Packing Disks into Disks with Optimal Worst-Case Density

Sándor P. Fekete
Department of Computer Science, TU Braunschweig
Mühlenpfordtstr. 23, 38106 Braunschweig, Germany
s.fekete@tu-bs.de

Phillip Keldenich
Department of Computer Science, TU Braunschweig
Mühlenpfordtstr. 23, 38106 Braunschweig, Germany
p.keldenich@tu-bs.de

Christian Scheffer
Department of Computer Science, TU Braunschweig
Mühlenpfordtstr. 23, 38106 Braunschweig, Germany
c.scheffer@tu-bs.de

Abstract

We provide a tight result for a fundamental problem arising from packing disks into a circular container: The critical density of packing disks in a disk is 0.5. This implies that any set of (not necessarily equal) disks of total area $\delta \leq 1/2$ can always be packed into a disk of area 1; on the other hand, for any $\varepsilon > 0$ there are sets of disks of area $1/2 + \varepsilon$ that cannot be packed. The proof uses a careful manual analysis, complemented by a minor automatic part that is based on interval arithmetic. Beyond the basic mathematical importance, our result is also useful as a blackbox lemma for the analysis of recursive packing algorithms.

2012 ACM Subject Classification Theory of computation → Packing and covering problems; Theory of computation → Computational geometry

Keywords and phrases Disk packing, packing density, tight worst-case bound, interval arithmetic, approximation

Digital Object Identifier 10.4230/LIPIcs.SoCG.2019.35

Related Version A full version of this paper can be found at `http://arxiv.org/abs/1903.07908` [3].

Supplement Material `https://github.com/phillip-keldenich/circlepacking`

Funding *Phillip Keldenich*: Supported by the German Research Foundation under Grant No. FE 407/17-2.

Acknowledgements We thank Sebastian Morr for joint previous work.

1 Introduction

Deciding whether a set of disks can be packed into a given container is a fundamental geometric optimization problem that has attracted considerable attention. Disk packing also has numerous applications in engineering, science, operational research and everyday life, e.g., for the design of digital modulation schemes [24], packaging cylinders [1, 10], bundling tubes or cables [29, 27], the cutting industry [28], or the layout of control panels [1], or radio tower placement [28]. Further applications stem from chemistry [30], foresting [28], and origami design [16].

Like many other packing problems, disk packing is typically quite difficult; what is more, the combinatorial hardness is compounded by the geometric complications of dealing with irrational coordinates that arise when packing circular objects. This is reflected by the limitations of provably optimal results for the optimal value for the smallest sufficient disk container (and hence, the densest such disk packing in a disk container), a problem that was

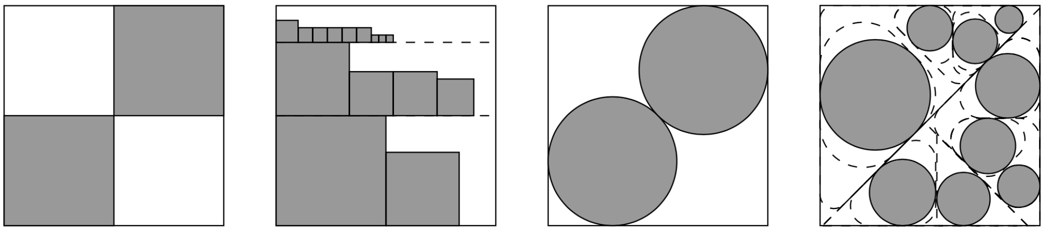

Figure 1 (1) An instance of critical density for packing squares into a square. (2) An example packing produced by Moon and Moser's shelf-packing. (3) An instance of critical density for packing disks into a square. (4) An example packing produced by Morr's Split Packing.

discussed by Kraviz [15] in 1967: Even when the input consists of just 13 unit disks, the optimal value for the densest disk-in-disk packing was only established in 2003 [9], while the optimal value for 14 unit disks is still unproven. The enormous challenges of establishing densest disk packings are also illustrated by a long-standing open conjecture by Erdős and Oler from 1961 [23] regarding optimal packings of n unit disks into an equilateral triangle, which has only been proven up to $n = 15$. For other examples of mathematical work on densely packing relatively small numbers of identical disks, see [11, 19, 7, 8], and [25, 18, 12] for related experimental work. Many authors have considered heuristics for circle packing problems, see [28, 13] for overviews of numerous heuristics and optimization methods. The best known solutions for packing equal disks into squares, triangles and other shapes are continuously published on Specht's website `http://packomania.com` [26].

For the case of packing not necessarily equal disks into a square container, Demaine, Fekete, and Lang in 2010 [2] showed that deciding whether a given set of disks can be packed is NP-hard by using a reduction from 3-Partition. This means that there is (probably) no deterministic polynomial-time algorithm that can decide whether a given set of disks can be packed into a given container.

On the other hand, the literature on exact approximation algorithms which actually give performance guarantees is small. Miyazawa et al. [20] devised asymptotic polynomial-time approximation schemes for packing disks into the smallest number of unit square bins. More recently, Hokama, Miyazawa, and Schouery [14] developed a bounded-space competitive algorithm for the online version of that problem.

The related problem of packing square objects has also been studied for a long time. The decision problem whether it is possible to pack a given set of squares into the unit square was shown to be strongly NP-complete by Leung et al. [17], also using a reduction from 3-Partition. Already in 1967, Moon and Moser [21] found a sufficient condition. They proved that it is possible to pack a set of squares into the unit square in a shelf-like manner if their combined area, the sum of all squares' areas, does not exceed $\frac{1}{2}$. At the same time, $\frac{1}{2}$ is the *largest upper area bound* one can hope for, because two squares larger than the quarter-squares shown in Figure 1 cannot be packed. We call the ratio between the largest combined object area that can always be packed and the area of the container the problem's *critical density*, or *optimal worst-case density*.

The equivalent problem of establishing the critical packing density for disks in a square was posed by Demaine, Fekete, and Lang [2] and resolved by Morr, Fekete and Scheffer [22, 4]. Making use of a recursive procedure for cutting the container into triangular pieces, they proved that the critical packing density of disks in a square is $\frac{\pi}{3+2\sqrt{2}} \approx 0.539$.

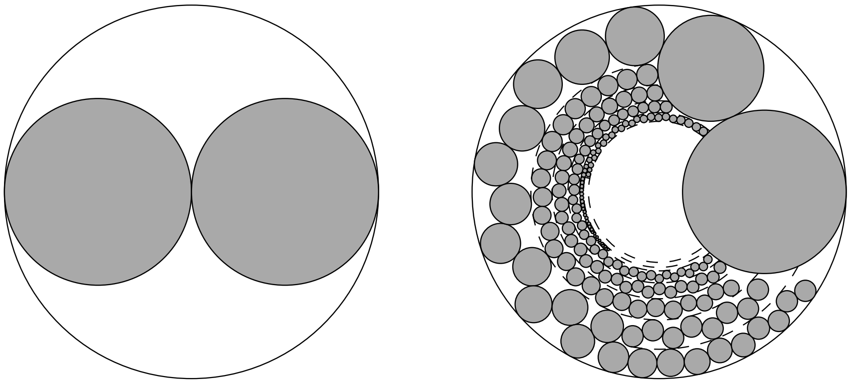

Figure 2 (1) A critical instance that allows a packing density no better than $\frac{1}{2}$. (2) An example packing produced by our algorithm.

It is quite natural to consider the analogous question of establishing the critical packing density for disks in a disk. However, the shelf-packing approach of Moon and Moser [21] uses the fact that rectangular shapes of the packed objects fit well into parallel shelves, which is not the case for disks; on the other hand, the split packing method of Morr et al. [22, 4] relies on recursively splitting triangular containers, so it does not work for a circular container that cannot be partitioned into smaller circular pieces.

Note that the main objective of this line of work is to compute tight worst-case bounds. For specific instances, a packing may still be possible, even if the density is higher; this also implies that proofs of infeasibility for specific instances may be trickier. However, the idea of using the total item volume for computing packing bounds can still be applied. See the work by Fekete and Schepers [5, 6], which shows how classes of functions called *dual-feasible* can be used to compute a *modified* volume for geometric objects, yielding good lower bounds for one- or higher-dimensional scenarios.

1.1 Results

We prove that the critical density for packing disks into a disk is 1/2: Any set of not necessarily equal disks with a combined area of not more than half the area of a circular container can be packed; this is best possibly, as for any $\varepsilon > 0$ there are instances of total area $1/2 + \varepsilon$ that cannot be packed. See Fig. 2 for the critical configuration.

Our proofs are constructive, so they can also be used as a constant-factor approximation algorithm for the smallest-area container of a given shape in which a given set of disks can be packed. Due to the higher geometric difficulty of fitting together circular objects, the involved methods are considerably more complex than those for square containers. We make up for this difficulty by developing more intricate recursive arguments, including appropriate and powerful tools based on *interval arithmetic*.

2 Preliminaries

Let $r_1, \ldots, r_n$ be a set of disks in the plane. Two point sets $A, B \subset \mathbb{R}^2$ *overlap* if their interiors have a point in common. A *container disk* $\mathcal{C}$ is a disk that may overlap with disks from $\{r_1, \ldots, r_n\}$. The *original* container disk O is the unit disk. Due to recursive calls of our algorithm there may be several container disks that lie nested inside each other. Hence, the largest container disk will be the unit disk O. For simplification, we simultaneously denote by r_i or $\mathcal{C}$ the disk with radius r_i or $\mathcal{C}$ and its radius. Wl.o.g., we assume $r_1 \geq \cdots \geq r_n$. We *pack* the disks $r_1, \dots, r_n$ by positioning their centers inside a container disk such that r_i lies inside $\mathcal{C}$ and two disks from $\{r_1, \dots, r_n\}$ do not overlap. Given two sets $A \subseteq B \subseteq \mathbb{R}^2$, we say that A is a *sector* of B. Furthermore, we denote the volume of a point set A by $|A|$.

3 A Worst-Case Optimal Algorithm

▶ **Theorem 1.** *Every set of disks with total area $\frac{\pi}{2}$ can be packed into the unit disk O with radius 1. This induces a worst-case optimal packing density of $\frac{1}{2}$, i.e., a ratio of $\frac{1}{2}$ between the area of the unit disk and the total area to be packed.*

The worst case consists of two disks D_1, D_2 with radius $\frac{1}{2}$, see Fig. 2. The total area of these two disks is $\frac{\pi}{4} + \frac{\pi}{4} = \frac{\pi}{2}$, while the smallest disk containing D_1, D_2 has an area of π.

In the remainder of Section 3, we give a constructive proof for Theorem 1. Before we proceed to describe our algorithm in Section 3.4, we give some definitions and describe *Boundary Packing* and *Ring Packing* as two subroutines of our algorithm.

3.1 Preliminaries for the Algorithm

We make use of the following definitions, see Fig. 3.

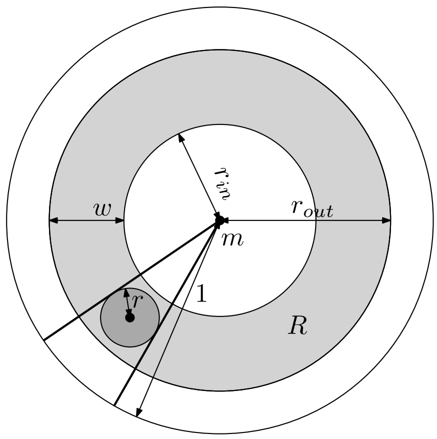

Figure 3 A ring $R \subset O$ with width w and a disk with its corresponding tangents.

For $r_{\text{out}} > r_{\text{in}} > 0$ and a container disk $\mathcal{C}$ such that $r_{\text{out}} \leq 2r_{\text{in}}$, we define a *ring* $R := R[r_{\text{out}}, r_{\text{in}}]$ of $\mathcal{C}$ as the closure of $r_{\text{out}} \setminus r_{\text{in}}$, see Fig. 3. The boundary of R consists of two connected components. The *inner boundary* is the component lying closer to the center m of r_{out} and the *outer boundary* is the other component. The *inner radius* and the *outer radius* of R are the radius of the inner boundary and the radius of outer boundary. Each ring is associated with one of three states $\{\text{OPEN}, \text{CLOSED}, \text{FULL}\}$. Initially, each ring is OPEN.

Let r be a disk inside a container disk $\mathcal{C}$. The two *tangents* of r are the two rays starting in the midpoint of $\mathcal{C}$ and touching the boundary of r. We say that a disk lies *adjacent* to r_{out} when the disk is touching the boundary of r_{out} from the inside of r_{out}.

3.2 Boundary Packing: A Subroutine

Consider a container disk $\mathcal{C}$, a (possibly empty) set S of already packed disks that overlap with $\mathcal{C}$, and another disk r_i to be packed, see Fig. 4. We *pack* r_i into $\mathcal{C}$ adjacently to the boundary of $\mathcal{C}$ as follows: Let α be the maximal polar angle realized by a midpoint of a disk

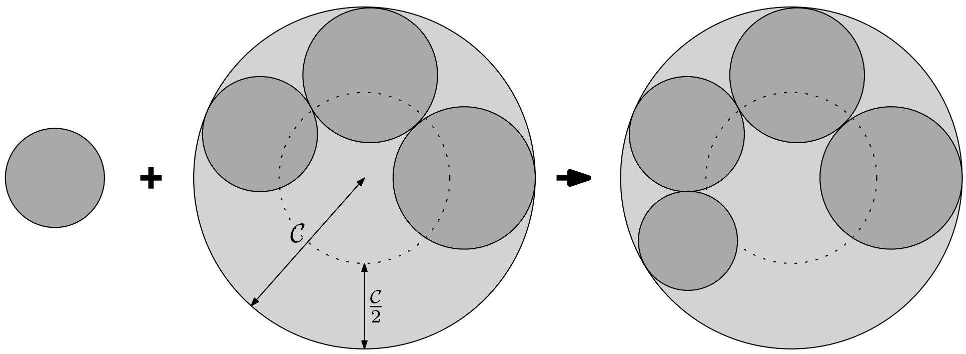

Figure 4 Boundary Packing places disks into a container disk $\mathcal{C}$ adjacent to the boundary of $\mathcal{C}$ as long as the diameter of the disks to be packed is at least as large as a given threshold $\mathcal{T}$ or until the current disk does no longer fit into $\mathcal{C}$. Initially, we have $\mathcal{T} = \frac{1}{4}$.

from S. We choose the midpoint of r_i realizing the smallest possible polar angle $\beta \geq \alpha$ such that r_i touches the outer boundary of $\mathcal{C}$ from the interior of $\mathcal{C}$ without overlapping another disk from S, see Fig. 4. If r_i cannot be packed into $\mathcal{C}$, we say that r_i *does not fit into* R.

Let $0 < \mathcal{T} \leq \frac{1}{4}$, called the *threshold*. *Boundary Packing* iteratively packs disks in decreasing order into $\mathcal{C}$ until the current disk r_i does not fit into $\mathcal{C}$ or the radius of r_i is smaller than $\mathcal{T}$.

3.3 Ring Packing: A Subroutine

Consider a ring $R := R[r_{\text{out}} r_{\text{in}}]$ with inner radius r_{in} and outer radius r_{out}, a (possibly empty) set S of already packed disks that overlap with R, and another disk r_i to be packed, see Fig. 5. We *pack* r_i into R adjacent to the outer (inner) boundary of R as follows: Let α be the maximal polar angle realized by a midpoint of a disk from S. We choose the midpoint of r_i realizing the smallest possible polar angle $\beta \geq \alpha$ such that r_i touches the outer (inner) boundary of R from the interior of R without overlapping another disk from S. If r_i cannot be packed into R, we say that r_i *does not fit into* R *(adjacent to the outer (inner) boundary)*.

Ring Packing iteratively packs disks into R alternating adjacent to the inner and outer boundary. If the current disk r_i does not fit into R Ring Packing stops and we declare R to be FULL. If r_{i-1} and r_i could pass each other, i.e., the sum of the diameters of r_{i-1} and r_i are smaller than the width of R, Ring Packing stops and we declare R to be CLOSED.

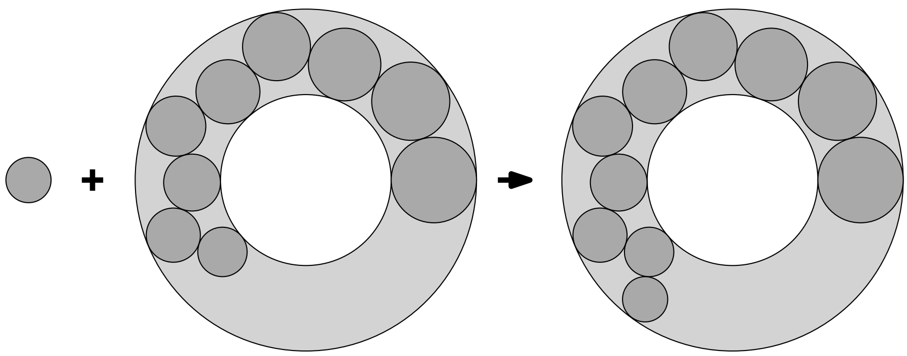

Figure 5 Ring Packing packs disks into a ring $R[r_{\text{out}}, r_{\text{in}}]$, alternating adjacent to the outer and to the inner boundary of R.

3.4 Description of the Algorithm

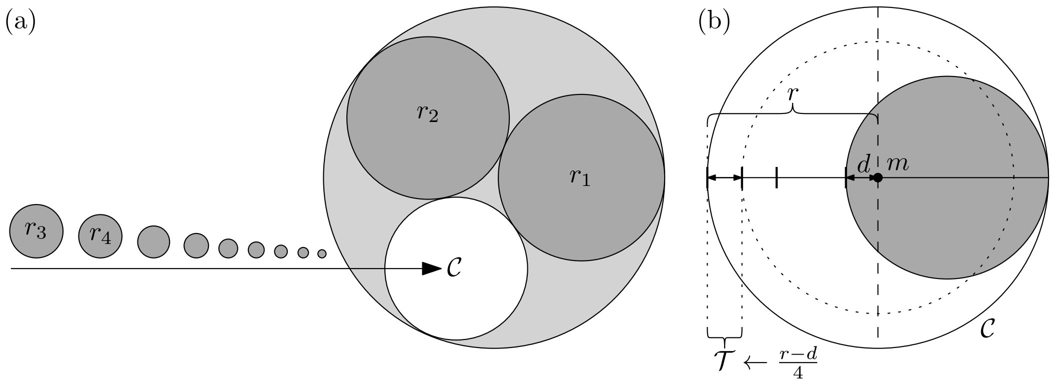

Figure 6 (a): If $r_1, r_2 \geq 0495\mathcal{C}$, Boundary Packing packs r_1, r_2 into $\mathcal{C}$. We update the current container disk $\mathcal{C}$ as the largest disk that fits into $\mathcal{C}$ and recurse on $\mathcal{C}$ with $r_3, \ldots, r_n$. (b): Determining the threshold $\mathcal{T}$ for disks packed by Boundary Packing.

Our algorithm *creates* rings. A ring only exists after it is created. We stop packing at any point in time when all disks are packed. Furthermore, we store the current threshold $\mathcal{T}$ for Boundary Packing and the smallest inner radius $r_{\min}$ of a ring created during the entire run of our algorithm. Initially, we set $\mathcal{T} \leftarrow \frac{1}{4}, r_{\min} \leftarrow 1$. Our algorithm works in five phases:

- **Phase 1 – Recursion:** If $r_1, r_2 \geq 0.495\mathcal{C}$, apply Boundary Packing to r_1, r_2, update $\mathcal{C}$ as the largest disk that fits into $\mathcal{C}$ and $\mathcal{T}$ as the radius of $\mathcal{C}$, and recurse on $\mathcal{C}$, see Fig. 6.
- **Phase 2 – Boundary Packing:** Let r be the radius of $\mathcal{C}$. If the midpoint m of $\mathcal{C}$ lies inside a packed disk r_i, let d be the minimal distance of m to the boundary of r_i, see Fig. 6(b). Otherwise, we set $d = 0$.
  We apply Boundary Packing to the container disk $\mathcal{C}$ with the threshold $\mathcal{T} \leftarrow \frac{r-d}{4}$.
- **Phase 3 – Ring Packing:** We apply Ring Packing to the ring $R := R[r_{\text{out}}, r_{\text{in}}]$ determined as follows: Let r_i be the largest disk not yet packed. If there is no open ring inside $\mathcal{C}$, we create a new open ring $R[r_{\text{out}}, r_{\text{in}}] \leftarrow R[r_{\min}, r_{\min} - 2r_i]$. Else, let $R[r_{\text{out}}, r_{\text{in}}]$ be the open ring with the largest inner radius r_{in}.
- **Phase 4 – Managing Rings:** Let $R[r_{\text{out}}, r_{\text{in}}]$ be the ring filled in Phase 3. We declare $R[r_{\text{out}}, r_{\text{in}}]$ to be closed and proceed as follows: Let r_i be the largest disk not yet packed. If r_i and r_{i+1} can pass one another inside $R[r_{\text{out}}, r_{\text{in}}]$, i.e., if $2r_i + 2r_{i+1} \leq r_{\text{out}} - r_{\text{in}}$, we create two new open rings $R[r_{\text{out}}, r_{\text{out}} - 2r_i]$ and $R[r_{\text{out}} - 2r_i, r_{\text{in}}]$.
- **Phase 5 – Continue:** If there is an open ring, we go to Phase 3. Otherwise, we set $\mathcal{C}$ as the largest disk not covered by created rings, set $\mathcal{T}$ as the radius of $\mathcal{C}$, and go to Phase 2.

4 Analysis of the Algorithm

4.1 Analysis of Phase 1 - The Recursion

If $r_2 \geq 0.495$, Lemma 2 allows us to recurse on $\mathcal{C}$ as required by Phase 1.

▶ **Lemma 2.** *If $r_1, r_2 \geq 0.495\mathcal{C}$, the volume of the largest container disk that fits into $\mathcal{C}$ after packing r_1, r_2 is at least twice the total volume of $r_3, \ldots, r_n$, see Fig. 7.*

Proof. W.l.o.g., assume that the original container disk is the unit disk. Lemma 3 implies $r_1 + r_2 \leq 1$, which means $r_1, r_2 \leq 0.505$, because $r_2 \geq 0.495$. Furthermore, $r_1 + r_2 \leq 1$ implies that we can move (w.l.o.g.) r_1, r_2 into two disks D_1, D_2 with radius 0.505, touching the

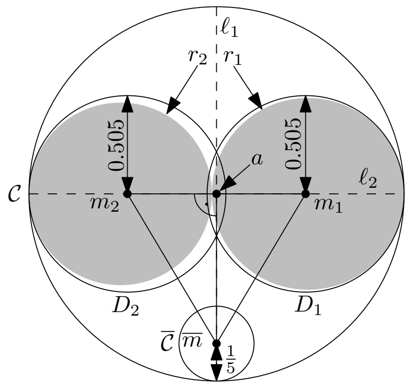

Figure 7 If $r_2 \geq 0.495$, we can pack r_1, r_2 into container disks D_1, D_2 and recurse on a third disk $\overline{c}$ whose area is twice the total area of the remaining disks.

boundary of $\mathcal{C}$ and with their midpoints m_1, m_2 on the horizontal diameter of $\mathcal{C}$, see Fig. 7. This decreases the volume of the largest disk that still fits into $\mathcal{C}$. Consider the disk $\overline{\mathcal{C}} := \frac{1}{5}$ lying adjacent to $\mathcal{C}$ and with its midpoint $\overline{m}$ on the vertical diameter ℓ_1 of $\mathcal{C}$. Pythagoras' Theorem implies that $|m_1\overline{m}| = \sqrt{(1-0.505)^2 + \left(1-\frac{1}{5}\right)^2} \approx 0.94075 > 0.505 + \frac{1}{5}$. Finally, we observe that the area of $\overline{\mathcal{C}}$ is $\frac{\pi}{25} = 0.4\pi > 0.0199 = 2\left(\frac{\pi}{2} - 2 \cdot \pi 0.495^2\right)$. This means that the area of $\overline{\mathcal{C}}$ is twice the total area of the remaining disks $r_3, r_4, r_5, \ldots$, concluding the proof. ◀

A technical key ingredient in the proof of Lemma 2 is the following lemma:

▶ **Lemma 3.** *The area of two disks r_1, r_2 is at least $\frac{\pi}{2}(r_1 + r_2)^2$.*

Proof. The first derivative of the function mapping a radius onto the area of the corresponding disk is the periphery of the corresponding circle. As $r_1 \geq r_2$, decreasing r_1 and increasing r_2 by the same value δ reduces the total area of r_1, r_2, while the value $r_1 + r_2$ stays the same. Hence, we assume w.l.o.g. that $r_1 = r_2$. This implies that the total area of r_1, r_2 is $2\pi r_1^2 = \frac{\pi}{2}(r_1 + r_2)^2$, concluding the proof. ◀

This allows us to assume $r_2 < 0.495\mathcal{C}$ during the following analysis.

4.2 Outline of the Remaining Analysis

Once our algorithm stops making recursive calls, i.e., stops applying Phase 1, Phase 1 is never applied again. W.l.o.g., let $r_1, \ldots, r_n$ be the remaining disks and O the container disk after the final recursion call.

The main idea of the remaining analysis is the following: We cover the original container disk O by a set of *sectors* that are subsets of O. Let r_i be a disk packed by Boundary Packing into the current container disk $\mathcal{C}$. We define the *cone* induced by r_i as the area of $\mathcal{C}$ between the two tangents of r_i. We say that $\mathcal{C}$ is the radius of the cone. textA *sector* is a subset of O.

Each disk *pays* portions, called *atomic potentials*, of its volume to different sectors of O. The total atomic potential paid by a disk r will be at most the volume of the disk r. Let $A_1, \ldots, A_k$ be the total atomic potentials paid to the sectors $S_1, \ldots, S_k \subset O$. The *potential* of a sector $S \subseteq O$ is the sum of the proportionate atomic potentials from $S_1, \ldots, S_k$, i.e., the sum of all $\frac{|S_i \cap S|}{|S_i|} A_i$ for $i = 1, \ldots, k$. The *(virtual packing) density* $\rho(S)$ of the sector S is defined as the ratio between the potential of S and the volume of S. If a sector achieves a density of $\frac{1}{2}$, we say that the sector is *saturated*, otherwise its *unsaturated*.

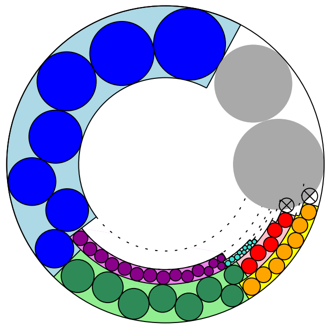

Figure 8 Different sequences of rings packed by different applications of Ring Packing. The minimal rings into which the orange and red disks are packed are full. The minimal ring into which the turquoise disks are packed is open. The uncolored, crossed-out circles illustrate that the corresponding disk did not fit into the current ring, causing it to be declared full.

Our approach for proving Theorem 1 is by induction over n. In particular, we assume that $O \setminus \mathcal{C}$ is saturated; we show that each disk r_i can be packed by our algorithm, as long as $\mathcal{C}$ is unsaturated implying that each set of disks with total volume of at most $\frac{|O|}{2}$ is packed. We assume for the remainder of the paper that $\mathcal{C}$ is the unit disk, i.e., $\mathcal{C} = 1$.

We consider the configuration achieved after termination.

If there is a ring that is neither full nor closed, all disks are packed.

Thus, we assume that all rings computed by our algorithm are full or closed. In order to avoid that Boundary Packing stops due to a disk r not fitting, we consider the gap that is left by Boundary Packing, see Fig. 9. This gap achieves its maximum for $r = \frac{1}{4}$. We guarantee that $\mathcal{C}$ has a density of

$$\rho := \frac{180^\circ}{360^\circ - 2\arcsin\left(\frac{1/4}{3/4}\right)} < 0.56065.$$

4.3 Analysis of Boundary Packing

The following lemma is the key ingredient for the analysis of Boundary Packing.

▶ **Lemma 4.** *Let $r \in [0.2019, \frac{1}{2}]$ be a disk lying adjacent to $\mathcal{C}$. The cone C induced by r has a density better than ρ if $r \in [\frac{1}{4}, 0.495]$ and at least $\frac{1}{2}$ if $r \in [0.2019, \frac{1}{2}]$, see Fig. 10.*

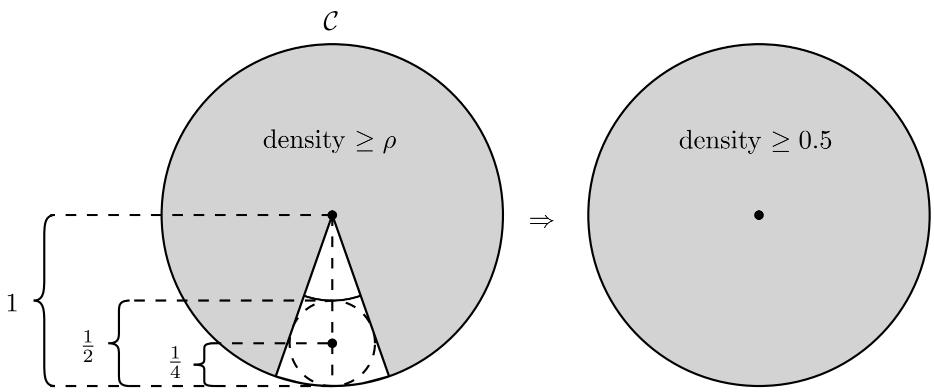

Figure 9 Ensuring a density of at least 0.5 for a ring R needs a density of 0.5606 for $R \setminus C$.

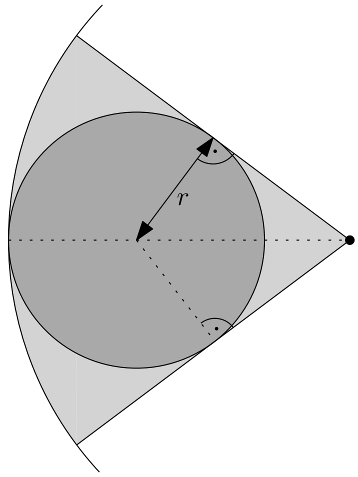

Figure 10 A disk $r \in [\frac{1}{4}, 0.495]$ lying adjacent to $\mathcal{C}$ induces a cone with density of at least 0.56127 if $r \in [\frac{1}{4}, 0.495]$ and of least $\frac{1}{2}$ if $r \in [\frac{1}{4}, \frac{1}{2}]$.

Proof. Let $f(r) := \frac{\pi r^2}{\arcsin\left(\frac{r}{1-r}\right)}$ for $\frac{1}{4} \leq r \leq \frac{1}{2}$. Thus we have

$$f'(r) = \frac{2\pi r}{\arcsin\left(\frac{r}{1-r}\right)} - \frac{\pi r^2 \left(\frac{1}{1-r} + \frac{r}{(1-r)^2}\right)}{\arcsin\left(\frac{r}{1-r}\right)^2 \sqrt{1 - \frac{r^2}{(1-r)^2}}}.$$

Solving $f'(r) = 0$ yields $r \approx 0.39464$. Furthermore, we have $f(\frac{1}{4}) \approx 0.57776$, $f(0.39464) = 0.68902$, $f(\frac{1}{2}) = 0.5$, and $f(0.495) \approx 0.56127$. Thus, f restricted to $[\frac{1}{4}, 0.495]$ achieves at 0.495 its global minimum 0.56127. A similar approach implies that f restricted to $[0.2019, \frac{1}{2}]$ attains its global minimum $\frac{1}{2}$ at $\frac{1}{2}$. ◀

The following lemma proves that all disks $r_i \geq \frac{\mathcal{C}}{4}$ that are in line to be packed into a container disk $\mathcal{C}$ can indeed be packed into $\mathcal{C}$.

▶ **Lemma 5.** *All disks $r_i \geq \frac{1}{4}$ that are in line to be packed into $\mathcal{C}$ by Boundary Packing do fit into $\mathcal{C}$.*

Proof. Assume that there is a largest disk $r_k \geq \frac{1}{4}$ not packed adjacent to $\mathcal{C}$. Each disk r_i from $r_1, \ldots, r_{k-1}$ pays its entire volume to the cone induced by r_i. Lemma 4 implies that each cone is saturated. As r_k does not fit between r_1, r_{k-1} and is adjacent to $\mathcal{C}$, Lemma 4 implies that the area of $\mathcal{C}$ that is not covered by a cone induced by $r_1, \ldots, r_{k-1}$ has a volume smaller than twice the volume of r_k. This implies that the total volume of $r_1, \ldots, r_k$ is larger than half of the volume of C. This implies that the total input volume of $r_1, \ldots, r_n$ is larger than twice the volume of the container. This is a contradiction, concluding the proof. ◀

▶ **Corollary 6.** *If $r_n \geq \frac{1}{4}$, our algorithm packs all input disks.*

Thus, we assume w.l.o.g. $r_n < \frac{1}{4}$, implying that our algorithm creates rings.

4.4 Analysis of Ring Packing

For the following definition, see Fig. 11 (Middle).

▶ **Definition 7.** *A* zipper *Z is a (maximal) sequence $\langle r_k, \ldots, r_\ell \rangle$ of disks that are packed into a ring R during an application of Ring Packing. The* length *of Z is defined as $k - \ell + 1$.*

Consider a zipper $\langle r_k, \ldots, r_\ell \rangle$ packed into a ring R. For a simplified presentation, we assume in Section 4.4 that the lower tangent of r_k realizes a polar angle of zero, see Fig. 11.

We refine the potential assignments of zippers as follows. Let $Z = \langle r_k, \ldots, r_\ell \rangle$ be an arbitrary zipper and R the ring into which Z is packed. In order to subdivide R into sectors corresponding to specific parts of the zipper, we consider for each disk r_i the *center ray*, which is the ray starting from m and passing the midpoint of r_i. Let t_1, t_2 be two rays starting in m. We say that t_1 lies above t_2 when the polar angle realized by t_1 is at least as large as the polar angle realized by t_2. t_1 is the *minimum* (*maximum*) of t_1, t_2 if t_1 does not lie above (below) t_2. Furthermore, the *upper tangent* (*lower tangent*) of a disk r_i is the maximal (minimal) tangent of r_i.

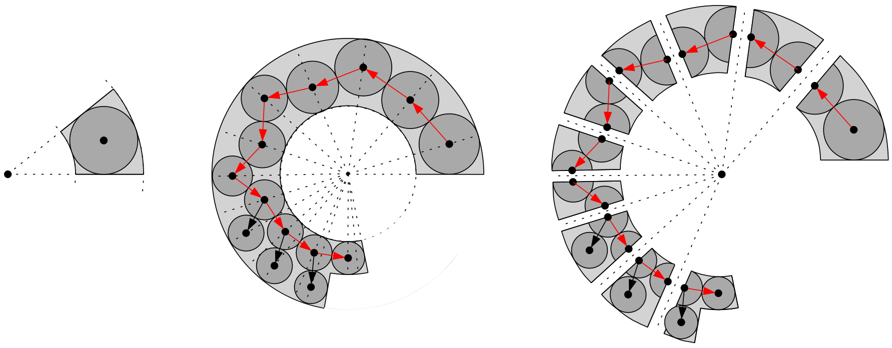

Figure 11 A maximal sequence of disks that are packed into a ring during an application of Boundary Packing. The corresponding sectors are illustrated in light gray. **Left:** A zipper of size one and the corresponding sector. **Middle:** A zipper of size 14, the resulting directed adjacency graph (black/red), and the path (red) leading from the largest disk to the smallest disk. The first seven edges of P are diagonal and the remaining edges of P are vertical. **Right:** The zipper and the sector disassembled into smaller sectors corresponding to the edges of the red path.

If the zipper Z consists of one disk r_k, the *sector* S of Z is that part of R between the two tangents to r_k and r_k pays its entire volume to S.

▶ **Lemma 8.** *The density of the sector S of a zipper of length one is at least* 0.77036*.*

Proof. As the zipper consists of only one disk r_k, r_k touches both the inner and the outer boundary of R. Hence, the density of S is not increased by assuming that the inner radius of R is equal to the diameter of r_k. Hence, the density of S is at least $\frac{\pi}{12 \arcsin(1/3)} \approx 0.77036$. ◀

Assume the zipper $\langle r_k, \ldots, r_\ell \rangle$ consists of at least two disks. We define the *adjacency graph* $G = (\{r_k, \ldots, r_\ell\}, E)$ as a directed graph as follows: There is an edge (r_j, r_i) if (1) $r_i \leq r_j$ and (2) r_i, r_j are touching each other, see Fig. 11 (Right). As Ring Packing packs each disk r_i with midpoint m_i such that m_i realizes the smallest possible polar angle, there is a path $e_k, \ldots, e_{\ell-1} =: P$ connecting r_k to r_ℓ in the adjacency graph G, see Fig. 11 (Middle). e_k is the *start* edge of P and $e_{\ell-1}$ is the end edge of P. The remaining edges of P that are neither the start nor the end edge of G, are *middle* edges of P. Furthermore, an edge $(r_j, r_m) = e_i \in P$ is *diagonal* if r_j, r_m are touching different boundary components of R. Otherwise, we call e_i *vertical*.

Depending on whether e_i is a start, middle, or an end edge and on whether e_i is diagonal or vertical, we classify the edges of the path P by eight different types T1-T8. For each type we individually define the sector A_i belonging to an edge $(r_j, r_m) = e_i \in P$ and the potential assigned to A_i, called the *potential of* e_i. Let t_{lower} be the minimum of the lower tangents of r_j, r_m and t_{upper} the maximum of the upper tangents of r_j, r_m, see Fig. 12 (a). Furthermore, let t_1, t_2 be the center rays of r_j, r_m, such that t_1 does not lie above t_2.

For the case that $e_i = (r_j, r_m)$ is a vertical edge, we consider additionally the disk r_p that is packed into R after r_j and before r_m, see Fig. 12 (f). Let t_3 be the maximum of t_2 and the upper tangent of r_p, see Fig. 12.

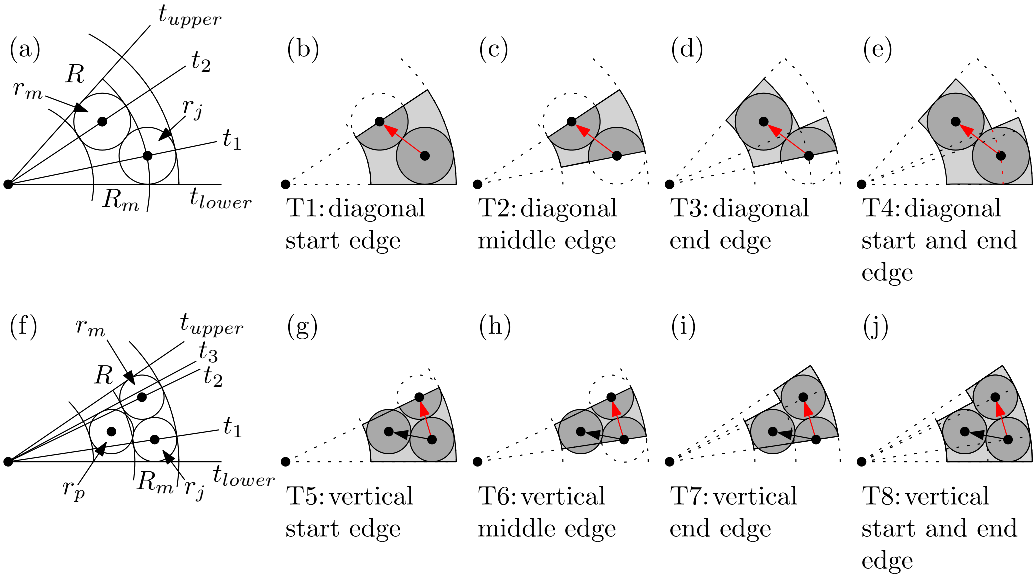

Figure 12 The eight possible configurations of an edge e_i (red) of P, the corresponding sectors (light gray), and the potentials (dark gray) payed by the involved disks to the sector.

T1 The sector of e_i: If $e_i = (r_j, r_m)$ is a diagonal start edge (as shown in Fig. 12(b)), the sector of e_i is that part of R that lies between t_{lower} and t_2.
The potential of e_i: r_j pays its entire volume and r_m the half of its volume to the sector of e_i.

T2 The sector of e_i: If $e_i = (r_j, r_m)$ is a diagonal middle edge, (as shown in Fig. 12(c)), the sector of e_i is that part of R that lies between t_1 and t_2.
The potential of e_i: r_j and r_m pay the half of its volume to the sector of e_i.

T3 The sector of e_i: If $e_i = (r_j, r_m)$ is a diagonal end edge, (as shown in Fig. 12(d)), the sector of e_i consists of two parts: (1) The first is the part of R that lies between the upper tangent and the center ray of r_j. (2) Let R_m be the smallest ring enclosing r_m. The second part of the sector is that part of R_m that lies between the upper tangent of r_m and the minimum of t_1 and the lower tangent of r_m.
The potential of e_i: r_j pays the half of its volume and r_m its entire volume to the sector of e_i.

T4 The sector of e_i: If $e_i = (r_j, r_m)$ is a diagonal start and end edge, (as shown in Fig. 12(e)), the sector of e_i is the union of two sectors: (1) The first is the part of R that lies between the lower and the upper tangent of r_j. (2) The second is that part of R_m that lies between the lower and the upper tangent of r_m.
The potential of e_i: r_j, r_m pay their entire volume to the sector of e_i.

T5 The sector of e_i: If $e_i = (r_j, r_m)$ is a vertical start edge, (as shown in Fig. 12(g)), the sector of e_i is that part of R that lies between the minimum of the lower tangents of r_j, r_p and the center ray of r_m.
The potential of e_i: r_j, r_p pay their entire volume and r_m the half of its volume to the sector of e_i.

T6 **The sector of e_i**: If $e_i = (r_j, r_m)$ is a vertical middle edge, (as shown in Fig. 12(h)), the sector of e_i is that part of R that lies between the center rays of r_j, r_m.
The potential of e_i: r_p pays its entire volume and r_j, r_m pay half of their respective volume to the sector of e_i.

T7 **The sector of e_i**: If $e_i = (r_j, r_m)$ is a vertical end edge, (as shown in Fig. 12(i)), the sector of e_i consists of two parts: (1) The first is that part of R that lies between the center ray of r_j and the upper tangent of r_p. (2) Let R_m be the smallest ring enclosing r_m. The second part of the sector is the part of R_m that lies between the center ray of r_j and the upper tangent of r_m.
The potential of e_i: r_j pays the half of its volume and r_p, r_m their entire volumes to the sector of e_i.

T8 **The sector of e_i**: If $e_i = (r_j, r_m)$ is a vertical start and end edge, (as shown in Fig. 12(j)), the sector of e_i is consists of two parts: (1) The first is that part of R that lies between the minimum of the lower tangents of r_j, r_p and the maximum of the upper tangents r_j, r_p. (2) Let R_m be the smallest ring enclosing r_m. The second part of the sector is that part of R_m that lies between the lower and the upper tangent of r_m.
The potential of e_i: r_j, r_p, r_m pay their entire volume to the sector of e_i.

For simplicity, we also call the density of the sector of an edge $e_i \in P$ the *density of* e_i. The *sector* of a zipper is the union of the sectors of the edges of P.

▶ **Lemma 9.** *Let $Z = \langle r_k, \dots, r_\ell \rangle$ be a zipper of length at least two and P a path in the adjacency graph of Z connecting r_k with r_ℓ. Each edge $e_i \in P$ has a density of at least ρ.*

The proof of Lemma 9 is the only computer-assisted proof. All remaining proofs are analytic. Due to space constraints, the proof of Lemma 9 is given in the full version of the paper [3]. Combining Lemmas 8 and 9 yields the following.

▶ **Corollary 10.** *Sectors of zippers have a density of at least ρ.*

Ring Packing stops when the sum of the diameters of the current disk r_i and the disk packed last r_{i-1} is smaller than the width w of the current ring, i.e., if $2r_{i-1} + 2r_i < w$. If $2r_{i-1} + 2r_i < w$, Phase 5 partitions the current ring into two new open rings with widths $2r_i, w - 2r_i$. Hence, the sectors of zippers packed by Ring Packing become firmly interlocked without leaving any gaps between two zippers, see Fig. 13. The only sectors that we need to

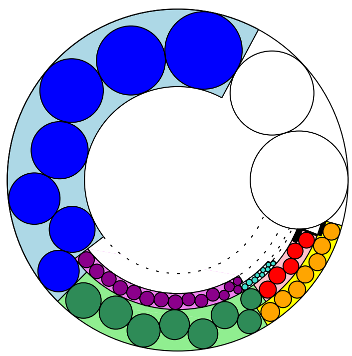

Figure 13 The sectors of rings packed by Ring Packing become firmly interlocked without leaving any gaps between two sectors. The minimal rings into which the orange and the red zippers are packed are full. The minimal ring into which the turquoise zipper is packed is open.

care about are the gaps that are left by Ring Packing due to the second break condition, i.e., the current disk does not fit into the current ring, see the black sectors in Fig. 13.

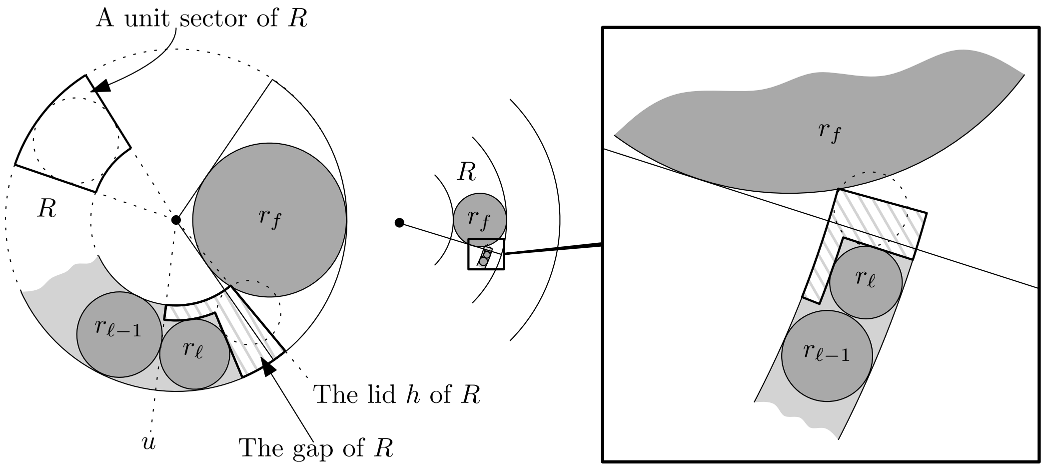

Figure 14 The lid, the gap (shaded white-gray), and a unit sector of a ring R.

▶ **Corollary 11.** *Let R be a minimal ring and G its gap. $R \setminus G$ has a density of at least ρ.*

In order to analyze the gaps left by Ring Packing, we first need to observe for which rings we need to consider gaps. In particular, we have two break conditions for Ring Packing:

(1) The current disk r_i does not fit into the current ring R, causing us to close the ring and disregard it for the remainder of the algorithm?

(2) The current and the last disk r_{i-1} packed into R can pass one another, resulting in R to be partitioned into several rings with smaller widths. Thus, we obtain that two computed rings R_1, R_2 either do not overlap or R_1 lies inside R_2.

▶ **Definition 12.** *Consider the set of all rings $R_1, \dots, R_k$ computed by our algorithm. A ring R_i is* maximal *if there is no ring R_j with $R_i \subset R_j$. A ring R_i is* minimal *if there is no ring R_j with $R_i \supset R_j$.*

By construction of the algorithm, each ring is partitioned into minimal rings. Thus, we define gaps only for minimal rings, see Figure 14 and Definition 13.

▶ **Definition 13.** *Let $Z = \langle \dots, r_{\ell-1}, r_\ell \rangle$ be a zipper of length at least 2 inserted into a minimal ring R. The* lid *h of R is the ray above the upper tangent u of r_ℓ such that h realizes a maximal polar angle while $h \cap R$ does not intersect an already packed disk r_f with $f \le \ell - 1$, see Fig. 14. The* gap *of R is the part of R between the upper tangent u of $r_{\ell-1}$ and the lid of R which is not covered by sectors of Z, see the white-gray striped sectors in Fig. 14.*

A unit sector *of R is a sector of R that lies between the two tangents of a disk touching the inner and the outer boundary of R, see Fig. 14. The* unit volume *U_R of R is the volume of a unit sector of R.*

The lid of a gap lies either inside a cone induced by a disk packed by Boundary Packing, see Fig. 14 (Left), or inside the sector of a zipper packed by Ring Packing, see Fig. 14 (Right). This leads to the following observation: Each minimal ring R is covered by the union of cones induced by disks packed by Boundary Packing into R, sectors of zippers packed by Ring Packing into R, and the gap of R.

Next, we upper bound the volume of the gap of minimal rings.

▶ **Lemma 14.** *The gap of a minimal ring R has a volume of at most $1.07024 U_R$.*

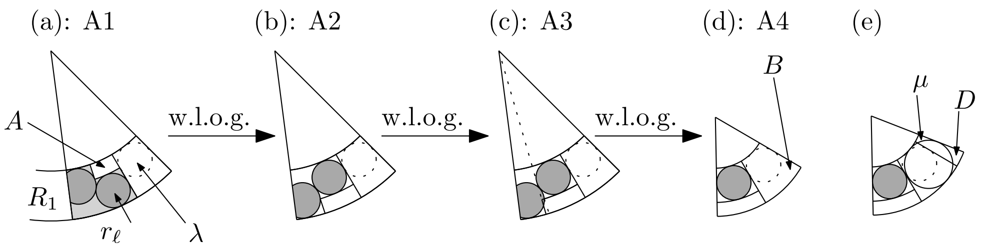

Figure 15 Simplifying assumptions that do not increase the density.

Proof. As we want to upper bound the volume of the gap w.r.t. the unit volume U_R of R, w.l.o.g. we make the following assumptions (A1)-(A4), see Fig. 15:

- (A1) The largest disk λ inside R touching h from below, the upper tangent of r_ℓ from above, and the inner boundary of R, such that λ does not overlap with any other disks from below, has the same radius as r_ℓ, see Fig. 15(a).
- (A2) The last disk r_ℓ packed into R touches the inner boundary of R, see Fig. 15(b).
- (A3) The empty pocket A left by the sector of the end edge of the zipper inside R is bounded from below by the lower tangent of r_ℓ but not by the upper tangent of $r_{\ell-1}$, see Fig. 15(c).
- (A4) $r_{\text{out}} = 1, r_{\text{in}} = \frac{1}{2}$, see Fig. 15(d).

Let B be the sector of R that lies between the two tangents of λ, see Fig. 15(d). We upper bound the volume of the gap of R as $|A| + |B| \leq 1.07024U_R$, as follows.

Let $\mu \subset R$ be the disk touching the inner and the outer boundary of R_1 and the upper tangent of r_ℓ from above, see Fig. 15(e). Furthermore, let D be the part of the cone induced by μ which lies inside R and between the upper and lower tangent of μ, see Fig. 15(e).

In the following, we show that $|A| - |D| \leq 0.07024U_R$.

$$\begin{aligned}
|A| - |D| \quad &\leq \quad \frac{2\arcsin\left(\frac{\lambda}{\frac{1}{2}+\lambda}\right)}{2\pi}\pi\left(1 - \left(\frac{1}{2} + 2\lambda\right)^2\right) \\
&\quad - \frac{2\arcsin\left(\frac{1}{3}\right) - 2\arcsin\left(\frac{\lambda}{\frac{1}{2}+\lambda}\right)}{2\pi}\pi\left(\frac{3}{4}\right) \\
&= \quad \arcsin\left(\frac{\lambda}{\frac{1}{2}+\lambda}\right)\left(\frac{7}{4} - \left(\frac{1}{2} + 2\lambda\right)^2\right) \\
&\quad - \frac{3}{4}\arcsin\left(\frac{1}{3}\right) =: V_{AD}.
\end{aligned}$$

The first derivative of V_{AD} is

$$\begin{aligned}
\frac{d\, V_{AD}\, \lambda}{d\, \lambda} \quad &= \quad \frac{\left(\frac{1}{\frac{1}{2}+\lambda} - \frac{\lambda}{\left(\frac{1}{2}+\lambda\right)^2}\right)\left(\frac{7}{4} - (2\lambda + \frac{1}{2})^2\right)}{\sqrt{1 - \frac{\lambda^2}{\left(\frac{1}{2}+\lambda\right)^2}}} \\
&\quad -4\arcsin\left(\frac{\lambda}{\frac{1}{2}+\lambda}\right)\left(2\lambda + \frac{1}{2}\right).
\end{aligned}$$

Solving $\frac{d\, V_{AD}\, \lambda}{d\, \lambda} = 0$ yields $\lambda \approx 0.196638$. Finally, we observe that $V_{AD}\left(\frac{1}{8}\right) \approx -0.01576$, $V_{AD}\left(0.196638\right) \approx 0.01756$, $V_{AD}\left(\frac{1}{4}\right) = 0$. This implies that $|A| - |D| \leq 0.01756 \leq 0.07024U_R$, because $U_R \geq \frac{1}{4}$. ◀

4.5 Analysis of the Algorithm for the Case $r_1 \leq 0.495$

We show that each computed minimal ring is saturated, see Corollary 17. Let $R_1, \dots, R_h \subseteq \mathcal{C}$ be the created minimal rings ordered decreasingly w.r.t. their outer radii. The inner boundary of R_i is the outer boundary of R_{i+1} for $i = 1, \dots, h-1$.

We show by induction over h that $R := R[r_{\text{out}}, r_{\text{in}}] := R_h$ is saturated. Thus, we assume that $R_1, \dots, R_{h-1}$ are saturated, implying that $\mathcal{C} \setminus r_{\text{out}}$ is saturated, where r_{out} is the outer radius of R_h.

For the remainder of Section 4.5, each disk r_i packed by Boundary Packing pays its entire volume to the cone induced by r_i.

▶ **Lemma 15.** *Assume $r_n < \frac{1}{4}$. There is at least one disk r_k packed into R and touching both the inner and the outer boundary of R.*

Proof. Assume that our algorithm did not pack a disk with radius smaller than $\frac{1}{4}$ adjacent to $\mathcal{C}$. Let r_k be the largest disk not packed adjacent to $\mathcal{C}$ into R.

By Lemma 5, we obtain that r_k is smaller than $\frac{1}{4}$. This implies that the volume of the sector that is not covered by the cones induced by $r_1, \dots, r_{k-1}$ is upper bounded by $\arcsin\left(\frac{1}{3}\right)$, see Fig. 16.

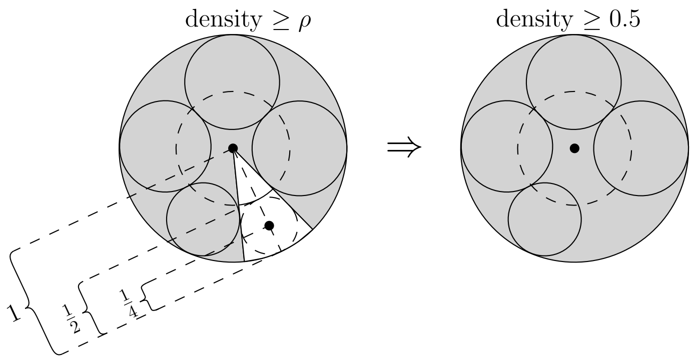

Figure 16 Ensuring density of at least ρ for all cones induced by disks packed by Boundary Packing implies a density of at least 0.5 for the entire container disk.

Each disk r_i from $r_1, \dots, r_{k-1}$ pays its entire volume to the cone induced by r_i. Lemma 4 implies that each cone has a density of at least ρ, because $r_1, \dots, r_n \leq 0.495$. This implies that the total volume of $r_1, \dots, r_{k-1}$ is at least $\pi \cdot \rho \cdot \frac{2\pi - 2\arcsin(1/3)}{2\pi} = \rho(\pi - \arcsin(1/3)) > \frac{\pi}{2}$ contradicting the assumption that the total input volume is no larger than $\frac{\pi}{2}$. ◀

▶ **Lemma 16.** *R_h is saturated.*

Proof. Let S_1 be the sector of R_h that is covered by cones induced by disks packed by Boundary Packing or by sectors of zippers packed by Ring Packing. Lemma 15 implies that there is a disk r_k packed into R_h such that r_k touches the inner and the outer boundary of R_h. Let S_2 be the sector of R_h between the lower and the upper tangent of r_k.

We move potentials δ_1, δ_2 from S_1, S_2 to a potential variable Δ and guarantee that Δ is at least $\frac{1}{2}$ times the volume of the gap G of R_h. Finally, we move Δ to G, implying that G is saturated, which in turn implies that R_h is saturated.

Lemma 8 implies that the density of S_2 is at least 0.77036. We move a potential $\delta_2 := (0.77036 - \rho)\,|S_1| > 0.20971 U_{R_h}$ from S_2 to Δ, implying that S_2 has still a density of ρ.

Combining Lemma 4 and Corollary 10 yields that S_1 has a density of at least ρ. Lemma 14 implies that the volume of the gap of R_h is at most $1.07024U_R$. The volume of R_h is at least $\frac{2\pi}{2\arcsin\left(\frac{1}{3}\right)}U_{R_h} > 9.24441U_{R_h}$. Thus, the volume of S_1 is at least $(9.24441 - 1.07024)U_{R_h} = 8.17417U_{R_h}$. Hence, we move a potential $\delta_1 := \left(\rho - \frac{1}{2}\right) 8.17417U_{R_h} > 0.49576U_{R_h}$ to Δ.

We have $\Delta = \delta_1 + \delta_2 > 0.49576 + 0.20971 = 0.70547$, which is large enough to saturate a sector of volume $V_\Delta = 2 \cdot 0.70547 = 1.41094U_{R_h}$. As $|G| \leq 1.07024$, moving Δ to G yields that G is saturated, which implies that R_h is saturated. This concludes the proof. ◀

▶ **Corollary 17.** *Each minimal ring is saturated.*

As each ring can be partitioned into minimal rings, we obtain the following.

▶ **Corollary 18.** *All rings are saturated.*

Combining Lemma 5 and Corollary 18 yields that all disks are packed.

▶ **Lemma 19.** *Our algorithm packs all input disks.*

Proof. By induction assumption we know that $O \setminus \mathcal{C}$ is saturated and Corollary 18 implies that all rings inside $\mathcal{C}$ are also saturated.

Let $\overline{\mathcal{C}}$ be the disk left after removing all rings from $\mathcal{C}$, implying that $\overline{\mathcal{C}}$ is empty. Lemma 5 implies that a final iteration of Boundary Packing to $\overline{\mathcal{C}}$ yields that all remaining disks are packed into $\overline{\mathcal{C}}$. This concludes the proof. ◀

4.6 Analysis of the Algorithm for the Case $0.495 \leq r_1$

We prove that all disks are packed if $0.495 \leq r_1$ by distinguishing whether $0.495 \leq r_1 \leq \frac{1}{2}$ or $\frac{1}{2} < r_1$. If $0.495 \leq r_1 \leq \frac{1}{2}$, we apply a similar approach as used for the case $r_1 \leq 0.495$. The additional difficulty for the case of $0.495 \leq r_1 \leq \frac{1}{2}$ is that the cone induced by r_1 may have a density of $\frac{1}{2}$. Thus, we have to generate some extra potential from the remaining sectors in order to ensure that the gaps of the rings are saturated, see [3] for details.

▶ **Lemma 20.** *If $0.495 \leq r_1 \leq \frac{1}{2}$, our algorithm packs all disks into the container disk.*

If $\frac{1}{2} < r_1$, we need to refine our analysis because the midpoint of the container disk $\mathcal{C}$ lies inside r_1. In particular, we consider a *half disk* H lying inside $\mathcal{C}$ such that H and r_1 are touching each other. The volume of H is at least twice the volume of the remaining disks to be packed, see Figure 17. Finally, applying a similar approach as used in the case of $0.495 \leq r_1 \leq \frac{1}{2}$ to H yields that all disks are packed, see [3] for details.

▶ **Lemma 21.** *If $\frac{1}{2} < r_1$, our algorithm packs all disks into the original container disk.*

Lemma 21 concludes the proof of Theorem 1.

5 Hardness

It is straightforward to see that the hardness proof for packing disks into a square can be adapted to packing disks into a disk, as follows.

▶ **Theorem 22.** *It is* NP*-hard to decide whether a given set of disks fits into a circular container.*

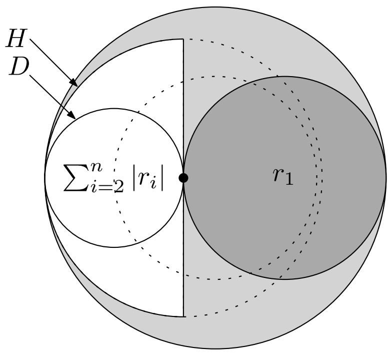

Figure 17 The total volume of the remaining disks to be packed is smaller than the volume of the white disk D. As $|H| = 2|D|$, it suffices to guarantee that H is saturated.

The proof is completely analogous to the one by Demaine, Fekete, and Lang in 2010 [2], who used a reduction from 3-Partition. Their proof constructs a disk instance which first forces some symmetrical free "pockets" in the resulting disk packing. The instance's remaining disks can then be packed into these pockets if and only if the related 3-Partition instance has a solution. Similar to their construction, we construct a symmetric triangular pocket by using a set of three identical disks of radius $\frac{\sqrt{3}}{2+\sqrt{3}}$ that can only be packed into a unit disk by touching each other. Analogous to [2], this is further subdivided into a sufficiently large set of identical pockets. The remaining disks encode a 3-Partition instance that can be solved if and only if the disks can be partitioned into triples of disks that fit into these pockets.

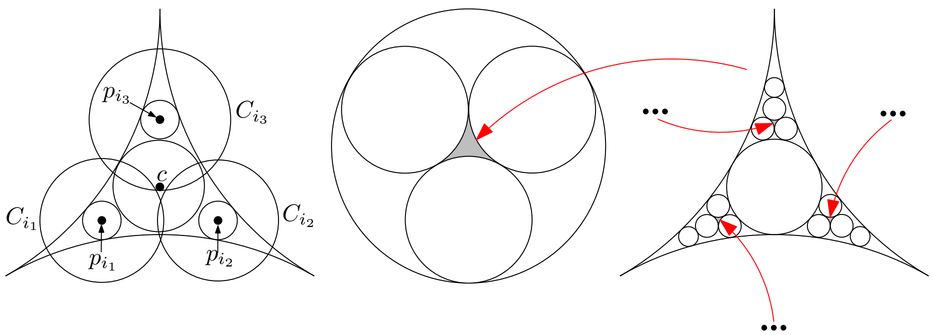

Figure 18 Elements of the hardness proof: (1) A symmetric triangular pocket from [2], allowing three disks with centers p_{i_1}, p_{i_2}, p_{i_3} to be packed if and only if the sum of the three corresponding numbers from the 3-Partition instance is small enough. (2) Creating a symmetric triangular pocket in the center by packing three disks of radius $\frac{\sqrt{3}}{2+\sqrt{3}}$ and the adapted argument from [2] for creating a sufficiently large set of symmetric triangular pockets.

6 Conclusions

We have established the critical density for packing disks into a disk, based on a number of advanced techniques that are more involved than the ones used for packing squares or disks into a square. Numerous questions remain, in particular the critical density for packing disks of bounded size into a disk or the critical density of packing squares into a disk. These remain for future work; we are optimistic that some of our techniques will be useful.

References

1 I. Castillo, F. J. Kampas, and J. D. Pintér. Solving circle packing problems by global optimization: numerical results and industrial applications. *European Journal of Operational Research*, 191(3):786–802, 2008.

2 E. D. Demaine, S.P. Fekete, and R. J. Lang. Circle Packing for Origami Design is Hard. In *Origami^5: 5th International Conference on Origami in Science, Mathematics and Education*, AK Peters/CRC Press, pages 609–626, 2011. `arXiv:1105.0791`.

3 S. P. Fekete, P. Keldenich, and C. Scheffer. Packing Disks into Disks with Optimal Worst-Case Density. *Computing Research Repository (CoRR)*, 2019. `arXiv:1903.07908`.

4 S. P. Fekete, S. Morr, and C. Scheffer. Split Packing: Algorithms for Packing Circles with Optimal Worst-Case Density. *Discrete & Computational Geometry*, 2018. `doi:10.1007/s00454-018-0020-2`.

5 S. P. Fekete and J. Schepers. New Classes of Fast Lower Bounds for Bin Packing Problems. *Math. Program.*, 91(1):11–31, 2001.

6 S. P. Fekete and J. Schepers. A General Framework for Bounds for Higher-Dimensional Orthogonal Packing Problems. *Math. Methods Oper. Res.*, 60:311–329, 2004.

7 F. Fodor. The Densest Packing of 19 Congruent Circles in a Circle. *Geometriae Dedicata*, 74:139–145, 1999.

8 F. Fodor. The Densest Packing of 12 Congruent Circles in a Circle. *Beiträge zur Algebra und Geometrie (Contributions to Algebra and Geometry)*, 41:401–409, 2000.

9 F. Fodor. The Densest Packing of 13 Congruent Circles in a Circle. *Beiträge zur Algebra und Geometrie (Contributions to Algebra and Geometry)*, 44:431–440, 2003.

10 H. J. Fraser and J. A. George. Integrated container loading software for pulp and paper industry. *European Journal of Operational Research*, 77(3):466–474, 1994.

11 M. Goldberg. Packing of 14, 16, 17 and 20 circles in a circle. *Mathematics Magazine*, 44:134–139, 1971.

12 R.L. Graham, B.D. Lubachevsky, K.J. Nurmela, and P.R.J. Östergøard. Dense Packings of Congruent Circles in a Circle. *Discrete Mathematics*, 181:139–154, 1998.

13 M. Hifi and R. M'hallah. A literature review on circle and sphere packing problems: models and methodologies. *Advances in Operations Research*, 2009. Article ID 150624.

14 P. Hokama, F. K. Miyazawa, and R. C. S. Schouery. A bounded space algorithm for online circle packing. *Information Processing Letters*, 116(5):337–342, May 2016.

15 S. Kravitz. Packing cylinders into cylindrical containers. *Mathematics Magazine*, 40:65–71, 1967.

16 R. J. Lang. A computational algorithm for origami design. *Proceedings of the Twelfth Annual Symposium on Computational Geometry (SoCG)*, pages 98–105, 1996.

17 J. Y. T. Leung, T. W. Tam, C. S. Wong, G. H. Young, and F. Y. L. Chin. Packing squares into a square. *Journal of Parallel and Distributed Computing*, 10(3):271–275, 1990.

18 B.D. Lubachevsky and R.L. Graham. Curved Hexagonal Packings of Equal Disks in a Circle. *Discrete & Computational Geometry*, 18:179–194, 1997.

19 H. Melissen. Densest Packing of Eleven Congruent Circles in a Circle. *Geometriae Dedicata*, 50:15–25, 1994.

20 F. K. Miyazawa, L. L. C. Pedrosa, R. C. S. Schouery, M. Sviridenko, and Y. Wakabayashi. Polynomial-time approximation schemes for circle packing problems. In *Proceedings of the 22nd European Symposium on Algorithms (ESA)*, pages 713–724, 2014.

21 J. W. Moon and L. Moser. Some packing and covering theorems. In *Colloquium Mathematicae*, volume 17, pages 103–110. Institute of Mathematics, Polish Academy of Sciences, 1967.

22 S. Morr. Split Packing: An Algorithm for Packing Circles with Optimal Worst-Case Density. In *Proceedings of the Twenty-Eighth Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 99–109, 2017.

23 N. Oler. A finite packing problem. *Canadian Mathematical Bulletin*, 4:153–155, 1961.

24 R. Peikert, D. Würtz, M. Monagan, and C. de Groot. Packing circles in a square: A review and new results. In *Proceedings of the 15th IFIP Conference*, pages 45–54, 1992.
25 G.E. Reis. Dense Packing of Equal Circles within a Circle. *Mathematics Magazine*, issue 48:33–37, 1975.
26 E. Specht. Packomania, 2015. URL: `http://www.packomania.com/`.
27 K. Sugihara, M. Sawai, H. Sano, D.-S. Kim, and D. Kim. Disk packing for the estimation of the size of a wire bundle. *Japan Journal of Industrial and Applied Mathematics*, 21(3):259–278, 2004.
28 P. G. Szabó, M. C. Markót, T. Csendes, E. Specht, L. G. Casado, and I. García. *New Approaches to Circle Packing in a Square*. Springer US, 2007.
29 H. Wang, W. Huang, Q. Zhangn, and D. Xu. An improved algorithm for the packing of unequal circles within a larger containing circle. *European Journal of Operational Research*, 141(2):440–453, September 2002.
30 D. Würtz, M. Monagan., and R. Peikert. The history of packing circles in a square. *Maple Technical Newsletter*, page 35–42, 1994.