

Packing a limited number of unequal circular objects in a rectangular container

Igor Litvinchev^{a,b}, Daniel Mosquera^b

^aComplex Systems Department, Computing Center,
Russian Academy of Sciences, Moscow, Russia.

^bFaculty of Mechanical and Electrical Engineering,
Nuevo Leon State University, Monterrey, Mexico.
Email: igorlitvinchev@gmail.com

Abstract. A problem of packing a limited number of unequal circular objects in a fixed size rectangular container is considered. The circular object is considered in a general sense, as a set of points that are all the same distance (not necessary Euclidean) from a center. The aim is to maximize the (weighted) number of objects placed into the container or minimize the waste. This problem has numerous applications in logistics, including production and packing for the textile, apparel, naval, automobile, aerospace and food industries. Frequently the problem is formulated as a nonconvex continuous optimization problem which is solved by heuristic techniques combined with local search procedures. We study a linear integer programming formulation based on approximating container by a regular grid. The nodes of the grid are considered as potential positions for assigning centers of the objects thus giving rise to a large scale linear 0-1 optimization problem with binary variables representing the assignment of centers to the nodes of the grid. Recursive packing allowing nesting circles inside one another is also considered. Numerical results are presented to demonstrate the efficiency of the proposed approach.

Keywords: circle packing, integer programming, large scale optimization.

1 Introduction

Packing a set of items of known dimensions into one or more large objects or containers to minimize a certain objective (e.g., the unused part of the objects or waste) constitutes a family of natural combinatorial optimization problems applied in logistics, computer science, industrial engineering, manufacturing and production processes, material science and nanotechnology, medicine [3, 6-8, 10, 30].

Packing problems for regular shapes (circles and rectangles) of objects and/or containers are well studied (see, e.g., a review [12] for circle packing). In circle packing problem the aim is to place a certain number of circles, each one with a fixed known radius inside a container. The circles must be totally placed in the container

without overlapping. The shape of the container may vary from a circle, a square, a rectangular, etc. Circle packing problems can be classified into two categories depending on whether the circle objects are equal [2, 5, 9, 22, 28] or unequal [1, 11, 13, 14, 16, 23-25, 27, 30, 32]. For packing unequal circles it is frequently assumed that exactly one circle of each radius is available [12]. However in many logistic applications [6] the number of circles packed has to be large enough to fulfil the demand while the supply of the circles may also be limited.

In this paper we study packing unequal circular-like objects in a rectangular container using a regular grid to approximate the container. It is assumed that (inside a given interval) multiple objects of the same radius can be packed. The circular-like object is considered in a general sense, as a set of points that are all the same distance (not necessary Euclidean) from a given point. Thus different shapes, such as ellipses, rhombuses, rectangles, octagons can be treated the same way by simply changing the norm used to define the distance. The nodes of the grid are considered as potential positions for assigning centers of the circles. The packing problem is then stated as a large scale linear 0-1 optimization problem. Valid inequalities are used to strengthening the original formulation. Numerical results on packing unequal circles, squares, rhombuses and regular octagons are presented to demonstrate efficiency of the proposed approach. Recursive packing, i.e. nesting circular objects inside one another is also considered. To the best of our knowledge, the idea to use a grid for approximated solution was first implemented by Beasley [4] in the context of cutting problems. Recently this approach was applied in [9, 17-21, 29] for packing problems. This work is a continuation of [20]. The rest of the paper is organized as follows. Section 2 presents the main constructions: basic binary model and valid inequalities. Results of a numerical experiment are presented in Section 3, while the last section concludes.

2 Basic model

Suppose we have non-identical circles C_k of known radius R_k , $k \in K = \{1, 2, \dots, K\}$. Here we consider the circle as a set of points that are all the same distance R_k (not necessary Euclidean) from a given point. In what follows we will use the same notation C_k for the figure bounded by the circle, $C_k = \{z \in \mathbb{R}^2 : \|z - z_{0k}\| \leq R_k\}$, assuming that it is easy to understand from the context whether we mean the curve or the figure. Suppose that at most M_k circles C_k are available for packing and at least m_k of them have to be packed. Denote by $i \in I = \{1, 2, \dots, n\}$ the node points of a regular grid covering the rectangular container. Let $F \subseteq I$ be the grid points lying on the boundary of the container. Denote by d_{ij} the distance (in the sense of norm used to define the circle) between points i and j of the grid. Define binary variables $x_i^k = 1$ if centre of a circle C_k is assigned to the point i ; $x_i^k = 0$ otherwise.

Since the circle C_k assigned to the point i has to be no-overlapping with other circles packed, then we have $x_j^l = 0$ for $j \in I, l \in K$, such that $d_{ij} < R_k + R_l$. For fixed i, k let $N_{ik} = \{j, l : i \neq j, d_{ij} < R_k + R_l\}$. Let n_{ik} be the cardinality of $N_{ik} : n_{ik} = |N_{ik}|$. Then the problem of optimal packing can be stated as follows:

$$z = \max \sum_{i \in I} \sum_{k \in K} S_k x_i^k \quad (1)$$

subject to

$$m_k \leq \sum_{i \in I} x_i^k \leq M_k, \quad k \in K, \quad (2)$$

$$\sum_{k \in K} x_i^k \leq 1, \quad i \in I \setminus F, \quad (3)$$

$$R_k x_i^k \leq \min_{j \in F} d_{ij}, \quad i \in I, k \in K, \quad (4)$$

$$x_i^k + x_j^l \leq 1, \text{ for } i \in I, k \in K, (j, l) \in N_{ik} \quad (5)$$

$$x_i^k \in \{0, 1\}, \quad i \in I, k \in K \quad (6)$$

Here S_k represent the importance of C_k in the packing pattern, e.g. S_k can be the area of C_k . Constraints (2) ensure that the number of circles packed is between m_k and M_k ; constraints (3) that at most one centre is assigned to any grid point; constraints (4) that the point i can not be a centre of the circle C_k if the distance from i to the boundary is less than R_k ; pair-wise constraints (5) guarantee that there is no overlapping between the circles; constraints (6) represent the binary nature of variables. More details on the problem (1)-(6) and its properties one can find in [18-20].

Note that all constructions proposed above remain valid for any norm used to define a circular-like object. In fact, changing the norm affects only the distance d_{ij} used in the definitions of the sets N_{ik} in the non-overlapping constraints (5). That is, by simple pre-processing we can use the basic model (1)-(6) for packing different geometrical objects of the same shape. It is important to note that the non-overlapping condition has the form $d_{ij} \geq R_k + R_l$ no matter which norm is used.

For example, a circular object in the maximum norm $\|z\|_\infty := \max_i \{|z_i|\}$ is represented by a square, taxicab norm $\|z\|_1 := \sum |z_i|$ yields a rhombus. In a similar way we may manage rectangles, ellipses, etc. Using a superposition of norms, we can consider more complex circular objects. For

$$\|z\| := \max_i \{|z_i|, \gamma \sum |z_i|\} \quad (7)$$

and a suitable $0.5 < \gamma < 1$ we get an octagon, an intersection of a square and a rhombus.

To tightening the LP-relaxation for (1)-(6), we consider valid inequalities aimed to ensure that no grid point is covered by two circles. Define matrix $\left[\alpha_{ij}^k \right]$ as follows. Let $\alpha_{ij}^k = 1$ for $d_{ij} < R_k$, $\alpha_{ij}^k = 0$ otherwise. By this definition, $\alpha_{ij}^k = 1$ if the circle C_k centred at i covers point j . The following constraints ensure that no points of the grid can be covered by two circles:

$$\sum_{k \in K} \sum_{j \in I} \alpha_{ij}^k x_j^k \leq 1, \quad i \in I. \quad (8)$$

We can treat (8) as a relaxed non-overlapping conditions and expect that refining the grid reduces overlapping. The valid inequalities (8) hold for any norm used to define the circular object. Numerical experiments presented in [18-20] demonstrate that aggregating valid inequalities (8) to the original problem (1)-(6) improves significantly the value of the corresponding LP-bound. Moreover, valid inequalities change the structure of the optimal LP-solution.

To consider nesting circular objects inside one another (recursive packing [25]), we only need to modify the non-overlapping constraints. In order to the circle C_k assigned to the point i be non-overlapping with other circles being packed (including circles places inside this circle), it is necessary that $x_j^l = 0$ for $j \in I, l \in K$, such the $R_k - R_l < d_{ij} < R_k + R_l$ for $R_k > R_l$. Let

$$\Omega_{ik} = \{j, l : i \neq j, R_k - R_l < d_{ij} < R_k + R_l, R_k > R_l\}.$$

Then the non-overlapping constraints for packing circular objects with nesting can be stated as

$$x_i^k + x_j^l \leq 1, \text{ for } i \in I; k \in K; (j, l) \in \Omega_{ik}. \quad (9)$$

Constraints (3) have to be omitted in the case of nesting.

3 Computational results

A numerical experiment was designed to evaluate the performance of the model (1)-(6) for packing different ($K \geq 1$) circular objects in a square rectangular container.

A rectangular uniform grid was used, such that all grid points are defined by the grid points on the edges of the square. Let L be a size of the square container, while N be a number of the equidistant grid points on its edge (and hence $n = N^2$). Similar to [23, 24, 26] the value of the radius was increased linearly with respect to the number of the circle. The following expression was used: $R_1 = L/6K$, $R_k = R_1(k - 0.8)$, $k = 2, 3 \dots K$.

For all problem instances at least one circle of each radius has to be packed ($m_k = 1$), while M_k were randomly generated integers, $M_k \in U[50, 200]$. For the size of the square container $L \in U[200, 300]$ was used. Four types of circular objects

were considered: circles, rhombuses, squares and octagons. For the octagon $\gamma = 1/\sqrt{2}$ was used in (7) giving the right octagon. We used $S_k = R_k^2$ for all problem instances and shapes.

Table 1. Packing circles

	1	2	3	4	5	6
L	242	243	275	229	263	238
N	35	27	35	34	35	30
K	13	13	13	12	14	13
\bar{z}	10450.4	14952.2	17970.5	12339.4	14333	14040
gap	90	12.8	28	34	53	17.7
\mathcal{X}	1(125/125) 2(193/193) 3(70/15) 4(52/2)	1(181/17), 2(82/29) 3(137/3), 4(199/4) 6(148/2), 7(85/2) 8(135/2), 10(149/3) 13(132/5)	1(69/39), 2(143/108) 3(79/29), 4(181/9) 5(124/22), 6(158/3) 7(51/2), 10(62/2) 11(105/3), 3(170/2)	1(185/162) 2(190/79) 3(55/38) 4(166/5) 8(165/6) 10(110/2)	1(95/95) 2(172/172) 3(109/70) 4(187/7)	1(92/15), 2(142/62) 3(176/13), 4(81/3) 5(99/5), 6(149/4) 8(175/3), 10(114/2) 11(195/3), 13(109/2)
\bar{z}	19185.8	20896.6	26402.4	15873.5	24334.2	25060.7
\mathcal{X}	1(125/125), 2(193/193) 3(70/70) 4(52/25) 5(114/11) 6(134/4) 7(52/3) 9(109/3)	1(181/181), 2(83/83) 3(137/99), 4(199/28) 5(165/5), 6(148/2) 7(85/3), 8(135/2) 9(109/3), 11(107/2) 12(196/2)	1(69/69), 2(143/143) 3(79/79), 4(181/38) 5(124/20), 6(158/4) 7(51/3), 8(192/2) 10(62/4), 12(106/2) 13(170/3)	1(185/185) 2(190/190) 3(55/55) 4(166/22) 5(174/3) 6(62/4) 7(51/3) 11(136/2)	1(95/95) 2(172/172) 3(109/109) 4(187/24) 5(107/8) 6(187/4) 7(187/5) 8(104/3) 9(128/4) 10(105/2)	1(92/92), 2(142/142) 3(176/91), 4(81/21) 5(99/5), 6(149/6) 7(76/2), 8(175/5) 10(114/2), 11(195/3) 12(124/2), 13(109/3)

Six problem instances for each circular object were generated and solved using valid inequalities (8) for non-recursive packing. All optimization problems were solved by the system CPLEX 12.61 [15]. The runs were executed on an Intel (R) Xeon (R) CPU E3-1245 v3 @ 3.4GHz with 8 cores and 16Gb RAM. The computation was interrupted after the computational time exceeded 10 hours CPU time.

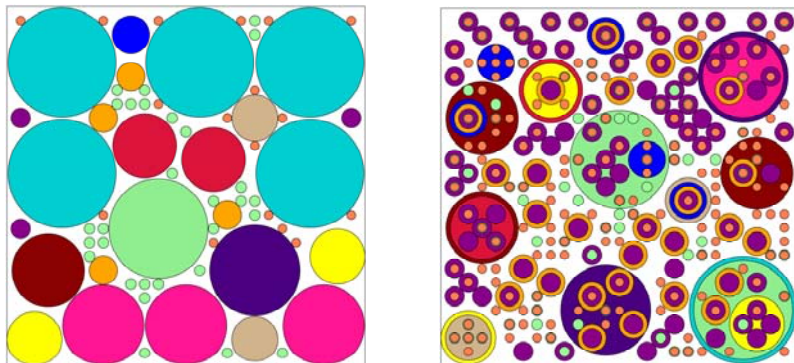


Fig. 1. Instance 2

The results of the numerical experiment are presented in Tables 1-4. Here lines 5 and 6 represent the best objective value z and the corresponding $mipgap$ in % obtained by CPLEX. The line 7 gives the best primal solution x in the form $k(M_k/x_k)$ where x_k denotes the number of the k -objects packed. For example, $3(70/15)$ means that from 70 objects C_3 available 15 objects were packed. The objects with $x_k = m_k = 1$ are omitted. The last two lines present the best objective value and the best primal solution for the case of recursive packing. For recursive packing the $mipgap$ is omitted since it was higher than 200% for all shapes and problem instances. Figures 1-4 provide examples of packing patterns.

Table2 . Packing rhombuses

	1	2	3	4	5	6
L	269	288	261	239	215	219
N	29	28	34	27	30	34
K	14	12	14	14	13	12
z	31564.4	33878.4	28795.7	23685	19662.4	19199.9
gap	5	5.9	13.3	1.7	4.1	19.3
x	3(120/5),4(127/127) 6(72/2),7(75/13) 9(83/2),10(162/5) 13(125/4),14(188/1)	2(126/7) 3(57/32) 6(77/46) 11(80/3)	1(185/3) 2(175/2) 3(174/58) 5(70/3) 6(99/76)	3(141/2) 4(142/5) 5(67/60) 10(123/13) 14(83/18)	2(134/3) 3(83/7) 4(51/42) 7(97/2) 8(151/49) 12(122/11)	1(189/16),2(193/38) 3(79/79),4(51/5) 5(147/43),6(159/2) 7(123/4),11(96/2) 12(162/2)
z	35730.4	49413.1	44888.8	28437.8	30501.9	28986.1
x	1(132/132),2(74/74) 3(120/120),4(127/127) 5(171/23),6(72/14) 7(75/4),8(108/2) 11(50/2)	1(151/151) 2(126/126) 3(57/57) 4(131/23) 5(113/28) 6(77/19) 7(164/8) 8(158/4) 9(117/2) 10(176/3) 11(80/2)	1(185/185) 2(175/175) 3(174/174) 4(139/29) 5(70/46) 6(99/31) 7(165/11) 8(130/40) 9(57/3) 10(103/2) 11(127/3)	1(97/97),2(68/68) 3(141/141),4(142/142) 5(67/67),6(57/29) 7(100/9),8(102/8) 9(166/6),10(123/2) 11(155/3),12(123/2)	1(199/199) 2(134/134) 3(83/83) 4(51/51) 5(68/58) 6(105/45) 7(97/42) 8(151/18) 9(165/6) 10(104/4) 11(118/2) 13(199/3)	1(189/189),2(193/193) 3(79/79),4(51/49) 5(147/20),6(159/6) 7(123/7),8(195/2) 9(108/4),10(152/3) 11(96/2),12(162/2)

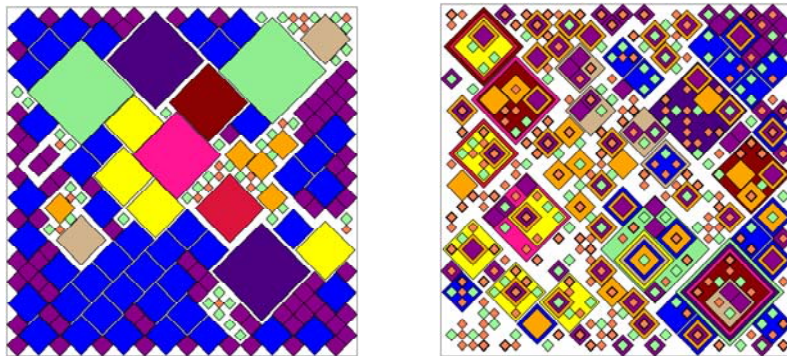


Fig. 2. Instance 6

As we can see from these tables, larger values of the *mipgap* correspond to “roundish” objects (circles and octagons) while for “angular” objects (squares and rhombuses) the *mipgap* is much smaller.

Table 3. Packing squares

	1	2	3	4	5	6
L	227	219	211	215	289	282
N	29	30	32	35	34	31
K	12	12	12	14	12	12
Z	11628	10542.2	10171	10326.7	18216.3	14040
<i>Gap</i>	7.7	6.3	6.9	12.7	11.6	17.7
x	2(181/12) 3(198/8) 4(75/2) 6(129/18) 9(115/3) 11(56/2)	1(132/2) 2(51/51) 3(59/6) 4(159/2) 7(165/5) 12(88/4)	1(51/13),2(156/9) 3(191/3),4(163/3) 6(179/3),9(82/2) 10(62/3),11(198/17)	2(69/35),3(175/4) 4(94/9),7(80/31) 9(161/2),10(124/2) 12(137/2),13(116/2)	1(100/15) 2(67/13) 5(90/3) 7(176/4) 8(94/3) 11(52/3) 12(184/2)	1(189/6) 2(82/5) 3(97/4) 4(149/5) 6(200/2) 7(97/34) 10(68/3)
Z	16693.9	15394	13974.4	14852.2	30162.6	26149.3
x	1(107/107) 2(181/181) 3(198/65) 4(75/13) 5(173/12) 6(129/6) 7(63/2) 8(154/4)	1(132/132) 2(51/51) 3(59/56) 4(159/10) 5(112/9) 6(104/9) 7(165/5) 11(88/2)	1(51/51),2(156/156) 3(191/145),4(163/62) 5(165/18),6(179/11) 7(89/5),8(73/4) 9(82/2),10(62/2) 11(198/3),12(116/2)	1(184/184),2(69/69) 3(175/98),4(94/37) 5(65/27),6(50/15) 7(80/11),8(160/3) 9(161/4),10(124/2) 12(137/2)	1(100/99), 2(67/67) 3(178/41) 4(142/24) 5(90/14) 6(101/2) 7(176/8) 8(94/2) 9(122/2) 10(153/2)	1(189/189),2(82/82) 3(97/60),4(149/23) 5(115/26),6(200/17) 7(97/9),8(165/2) 9(116/3),10(68/2) 11(64/2)

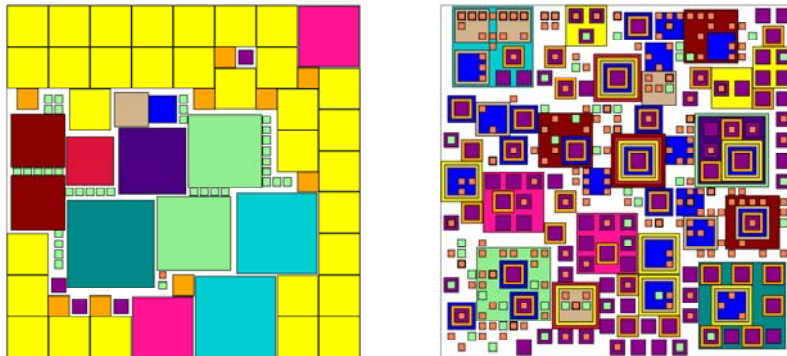


Fig. 3. Instance 4

We also note that although the computations were interrupted after 10 hours CPU time, both primal solution and *mipgap* remain almost unchanged after the first hour of computations. This effect was observed for all problem instances and shapes including recursive packing. Analyzing the final packing patterns we may conclude that for circles and octagons the small radii are used almost completely (close to the upper limit M_k), while for squares and rhombuses the distribution of radii used in the final

packing is more uniform. The patterns for recursive packing seem to be sparse, however note that the smallest objects were completely used, $\sum_i x_i^k = M_k$, $k = 1, 2$.

Table 4. Packing octagons

	1	2	3	4	5	6
L	275	262	262	256	214	202
N	32	34	35	33	34	27
K	13	14	14	14	12	12
Z	17508.2	16668.4	15575.8	14663.2	9938.24	9892.6
gap	23.1	23.6	35.9	34.9	49	16.6
x	1(134/28),2(79/71) 3(131/23),4(140/2) 5(71/2),6(187/5) 7(87/3),8(67/8) 9(190/4),12(81/3)	1(106/32),2(156/50) 3(61/24),4(163/6) 5(97/9),6(142/3) 7(69/2),9(119/3) 10(78/2),11(63/2) 14(176/2)	1(71/6) 2(196/99) 3(83/45) 4(173/35) 5(159/2) 13(149/2)	1(187/29) 2(179/150) 3(123/46) 4(89/4) 5(112/4) 6(141/3) 14(184/2)	1(64/64) 2(188/188) 3(105/69) 4(153/110) 6(142/3) 8(159/2) 9(112/2)	1(60/8) 2(173/21) 3(71/32) 4(81/12) 5(66/5) 6(53/5) 8(128/3) 11(141/2) 12(173/11)
Z	23891	27421.9	21323.9	30902.4	14899.7	12141.8
x	1(134/134),2(79/79) 3(131/101),4(140/28) 5(71/17),6(187/10) 7(87/3),8(67/3) 9(190/3),12(81/2)	1(106/106),2(156/156) 3(61/61),4(163/46) 5(97/15),6(142/7) 7(69/3),8(168/4) 10(78/3),12(128/2) 14(176/2)	1(71/71) 2(196/196) 3(83/83) 4(173/25) 5(159/11) 6(82/5) 7(162/4) 8(170/2) 9(119/2)	1(187/187),2(179/179) 3(123/81),4(89/23) 5(112/8),6(141/7) 7(71/4),8(109/4) 9(190/3),11(152/2) 12(170/3),13(190/2) 14(184/3)	1(64/64), 2(188/188) 3(105/105) 4(153/97) 5(53/24) 6(142/13) 7(57/4) 8(159/2) 9(112/4) 10(146/3) 11(105/2)	1(60/60) 2(173/173) 3(71/71) 4(81/81) 5(66/31) 6(53/5) 7(90/6) 8(128/5)

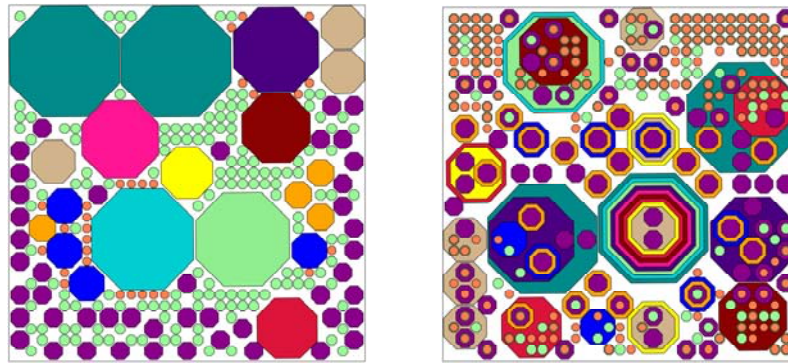


Fig. 4. Instance 4

1 Conclusions

We presented numerical experiment with integer model for approximate packing unequal circular objects. A formulation permitting nesting circles inside one another (recursive packing) was also considered and numerically evaluated. Recursive packing was mentioned in [8, 10] in the context of packing pipes of different diameters into a shipping container and has not received much attention so far [25]. Two types of circular objects were studied – “roundish” (circle and right octagon) and “angular” (square and rhombus). The computational study indicates that (inside 10 hours of CPU time limit) the best primal solution as well as the corresponding *mipgap* remain almost unchanged after the first hour of computation. This may be caused by a low quality of dual bounds generated by the commercial software. An interesting direction for the future research is to study the use of Lagrangian relaxation and corresponding heuristics [31] to cope with large dimension of arising problems and to improve the dual bounds. Some complements in this direction are in course. This work was partially supported by CONACYT, Mexico (167019).

References

1. Addis, B., Locatelli, M., Schoen, F.: Efficiently packing unequal disks in a circle: a computational approach which exploits the continuous and combinatorial structure of the problem. *Operations Research Letters*. 36, 37--42 (2008).
2. Akeb, H., Hifi, M.: Solving the circular open dimension problem using separate beams and look-ahead strategies. *Computers & Operations Research*. 40, 1243--1255 (2013).
3. Baltacioglu, E., Moore, J.T., Hill R.R.: The distributor's three-dimensional pallet-packing problem: a human-based heuristical approach", *International Journal of Operations Research*.1, 249--266 (2006).
4. Beasley, J. E.: An exact two-dimensional non-guillotine cutting tree search procedure. *Operations Research*. 33, 49--64 (1985).
5. Birgin, E.G., Gentil, J.M.: New and improved results for packing identical unitary radius circles within triangles, rectangles and strips", *Computers & Operations Research*. 37, 1318—1327 (2010).
6. Bortfeldt, A., Wäscher, G.: Constraints in container loading—A state-of-the-art review. *European Journal of Operational Research*. 229, 1--20 (2013).
7. Castillo, I., Kampas, F.J., Pinter, J.D.: Solving circle packing problems by global optimization: Numerical results and industrial applications, *European Journal of Operational Research*.191, pp. 786--802 (2008).
8. Frazer, H.J., George, J.A.: Integrated container loading software for pulp and paper industry. *European Journal of Operational Research*. 77, 466--474. (1994).
9. Galiev, S.I., Lisafina. M.S.: Linear models for the approximate solution of the problem of packing equal circles into a given domain. *European Journal of Operational Research*. 230, 505--514 (2013).
10. George, J.A.: Multiple container packing: a case study of pipe packing, *Journal of the Operational Research Society*. 47, 1098--1109 (1996).
11. He, K., Huang, M., Yang, C.: An action-space-based global optimization algorithm for packing circles into a square container. *Computers & Operations Research*. 58, 67--74 (2015).

12. Hifi, M., M'Hallah, R.: A literature review on circle and sphere packing problems: models and methodologies, *Advances in Operations Research*. 22, 1--23 (2009) doi:10.1155/2009/150624.
13. Huang, W.Q., Li, Y., Li, C.M., Xu, R.C.: New heuristics for packing unequal circles into a circular container. *Computers & Operations Research*. 33, 2125--2142 (2006).
14. Huang, W. Q., Li, Y., Akeb, H., Li, C. M.: Greedy algorithms for packing unequal circles into a rectangular container", *The Journal of the Operational Research Society*. 56, 539--548 (2005).
15. ILOG CPLEX, *Mathematical programming optimizers*. Version 12.6 (2013).
16. Kubach, T., Bortfeldt, A., Gehring, H.: Parallel greedy algorithms for packing unequal circles into a strip or a rectangle. *Central European Journal of Operations Research*. 17, 461--477 (2009).
17. Litvinche, I., Ozuna, L.: Approximate packing circles in a rectangular container: valid inequalities and nesting. *Journal of Applied Research and Technologies*. 12, 716-723 (2014).
18. Litvinchev, I., Infante, L., Ozuna, L.: Approximate circle packing in a rectangular container: integer programming formulations and valid inequalities. In: R.G. Gonzalez-Ramirez et al. (eds.), *ICCL 2014, LNCS*, vol. 8760, pp. 47--61, Springer, Berlin-Heidelberg, (2014).
19. Litvinchev, I., Infante, L., Ozuna, L.: Approximate packing: integer programming models, valid inequalities and nesting, in: G. Fasano, J.A. Pinter (eds.) *Optimized Packings with Applications Springer Optimization and Its Applications*, vol. 105, 117-135. Springer-Verlag, (2015)
20. Litvinchev, I., Infante, L., Ozuna, L.: Packing circular-like objects in a rectangular container. *Journal of Computer and Systems Sciences International*. 54, 259--267 (2015).
21. Litvinchev, I., Infante, L., Ozuna, L.: Using valid inequalities and different grids in LP-based heuristic for packing circular objects. In: N.T. Nguyen et al. (eds.), *ACIIDS 2016 LNCS*, vol. 9622, pp. 681--690, Springer, Berlin-Heidelberg, (2016).
22. Locatelli, M., Raber, U.: Packing equal circles in a square: a deterministic global optimization approach: *Discrete Applied Mathematics*. 122, 139-166 (2002).
23. Lopez, C.O., Beasley, J. E.: Packing unequal circles using formulation space search. *Computers & Operations Research*. 40, 1276--1288 (2013).
24. Lopez, C.O., Beasley, J.E.: A formulation space search heuristic for packing unequal circles in a fixed size circular container", *European Journal of Operational Research*. 251, 64--73 (2016).
25. Pedroso, J.P., Cunha, S., Tavares, J.N., Recursive circle packing problems, *International Transactions in Operational Research*. 23, 355--368 (2016).
26. Specht, E.: Packomania website (2016), www.packomania.com
27. Stoyan, Y.G., Yaskov, G.N.: A mathematical model and a solution method for the problem of placing various-sized circles into a strip. *European Journal of Operational Research*. 156, 590-600 (2004).
28. Szabó, P.G., Markót, M.C., Csendes, T., Specht, E., Casado, L.G., García, I.: *New Approaches to Circle Packing in a Square: With Program Codes*, (Springer Optimization and Its Applications.6, Springer-Verlag, (2007).
29. Toledo, F.M.B., Carravilla, M.A., Ribero, C., Oliveira, J.F., Gomes, A.M.: The dotted-board model: a new MIP model for nesting irregular shapes, *International Journal of Production Economics* 145, 478--487 (2013).
30. Wang, J.: Packing of unequal spheres and automated radiosurgical treatment planning. *Journal of Combinatorial Optimization*. 3 453--463 (1999).
31. Wolsey, L.A.: *Integer programming*. Wiley, New York, (1999).
32. Zeng, Z., Yu, X., He, K., Huang, W., Fu, Z.: Iterated Tabu Search and Variable Neighborhood Descent for packing unequal circles into a circular container. *European Journal of Operational Research*. 250, 615--627 (2016).