Investigation of the Possibilities of Optimizing the Model of Potential Containers to Increase the Speed of Placement of Orthogonal Polyhedra

Vladislav Chekanin [1,2], Alexander Chekanin [1]

[1] Moscow State University of Technology «STANKIN», 3a Vadkovsky lane, Moscow, 127055, Russia
[2] V.A. Trapeznikov Institute of Control Sciences of Russian Academy of Sciences (ICS RAS), 65 Profsoyuznaya street, Moscow, 117997, Russia

Abstract
An optimization problem of packing objects of arbitrary geometry with generalization in dimension is considered. It is proposed to use a discrete representation of objects of complex shape in the form of orthogonal polyhedra, which are compound objects obtained by combining rectangles or parallelepipeds depending on the dimension of the problem. The model of potential containers is used to form and describe the placement schemes of orthogonal polyhedra. The paper proposes algorithms that provide a qualitative increase in the speed of formation of the placement schemes by reducing the number of potential containers processed when placing each compound object. A fast algorithm for updating sets of potential containers is presented, which is based on the use of the set-theoretic operation of intersection. An additional increase in the speed of the potential containers model is achieved by removing potential containers that cannot be used to place new objects in all possible orientations. It is shown that with an increase in the number of objects placed using the proposed algorithms, the time spent on placing one object is reduced. The proposed optimization makes it possible to solve the problems of placing objects of complex shape, specified with a higher degree of detail, which will provide a denser packing in the allotted time. The results of the computational experiments carried out on the problems of packing flat and volumetric objects of irregular shape are presented, confirming the effectiveness of the developed algorithms.

Keywords
Packing problem, layout, orthogonal polyhedron, model of potential containers, optimization.

1. Introduction

The problems of packing objects of arbitrary shape are extremely complex and require significant computational resources to solve them, which is explained both by the fact that these problems belong to the class of NP-hard optimization problems [1, 2] and the need for a detailed description of the geometry of the placed objects. The solution of a large number of optimization problems of resource allocation of various geometry is reduced to packing problems, including: problems of irregular cutting of materials (metal sheets, plywood, cardboard, etc.) [2–4], layout problems (planning free spaces in workshops, rockets, airplanes, ships, tankers, etc.) [5, 6], modeling of microstructure composite materials [7, 8], geometric surface covering [9], placement of parts on the 3D printer platform [10–12].

The considered problem is to find the most rational way to place a given set of objects of arbitrary geometry inside a container. An indicator of the quality of the placement of objects is the height of the resulting placement scheme, which should be minimized. A solution of the problem can be presented in the form of a sequence of objects selected for their subsequent placement in a container with information about the orientation variant of each object. Such a sequence can be obtained using heuristic or metaheuristic optimization methods that reduce the number of considered alternative solutions to the

GraphiCon 2022: 32nd International Conference on Computer Graphics and Vision, September 19-22, 2022, Ryazan State Radio Engineering University named after V.F. Utkin, Ryazan, Russia
EMAIL: vladchekanin@rambler.ru (V. Chekanin); av.chekanin@stankin.ru (A. Chekanin)
ORCID: 0000-0001-9448-0583 (V. Chekanin); 0000-0003-0788-469X (A. Chekanin)

problem [13–15]. The final placement of objects in a container based on a given sequence is carried out using algorithms for geometric design of the packing.

The most common method of forming the placement of objects of complex shape is the method based on the construction of a hodograph vector function of dense placement [12, 16]. To obtain a locally optimal solution to the problems of packing some basic geometric shapes (for example, rectangles, circles, ellipses, parallelepipeds, spheres, etc.), the method of phi-functions and quasi-phi-functions developed by Professor Yu.G. Stoyan is used [17–19]. Despite the undoubted obvious advantages of these methods, associated primarily with the possibility of obtaining a layout of arbitrarily oriented objects, they have a number of limitations on its practical application. On the one hand, when placing an object of arbitrary geometry, it is necessary to first decompose this object into a set of simple shapes or develop a new mathematical model for its description with piecewise smooth functions (as is done, in particular, for ellipsoids and spheroconuses). For non-convex polyhedra, there is no possibility of constructing phi-functions, so there is a need to decompose them into a set of convex polyhedra. Obviously, with an increase in the degree of detail of the geometric description of the placed object, the complexity of the mathematical model being formed increases. Another disadvantage is associated with the need to use time-consuming methods of nonlinear programming to solve the problem [18, 20]. These problems occur when placing objects are given as polygonal models.

The discrete (voxel-based) representation makes it possible to describe objects of arbitrary geometry with a given degree of detail [7, 8, 21, 22]. The articles [23, 24] present a method of rapid placement of objects of complex shape, based on their representation in the form of orthogonal polyhedra (OP) [25, 26], obtained as a result of rasterization (voxelization) and subsequent decomposition of objects [27]. To describe a packing (placement scheme) and analyze the free spaces inside a container, the developed model of potential containers is used [15, 28].

Each OP O in a D-dimensional space consists of m orthogonal objects (parallelepipeds of dimension D) with a fixed position relative to each other. An orthogonal object with a number k is defined by a vector $\left\{w_k^1; w_k^2; \ldots; w_k^D\right\}$ of overall dimensions (the superscript in formulas is used to denote the number of the coordinate axis) and a vector $\left\{z_k^1; z_k^2; \ldots; z_k^D\right\}$ that determines its position in the local coordinate system of this OP. The classical orthogonal packing problem is a special case of the OP packing problem under consideration, in which each OP consists of only one object [29].

Computational experiments were carried out using the author's application software «Packer» [15], designed to solve practical problems of optimizing the allocation of resources of arbitrary shape and various dimensions (a personal computer based on an Intel Core i5-8400 processor with a frequency of 2.8 GHz and having 8 GB RAM was used).

2. Optimization the model of potential containers

2.1. Optimizing the process of updating a set of potential containers

The model of potential containers fully describes the entire free space of a D-dimensional container with a set of various virtual objects in the form of D-dimensional parallelepipeds, called potential containers (PC), which have the maximum possible overall dimensions. Each PC does not overlap any of the objects placed in the container and is within the boundaries of the container. We will denote by $\left\{p_i^1; p_i^2; \ldots; p_i^D\right\}$ the overall dimensions of the PC with the number i. The position of the PC inside the container will be determined by a vector $\left\{x_i^1; x_i^2; \ldots; x_i^D\right\}$ that contains the coordinates of the PC point that is closest to the origin of the container coordinates.

When trying to place one object inside the container, it is necessary to check the correctness of this placement. The model of potential containers guarantees the correct formation of the placement scheme when each placed orthogonal object is within the boundaries of the potential container selected for it. In this case, the time-consuming operation of detection possible collisions is reduced to a simple check of the placement of the considered object entirely inside one potential container, as a result of which the speed of geometric design of this packing increases.

In order to correctly describe all the free spaces inside the packing after placing each OP, it is necessary to perform the procedure for updating the set of PCs in the placement area. This procedure solves the problem of obtaining a new set of PCs of minimum cardinality, which describes all the existing free areas inside the container, provided that none of the PCs from this set even partially overlaps any of the placed objects. It is obvious that with an increase in the number of orthogonal objects that are part of the placed OP, the time spent on updating a set of PCs will increase, therefore, for the problems of packing objects of complex geometric shape, it is necessary to use the fastest algorithm for updating a set of PCs.

Two algorithms have been developed to update a set of PCs when placing an OP:

- sequential PC update algorithm;
- PC update algorithm based on the application of the intersection operation.

Figure 1 shows a block diagram of the sequential PC update algorithm, which updates the parameters of the overlapping PCs after placing each orthogonal object that is part of the placed OP.

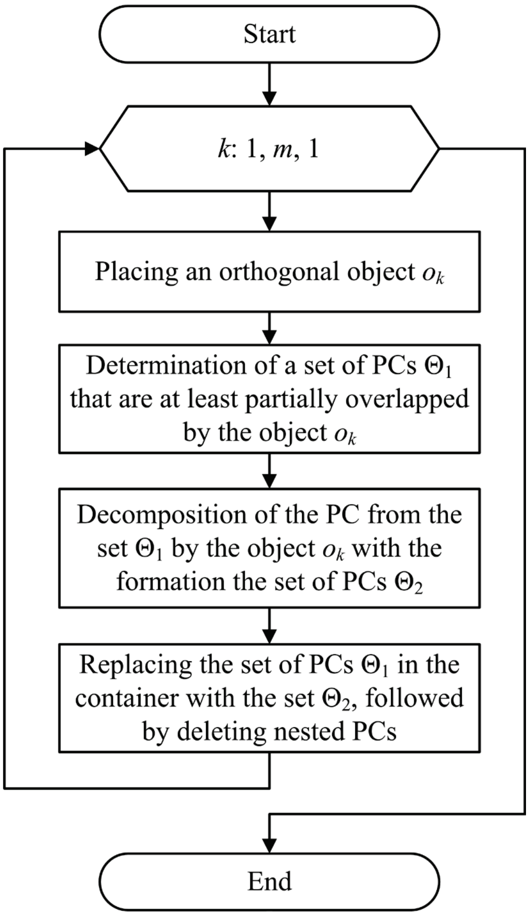


Figure 1: Updating a set of PCs with sequential placement of all orthogonal objects forming an OP

When creating a set of PCs Θ_1, the condition of its overlap with an orthogonal object o_k must be met for each PC $P_i \in \Theta_1$: $\left(x_i^d < X^d + z_k^d + w_k^d\right) \wedge \left(x_i^d + p_i^d > X^d + z_k^d\right) \quad \forall d \in \{1, \dots, D\}$, where $\left\{X^1, X^2, \dots, X^D\right\}$ are the coordinates of the placement point of the origin of the coordinates of the OP.

The set Θ_2 will contain PCs obtained as a result of decomposition by an orthogonal object o_k of all PCs $P_i \in \Theta_1$:

1. PCs formed when the overlap condition $\left(X^d + z_k^d > x_i^d\right) \wedge \left(X^d + z_k^d < x_i^d + p_i^d\right)\ \forall d \in \{1,\ldots,D\}$ is met, which are located at the point $\left\{x_i^1, x_i^2,\ldots,x_i^D\right\}$ of the container and have overall dimensions $\left\{p_i^1, p_i^2,\ldots,p_i^{d-1}, X^d + z_k^d - x_i^d, p_i^{d+1},\ldots,p_i^D\right\}$;
2. PCs formed when the overlap condition $\left(X^d + z_k^d + w_k^d > x_i^d\right) \wedge \left(X^d + z_k^d + w_k^d < x_i^d + p_i^d\right)$ $\forall d \in \{1,\ldots,D\}$ is met, which are located at points $\left\{x_i^1, x_i^2,\ldots,x_i^{d-1}, X^d + z_k^d + w_k^d, x_i^{d+1},\ldots,x_i^D\right\}$ and have overall dimensions $\left\{p_i^1, p_i^2,\ldots,p_i^{d-1}, x_i^d + p_i^d - X^d - z_k^d - w_k^d, p_i^{d+1},\ldots,p_i^D\right\}$.

When decomposing PCs that overlap with an orthogonal object, new PCs are formed which can be entirely located inside other PCs. As a result, the subsequent placing of other objects leads to a sharp increase in the number of PCs and a corresponding increase in the time spent on the formation of the packing. Therefore, in order to obtain a set of PCs of minimum cardinality, it is necessary to search and delete PCs nested in each other entirely after placing each orthogonal object.

The PC P_1 is nested in the PC P_2 (i.e. it is contained entirely inside the PC P_2) if the condition is met:

$$\left(x_1^d \ge x_2^d\right) \wedge \left(x_1^d + p_1^d \le x_2^d + p_2^d\right)\ \ \forall d \in \{1,\ldots,D\}. \tag{1}$$

At the first stage of searching the nested PCs, a set Θ' is formed consisting of PCs $P_i \in \Theta'$ for which $\exists d \in \{1,\ldots,D\} : x_i^d \le X^d + z_k^d + w_k^d$. Next, for each pair of PCs $P_1, P_2 \in \Theta', P_1 \ne P_2$, the condition (1) is checked, when fulfilled, the PC P_1 is removed from the set used to describe the free space of the packing.

The second developed algorithm for updating a set of PCs solves the problem of determining the areas that will remain unchanged in the container after placing the OP. The algorithm is based on the use of the intersection operation applied to the original set of PCs Ω_0 and the set of PCs formed in an empty container after placing the considered OP in it. This algorithm for OP O includes steps 1–4.

Step 1. Form a set $\Omega_0' \subset \Omega_0$ consisting of PCs for which $\exists d \in \{1,\ldots,D\} : x_i^d \le X^d + S^d$, where $S^d = \max\left(z_k^d + w_k^d\right),\ k = 1\ldots|O|$.

Step 2. Place the OP O at the point $\left\{X^1, X^2,\ldots,X^D\right\}$ of a new empty container (identical in shape and overall dimensions to the container used in the problem being solved), as a result of which a set Ω_1 of PCs will be formed in it. When placing OP, the algorithm described above for updating a set of PCs is used (based on the sequential placement of all orthogonal objects).

Step 3. Apply the intersection operation to the sets of PCs Ω_0' and Ω_1, as a result, a set $\Omega_0'' := \Omega_0' \cap \Omega_1$ describing the free space of the considered container in the neighborhood of the placed OP will be obtained. When performing the intersection operation, each set of PCs is considered as an OP consisting of orthogonal objects whose parameters coincide with the parameters of the corresponding PCs.

Step 4. Replace all PCs in set Ω_0 from set Ω_0' with PCs from the resulting set Ω_0''.

The scheme of updating a set of PCs based on the application of the intersection operation is shown in Figure 2.

A comparative analysis of the efficiency of the developed algorithms for updating PC sets was carried out on the example of solving the problem of placing a set of identical OP having external dimensions of the bounding parallelepiped AABB $\{62, 42, 24\}$ and consisting of 66 orthogonal objects (Figure 3, a). The container is represented by a parallelepiped with the overall dimensions $\{340, 340, 620\}$. Rotation of OP in the problem was not allowed. Figure 3, b shows an example of the resulting packing of a set of 250 OP.

(a) (b) (c) (d)

Figure 2: Updating a set of PCs based on the application of the intersection operation: (a) set of PCs Ω'_0 in the original container before placing the OP; (b) set of PCs Ω_1 in a new container after placing the OP; (c) illustration of the intersection operation; (d) set of PCs Ω''_0 after placing the OP

(a) (b)

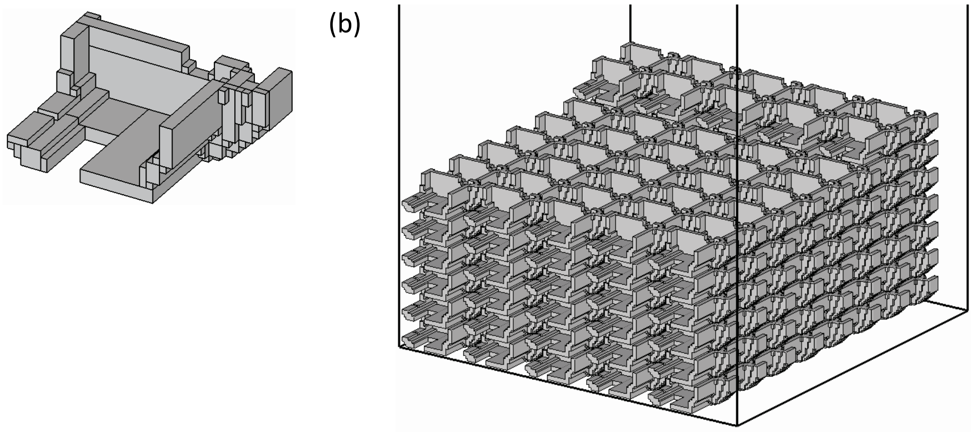


Figure 3: The test problem of packing a set of OP: (a) type of the placed OP; (b) result of placing a set of 250 OP

Summary results of solving the problem of placing sets of 50, 100, 150, 200 and 250 OP without a gap are presented in Table 1, which uses the following notation: T – the time spent on placing all OP; t – the average time spent on placing one OP, which is determined by the formula $t = T/N$, where N is the total number of all OP in the problem being solved. A computational experiment has shown that an algorithm based on the intersection of sets of PCs provides a qualitatively higher speed of placement of OP in comparison with the algorithm of sequential updating of PCs.

Table 1
Testing algorithms for updating PC sets

Number of OP in the problem	Number of PC after placing all OP	Sequential update algorithm		Algorithm based on the intersection operation	
		T, s	t, s	T, s	t, s
50	4369	18.3	0.366	7.7	0.154
100	9221	119.9	1.199	33.1	0.331
150	14073	416.0	2.773	85.2	0.568
200	18889	1082.6	5.413	177.6	0.888
250	23759	1800.2	7.201	274.5	1.098

2.2. Removing unused potential containers

An additional increase in the speed of the model of potential containers is achieved by removing PCs that cannot be used to place any of the specified objects in any of their possible orientations.

In the Table 2 the indicators of the speed of placement of sets of OP (Figure 3, a) when removing unused PCs and the use of an algorithm for updating a set of PCs based on the intersection operation are summarized. The last column of the table contains a parameter K showing how many times the speed of packing construction increases when unused PCs are removed.

Table 2

The speed of placement of OP sets when removing unused PCs

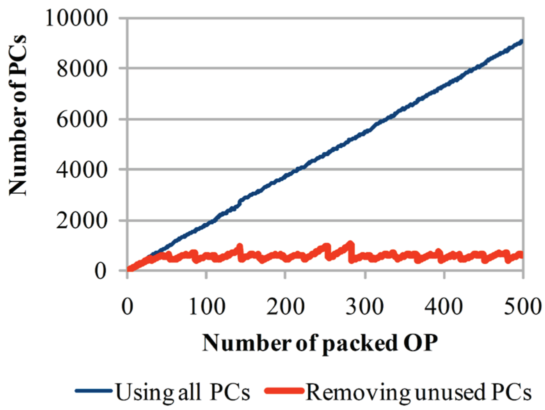
Number of OP in the problem	Number of PC after placing all OP	T, s	t, s	Increasing the speed of object placement (K)
50	3430	5.7	0.114	1.351
100	4444	21.8	0.218	1.518
150	5450	39.8	0.265	2.143
200	2559	57.8	0.289	3.073
250	3438	72.0	0.288	3.813

The obtained test results demonstrate a significant increase in the speed of object placement due to a fundamental reduction in the number of PCs processed. It is important to note that when removing unused PCs, the value of the object placement speed parameter K increases.

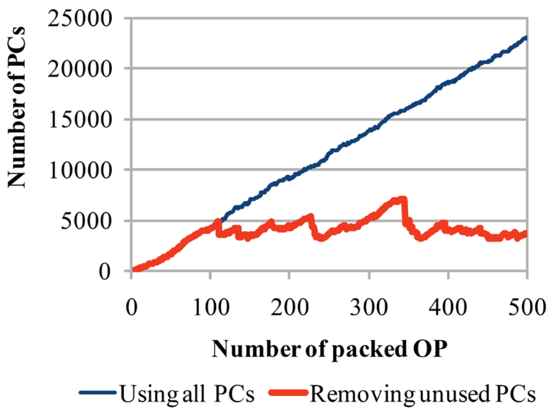
Figure 4 presents diagrams showing the dynamics of changes in the number of PCs in a container when solving problems of packing sets from 1 to 500 objects of the same type, shown in Figure 3, a (when placing objects, a gap of 10 was set):

- problem 1: packing of objects without rotation (Figure 5, a);
- problem 2: packing of objects with sequential assignment of one of six orientation options for each OP (Figure 5, b).

Figure 6 shows the dynamics of changes in the number of PCs and the speed of solving the packing problem when placing sets from 1 to 100 flat (two-dimensional) OP of two types without a gap. The formed placement scheme of a set of 100 OP and the free space of the container for the resulting packing are shown in Figure 7.

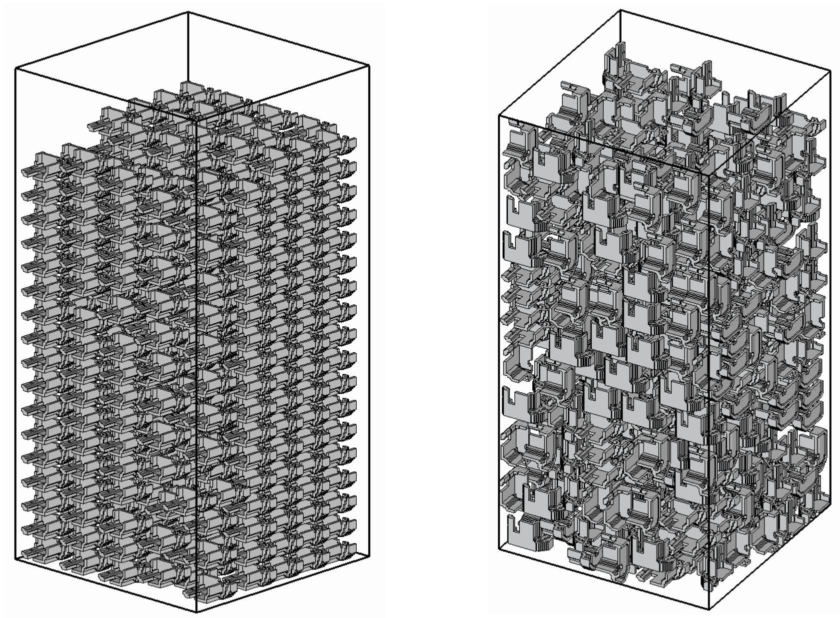


(a)



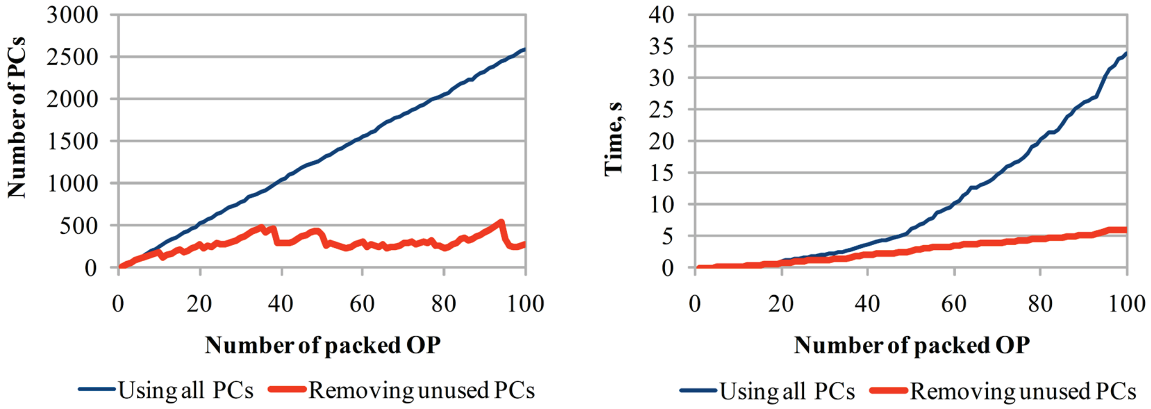
(b)

Figure 4: Dynamics of changes in the number of PCs when placing three-dimensional objects of complex shape: (a) problem 1; (b) problem 2



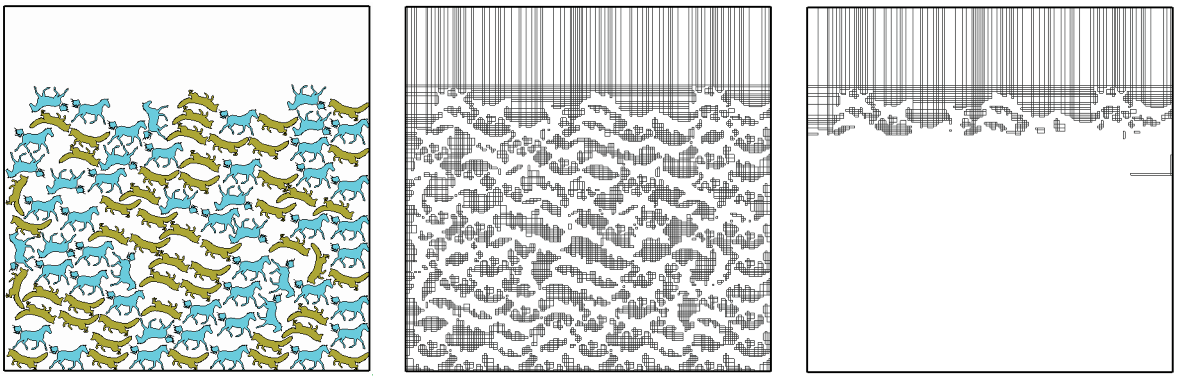
(a) (b)

Figure 5: Test problems of packing three-dimensional objects of complex shape: (a) problem 1; (b) problem 2



(a) (b)

Figure 6: Results of testing the model of potential containers on flat OP: (a) dynamics of change in the number of PCs; (b) dynamics of changes in the time spent on solving the problem



(a) (b) (c)

Figure 7: Packing of flat OP: (a) solution; (b) set of PCs in a packing; (c) set of PCs in a packing when removing unused PCs

3. Conclusion

Among the advantages of the method of solving the problems of packing objects in the form of an OP in comparison with the traditional methods of forming irregular packing of vector or polygonal objects, it should be noted the possibility of specifying arbitrary geometry of objects and containers to be placed, a complete analysis of all free areas inside the formed placement scheme, as well as the ability to control the speed and accuracy of the solution by changing the degree of detail of objects during their rasterization (voxelization) at the preprocessor stage of preparing the solution of the problem (this issue is considered in [30]).

The conducted studies have shown that both for updating sets of PCs of arbitrary dimension and for placing OP (this issue is discussed in the article [25]), algorithms based on the application of the set-theoretic intersection operation are the most effective, therefore they were chosen as the basic ones when implementing the irregular-shaped object packing module included in the developed application software «Packer».

The algorithms proposed in this paper make it possible to qualitatively increase the speed of geometric design of the placement schemes based on a certain given sequence of selecting objects for their placement in the container. It is obvious that the effect of using the described algorithms will be most clearly noticeable when using multi-pass metaheuristic optimization algorithms, the use of which is associated with processing of a large number of intermediate solutions to the problem.

As a promising direction for the subsequent development of research in the field of optimizing the work of the model of potential containers, one can single out the development of an algorithm for clustering a set of PCs into disjoint sets, followed by determining the free areas of the container that cannot be used to place new objects inside them.

4. References

[1] D. S. Johnson, A brief history of NP-completeness, 1954–2012, Documenta Mathematica Extra Volume ISMP (2012) 359–376. URL: https://elibm.org/article/10011465.

[2] A. A. Leao, F. M. Toledo, J. F. Oliveira, M. A. Carravilla, R. Alvarez-Valdés, Irregular packing problems: a review of mathematical models, European Journal of Operational Research 282.3 (2020) 803–822. doi:10.1016/j.ejor.2019.04.045.

[3] S. Plankovskyy, Y. Tsegelnyk, O. Shypul, A. Pankratov, T. Romanova, Cutting irregular objects from the rectangular metal sheet, in: M. Nechyporuk, V. Pavlikov, D. Kritskiy (Eds), Integrated Computer Technologies in Mechanical Engineering, volume 1113, Springer, Cham, 2020, pp. 150-157. doi:10.1007/978-3-030-37618-5_14.

[4] J. A. Bennell, J. F. Oliveira, The geometry of nesting problems: A tutorial, European journal of operational research 184.2 (2008) 397–415. doi:10.1016/j.ejor.2006.11.038.

[5] Y. W. Tun, L. V. Markin, Methods of formation of receptor (voxel) geometric models for automated layout tasks, IOP Conference Series: Materials Science and Engineering 687.4 (2019) 044050. doi:10.1088/1757-899X/687/4/044050.

[6] L. V. Markin, Discrete geometric models in problems of automated assembling of objects, IOP Conference Series: Materials Science and Engineering 451.1 (2018) 012124. doi:10.1088/1757-899X/451/1/012124.

[7] A. C. J. De Korte, H. J. H. Brouwers, Random packing of digitized particles, Powder technology, 233 (2013) 319–324. doi:10.1016/j.powtec.2012.09.015.

[8] T. Byholm, M. Toivakka, J. Westerholm, Effective packing of 3-dimensional voxel-based arbitrarily shaped particles, Powder Technology 196.2 (2009) 139–146. doi:10.1016/j.powtec.2009.07.013.

[9] W. Hu, Z. Chen, H. Pan, Y. Yu, E. Grinspun, W. Wang, Surface mosaic synthesis with irregular tiles, IEEE transactions on visualization and computer graphics 22.3 (2015) 1302–1313. doi:10.1109/TVCG.2015.2498620.

[10] L. J. Araújo, E. Özcan, J. A. Atkin, M. Baumers, Analysis of irregular three-dimensional packing problems in additive manufacturing: a new taxonomy and dataset, International Journal of Production Research 57.18 (2019) 5920–5934. doi:10.1080/00207543.2018.1534016.

[11] Y. Che, K. Hu, Z. Zhang, A. Lim, Machine scheduling with orientation selection and two-dimensional packing for additive manufacturing, Computers & Operations Research, 130 (2021) 105245. doi:10.1016/j.cor.2021.105245.
[12] M. Verkhoturov, G. Verkhoturova, R. Yagudin, K. Danilov, R. Karimov, N. V. Kondratyeva, S. S. Valeev, Optimization of placement in the tasks of rapid prototyping and manufacturing of volumetric parts based on additive technologies, CEUR Workshop Proceedings, 2638 (2020) 298–305. URL: http://ceur-ws.org/Vol-2638/paper27.pdf.
[13] K. He, Y. Jin, W. Huang, Heuristics for two-dimensional strip packing problem with 90 rotations, Expert Systems with Applications 40.14 (2013) 5542–5550. doi:10.1016/j.eswa.2013.04.005.
[14] A. Martinez-Sykora, R. Alvarez-Valdés, J. A. Bennell, R. Ruiz, J. M. Tamarit, Matheuristics for the irregular bin packing problem with free rotations, European Journal of Operational Research 258.2 (2017) 440–455. doi:10.1016/j.ejor.2016.09.043.
[15] V. A. Chekanin, A. V. Chekanin, Algorithms for management objects in orthogonal packing problems, ARPN Journal of Engineering and Applied Sciences 11.13 (2016) 8436–8446. URL: http://www.arpnjournals.org/jeas/research_papers/rp_2016/jeas_0716_4620.pdf.
[16] N. N. Klevanskiy, S. I. Tkachev, L. A. Voloshchuk, R. B. Nourgaziev, V. S. Mavzovin, Regular two-dimensional packing of congruent objects: Cognitive Analysis of Honeycomb Constructions, Applied Sciences 11.11 (2021) 5128. doi:10.3390/app11115128.
[17] Y. Stoyan, A. Pankratov, T. Romanova, Quasi-phi-functions and optimal packing of ellipses, Journal of Global Optimization 65.2 (2016) 283–307. doi:10.1007/s10898-015-0331-2.
[18] T. Romanova, J. Bennell, Y. Stoyan, A. Pankratov, Packing of concave polyhedra with continuous rotations using nonlinear optimization, European Journal of Operational Research. 268.1 (2018) 37–53. doi:10.1016/j.ejor.2018.01.025.
[19] Y. Stoyan, T. Romanova, A. Pankratov, A. Chugay, Optimized object packings using quasi-phi-functions, in: G. Fasano, J. D. Pintér (Eds.), Springer Optimization and Its Applications, volume 105, Springer, Cham, 2015, pp. 265–293. doi:10.1007/978-3-319-18899-7_13.
[20] S. C. Leung, Y. Lin, D. Zhang, Extended local search algorithm based on nonlinear programming for two-dimensional irregular strip packing problem, Computers & Operations Research 39.3 (2012) 678–686. doi:10.1016/j.cor.2011.05.025.
[21] Y. Zhao, C. Rausch, C. Haas, Optimizing 3D irregular object packing from 3D scans using metaheuristics, Advanced Engineering Informatics, 47 (2021) 101234. doi:10.1016/j.aei.2020.101234.
[22] A. V. Tolok, N. B. Tolok, Mathematical programming problems solving by functional voxel method, Automation and Remote Control 79.9 (2018) 1703–1712. doi:10.1134/S0005117918090138.
[23] V. A. Chekanin, A. V. Chekanin, Development of algorithms for the formation and placement of N-dimensional orthogonal polyhedrons into containers of complex geometric shape, The International Journal of Advanced Manufacturing Technology 117 (2021) 2467–2479. doi:10.1007/s00170-021-06974-y.
[24] V. A. Chekanin, A. V. Chekanin, Algorithm for the placement of orthogonal polyhedrons for the cutting and packing problems, in: A. Evgrafov (Eds.), Advances in Mechanical Engineering, Lecture Notes in Mechanical Engineering, Springer, Cham, 2020, pp. 41–48. doi:10.1007/978-3-030-39500-1_5.
[25] V. Chekanin, A. Chekanin, Algorithms for working with orthogonal polyhedrons in solving cutting and packing problems, CEUR Workshop Proceedings 3027 (2021) 656–665. doi:10.20948/graphicon-2021-3027-656-665.
[26] V. A. Chekanin, A. V. Chekanin, Development of algorithms for the correct visualization of two-dimensional and three-dimensional orthogonal polyhedrons, in: A. Radionov, A. Karandaev (Eds.), Advances in Automation, RusAutoCon 2019, Lecture Notes in Electrical Engineering, volume 641, Springer, Cham, 2020, pp. 891–900. doi: 10.1007/978-3-030-39225-3_96.
[27] V. Chekanin, Solving the problem of packing objects of complex geometric shape into a container of arbitrary dimension, CEUR Workshop Proceedings, 2744 (2020). doi:10.51130/graphicon-2020-2-3-50.

[28] A. V. Chekanin, V. A. Chekanin, Effective data structure for the multidimensional orthogonal bin packing problems, Advanced Materials Research 962–965 (2014) 2868–2871. doi:10.4028/www.scientific.net/AMR.962-965.2868.
[29] V. A. Chekanin, A. V. Chekanin, Improved data structure for the orthogonal packing problem, Advanced Materials Research 945–949 (2014) 3143–3146. doi:10.4028/www.scientific.net/AMR.945-949.3143.
[30] V. A. Chekanin, A. V. Chekanin, Solving the problem of dense packing of objects of complex geometry, in: A. N. Evgrafov (Eds.), Advances in Mechanical Engineering, Lecture Notes in Mechanical Engineering, Springer, Cham, 2022, pp. 108–116. doi:10.1007/978-3-030-91553-7_12.