This is a self-archived preprint of a paper that has been published in: Proceedings Operations Research 2022 (OR 2022 Karlsruhe), Springer, Berlin, 2023, pp. 523-529, https://doi.org/10.1007/978-3-031-24907-5

A 2D Convex Shapes Bin Packing Problem in the Production of Laminated Safety Glass

Steffen Goebbels, Thomas Lühring, and Jochen Rethmann

Niederrhein University of Applied Sciences, Faculty of Electrical Engineering and Computer Science, iPattern Institute, 47805 Krefeld, Germany
{steffen.goebbels,jochen.rethmann}@hsnr.de, thomas.luehring@stud.hn.de

Abstract. The discussed two-dimensional nesting problem is motivated by the production of differently shaped tiles of laminated safety glass that can be represented by primitive, convex polygons. Within as few rectangular bins as possible, representing the space of a furnace, tiles must be placed without overlapping. While the primary problem is to minimize the number of occupied bins, distances between adjacent tiles or a tile and an adjacent furnace boundary must be neither too small nor too large to ensure the stability of the furnace filling during a lamination process. To fulfill this condition, a minimum number of additional rectangular support plates must be added. These plates are considered equivalent to tiles when measuring distances. This is a new aspect that, to our knowledge, has not been covered in the literature so far. We represent the problem as a mixed integer linear program based on no-fit polygons and compare results with those of a greedy-type heuristic.

Keywords: Nesting problem; 2D-irregular shapes bin packing problem

1 Introduction

A variety of heuristics and optimization procedures including evolutionary algorithms and simulated annealing strategies have been developed to tackle several problems of nesting polygonal shapes within rectangular spaces, cf. [2]. This paper discusses a variant with additional constraints motivated by the automation of laminated glass tile production by the company HEGLA-HANIC GmbH. The tiles are homogeneous stacks of glass layers and intermediate foils. The composition of the layers can be optimized from a material point of view (cf. [9]), but that is not intended here. Rather, the tiles are given as simple, convex 2D polygons, which thus do not have to be generated by guillotine cuts. As a primary optimization goal, the tiles must be arranged on a minimum number of rectangular furnace bins without overlaps. During the laminating process, a plate is placed from above the tiles with great pressure. The tiles must not be too close to each other, but also not too far apart, so that the pressure does not cause any damage. The lower distance bound can be easily achieved by enlarging the polygons through scaling. To fulfill the upper distance condition, rectangular support plates can be added. The secondary optimization goal is to minimize their number. This problem is strongly NP hard since the classical bin packing problem is reducible to it. The survey [10] summarizes modeling techniques for 2D nesting problems. Here, we state the problem as a mixed integer linear program (MILP) based on no-fit polygons. Then we discuss a simple greedy approach.

2 Mixed Integer Linear Program

Geometric Basics Let simple, convex polygons P_i, $i \in [n] := \{1, \dots, n\}$, representing tiles be given such that they can be traversed counter-clockwise by following the edges between m_i vertices $\mathbf{v}_{i,1}, \dots, \mathbf{v}_{i,m_i}$ and back to $\mathbf{v}_{i,m_i+1} := \mathbf{v}_{i,1}$ where $\mathbf{v}_{i,k} = (\mathbf{v}_{i,k}.x, \mathbf{v}_{i,k}.y) \in \mathbb{R}^2$. To guarantee a minimum distance between tiles in the final layout, the original tiles have already been enlarged. We also add N (also enlarged) rectangular support plates P_i, $i \in \{n+1, \dots, n+N\}$ of not necessarily different size and a rectangle P_{n+N+1} that will be used to limit maximum distances. The model also allows simple, convex polygons instead of rectangles. For each pair $(i, j) \in [n + N] \times [n + N + 1]$ with $i < j$ we compute a no-fit polygon (NFP, see [1,3]) $F_{i,j}$ with vertices $\mathbf{f}_{i,j,1}, \dots, \mathbf{f}_{i,j,m_{i,j}}, \mathbf{f}_{i,j,m_{i,j}+1} := \mathbf{f}_{i,j,1}$ that are also arranged counter-clockwise. Here, this polygon describes the curve of reference point $\mathbf{v}_{j,1}$ when P_j traverses around the edges of the fixed polygon P_i. Note that with P_i and P_j, also the NFP is simple and convex. This follows directly from the standard algorithm to obtain the shape of an NFP for two convex polygons by orienting P_i counter-clockwise, P_j clockwise, translating all directed edges of both polygons to a single point and then concatenating the edges counter-clockwise giving a polygon $\tilde{F}_{i,j}$ with vertices $\tilde{\mathbf{f}}_{i,j,k}$, see [4]. To obtain the NFP $F_{i,j}$ one only has to translate this shape $\tilde{F}_{i,j}$ according to the reference point and the position of P_i with vector $(\Delta x_{i,j}, \Delta y_{i,j})$,

$$\Delta x_{i,j} := \min_{k \in [m_i]} \mathbf{v}_{i,k}.x - \max_{k \in [m_j]} (\mathbf{v}_{j,k}.x - \mathbf{v}_{j,1}.x) - \min_{k \in [m_{i,j}]} \tilde{\mathbf{f}}_{i,j,k}.x,$$

and $\Delta y_{i,j}$ defined accordingly with x replaced by y. Whereas we restrict ourselves to convex polygons, many algorithms were developed to also compute NFPs for non-convex polygons, see [5,12] and the literature cited there.

To shift a polygon to a certain position, we use an offset $\mathbf{s}_i = (\mathbf{s}_i.x, \mathbf{s}_i.y)$. Shifted polygons $\mathbf{s}_i + P_i$ and $\mathbf{s}_j + P_j$ do not overlap if and only if $\mathbf{s}_j + \mathbf{v}_{j,1}$ lies outside $\mathbf{s}_i + F_{i,j}$. The NFPs have to be computed in advance. As a result, no intersections need to be calculated later.

MILP Let $B \leq n$ be the maximum number of furnace rectangles (bins) to be considered. To choose B sufficiently small, one can use the number of occupied bins of any feasible solution computed with a heuristic, cf. Section 3. Binary variables $x_{i,k} \in \{0, 1\}$ indicate whether a polygon P_i is placed within the furnace rectangle with index $k \in [B]$ (then $x_{i,k} = 1$) or not ($x_{i,k} = 0$). Then the primary goal of the nesting problem is to minimize the number of occupied bins such that all tiles can be placed without overlaps. The secondary goal is to use a minimum number of support plates to fulfill the maximum distance restriction:

With binary variables b_k indicating the use of bin k, the goal is then

$$\text{minimize} \sum_{k\in[B]} b_k + \frac{1}{2N} \sum_{k\in[B]} \sum_{i=n+1}^{n+N} x_{i,k}, \text{ s.t. } \forall_{k\in[B]} \sum_{i\in[n]} x_{i,k} \leq n \cdot b_k$$

and several further restrictions described in what follows.

The objective function is lower bounded by the area of all tiles divided by the furnace area. All coordinates plus offsets, i.e., coordinates of points $\mathbf{v}_{i,k}+\mathbf{s}_i$, have to be within the range of the furnace rectangle coordinates. Each polygon has to be placed within at most one bin (cf. (1)): $\forall_{i\in[N+n]} \sum_{k\in[B]} x_{i,k} \leq 1$.

There must be no overlaps between polygons $\mathbf{s}_i+P_i$ and $\mathbf{s}_j+P_j$ placed within the same bin (i.e., $x_{i,k} = x_{j,k} = 1$), i.e., by considering convexity of the NFP $F_{i,j}$, the reference point $\mathbf{s}_j + \mathbf{v}_{j,1}$ must lie in at least one half-plane bounded by a straight line through an edge of the NFP $\mathbf{s}_i + F_{i,j}$ and in which the NFP is not located. For such a half-plane, $y_{i,j,k} \in \{0,1\}$ is set to one. By applying the inner product "$\cdot$" and by considering the Hesse normal form of lines (the absolute value of the inner product between a point on a line and a normal of the line is the distance to the origin, here the outer normal of the occupied half plane is chosen to compare signed distances), one gets conditions (cf. [7,10])

$$\begin{aligned}
&\forall_{i,j\in[n+N],i<j}\forall_{k\in[m_{i,j}]}\forall_{l\in[b]}\\
&\quad(\mathbf{f}_{i,j,k+1}.y - \mathbf{f}_{i,j,k}.y, -\mathbf{f}_{i,j,k+1}.x + \mathbf{f}_{i,j,k}.x)\\
&\quad\cdot[(\mathbf{s}_i.x + \mathbf{f}_{i,j,k}.x, \mathbf{s}_i.y + \mathbf{f}_{i,j,k}.y) - (\mathbf{s}_j.x + \mathbf{v}_{j,1}.x, \mathbf{s}_j.y + \mathbf{v}_{j,1}.y)]\\
&\qquad\leq M(2 - x_{i,l} - x_{j,l}) + M(1 - y_{i,j,k}),\\
&\forall_{i,j\in[n+N],i<j} \sum_{k\in[m_{i,j}]} y_{i,j,k} \geq 1.
\end{aligned}$$

The constant $M > 0$ has to be chosen sufficiently large. For non-convex polygons, checking with convex regions outside the NFP can be done instead of checking with half planes, see [6].

So far, we have not described how to enable rotations. In the application under consideration, only rotations by multiples of 90° are to be discussed (orthogonal rotation). Rotations by a finite number of angles can be easily represented by adding rotated tile polygons (of different shape) to the list of polygons P_i and by assuring that exactly one rotated instance of a polygon has to be placed in exactly one bin, i.e., for each index set $I \subset [n]$, representing all rotated instances of a tile, we require

$$\sum_{i\in I} \sum_{k\in[B]} x_{i,k} = 1. \tag{1}$$

We model a maximum distance condition by placing a grid with g points over all furnace rectangles, i.e., bins, see Fig. 1. Let $\mathbf{g}_i \in \mathbb{R}^2$, $i \in [g]$, be offset vectors that shift predefined rectangle P_{n+N+1} to have a center point at a corresponding grid point. The condition is that, for each grid point indexed by $i \in [g]$, in each bin at least one intersection between a placed tile or support plate polygon $\mathbf{s}_j + P_j$ and

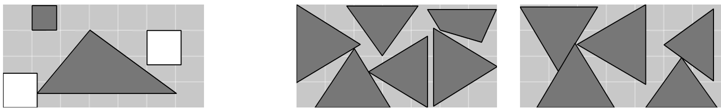

Fig. 1. The distance condition (2, 3) requires that each square of the background grid has to be at least partially covered. Left: A feasible one-bin layout for two tiles (grey) with two support plates (white). Right: An optimal solution using two bins without support plates (instance 3 in Table 1).

this shifted rectangle $\mathbf{g}_i + P_{n+N+1}$ has to occur. Such an intersection is indicated by setting a binary variable $z_{i,j,l} \in \{0,1\}$, $i \in [g]$, $j \in [n+N]$, $l \in [B]$, to one. It occurs if and only if the reference point $\mathbf{g}_i + \mathbf{v}_{n+N+1,1}$ lies inside each half plane that is bounded by a line through an edge of the NFP $\mathbf{s}_j + F_{j,n+N+1}$ and that is occupied by the NFP.

$$
\begin{aligned}
&\forall_{i\in[g]}\forall_{j\in[n+N]}\forall_{k\in[m_{j,n+N+1}]}\forall_{l\in[B]}\\
&\qquad(\mathbf{f}_{j,n+N+1,k+1}.y - \mathbf{f}_{j,n+N+1,k}.y, -\mathbf{f}_{j,n+N+1,k+1}.x + \mathbf{f}_{j,n+N+1,k}.x)\\
&\qquad\cdot [(\mathbf{s}_j.x + \mathbf{f}_{j,n+N+1,k}.x, \mathbf{s}_j.y + \mathbf{f}_{j,n+N+1,k}.y)\\
&\qquad\qquad - (\mathbf{g}_i.x + \mathbf{v}_{n+N+1,1}.x, \mathbf{g}_i.y + \mathbf{v}_{n+N+1,1}.y)]\\
&\qquad\geq -M(1 - x_{j,l}) - M(1 - z_{i,j,l}), \qquad (2)\\
&\forall_{i\in[g]}\forall_{l\in[B]} \sum_{j\in[n+N]} z_{i,j,l} > \sum_{j\in[n+N]} (1 - x_{j,l}). \qquad (3)
\end{aligned}
$$

If $x_{j,l} = 0$, one can choose $z_{i,j,l} = 1$, i.e., $\forall_{i\in[g]}\forall_{j\in[n+N]}\forall_{l\in[B]}\, z_{i,j,l} \geq 1 - x_{j,l}$.

Certain solver heuristics appear to work better if shifted rectangles $\mathbf{g}_i + P_{n+N+1}$ slightly overlap such that placement in overlap regions is preferred.

3 Greedy Approach

In up to 10,000 (nearly) random orders (permutations), we iteratively position the tiles in a bottom-left strategy. Then, among all results with the smallest number of bins, we select a result that intersects with the largest number of rectangles $\mathbf{g}_i + P_{n+N+1}$ so that a small number of support plates is needed. Motivated by the instability of the problem, this stochastic experiment replaces a local search to find a good order. To further reduce the required number of support plates, tiles small enough to fit into the distance rectangle P_{n+N+1} are always placed at the end of each permutation so that they can be inserted into empty distance rectangles with priority. We basically use steps 1–5 of the genetic algorithm in [8] in the implementation of the bottom-left strategy. However, we do not only attach to the last placed tile polygon but to all polygons. We shift each attached polygon as far as possible to the left and to the bottom by using a binary search for feasible positions that also allows to fill gaps. After placing

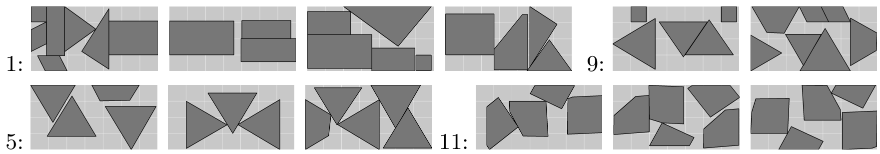

Fig. 2. Feasible solutions of four problem instances computed by CPLEX on 12 threads within a limit of one hour elapsed time (instances 1, 5, 9, and 11 in Table 1)

the tiles, support plates are added to fulfill (2, 3). As long as each support plate fits into the rectangle P_{n+N+1} of the distance condition, and if enough support plates are provided, this is always possible. Since we use distance rectangles that slightly overlap, we greedily search for a vertex of these rectangles that is covered by a maximum number of so far empty distance rectangles. Then we place a support plate there (if it fits). Finally, we remove some of the support plates by re-arranging tiles: For each rectangle $\mathbf{g}_i + P_{n+N+1}$ in which a support plate is placed, we try to shift a tile from the left or from the bottom to the border of this rectangle such that (2, 3) holds without the support plate.

4 Results

Results for exemplary problem instances[1] provided by HEGLA-HANIC GmbH are listed in Table 1. Small instances can be solved with our MILP to optimality, e.g., see Fig. 1. However, for most instances up to 20 tiles, CPLEX 12.8 was able to find feasible (but not necessarily optimal) solutions within 60 minutes, cf. Fig. 2. The greedy approach found feasible solutions for all instances within less than two minutes when working with 10,000 permutations, but in most cases 1,000 permutations led to similar results in a fraction of time. For instances that could be solved with the MILP, the greedy heuristic was able to obtain the same number of bins as the MILP and reduce the number of support plates to one on average, while the feasible solutions found by MILPs within the time limit had zero support plates on average. For some instances, we could further reduce the support plates manually, see upper bounds for the optimum in brackets.

5 Conclusions

Although the greedy approach often fails to find a minimum number of support plates, it is apparently sufficient for practical use. Future work may test other strategies. For example, the assignment to bins could be separated from the placement of tiles and support plates within the bins in a branch-and-bound approach. The prerequisites of the framework in [11] are fulfilled. Grouping tiles into classes could help doing the tile assignment.

[1] Data available at https://www.hs-niederrhein.de/fileadmin/dateien/FB03/Personen/goebbels/Publikationen/dataset.zip

Table 1. Results of the MILP and the greedy strategy (10,000 permutations): A best feasible solution of the MILP is considered if the time limit of 60 minutes is exceeded.

		MILP (CPLEX, 12 threads)			Greedy Strategy (one thread)		
instance	tiles	bins	support	time [s]	bins	support	time [s]
1	19	4	0	exceeded	4	0	13
2	47	–	–	exceeded	9	8 (≤ 5)	86
3	11	2	0	18	2	1 (0)	5
4	12	–	–	exceeded	4	4 (≤ 3)	15
5	12	3	0	exceeded	3	3 (0)	7
6	12	–	–	exceeded	8	7 (≤ 5)	20
7	28	–	–	exceeded	9	3 (≤ 1)	18
8	36	–	–	exceeded	7	5 (≤ 4)	51
9	12	2	0	exceeded	2	0	6
10	14	–	–	exceeded	4	1 (0)	8
11	14	3	0	exceeded	3	1 (0)	12
12	46	–	–	exceeded	16	17 (≤ 9)	73

References

1. Adamowicz, M., Albano, A.: Nesting two-dimensional shapes in rectangular modules. Comput. Aided Des. **8**, 27–33 (1976)
2. Albano, G.: Optimal allocation of two-dimensional irregular shapes using heuristic search methods. IEEE Trans Syst Man Cybern **10**, 242–248 (2010)
3. Art, R.: An approach to the two dimensional irregular cutting stock problem. Technical Report IBM Cambridge Scientific Centre **36-Y08** (1966)
4. Cunninghame-Green, R.: Geometry, shoemaking and the milk tray problem. New Sci. **1677**, 50–53 (1989)
5. Dean, H.T., Tu, Y., Raffensperger, J.F.: An improved method for calculating the no-fit polygon. Comput. Oper. Res. **33**, 1526–1539 (2006)
6. Fischetti, M., Luzzi, I.: Mixed-integer programming models for nesting problems. J. Heuristics **15**(3), 201–226 (2009)
7. Gomes, A., Oliveira, J.: Solving irregular strip packing problems by hybridising simulated annealing and linear programming. Eur. J. Oper. Res. **171**, 811–829 (2006)
8. Junior, B.A., Pinheiro, P.R., Saraiva, R.D.: A hybrid methodology for tackling the irregular strip packing problem. IFAC Proceedings Volumes **46**(7), 396–401 (2013)
9. Kaveh, A., Dadras, A., Geran Malek, N.: Robust design optimization of laminated plates under uncertain bounded buckling loads. Struct. Multidiscip. Optim. **59**(3), 877–891 (2019)
10. Leao, A.A., Toledo, F.M., Oliveira, J.F., Carravilla, M.A., Alvarez-Valdés, R.: Irregular packing problems: A review of mathematical models. Eur. J. Oper. Res. **282**(3), 803–822 (2020)
11. Rebennack, S., Kallrath, J., Pardalos, P.M.: Column enumeration based decomposition techniques for a class of non-convex minlp problems. J Glob Optim **43**(2-3), 277–297 (2009)
12. Rocha, P.: Robust NFP generation for nesting problems. arXiv preprint **arXiv:1903.11139v1** (2019)