Approximating Smallest Containers for Packing Three-dimensional Convex Objects

Helmut Alt* Nadja Scharf*

Abstract

We investigate the problem of computing a minimum volume container for the non-overlapping packing of a given set of three-dimensional convex objects. Already the simplest versions of the problem are $\mathcal{NP}$-hard so that we cannot expect to find exact polynomial time algorithms. We give constant ratio approximation algorithms for packing axis-parallel (rectangular) cuboids under translation into an axis-parallel (rectangular) cuboid as container, for cuboids under rigid motions into an axis-parallel cuboid or into an arbitrary convex container, and for packing convex polyhedra under rigid motions into an axis-parallel cuboid or arbitrary convex container. This work gives the first approximability results for the computation of minimum volume containers for the objects described.

1 Introduction

The problem of efficiently packing objects arises in a large variety of contexts. Apart from the obvious ones, like transportation or storage, there are more abstract ones, for example cutting stock or scheduling. Consequently, packing problems have been investigated in mathematics and operations research for a long time (for a survey and references, see [1]).

In this work, we consider the problem of packing three-dimensional convex polyhedra into a minimum-volume container. All variants of this problem are NP-hard. We will develop constant factor approximation algorithms for some of them. The worst case constant factors are still very high, but probably they will be much lower for realistic inputs. The major aim of this paper, however, is to show the existence of constant factors at all, i.e., that the problems belong to the complexity class APX. For a complete version, see [3].

Related Work. So far, there are only few results about finding containers of minimum volume. Related problems include strip packing and bin packing. In two-dimensional strip packing the width of a strip is given and the objects should be packed in order to minimize the length of the strip used. In three dimensions, the rectangular cross section of the strip is fixed. Bin-packing is the problem where the complete container is fixed and the objective is to minimize the number of containers to pack all objects.

Approximation algorithms have been developed for two- and three-dimensional bin and strip packing (e.g. [4, 5, 6, 7]). Approximation algorithms for minimum area containers in two dimensions were given by v.Niederhäusern [8] and Alt et al. [2].

The well-known $\mathcal{NP}$-complete problem PARTITION can be reduced to our problem showing $\mathcal{NP}$-hardness.

2 Preliminaries and Notation

For most algorithms considered here, the input is a set of rectangular boxes $\mathcal{B} = \{b_1, b_2, \ldots b_n\}$. We denote a box b_i in axis-parallel orientation by its height, width and depth (h_i, w_i, d_i).

We define:

$$\begin{aligned} h_{\max} &= \max\{h_i \mid b_i \in \mathcal{B}\}, \\ w_{\max} &= \max\{w_i \mid b_i \in \mathcal{B}\}, \text{ and} \\ d_{\max} &= \max\{d_i \mid b_i \in \mathcal{B}\}. \end{aligned}$$

Definition 1 (OMCOP) *An instance of orthogonal minimal container packing (OMCOP) is a set of convex polyhedra. The aim is to pack these polyhedra non-overlapping such that the minimal axis-parallel container has minimal volume V_{opt}. Variants include the kind of motions allowed or that more specialized objects are to be packed.*

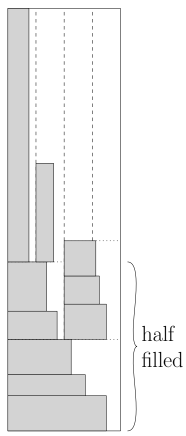

Figure 1: Result of Alg. 1

Algorithm 1 was first given in [8]. We describe it here in detail since it will be used later as a subroutine. For an example see Figure 1.

Observation 1 *The resulting strip of Algorithm 1 is half filled with rectangles up to the bottom edge of the highest rectangle r_i touching the upper end of the packing. Otherwise, r_i could have been placed lower. Thus, the strip is half filled except for a part with area at most $w \cdot h_{\max}$.*

*Institute of Computer Science, Freie Universität Berlin, alt@mi.fu-berlin.de, nadja.scharf@fu-berlin.de. This research was partially funded by DFG (Deutsche Forschungsgemeinschaft) under grant no. AL 253/7-2.

Algorithm 1:

Input: List $\mathcal{S}$ of rectangles $r_i = (w_i, h_i)$, a width for the strip w

1. Split $\mathcal{S}$ in sublists $\mathcal{S}_j = \left\{r_i \in \mathcal{S} \mid \frac{w}{2^{j-1}} \geq w_i > \frac{w}{2^j}\right\}$ for $j \geq 1$.
2. Start with packing the rectangles in $\mathcal{S}_1$ on top of each other in the strip $[0, w] \times [0, \infty)$. Split the remaining strip in two substrips with width $\frac{w}{2}$ and pack the rectangles in $\mathcal{S}_2$ one after another into these substrips. Each r_i is packed in the substrip with current minimal height.
3. Again split the substrips into two and pack S_3. Iterate that process until everything is packed.

3 Reduction from 3D-OMCOP to Strip Packing

In this section we consider the version of OMCOP where the given objects are axis-parallel boxes that are to be packed under translation. The idea of the reduction of OMCOP to strip packing is to test different base areas for the strip and to return the result with minimal volume. The base area of an optimal solution is a rectangle of width within the interval $\left[w_{\max}, W_{\sum}\right]$ and depth within the interval $\left[d_{\max}, D_{\sum}\right]$, where $W_{\sum}$ ($D_{\sum}$) denotes the sum of width (depth) of all boxes to be packed. We subdivide these intervals logarithmically depending on some parameter ε and call for all resulting width-depth-pairs as base area a strip packing algorithm with the given boxes. For a possible subdivision see Figure 2. With a more detailed elaboration and analysis (see [3]) we obtain the following theorem.

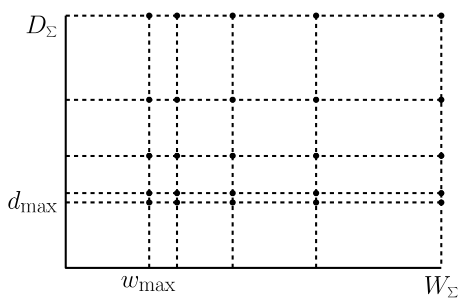

Figure 2: Example for a subdivision. The tested base areas have their lower left corner in common, candidates for the upper right corner are the grid points.

Theorem 1 *If we use an α-approximation algorithm to pack the boxes under translation into the strips with the base areas defined above, we obtain for any fixed $\varepsilon > 0$ an $(\alpha + \varepsilon)$-approximation for the OMCOP variant where n axis aligned boxes are to be packed under translation. Its runtime is $\mathcal{O}\left(T(n)\frac{\log^2 n}{\varepsilon^2}\right)$ where $T(n)$ is the runtime of the strip packing algorithm.*

If we use the algorithm given by Diedrich et al. [5] which gives a $\frac{29}{4}$-approximation for three-dimensional strip packing, we obtain the following corollary.

Corollary 2 *There exists a $(7.25 + \varepsilon)$-approximation algorithm for packing axis-parallel boxes under translation into a minimum volume axis-parallel box with runtime polynomial in the input size and $\frac{1}{\varepsilon}$.*

4 Algorithms for Variants of OMCOP

In this section we will give algorithms for variants of OMCOP. The basic idea is to get rid of the third dimension by partitioning the set of objects into sets of objects with similar height and then packing those using an algorithm for two-dimensional boxes. These containers then get cut into pieces with equal base area. The pieces will be stacked on top of each other, see Algorithm 2.

4.1 Cuboids under Translation

Although this algorithm gets outperformed by the construction in the previous section, we state it here as base for the algorithms for the other variants. For an illustration of steps 3 to 5 see Figure 3.

Algorithm 2:

Input: Set of axis parallel boxes $\mathcal{B} = \{b_1, \ldots, b_n\}$, $\alpha \in (0, 1)$, $c > 1$

1. Partition $\mathcal{B}$ into subsets of boxes that have almost the same height: $\mathcal{B}_j = \left\{b_i \in \mathcal{B} \mid h_{\max} \cdot \alpha^j < h_i \leq h_{\max} \cdot \alpha^{j-1}\right\}$
2. Use Algorithm 1 to pack every $\mathcal{B}_j$ into a strip with width $w_{\max}$ and height $h_{\max} \cdot \alpha^{j-1}$ by taking the base areas of the boxes as rectangles and applying Algorithm 1 to them.
3. Divide the strips into pieces with depth $(c - 1) \cdot d_{\max}$, ignoring the last part of the strip of depth $d_{\max}$. (Parts of boxes contained in this part of the strip will be covered in step 4.)
4. Extend each piece to depth $c \cdot d_{\max}$ such that every box lies entirely in the piece its front lies in.
5. Stack the pieces on top of each other.

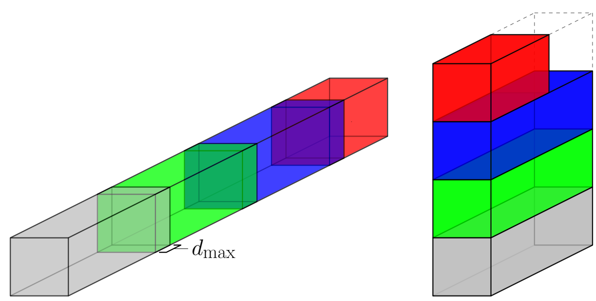

Figure 3: Cut strip and obtained pieces stacked.

Theorem 3 *For suitable values of c and α Algorithm 2 computes a 11.542-approximation for the variant of* OMCOP *where n axis parallel cuboids are packed under translation in $\mathcal{O}(n \log n)$ time.*

Proof. Let D_j denote the depth of the strip obtained in step 2 for the boxes in $\mathcal{B}_j$. Then we get by step 3 $k = \left\lceil \frac{D_j - d_{\max}}{(c-1)d_{\max}} \right\rceil$ pieces. After step 4 each piece has volume $c \cdot d_{\max} w_{\max} h_{\max} (\alpha)^{j-1}$. Then the total volume of the pieces obtained for the subset $\mathcal{B}_j$ is:

$$\begin{aligned} V_j =& k \cdot c \cdot d_{\max} w_{\max} h_{\max} (\alpha)^{j-1} \\ <& \frac{c}{c-1} (D_j - d_{\max}) w_{\max} h_{\max} (\alpha)^{j-1} \\ &+ c \cdot d_{\max} w_{\max} h_{\max} (\alpha)^{j-1}. \end{aligned}$$

We know from Algorithm 1 that the base area of the strip is half filled with boxes except for the last part of depth $d_{\max}$ (Observation 1), so $(D_j - d_{\max}) w_{\max} \le 2 \sum_{b_i \in \mathcal{B}_j} A_B(b_i)$ where $A_B(b)$ denotes the base area of box b. Also, for every $b_i \in \mathcal{B}_j$ the inequality $h_{\max} (\alpha)^{j-1} < \frac{h_i}{\alpha}$ holds. Thus, we get for the total volume of the packing V that

$$\begin{aligned} V \le& \sum_{j=1}^{\infty} \left(\frac{c}{c-1} (D_j - d_{\max}) w_{\max} h_{\max} \alpha^{j-1} \right. \\ & \left. + c \cdot d_{\max} w_{\max} h_{\max} \cdot \alpha^{j-1} \right) \\ \le& \sum_{j=1}^{\infty} \left(\frac{2c}{\alpha(c-1)} \sum_{b_i \in \mathcal{B}_j} V(b_i) \right. \\ & \left. + c \cdot w_{\max} d_{\max} h_{\max} \cdot \alpha^{j-1} \right) \\ \le& \frac{2c}{\alpha(c-1)} \underbrace{\sum_{b \in \mathcal{B}} V(b)}_{\le V_{\text{opt}}} \\ & + c \cdot \underbrace{w_{\max} \cdot d_{\max} \cdot h_{\max}}_{\le V_{\text{opt}}} \cdot \sum_{l=0}^{\infty} \alpha^l \qquad (1) \\ \le& \left(\frac{2c}{\alpha(c-1)} + \frac{c}{1-\alpha} \right) V_{\text{opt}}. \qquad (2) \end{aligned}$$

The factor before V_{opt} in term (2) is minimized if the partial derivatives with respect to c and α are 0. This gives an approximation ratio of $\frac{3}{\sqrt[3]{2}-1} \approx 11.542$. $\square$

4.2 Cuboids under Rigid Motions

Now we consider the variant of OMCOP where the objects to be packed are boxes and rigid motions are allowed. We use the algorithm stated above but with an extra preprocessing step, namely rotating every box $b_i \in \mathcal{B}$ such that it becomes axis parallel and $h_i \ge w_i \ge d_i$. This can be done in $\mathcal{O}(n)$ time. To prove the performance bound we need Lemma 4.

Lemma 4 *If every $b_i \in \mathcal{B}$ is oriented such that $h_i \ge w_i \ge d_i$, then $h_{\max} \cdot w_{\max} \cdot d_{\max} \le \sqrt{6} \cdot V_{\text{opt}}$.*

Proof. An optimal container has to contain the box determining $h_{\max}$, so it contains a line segment l_1 of length $h_{\max}$. The projection of l_1 on one of the axes has a length of at least $\frac{1}{\sqrt{3}} h_{\max}$. W.l.o.g. let this axis be the x-axis. Thus, the optimal container has an expansion of at least $\frac{1}{\sqrt{3}} h_{\max}$ in x-direction. Since every box is higher then wide, a box with width $w_{\max}$ contains a disk D with diameter $w_{\max}$ and so the optimal container does. D contains a diametric line segment l_2 which is parallel to the y-z-plane. The projection of l_2 and thus the one of the whole box on the y-axis or on the z-axis has a length of at least $\frac{1}{\sqrt{2}} w_{\max}$. W.l.o.g. let this be the y-axis. A box with depth $d_{\max}$ contains a sphere s with diameter $d_{\max}$. The projection of s on any axis, in particular the z-axis, has length at least $d_{\max}$. $\square$

Observe that every argument leading to inequality (1) still holds for this variant of the algorithm. Using Lemma 4 to estimate $h_{\max} \cdot w_{\max} \cdot d_{\max}$ we get an approximation factor of $\frac{2c}{\alpha(c-1)} + \frac{c \cdot \sqrt{6}}{1-\alpha}$. Minimizing this expression as before yields:

Theorem 5 *The given algorithm computes a 17.738-approximation for the variant of 3D* OMCOP *where n axis parallel cuboids are packed under rigid motions in $\mathcal{O}(n \log n)$ time.*

Convex Container. If we allow a convex container instead of an orthogonal container, we can use the same algorithm but adapt the analysis. The arguments leading to inequality (1) still hold since they only use the total volume of the boxes as estimate for the volume of an optimal container. But we can only show $h_{\max} \cdot w_{\max} \cdot d_{\max} \le 6 \cdot V_{\text{opt}}$, so we get with a detailed analysis the following theorem.

Theorem 6 *Using the algorithm described in section 4.2 we get a 29.135-approximation for packing n axis parallel boxes under rigid motions into a smallest-volume convex container in time $\mathcal{O}(n \log n)$.*

4.3 Convex Polyhedra under Rigid Motions

We use the algorithm from section 4.2 to pack convex polyhedra under rigid motions into an axis-parallel minimal volume box. To do so, we add another preprocessing step where we compute an enclosing box for every polyhedron. We then pack these boxes with the algorithm discussed in section 4.2. For a convex polyhedron p the enclosing box is built as follows: Let B and T be two points of p with largest distance h and π a hyperplane normal to the line segment $\overline{BT}$. Let p' be the orthogonal projection of p onto π, R' and L'

be two points of p' with largest distance w, and R and L its preimages. Let l be a line normal to $\overline{R'L'}$, the projection of p' onto l a line segment of length d with endpoints F'' and D'', and F and D its perimages in p. Then the enclosing box is $B_p = (h, w, d)$. See Figure 4 for an example. Checking all pairs of vertices as candidates for B and T, and R' and L', we get a total running time of $\mathcal{O}\left(m^2\right)$ for computing the bounding boxes of polyhedra with m vertices in total. For the analysis of this algorithm we need two lemmata that follow.

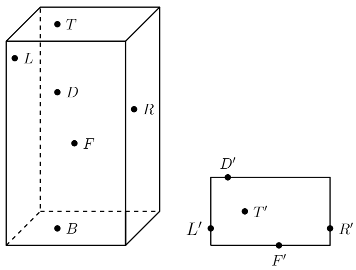

Figure 4: Box with a point of the enclosed polyhedron in every facet and the projection of the box on its base.

Lemma 7 *Let $b = (h, w, d)$ with $h \geq w \geq d$ be the enclosing box obtained for polyhedron p. Then, parallel to any given plane, p contains a line segment of length at least $w \cdot \frac{1}{\sqrt{5}}$.*

This lemma can be proven by showing that either each height in triangle (TBL) or triangle (TBR) is at least $\frac{w}{\sqrt{5}}$. The complete proof can be found in [3].

Lemma 8 *Let $b = (h, w, d)$ be the enclosing box obtained for a polyhedron p. The projection of p onto an arbitrary line g has length at least $\frac{d}{8\sqrt{3}}$.*

This Lemma is shown by an elaborate construction, where we find four line segments inside p such that the projection of at least one of them onto g has length at least $\frac{1}{8\sqrt{3}}d$. The complete proof can be found in [3].

Just as in the proof of Lemma 4 any container, in particular the optimal one, must contain a line segment of length $h_{\max}$ whose projection on one axis, say the x-axis, has length at least $\frac{h_{\max}}{\sqrt{3}}$. Applying Lemma 7 to the y-z-plane and the polyhedron defining $w_{\max}$ gives a line segment of length at least $\frac{w_{\max}}{\sqrt{5}}$ whose projection onto at least one axis, say the y-axis, has length at least $\frac{w_{\max}}{\sqrt{10}}$. By Lemma 8, the projection of the polyhedron defining $d_{\max}$ onto the z-axis has length at least $\frac{d_{\max}}{8\sqrt{3}}$. Summarizing, we obtain that $V_{\text{opt}} \geq \frac{1}{24\sqrt{10}} h_{\max} \cdot w_{\max} \cdot d_{\max}$. Using this inequality and the fact that the volume of each enclosing box is at most 6 times the volume of the enclosed polyhedron, we derive the following approximation ratio from inequality (1): $\frac{12c}{\alpha(c-1)} + \frac{c \cdot 24\sqrt{10}}{1-\alpha}$. We get by minimization:

Theorem 9 *The given algorithm computes a* 277.59*-approximation for the variant of* OMCOP *where n convex polyhedra having m vertices in total are to be packed under rigid motions in time $\mathcal{O}\left(m^2 + n \log n\right)$.*

Convex Container. We use the the result of the algorithm given in Section 4.3 to compute an approximation for a minimum volume arbitrary convex container. The approximation ratio becomes a different expression since we can only show $h_{\max} \cdot w_{\max} \cdot d_{\max} \leq 24\sqrt{60}V_{\text{opt}}$. A detailed analysis yields the following theorem.

Theorem 10 *The algorithm given in section 4.3 computes a convex container with volume at most* 511.37 *times the volume of an optimal convex container for packing n convex polyhedra having m vertices in total under rigid motions in time $\mathcal{O}\left(m^2 + n \log n\right)$.*

References

[1] H. Alt. Computational aspects of packing problems. *Bulletin of the EATCS*, 118, 2016. forthcoming.

[2] H. Alt, M. de Berg, and C. Knauer. Approximating minimum-area rectangular and convex containers for packing convex polygons. In *Proc. 23th Annu. European Sympos. Algorithms (ESA)*, pages 25–34, 2015.

[3] H. Alt and N. Scharf. Approximating Smallest Containers for Packing Three-dimensional Convex Objects. *ArXiv e-prints*, January 2016.

[4] N. Bansal and A. Khan. Improved approximation algorithm for two-dimensional bin packing. In *Proc. 25th Annu. ACM-SIAM Sympos. Discrete Algorithms (SODA)*, pages 13–25, 2014.

[5] F. Diedrich, R. Harren, K. Jansen, R. Thöle, and H. Thomas. Approximation algorithms for 3D orthogonal knapsack. *J. Comput. Sci. Tech.*, 23(5):749–762, 2008.

[6] K. Jansen and L. Prädel. A new asymptotic approximation algorithm for 3-dimensional strip packing. In *Proc. 40th International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM)*, pages 327–338, 2014.

[7] F. K. Miyazawa and Y. Wakabayashi. Three-dimensional packings with rotations. *Comput. Oper. Res.*, 36(10):2801–2815, 2009.

[8] L. von Niederhäusern. Packing polygons: Research of approximation algorithms. Master's thesis, Freie Universität Berlin and École Polytechnique Fédérale de Lausanne, 2014.