Yugoslav Journal of Operations Research
21 (2011), Number 1, 93-102
DOI: 10.2298/YJOR1101093S

SOLVING THE TWO-DIMENSIONAL PACKING PROBLEM WITH m-M CALCULUS

Aleksandar SAVIĆ,
Faculty of Mathematics, University of Belgrade
aleks3rd@gmail.com
Tijana ŠUKILOVIĆ,
Faculty of Mathematics, University of Belgrade
tijana@matf.bg.ac.rs
Vladimir FILIPOVIĆ
Faculty of Mathematics, University of Belgrade
vladofilipovic@hotmail.com

Received: July 2010 / Accepted: May 2011

Abstract: This paper considers the two dimensional rectangular packing problem. The mathematical formulation is based on the optimization of a non-linear function with piecewise linear constraints with a small number of real variables. The presented method of m-M calculus finds all optimal solutions on small instances. Computational performance is good on smaller instances.

Key words: Non-linear optimization, m-M calculus, two dimensional packing.

MSC: 90C30, 90C56

1. INTRODUCTION

Packing problem and the closely related problem of cutting, encompass a wide range of industry originated problems. A successful optimal solution, or even finding an approximately good solution, significantly facilitate in both saving money and raw materials.

Most of the contributions in literature are devoted to the case where the items to be packed have a fixed orientation with respect to the stock unit(s), i.e., it is not allowed to rotate them. This case, which is the object of the present article, reflects a number of practical contexts, such as the cutting of corrugated or decorated material (wood, glass, clothing stripes), or the newspapers paging.

For variants allowing rotations (usually by 90º) and/or constraints on the items placement (such as the ‘‘guillotine cuts’’) the reader is referred to [1, 8], where a three-field classification of the area is also introduced. General surveys on cutting and packing problems can be found in [10, 11, 12]. Results of the probabilistic analysis of packing algorithms can be found in [6] and [7].

Let us introduce the problems in a more formal way. We are given a set of n rectangular *items* $j \in J = \{1,\dots,n\}$ each defined by a width, w_j, and a height, h_j:

(i) in the *Two-Dimensional Bin Packing Problem* (2BP), we are further given an unlimited number of identical rectangular bins of width W and height H, and the objective is to allocate all the items to the minimum number of bins;

(ii) in the *Two-Dimensional Strip Packing Problem* (2SP), we are further given a bin of width W and infinite height (hereafter called strip), and the objective is to allocate all the items to the strip by minimizing the height to which the strip is used.

In both cases, the items have to be packed with their w-edges parallel to the W-edge of the bins (or strip). We will assume, without loss of generality, that all input data are positive integers, and that $w_j \leq W$ and $h_j \leq H$ ($j = 1, \dots, n$).

Both problems are strongly NP-hard, as is easily seen by transformation from the strongly NP-hard (one-dimensional) Bin Packing Problem (1BP), where n items, each having an associated size h_j, have to be partitioned into the minimum number of subsets, so that the sum of the sizes in each subset does not exceed a given capacity H [20].

A special case of the aforementioned problems is packing a set of rectangles into a bounding rectangle.

The problem of determining whether a set of rectangles can be packed into a bounding rectangle can be related to some practical applications in floor planning, placement, or job scheduling problems [9]. The rectangle packing problem belongs to a subset of classical cutting and packing problems and has been shown to be NP-complete [17, 19].

Cutting and packing problems [1, 2, 4, 8, 15, 25] have a wide range of applicability that has been studied for more than 40 years. The two-dimensional cutting stock problems with rectangle shapes are closely related to our problem. In [5], the basic formulation issues and solution procedures of linear programming, sequential heuristic and hybrid solution procedures are addressed. Based on the usage of guillotine or non-guillotine cuts, the problems can be roughly divided into two styles of cutting. Constrained cutting refers to the employment of orthogonal and guillotine cuts under certain copy constraints of the rectangles [13]. Optimal algorithms for orthogonal two-dimension cutting were proposed in [2, 15]. However, they might not be practical for large problems. In the orthogonal packing problem, a set of rectangles are to be packed into a rectangle board with rectangle edges parallel to the x- and y-axes of the board, respectively, and the height is to be minimized. In order to reduce the number of possible orthogonal packing patterns, a so-called BL-condition was introduced, and a combination algorithm using genetic and deterministic methods was proposed [3, 14, 18]. In [22], a new and improved algorithm with level heuristics was suggested. In [16, 21, 25], a various range of metaheuristics was suggested for solving the two dimensional rectangle packing problem. A very effective quasi-human heuristic was presented in [26]

This work presents an application of *m-M calculus* [23] on the mathematical model given in [24]. The proposed method of solving will allow finding all optimal solutions for the problem, which can contribute to the quality of the solution, because some different optimal solutions can be more or less preferable by some other criterion.

2. MODEL

The problem expounded is the two dimensional packing of a big rectangle with a given number of small rectangles. It is important to consider that the sides of small rectangles must be parallel with the sides of a big rectangle. Two cases are possible: in the first case, a 90° degree rotation for smaller rectangles is not allowed, while the second case allows rotation. Smaller rectangles must be placed completely inside the big rectangle and they must not overlap. Let the dimensions of the big rectangle be respectively A and B. There are n small rectangles. Let their dimensions be respectively a_i and b_i $i = 1,. . . ., n$. It is allowed for the small rectangles to have the same dimensions.

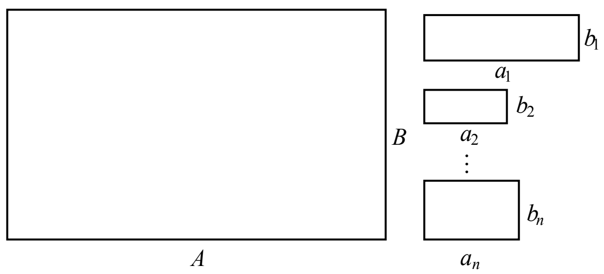

Figure 1 Big rectangle and array of smaller rectangles

Let us consider a strip of width A and infinite length. Rectangles will be packed on this strip, and let bounding rectangle be at the beginning of the strip, as shown on Figure 2.

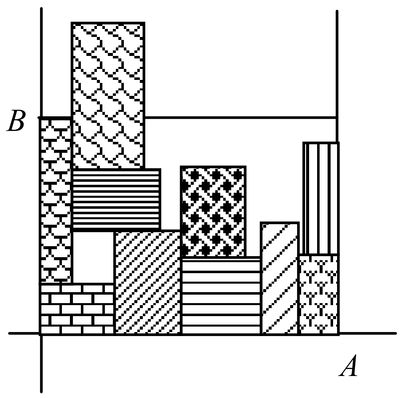

Figure 2 Infinite strip with big rectangle at its beginning packed with smaller rectangles

Let x_i and y_i, $i = 1, \dots, n$ be coordinates of the centers of rectangles in the coordinate system, where strip starts from x-axis and is spreading in the direction of y-axis. Bounding rectangle then has vertices in points with coordinates (0,0), (0,B), (A,B) and (A,0). Model is presented and proofed in [24]. Now the mathematical model for packing rectangles into bounding rectangle is as follows.

$$\max \sum_{i=1}^{n} \frac{\max(0, B - y_i)}{B - y_i} a_i b_i \tag{1}$$

$$\left| \frac{x_j - x_i}{a_i + a_j} - \frac{y_j - y_i}{b_i + b_j} \right| + \left| \frac{x_j - x_i}{a_i + a_j} + \frac{y_j - y_i}{b_i + b_j} \right| \geq 1, \quad i = 1, 2, \dots, n-1, \quad j = i+1, \dots, n \tag{2}$$

$$|y_i - B| \geq \frac{b_i}{2}, \quad i = 1, 2, \dots, n \tag{3}$$

$$\frac{a_i}{2} \leq x_i \leq A - \frac{a_i}{2}, \quad i = 1, 2, \dots, n \tag{4}$$

$$y_i \geq \frac{b_i}{2}, \quad i = 1, \dots, n \tag{5}$$

The model is based on the fact that the rotation of rectangles is not allowed. Conditions (2) do not allow rectangles to overlap. Conditions (3) stipulate that a rectangle should be put either in bounding rectangle, or outside without overlapping the boundaries. Conditions (4) and (5) stipulate that rectangles stay inside the strip of width A and infinite length without overlapping the boundaries. Objective function (1) is such that only rectangles put in bounding rectangle are considered.

It is easy to see that this is a non-linear mathematical model. All conditions are at the most piecewise linear functions of variables x_i and y_i. The number of variables is relatively small and is equal to $2n$. The number of conditions equals

$$\frac{n(n-1)}{2} + 3n = \frac{1}{2}(n^2 + 5n) \tag{6}$$

The number of conditions is relatively large; however, this fact is not significant in proposing a method for solving the problem.

3. *m-M* CALCULUS

The method of m-M calculus is presented in [23]. This paper will briefly present its general ideas. Let the variables x_1, x_2, …, x_n be from some segment $D = [a_1, b_1] \times [a_2, b_2] \times \dots \times [a_n, b_n] \subseteq R^n$. Then let there a system of inequalities be given

$$f_1(x_1, x_2, \dots, x_n) \geq 0, f_2(x_1, x_2, \dots, x_n) \geq 0, \dots, f_k(x_1, x_2, \dots, x_n) \geq 0 \tag{7}$$

where functions $f_i : D \to R, \quad i = 1, 2, \dots, k$.

Now, for any $\Delta \subseteq D$, $m(f_i)(\Delta), M(f_i)(\Delta)$ can be determined. By definitions given in [23], segment Δ is feasible if and only if

$$(\forall i)(M(f_i)(\Delta) \geq 0) \tag{8}$$

Also, segment Δ is solutional if and only if

$$(\forall i)(m(f_i)(\Delta) \geq 0) \tag{9}$$

and finally, segment Δ is indeterminate if and only if

$$(\forall i)(M(f_i)(\Delta) \geq 0), \quad (\exists i)(m(f_i)(\Delta) < 0) \tag{10}$$

Now, idea of solving with m-M calculus is that in a tree cell strategy starting set D is decomposed in segments. In every iteration, all infeasible segments are rejected, until either there are no more segments left, which means there is no solution; or there are no more indeterminate segments, which means all that is left are solely solutional segments. A union of solutional segments gives the solution set. As decomposition progresses, iteration includes a better approximation of solution set. Finally, the results of this procedure are all the solutions for the system of inequalities (7).

This last fact determined using this method for solving problems (1)-(5).

Based on the previously presented mathematical model for packing, we had to solve a problem of constrained optimization on an n-dimensional segment.

For that purpose, first we had to write the appropriate m-M functions for inequality formulas. m-function for objective function and inequalities (2) and (3) are:

$$\sum_{i=1}^{n} \frac{\max\left(0,\ B-by_i\right)}{B-ay_i} a_i b_i - AB \tag{11}$$

$$\begin{aligned}&1-\max\left(\frac{bx_j-ax_i}{a_i+a_j}-\frac{ay_j-by_i}{b_i+b_j}, \frac{ax_j-bx_i}{a_i+a_j}-\frac{by_j-ay_i}{b_i+b_j}\right)-\\&-\max\left(\frac{bx_j-ax_i}{a_i+a_j}+\frac{ay_j-by_i}{b_i+b_j}, -\frac{bx_j-ax_i}{a_i+a_j}-\frac{ay_j-by_i}{b_i+b_j}\right)\end{aligned} \tag{12}$$

$$\frac{b_i}{2}-\max\left(\left|by_i-B\right|,\left|B-ay_i\right|\right). \tag{13}$$

and respectively M-functions are

$$\sum_{i=1}^{n} \frac{\max\left(0,\ B-ay_i\right)}{B-by_i} a_i b_i - AB \tag{14}$$

$$\begin{aligned} &1-\min\left(\frac{bx_j-ax_i}{a_i+a_j}-\frac{ay_j-by_i}{b_i+b_j},\frac{ax_j-bx_i}{a_i+a_j}-\frac{by_j-ay_i}{b_i+b_j}\right)- \\ &-\min\left(\frac{bx_j-ax_i}{a_i+a_j}+\frac{ay_j-by_i}{b_i+b_j},-\frac{bx_j-ax_i}{a_i+a_j}-\frac{ay_j-by_i}{b_i+b_j}\right) \end{aligned} \tag{15}$$

$$\frac{b_i}{2}-\min\left(\left|by_i-B\right|,\left|B-ay_i\right|\right) \tag{16}$$

where ay_i and by_i are lower-left and upper-right y-coordinate of i^{th} rectangular, respectively. Constraints (4) and (5) are simple inequalities, and their m and M functions are the same as constraints.

Also, since we have been using dyadic tree cell-decomposition strategy, we had to establish some feasibility criteria for the n-dimensional segment. The Cartesian product of $[X_1,Y_1] \times [X_2,Y_2] \times \ldots \times [X_n,Y_n]$ n-cells is *feasible* if it satisfies two conditions:

(1) $(\forall i \in \{1,\ldots,k\})\ m(f_i)([X_1,Y_1]\times[X_2,Y_2]\times\ldots\times[X_n,Y_n] \geq 0$

(2) $(\forall i \in \{1,\ldots,k\})\ m(f_i)([X_1,Y_1]\times[X_2,Y_2]\times\ldots\times[X_n,Y_n] < 0 \Rightarrow$
$M(f_i)([X_1,Y_1]\times[X_2,Y_2]\times\ldots\times[X_n,Y_n] \geq 0$

Therefore, our algorithm is simple: we start from a reshaped bounding box and perform cell-decomposition until there are no more feasible cells, or some predefined level of accuracy is accomplished.

4. EXPERIMENTAL RESULTS

All experiments were performed on a PC with P4 2G CPU and 960Mb RAM. Experiments were performed on a small number of rectangles, to be more specific, 4 and 5. The bounding rectangle in all tests had dimensions 5 by 7.

In the first test we packed 4 rectangles with dimensions 3×4, 2×6, 3×2 and 5×1 into the 5×7 rectangle. There are 8 possible solutions shown in Figure 3, and our proposed method was able to detect all of them. The results are presented in Table 1.

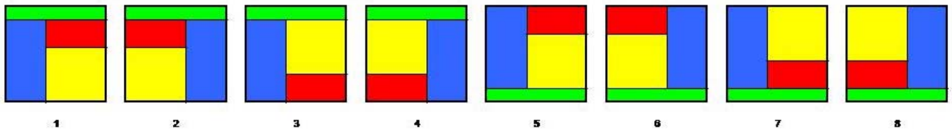

Figure 3 Packing 4 rectangles

Table 1 Centers' coordinates of 4 packed rectangles for all possible solutions

rectangle number / solution number	1 (yellow, 3×4)	2 (blue, 2×6)	3 (red, 3×2)	4 (green, 5×1)
1	4.5, 2.0.	1.0, 3.0.	4.5, 5.0.	3.5, 6.5.
2	1.5, 2.0.	4.0, 3.0.	1.5, 5.0.	3.5, 6.5.
3	4.5, 4.0.	1.0, 3.0.	4.5, 1.0.	3.5, 6.5.
4	1.5, 4.0.	4.0, 3.0.	1.5, 1.0.	3.5, 6.5.
5	4.5, 3.0.	1.0, 4.0.	4.5, 6.0.	3.5, 0.5.
6	1.5, 3.0.	4.0, 4.0.	1.5, 6.0.	3.5, 0.5.
7	4.5, 5.0.	1.0, 4.0.	4.5, 2.0.	3.5, 0.5.
8	1.5, 5.0.	4.0, 4.0.	1.5, 2.0.	3.5, 0.5.

Total processing time was 61 second, while the memory cost was 398Mb.

As shown in Figure 3, most solutions were obtained from one solution using symmetric properties. Because of this, it was necessary to make the method modification so that the stopping criterion would be finding the first optimal solution.

Results of the method modified with this criterion performed on two sets of rectangles, first with 4 and second with 5 rectangles can be seen on Figure 4. The depicted solutions are the first found solutions in both cases.

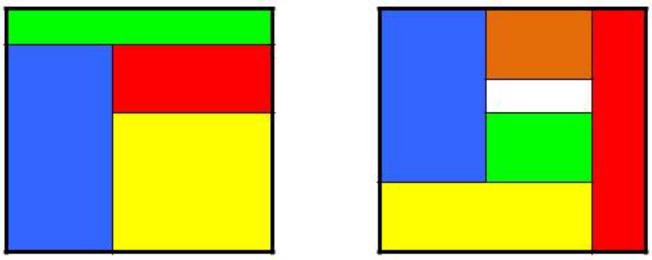

Figure 4 Left: first optimal solution for packing 4 rectangles; right: first optimal solution for packing 5 rectangles (the gap between orange and green rectangle is unoccupied space)

Results obtained by finding the first optimal solution are presented in Table 2.

Table 2 Results of packing with modified stopping criterion

	Test 1	Test 2
Number of rectangles	4	5
Dimensions of rectangles	2×6 3×2 3×4 5×1	1×7 2×2 2×2 2×5 4×2
Time for detecting first optimal solution (s)	2	69
Memory cost (MB)	61	681
Detected centers	1.0, 3.0. 4.5, 5.0. 4.5, 2.0. 3.5, 6.5.	4.5, 3.5. 3.0, 4.5. 3.0, 6.0. 1.0, 4.5. 2.0, 1.0.

As shown in Table 2, memory requirements are greatly increasing with the number of small rectangles. Because of this, each cell was represented by only three pieces of information. Each cell consists of two pointers to the lower left and upper right end of the cell and *id* of the cell – indicator to efficiently determine whether the solution has been detected, and from which cell the decomposition solving process should continue. Endpoints of a cell were structured as two *n*-dimensional arrays of floats and *id* is the integer type. Although the process of cutting-off non-feasible cells is very fast, number of solutional segments grows exponentially, and so does the memory cost. Build-up of the number of solutional segments is in one moment drastically cut-off, and after that point in process the number of feasible segments plummets. If procedure reaches this point, all optimal solutions are quickly found afterwards. Otherwise, memory requirements stopped the execution of the algorithm.

5. CONCLUSION

This paper presented a method for finding all optimal solutions for the two dimensional rectangle packing problem. The method applied m-M calculus and dyadic decomposition of starting bounding rectangle. Performance was relatively good for small instances, but for greater instances, the method needs much memory. The modification for finding the first optimal solution was presented as well.

Based on the presented results, we can conclude that the method is applicable for finding all solutions, but has drawbacks. The most promising direction would be in the better optimization of the decomposition process, which can lead to improvement in method performance, especially for greater instances.

REFERENCES

[1] Baldacci, R., and Boschetti, M.A., "A cutting-plane approach for the two-dimensional orthogonal non-guillotine cutting problem", *European Journal of Operational Research*, 183 (3) (2007) 1230-1248.

[2] Beasley, J.E., "An exact two-dimensional non-guillotine cutting tree search procedure", *Operations Research,* 33 (1985) 49–64.

[3] Binkley, K.J., and Hagiwara, M., "Applying self-adaptive evolutionary algorithms to two-dimensional packing problems using a four corners' heuristic", *European Journal of Operational Research*, 183 (3) (2007) 1230-1248.

[4] Christofides, N., and Hadjiconstantinou, E., "An exact algorithm for orthogonal 2-D cutting problems using guillotine cuts", *European Journal of Operational Research,* 83 (1995) 21–38.

[5] Cintra, G.F., Miyazawa, F.K., Wakabayashi, Y., and Xavier, E.C., "Algorithms for two-dimensional cutting stock and strip packing problems using dynamic programming and column generation", *European Journal of Operational Research*, 191 (1) (2008) 59-83.

[6] Coffman, E.G., Jr., and Lueker, G.S., *Probabilistic Analysis of Packing and Partitioning Algorithms*, Wiley, Chichester, 1992.

[7] Coffman, E.G., Jr., and Shor, P.W., "Average-case analysis of cutting and packing in two dimensions", *European Journal of Operational Research,* 44 (1990) 134–144.

[8] Correa, J.R., "Near-optimal solutions to two-dimensional bin packing with 90 degree rotations", *Electronic Notes in Discrete Mathematics*, 18 (2004) 89-95.

[9] Diegert, C., "Practical automatic placement for standard-cell integrated circuit", *American Journal of Mathematical and Management Sciences,* 8 (34) (1988) 309–328.

[10] Dowsland, K.A., and Dowsland, W.B., "Packing problems", *European Journal of Operational Research,* 56 (1992) 2–14.

[11] Dyckhoff, H., Finke, U., *Cutting and Packing in Production and Distribution*, Physica Verlag, Heidelberg, 1992.

[12] Dyckhoff, H., Scheithauer, G., and Terno J., "Cutting and packing (C&P)", in: M., Dell'Amico, F., Maffioli, S., Martello, (eds.), *Annotated Bibliographies in Combinatorial Optimization*, Wiley, Chichester, 1997, 393–413.

[13] Epstein, L., Levin, A., and Van Stee, R., "Two-dimensional packing with conflicts", *Acta Informatica*, 45 (3) (2008) 155-175.

[14] Gonzalves, J.F., and Resende, M.G.C., "A parallel multi-population genetic algorithm for a constrained two-dimensional orthogonal packing problem", *Journal of Combinatorial Optimization*, (2010) 1-22 Article in Press.

[15] Hadjiconstantinou, E., and Christofides N., "An optimal algorithm for general orthogonal 2-D cutting problems", Technical report MS-91/2, Imperial College, London, UK.

[16] Harwig, J.M., Barnes, J.W., and Moore, J.T., "An adaptive tabu search approach for 2-dimensional orthogonal packing problems", *Military Operations Research*, 11 (2) (2006) 5-26.

[17] Hochbaum, D.S., and Maass, W., "Approximation schemes for covering and packing problems in image processing and VLSI", *Journal of the Association for Computing Machinery*, 32 (1) (1985) 130–136.

[18] Kim, J., Moon, B.-R., "A hybrid genetic algorithm for a variant of two-dimensional packing problem", *Proceedings of the 11th Annual Genetic and Evolutionary Computation Conference*, *GECCO-2009*, 287-292.

[19] Leung J., Tam, T., Wong, C.S., Young, G., and Chin, F., "Packing squares into square", *Journal of Parallel and Distributed Computing*, 10 (1990) 271–275.

[20] Lodi, A., Martello, S., and Monaci, M., "Two-dimensional packing problems: A survey", *European Journal of Operational Research*, 141 (2002) 241–252.

[21] Lu, S., Lu, Y., Zha, J., and Lu, S., “Cross-entropy method for two-dimensional packing problem with rectangle pieces”, *Beijing Jiaotong Daxue Xuebao/Journal of Beijing Jiaotong University,* 33 (2) (2009) 39-43.

[22] Ortmann, F.G., Ntene, N., van and Vuuren, J.H., “New and improved level heuristics for the rectangular strip packing and variable-sized bin packing problems”, *European Journal of Operational Research*, 203 (2) (2010) 306-315.

[23] Prešić S.B., “A survey of the m-M calculus”, *Yugoslav Journal of Operations Research*, 8 (1) (1998) 137-168.

[24] Savić A., “Packing and cutting problems“, Master thesis, Faculty of Mathematics, University of Belgrade, 2000.

[25] Toro, E.M., Garcs, A., and Ruiz, H., “Two dimensional packing problem using a hybrid constructive algorithm of variable neighborhood search and simulated annealing”, *Revista Facultad de Ingenieria,* 46 (2008) 119-131.

[26] Wu, Y.L., Huang, W., Lau, S., Wong, C.K., and Young, G.H., “An effective quasi-human based heuristic for solving the rectangle packing problem”, *European Journal of Operational Research*, 141 (2002) 341–358.