Applied Mathematics 123 (1–2) (2002) 513–577.
[4] K. Anstreicher, N. Brixius, J.-P. Goux, J. Linderoth, Solving large quadratic assignment problems on computational grids, Mathematical Programming 91 (3, Ser. B) (2002) 563–588.
[5] G.C. Armour, E.S. Buffa, A heuristic algorithm and simulation approach to relative location of facilities, Management Sciences 9 (1963) 294–309.
[6] J. Balakrishnan, C.-H. Cheng, K.-F. Wong, FACOPT: a User friendly FACility layout OPTimization system, Computers and Operation Research 30 (2003) 1625–1641.
[7] M.S. Bazaraa, Computerized layout design: A branch and bound approach, AIIE Transactions 7 (4) (1975) 432–438.
[8] S. Bock, K. Hoberg, Detailed layout planning for irregularly-shaped machines with transportation path design, European Journal of Operational Research 177 (2007) 693–718.
[9] Y.A. Bozer, R.D. Meller, A reexamination of the distance-based facility layout problem, IIE Transactions 29 (7) (1997) 549–560.
[10] I. Castillo, T. Sim, A spring-embedding appraoch for the facility layout problem, Journal of the Operations Research Society 55 (2004) 73–81.
[11] I. Castillo, J. Westerlund, S. Emet, T. Westerlund, Optimization of block layout design problems with unequal areas: A comparison of MILP and MINLP optimixation methods, Computers and Chemical Engineering 30 (1) (2005) 54–69.
[12] I. Castillo, T. Westerlund, An ε-accurate model for optimal unequal-area block layout design, Computers and Operations Research 32 (3) (2005) 429–447.
[13] Y.-K. Chen, S-W. Li, S.-Y. Chou, An efficient two-staged approach for generating block layouts, Computers and Operations Research 29 (2002) 489–504.
[14] E. de Klerk, Aspects of semidefinite programming, Applied Optimization, vol. 65, Kluwer Academic Publishers, Dordrecht, 2002.
[15] Z. Drezner, DISCON: A new method for the layout problem, Operations Research 28 (6) (1980) 1375–1384.
[16] M. Enea, G. Galante, E. Panascia, The facility layout problem approached using a fuzzy model and a genetic search, Journal of Intelligence Manufacturing 16 (2005) 303–316.
[17] L.R. Foulds, Graph Theory Applications, Springer-Verlag, New York, 1991.
[18] K.Y. Gau, R.D. Meller, An iterative facility layout algorithm, International Journal of Production research 37 (16) (1999) 3739–3758.
[19] M. Grötschel, L. Lovász, A. Schrijver, Geometric algorithms and combinatorial optimization, Algorithms and Combinatorics, vol. 2, Springer-Verlag, Berlin, 1993. Second ed..
[20] C. Helmberg, <http://www-user.tu-chemnitz.de/∼helmberg/semidef.html>.
[21] C.L. Huntley, D.E. Brown, A parallel heuristic for quadratic assignment problems, Computers and Operations Research 18 (3) (1991) 275–289.
[22] I. Jankovits. An improved convex optimization model for two-dimensional facility layout. Master's thesis, University of Waterloo, 2006.
[23] J.G. Kim, Y. D Kim, A space partitioning method for facility layout problems with shape constraints, IIE Transactions 30 (10) (1998) 947–957.
[24] J.G. Kim, Y. D Kim, Branch and bound algorithm for locating input and output points of departments on the block layout, Journal of the Operations Research Society 50 (5) (1999) 517–525.
[25] J.S. Kochhar, B.T. Foster, S.S. Heragu, Hope: A genetic algorithm for the unequal area facility layout problem, Computers and Operations Research 25 (1998) 583–594.
[26] M. Laurent, F. Rendl, Semidefinite programming and integer programming, in: K. Aardal, G. Nemhauser, R. Weismantel (Eds.), Handbook on Discrete Optimization, Elsevier, 2005, pp. 393–514.
[27] K.Y. Lee, M.I. Rohb, H.S. Jeong, An improved genetic algorithm for multi-floor facility layout problems having inner structure walls and passages, Computers and Operations Research 32 (2005) 879–899.
[28] Y.H. Lee, M.H. Lee, A shape-based block layout approach to facility layout problems using hybrid genetic algorithm, Computers and Industrial Engineering 42 (2002) 237–248.
[29] Q. Liu, R.D. Meller, A sequence-pair representation and mip-model-based heuristic for the facility layout problem with rectangular departments, IIE Transactions 39 (2007) 377–394.
[30] T.D. Mavridou, P.M. Pardalos, Simulated annealing and genetic algorithms for the facility layout problem: A survey, Computational Optimization and Applications 7 (1) (1997) 111–126.
[31] R.D. Meller, Y.A. Bozer, A new simulated annealing algorithm for the facility layout problem, International Journal of Production research 34 (6) (1996) 1675–1692.
[32] R.D. Meller, W. Chen, H.D. Sherali, Applying the sequence-pair representation to optimal facility layout designs, Operations Research Letters 35 (5) (2007) 651–659.
[33] R.D. Meller, V. Narayanan, P.H. Vance, Optimal facility layout design, Operations Research Letters 23 (1999) 117–127.
[34] M. Mir, M.H. Imam, A hybrid optimization approach for layout design of unequal-area facilities, Computers and Industrial Engineering 39 (2001) 49–63.
[35] B. Montreuil, A modeling framework for integrating layout design and flow network design, in: Proceedings of the material handling research colloquium, pp. 43–58, 1990.
[36] C.E. Nugent, T.E. Vollman, J. Ruml, An experimental comparison of techniques for the assignment of facilities to locations, Operations Research 16 (1968) 150–173.
[37] P. Pardalos, H. Wolkowicz, Wolkowicz (Eds.), Quadratic assignment and related problems, American Mathematical Society, Providence, RI, 1994. Papers from the workshop held at Rutgers University, New Brunswick, New Jersey, May 20–21, 1993..
[38] M. Scriabin, R.C. Vergin, Comparison of computer algorithms and visual based methods for plant layout, Management Sciences 22 (2) (1975) 172–181.
[39] H.D. Sherali, B.M.P. Fraticelli, R.D. Meller, Enhanced model formulation for optimal facility layout, Operations Research 51 (4) (2003) 629–644.
[40] S.P. Singh, R.R.K. Sharma, A review of different approaches to the facility layout problems, International Journal of Advanced Manufacturing Technology 30 (5–6) (2006) 425–433.
[41] J.F. Sturm, Using SeDuMi 1.02, a MATLAB toolbox for optimization over symmetric cones, Optimization Methods Software 11/12 (1–4) (1999) 625–653.
[42] P.L. Takouda, M.F. Anjos, A. Vannelli. Global lower bounds for the VLSI macrocell floorplanning problem using semidefinite optimization, in: Proceedings of IWSOC 2005, pp. 275–280, 2005.
[43] Y. Takuya, I. Takashi, Detailed layout design methodology using mixed integer programming and simulated annealing, J. Japan Ind. Manag. Ass.*** 57 (1) (2006) 39–50.
[44] K.A. Tam, Genetic algorithms, function optimization, and facility layout design, European Journal of Operational Research 63 (1992) 322–346.
[45] K.A. Tam, A simulated annealing algorithm for allocating space to manufacturing cells, International Journal of Production research 30 (1) (1992) 63–87.
[46] D.M. Tate, A.E. Smith, Unequal-area facility layout by genetic search, IIE Transactions 27 (1995) 465–472.
[47] X. Tong, SECOT: A sequential construction technique for facility design. Ph.D. thesis, University of Pittsburgh, 1991.
[48] D.J. van Camp, M.W. Carter, A. Vannelli, A nonlinear optimization approach for solving facility layout problems, European Journal of Operations Research 57 (1991) 174–189.
[49] H. Wolkowicz, R. Saigal, L. Vandenberghe (Eds.), Handbook of Semidefinite Programming, Kluwer Academic Publishers, Boston, MA, 2000.