Improved Bound for Online Square-into-Square Packing *

Brian Brubach

Department of Computer Science,
University of Maryland–College Park, College Park, MD, USA
bbrubach@cs.umd.edu

Abstract. We show a new algorithm and improved bound for the online square-into-square packing problem using a hybrid shelf-packing approach. This 2-dimensional packing problem involves packing an online sequence of squares into a unit square container without any two squares overlapping. We seek the largest area α such that any set of squares with total area at most α can be packed. We show an algorithm that packs any online sequence of squares with total area $\alpha \leq 2/5$, improving upon recent results of 11/32 [3] and 3/8 [8]. Our approach allows all squares smaller than a chosen maximum height h to be packed into the same fixed height shelf. We combine this with the introduction of variable height shelves for squares of height larger than h. Some of these techniques can be extended to the more general problems of rectangle packing with rotation and bin packing.

Keywords: Packing · online problems · packing squares · packing rectangles

1 Introduction

In packing problems, we wish to place a set of objects into a container such that no two objects overlap. These problems have been studied extensively and have numerous applications. However, even common one-dimensional versions of packing problems, such as the Knapsack problem, are NP-hard. For such difficult problems, it is often important to know whether it is even feasible to pack a given set into a particular container. In the two-dimensional case, it is worth noting that merely checking whether a given set of squares can be packed into a unit square was shown to be NP-hard by Leung, et al. [4].

To address this fundamental feasibility question, the square-into-square packing problem asks, "What is the largest area α such that any set of squares with total area α can be packed into a unit square without overlapping?" It is trivial to show that an upper bound is $\alpha \leq 1/2$. Two squares of height $1/2 + \epsilon$ cannot be packed into a unit square container. In addition, Moon and Moser [1] showed in 1967 that the bound of 1/2 is tight in the offline case. Squares can be sorted

* This work has been supported partially by NSF award CCF-1218620. I also want to thank my mentor Prof. Gandhi for all his support.

in decreasing order and packed left-to-right into horizontal "shelves" starting along the bottom of the container. The height of each shelf is set by the largest object in the shelf and when a shelf fills, a new one is opened directly above it.

In the online version of the problem, we have no knowledge of the full set of squares to be packed. Squares are received one at a time and each must be packed before seeing the next square. Once a square is packed, it cannot be moved. Thus, the successful offline approach cannot be used as it requires sorting the set. Although various techniques have been employed to achieve lower bounds for the online case, the most recent work revisits the idea of shelves in a novel way. The current best lower bound for the online version is $\alpha \geq 11/32$ by Fekete and Hoffmann [3] in 2013. They took a dynamic, multi-directional shelf-packing approach that allocates perpendicular shelves within other larger shelves.

1.1 Related work

Offline Square Packing. Early related work involved packing a set of objects into the smallest possible rectangle container. Moser [5] posed this question in 1966: "What is the smallest number A such that any family of objects with total area at most 1 can be packed into a rectangle of area A?" Since then, there have been many results for the offline packing of squares into rectangle containers.

In 1967, Moon and Moser [1] showed that any set of squares with total area 1 can be packed into a square of height $\sqrt{2}$. This established $A \leq 2$ or in the terms of our problem, $\alpha \geq 1/2$. This result was followed by several improvements on the value of A using rectangular containers. The current best upper bound is 1.3999 by Hougardy [6] in 2011.

Online Square Packing. In 1997, Januszewski and Lassak [7] considered the online variant in many dimensions. For two-dimensional square-into-square packing, their work showed a bound of $\alpha \geq 5/16$ by recursively dividing a unit square container into rectangles of aspect ratio $\sqrt{2}$. In 2008, Han et al. [2] used a similar approach to improve the lower bound to 1/3. Januszewski and Lassak [7] also considered the general problem of packing d-dimensional cubes into a unit cube and for for $d \geq 5$, showed a tight bound of $2\left(\frac{1}{2}\right)^d$. For $d = 3$ or 4, they showed cubes of total volume $3/2(1/2)^d$ could be packed.

Fekete and Hoffmann [3] provided a new approach in 2013 which uses multi-directional shelves (horizontal and vertical) that are allocated dynamically within other larger shelves. Using this technique, they were able to improve the lower bound further to 11/32. More recently, the author of this paper improved the bound to 3/8 in an unpublished paper [8]. That paper used a multi-directional shelf approach similar to [3] as well as some new techniques which will be included in this paper.

1.2 Our Contributions

We show a new lower bound of $\alpha \geq 2/5$ for online square-into-square packing, improving upon the previous results. Our algorithm combines dynamic, multi-directional *fixed* height shelves with *variable* height shelves. Fixed height shelves are assigned a height at the moment they are opened. Variable height shelves are not assigned a height until they are filled, at which point their height is considered to be the height of the tallest square in the shelf.

One of the key challenges in this problem is packing squares of greatly varying heights. In particular, if squares differ in height by a factor greater than 2, it is difficult to pack them together without creating a lot of wasted space. However, our new approach to multi-directional fixed height shelves allows us to pack all squares with height at most h (in our application $h = 1/6$) together into the same fixed height shelves of height h with very little expense of wasted space. Similarly, we show how to use variable height shelves to pack all squares with height greater than h together.

We introduce new criteria for determining the dimensions of vertical shelves which are dynamically allocated within the horizontal shelves. This allows us to completely avoid the use of *buffer regions* such as those added to the ends of shelves in [3] and minimize wasted space. Our use of variable shelves shared by all larger squares handles many such squares and with fewer special cases. These results may be of independent interest for other 2-dimensional packing problems. Our technique for multi-directional shelf packing can be used for rectangles (if rotation is allowed) as well as squares.

Another interesting future direction for this work would be bin packing problems. Many algorithms for these problems take a natural approach of segregating squares of different sizes into separate bins and maintaining several open bins, each devoted to a different size class. Combining all or most sizes of squares or rectangles into the same bin could be useful in versions of this problem with special restrictions. For instance, there may be a cost associated with switching back-and-forth between different bins or we may require a parallel algorithm in which multiple bins are packed simultaneously.

Finally, we note that our ratio of lower to upper bounds $\left(\frac{2/5}{1/2}\right)$ is the first 2-dimensional result which is tighter than the bounds shown in [7] for the 3 and 4-dimensional versions of this problem. This combined with the aforementioned tight results for dimension at least 5, suggests that new improvements may be possible in the 3 and 4-dimensional cases as well.

1.3 Outline

In section 2, we discuss preliminaries including terminology and notation. In section 3, we present our algorithm for the online packing of squares into a unit square container. In section 4, we analyze our algorithm and show that it successfully packs any online series of squares with total area at most 2/5.

2 Preliminaries

2.1 Terminology

We define a *shelf* S as a subrectangle in the container with height h and length ℓ. For *variable* height shelves, h is equal to the height of the largest square in the shelf. For *fixed* height shelves, the height is determined in advance along with a packing ratio r, $0 < r < 1$. The *packing ratio* is the ratio of the smallest possible height to the largest possible height of squares that can be packed into S. Any square packed into fixed shelf S must have height k, $h \geq k > hr$.

When *packing* a shelf S, squares are added side-by-side to S. In our algorithm, we also pack small vertical fixed shelves into larger horizontal fixed shelves. These vertical shelves are also packed side-by-side with squares and other vertical shelves as seen in Figure 1.

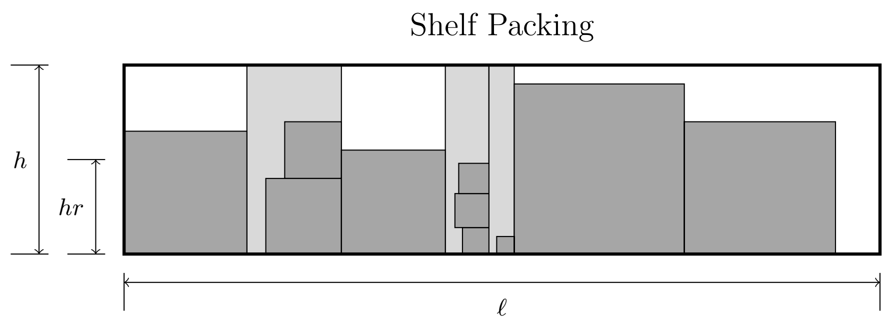

Fig. 1. Illustration of shelf packing with $r = 1/2$ for the horizontal shelf. Vertical shelves are designated with a light gray background and have different packing ratios.

In addition, shelves may be considered *open* or *closed*. A shelf S is initially considered open. As squares are added to S, we may receive a square Q with height h_Q, $h \geq h_Q > hr$, such that packing Q into S would exceed the length of S. At this point, we say that S is closed. The new square Q is then packed into some other shelf which is open. When analyzing shelves, we refer to the *used length* l_u of a shelf to describe the length of that shelf which is occupied by squares or vertical shelves.

In the analysis, we will *assign* fractions of the input area to regions or shelves within the container. Typically, the area assigned to a region represents squares which have been packed into that region. However, area assigned to a region may also come from a square packed into some other region subject to the following commonsense rule: A single square may have parts of its area assigned to different regions as long as the sum of those parts is at most the area of the square itself.

We use the term *density* to describe the ratio of the assigned area of a region to the total area of that region. In this paper, we will often count the total area packed into the container in the following way. Let there be a region R with available area A. We show that most of R has been assigned area to a density of $1/2$ with the exception of some small portion of wasted space with area at most W and a density of 0. We then calculate the total area assigned to R as $A/2 - W/2$. We call this $W/2$ value the *waste*.

2.2 Size Classes of Squares

We divide possible input squares into four classes based on height:

- **Large:** height $> \frac{1}{3}$ (also $< 2/3$ due to our input having area at most $2/5$)
- **Medium:** height $\leq \frac{1}{3}$ and $> \frac{1}{6}$
- **Small:** height $\leq \frac{1}{6}$ and $> \frac{1}{12}$
- **Very Small:** height $\leq \frac{1}{12}$

We also refer to small squares as class c_0 and subdivide the very small squares into subclasses c_i, $i \geq 1$. Squares in c_i are packed into shelves with max height h_i and packing ratio r_i. They have height k_i, $h_i \geq k_i > h_i r_i$. We use the notation c_{j+} to refer to all c_i, $i \geq j$.

In our algorithm, we assign ratios as follows: $r_0 = 0.5$, $r_1 = 0.71$, $r_2 = 0.65$, and $r_{3+} = 0.58$. To account for all small and very small squares with height $\leq \frac{1}{6}$, we let $h_0 = \frac{1}{6}$ and for all $i \geq 1$, we set $h_i = h_{i-1} r_{i-1}$. In section 4.2, we will show that these ratios ensure closed vertical shelves for very small squares will have a density greater than 0.5, which is the packing ratio for small squares.

3 Algorithm

For each square, we pack it according to a subroutine based on its size. Very small squares are a special case. They are packed based on their subclass c_i, $i \geq 1$. For each subclass, we maintain exactly one open vertical shelf at any given time. When we receive a very small square, we attempt to add it to the appropriate vertical shelf for its height class. If this new square does not fit, we close that shelf and open a new one. The new vertical shelf itself is packed into the container as if it is a small square. We will show in the analysis that these new vertical shelves can be treated the same as small squares. As such, we only discuss large, medium, and small squares in the description of our algorithm (Figure 2).

Small: We first alternate packing small squares from left-to-right into the top and bottom shelves of the initial packing region. Each time, we choose the shelf with the shortest used length. Eventually, these shelves will fill up. The first square we receive which doesn't fit into these shelves is packed into shelf M_0 in the main packing region. The second such square is packed into M_1. This can be seen in part (B) of Figure 2.

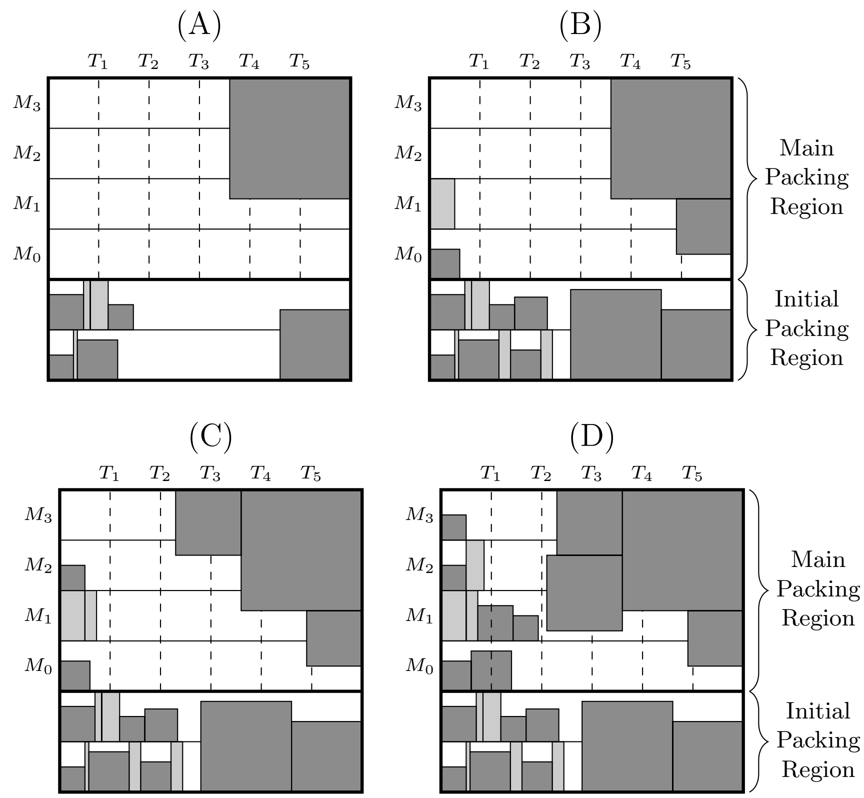

Fig. 2. Illustration of packing algorithm. The light gray rectangles represent vertical shelves for very small squares packed into the shelves for small squares. The small shelves in the main region are labeled M_0 through M_3 with thresholds labelled T_1 through T_5. (A) shows a large square in the upper right corner, a medium square in the lower right, and a set of small and very small squares in the lower left. (B) shows the first two "small" squares added to the main region on the left as well as a medium square added to the variable shelf on the right. (C) and (D) illustrate the continuation of packing into the main region.

After those first two small squares have been packed into the main region, all future small squares are packed left-to-right into the main region according to the following rule. We start packing in the shelf M_1 up to threshold T_1. Let M_i be the current shelf and T_j be the current threshold. When we receive a small square Q that would cause the used length of M_i to exceed threshold T_j, we pack Q into the shelf $M_{(i+1) \bmod 4}$ and that shelf becomes the current shelf. Each time we return to M_0 we increment the threshold to T_{j+1}. Each threshold T_j is at a distance of $j/6$ from the left side of the container.

Medium: Medium squares are first packed from right-to-left into the initial region. When we receive medium squares which don't fit into this region, we pack them into main packing region.

In the main region, medium squares will be packed together with large squares in vertical variable shelves from top-to-bottom, starting in the upper right corner. Each time we close one of these shelves, the new one is opened immediately to the left of the previous shelf. We continue adding shelves from right-to-left until the remainder of the input is small enough that no more medium or large squares can be received.

Large: Except in one special case, large squares are packed together with medium squares in the main region as described above. The special case is when we receive a third large square (there can be at most three since the input is at most $2/5$). In this case, the third large square is packed into the initial region as if it is a medium square.

4 Analysis

We will show that any input which cannot be packed by our algorithm must have total area greater than $2/5$. In Section 4.1, we cover basic shelf packing and how we assign area to fixed shelves. In Section 4.2, we focus on vertical shelves, showing why they can be treated as small squares (given a sacrifice of $0.235/12$ waste due to maintaining one open vertical shelf for each size class). In Section 4.3, we bound the waste in the initial region under different circumstances. In Section 4.4, we bound the density of variable shelves for medium/large squares and the waste due to small shelves in the main region. In Section 4.5, we show that $\alpha \geq 2/5$ for the online square-into-square packing problem.

4.1 Shelf Packing

In our analysis, it is important to determine the area assigned to open and closed shelves. The foundation for many of our lemmas is a generalization of a lemma due to Moon and Moser [1]. The results in this section and Section 4.2 can be extended to the setting of rectangles with rotation at the expense of some additional wasted space. We show this in the appendix.

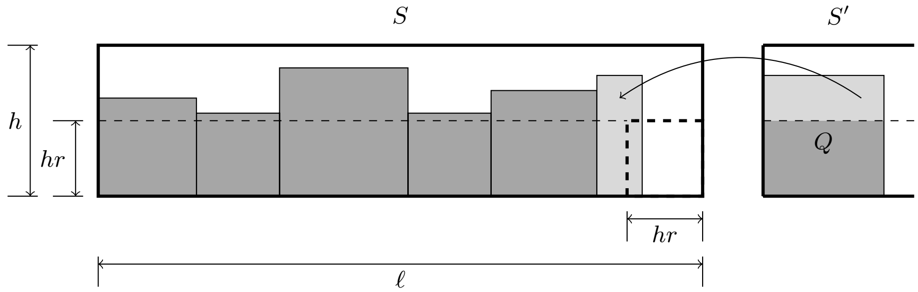

Fig. 3. Illustration of Lemma 1 and Corollary 2 for the case when $r = 1/2$. The upper portion of the square Q is assigned to S, while the lower portion is assigned to S'.

Lemma 1. *Let S be a shelf with height h, length ℓ and ratio r, $0 < r < 1$, that is packed with a set P of squares with height $\leq h$ and $> hr$. Let Q be the first square with height h_Q, $h \geq h_Q > hr$, that does not fit into S. The total area of all the squares packed into S plus the area of Q is greater than $\ell hr - (hr)^2 + h_Q hr$.*

Proof. See Appendix. □

Corollary 2. *We can assign an area of $\ell hr - (hr)^2$ to every closed shelf S.*

Proof. See Appendix. □

4.2 Small and Very Small Shelves

In this section, we show that vertical shelves for very small squares can be packed into small shelves as if they are small squares. Formally, we extend Lemma 1 and Corollary 2 to small shelves containing closed vertical shelves and show that we can bound the waste due to open vertical shelves. Refer to section 2.2 for an overview of how we subdivide the small and very small classes.

For Lemma 1, the used length of a shelf must have density equal to the packing ratio r. Since $r_0 = \frac{1}{2}$, any used section of a small shelf with only c_0 squares satisfies this trivially. We need vertical shelves to supply the same guarantee. Note that vertical shelves within small shelves have a length equal to h_0 (the height of small shelves), which is at least twice the height of any very small shelf.

Lemma 3. *Let S be a shelf with height h, length ℓ and packing ratio r, $0 < r < 1$. If $\ell \geq 2h$, we can choose a value for r, such that the area assigned to S is at least $\frac{\ell h}{2}$ when S is closed.*

Proof. See Appendix. □

Summary of Heights, Ratios, and Packing Densities:

Height	Ratio	Packing Density
$h_0 = 1/6$	$r_0 = 0.5$	> 0.5
$h_1 \approx 0.08333$	$r_1 = 0.71$	$0.71^2 > 0.5$
$h_2 \approx 0.05917$	$r_2 = 0.65$	$r - \frac{hr^2}{l} = 0.65 - \frac{0.05917 * 0.65^2}{0.25} > 0.5$
$h_3 \approx 0.03846$	$r_{3+} = 0.58$	$r - \frac{hr^2}{l} > 0.58 - \frac{0.03846 * 0.58^2}{0.25} > 0.5$

For simplicity of analysis, we've assigned a ratio of 0.58 for all c_{3+}. We can do this because the packing density only increases with shorter heights as long as the ratio and length remain the same. In short, for $i \geq 3$, we have $h_i = h_{i-1} r_{i-1}$, $r_i = 0.58$, and density greater than 0.5.

Lemma 4. *The waste due to open vertical shelves in the entire container is at most* $\frac{0.235}{12}$ *and subtracting this number from the sum of all assigned areas allows us to consider all vertical shelves closed.*

Proof. See Appendix. □

Lemma 5. *Lemma 1 and Corollary 2 can be extended to small horizontal shelves with vertical shelves packed into them at the expense of* $\frac{0.235}{12}$ *waste subtracted from the total area assigned to the whole container.*

Proof. See Appendix. □

4.3 Waste in the Initial Packing Region

Our algorithm packs this region from left-to-right with small and very small squares and right-to-left with medium squares and possibly one large square. In this section, we define the *empty length* E of the initial packing region as the distance between the rightmost small or very small square and the leftmost medium or large square. In other words, the distance between these two groups.

While used portions of these shelves will have a density of 1/2, there may be waste due to unevenly packed small shelves or the empty length. We start by analyzing the waste due to unevenly packing the two small shelves in alternating fashion. Then, we address the waste due to the empty length. Finally, we show that this waste is reduced once we pack small squares into the main region.

Lemma 6. *The waste due to unevenly packed small shelves is at most* $1/72$.

Proof. See Appendix. □

Lemma 7. *The waste due to empty length* E *is at most* $E/6$.

Proof. See Appendix. □

Lemma 8. *After receiving two small squares which do not fit in the initial region, the waste in the initial region is at most 1/72.*

Proof. See Appendix. □

4.4 The Main Packing Region

In this section, we first consider the section B in the main region which is occupied by variable shelves for medium and large squares. Then, we consider the area C occupied by fixed shelves for small and very small squares.

Lemma 9. *Let B be the rectangular section in the main packing region containing all closed variable height shelves. Let B also include the open variable shelf if that shelf contains one large square or at least two medium squares. The density of B is at least $1/2$.*

Proof. See Appendix. □

Lemma 10. *Let C be the rectangular section in the main packing region containing all small squares. If the small shelf with the longest used length is one of the top two shelves (M_2 or M_3) the waste in C is at most $5/144$. Otherwise, the waste in C is at most $9/144$. The remaining portion of C has a density of $1/2$.*

Proof. See Appendix. □

4.5 Improved Bound for Online Square-into-Square Packing

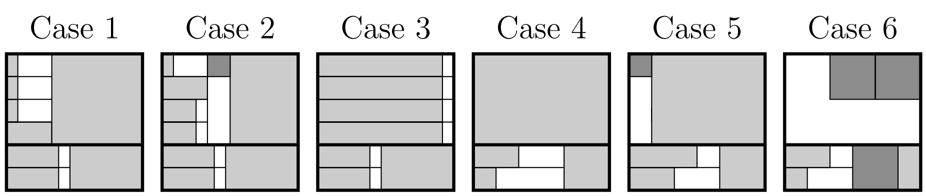

Fig. 4. Illustrations of each case in Theorem 11. Light gray sections have been assigned enough area to have a density of 1/2. Dark gray squares represent medium squares in Cases 2 and 5 and large squares in Case 6. The white sections are wasted space.

Theorem 11. *Any set of squares with total area at most $2/5$, which is received in an online fashion, can be packed into a unit square container.*

Proof. Assume for contradiction that the total area of the input is at most $2/5$ and some square Q does not fit. We divide the proof into cases based on which size classes of squares have been packed into the main region. As before, we use the term small to refer to both small and very small squares.

In cases 1 and 2, the main region has small squares as well as medium and/or large squares. In case 3, it has only small squares. In cases 4 and 5, it has medium squares and may have large squares. In case 6, it has only large squares. We will show that each case can only occur if the input is greater than 2/5.

Case 1: The main region contains small squares as well as medium and/or large squares and the most recent variable height shelf contains at least one large square or at least two medium squares.

Here, we consider the intersection of the most recent variable shelf and the fixed height shelf with the longest used length. By Lemmas 9 and 10, the main region has waste at most 9/144 due only to small shelves. By Lemmas 4 and 8, the waste in the initial region is at most $0.235/12 + 1/72$. Then the total area is

$$1/2 - 9/144 - 0.235/12 - 1/72 > 2/5$$

Case 2: The main region contains small squares as well as medium and/or large squares and the most recent variable height shelf contains exactly one medium square.

Again, we consider the intersection of the most recent variable shelf and the fixed height shelf with the longest used length. Let S be the intersecting variable shelf. By Lemma 9, the area to the right of S has no waste. Note that since S has exactly one medium square, it must intersect one of the top two small shelves. Then by Lemma 10, the area to the left of S has waste at most 5/144. We now consider the waste in S itself. Let M with height h_M be the medium square in S. The waste due to S is $(h_M \cdot 2/3)/2 - h_M^2$. As in the previous case, by Lemmas 4 and 8, the waste in the initial region is at most $0.235/12 + 1/72$.

$$1/2 - 5/144 - 0.235/12 - 1/72 - (h_M \cdot 2/3)/2 + h_M^2 > 2/5$$

Case 3: There are no medium or large squares in the main region.

As in previous cases, the initial region will account for an area of $1/6 - 0.235/12 - 1/72$. In the main region, we can apply Corollary 2 to the bottom three shelves and Lemma 1 to the topmost shelf. By design each time some square closes a shelf, there is room for that square in the shelf above it until we receive some square which would close the top shelf. Since Corollary 2 assigns an area less than Lemma 1, we can apply it to all four shelves to account for a total area of $4(\ell hr - (hr)^2) = 4(1 \cdot 1/6 \cdot 1/2 - 1/12^2) = 1/3 - 1/36$.

$$1/6 - 0.235/12 - 1/72 + 1/3 - 1/36 > 2/5$$

Case 4: The main region contains at least one medium square, may contain large squares, and contains no small squares. The final variable shelf contains at least one large square or at least two medium squares.

In this case, we consider an input which causes the variable shelves to extend beyond the left edge of the container. By Lemma 9, the area occupied by these variable shelves (which includes the entire main region) has a density of 1/2. However, unlike previous cases, there may be empty space in the initial region of at most 1/3 since that is the height of the biggest possible medium square.

By Lemma 7, the waste due to that empty space is at most $1/3 \cdot 1/6 = 1/18$. So the waste in the initial region is at most $0.235/12 + 1/72 + 1/18$.

$$1/2 - 0.235/12 - 1/72 - 1/18 > 2/5$$

Case 5: The main region contains at least one medium square, may contain large squares, and contains no small squares. The final variable shelf contains exactly one medium square.

Again, we consider an input which causes the variable shelves to extend beyond the left edge of the container. Let S be the open variable shelf containing a single medium square M with height h_M. By Lemma 9, the area to the right of S has no waste. As before, the waste due to S is $(h_M \cdot 2/3)/2 - h_M^2$. Since M was packed into the main region, the empty length in the initial region is at most h_M. Then by Lemma 7, the waste due to that empty space is at most $h_M \cdot 1/6 = h_M/6$. So the waste in the initial region is at most $0.235/12 + 1/72 + h_M/6$.

$$1/2 - 0.235/12 - 1/72 - h_M/6 - (h_M \cdot 2/3)/2 + h_M^2 > 2/5$$

Case 6: Three large squares are received before any small or medium squares are packed into the main region. This is the special case in which we need to pack a large square into the initial region.

Let Q be the third large square with height h_Q. Suppose there isn't room in the initial region for Q or some small/medium square received after Q. The first two large squares represent an area of at least $2/9$ and the area of Q is ${h_Q}^2 > 1/9$. As in case 3, the initial region can be assigned an area of at least $1/6$ minus waste. The empty length can be at most h_Q, otherwise Q would fit.

$$2/9 + {h_Q}^2 + 1/6 - h_Q/6 - 0.235/12 - 1/72 > 2/5$$

□

References

1. J. Moon and L. Moser. Some packing and covering theorems. *Colloq. Mathematicum* 17, 103–110, 1967
2. X. Han, K. Iwama, G. Zhang. Online removable square packing. *Theory of Comp. Systems* 43(1), 38–55, 2008
3. S. Fekete and H. Hoffmann. Online Square-into-Square Packing. *In Proc. of the 16th Intl. Workshop on Approximation Algorithms for Combinatorial Optimization Problems*, 126–141, 2013.
4. J.Y.-T. Leung, T.W. Tam, C.S. Wong, G.H. Young, F.Y.L. Chin. Packing squares into a square. *J. Parallel and Dist. Comp.* 10(3), 271–275, 1990
5. L. Moser. Poorly formulated unsolved problems of combinatorial geometry. Mimeographed, 1966
6. S. Hougardy. On packing squares into a rectangle. *Computational Geometry: Theory and Applications* 44(8), 456–463, 2011
7. J. Januszewski, M. Lassak. On-line packing sequences of cubes in the unit cube. *Geometriae Dedicata* 67(3), 285–293, 1997
8. B. Brubach. Improved Online Square-into-Square Packing. 2014 URL http://arxiv.org/abs/1401.5583

5 Appendix

Lemma 1. *Let S be a shelf with height h, length ℓ and ratio r, $0 < r < 1$, that is packed with a set P of squares with height $\leq h$ and $> hr$. Let Q be the first square with height h_Q, $h \geq h_Q > hr$, that does not fit into S. The total area of all the squares packed into S plus the area of Q is greater than $\ell hr - (hr)^2 + h_Q hr$.*

Proof. Given that Q does not fit into S, the used section of S must have length at least $\ell - h_Q$. Since every square in P has height greater than hr, it must also be the case that the set P covers an area of at least $hr(\ell - h_Q) = \ell hr - h_Q hr$. The area of Q is h_Q^2. Combining these, we get a covered area of at least $\ell hr - h_Q hr + h_Q^2 = \ell hr - h_Q hr + h_Q^2 + (hr)^2 - (hr)^2 + h_Q hr - h_Q hr > \ell hr - (hr)^2 + h_Q hr$

□

Corollary 2. *We can assign an area of $\ell hr - (hr)^2$ to every closed shelf S.*

Proof. By Lemma 1, the covered area of S plus the area of Q is greater than $\ell hr - (hr)^2 + h_Q hr$. Let S' be the open shelf holding Q. We can assign the area $\ell hr - (hr)^2$ to S and let $h_Q hr$ be assigned to S'. This ensures that the used section of S' maintains a density equal to its packing ratio. Thus we can repeat this process when we eventually receive some square Q' which closes S'.

This is illustrated in Figure 3. In the picture, the top portion of Q is assigned to S while the lower portion is assigned to S'.

□

Lemma 3. *Let S be a shelf with height h, length ℓ and packing ratio r, $0 < r < 1$. If $\ell \geq 2h$, we can choose a value for r, such that the area assigned to S is at least $\frac{\ell h}{2}$ when S is closed.*

Proof. We consider two cases:

Case 1: $\ell = 2h$.

Notice that if $r > 2/3$, any closed shelf S must contain exactly two squares. Then if $r \geq \sqrt{1/2} > 2/3$, the minimum packing density of a closed shelf S is $\frac{2(hr)^2}{\ell h} \geq \frac{2h^2(\sqrt{1/2})^2}{2h^2} = \frac{1}{2}$. So, for any $r \geq \sqrt{1/2}$, the total covered area is at least $\frac{\ell h}{2}$.

Case 2: $\ell \geq (2 + \epsilon)h$ for some $\epsilon > 0$.

In this case, $r - \frac{hr^2}{\ell} \geq r - \frac{r^2}{2+\epsilon}$ and we can choose an $r < 1$ such that $r - \frac{hr^2}{\ell} \geq \frac{1}{2}$. Multiplying through by ℓh gives us $\ell hr - (hr)^2 \geq \frac{\ell h}{2}$. Using Corollary 2, we can assign the area $\ell hr - (hr)^2$ to S when it closes. So S has been assigned an area of at least $\frac{\ell h}{2}$ when it is closed.

□

Lemma 4. *The waste due to open vertical shelves in the entire container is at most $\frac{0.235}{12}$ and subtracting this number from the sum of all assigned areas allows us to consider all vertical shelves closed.*

Proof. Recall that a new vertical shelf for a given very small size class is opened only when another must be closed. Thus, there is at most one open shelf for each size class at any time and the length of small shelf space wasted is at most the sum of the heights of all vertical shelves. Using the heights and ratios we have defined, this sum is at most 0.235. It follows that the space wasted is

$$\text{Sum of heights} \times \text{Height of shelf} \times \text{Density} \leq 0.235 \cdot \frac{1}{6} \cdot \frac{1}{2} \leq \frac{0.235}{12}$$

□

Lemma 5. *Lemma 1 and Corollary 2 can be extended to small horizontal shelves with vertical shelves packed into them at the expense of* $\frac{0.235}{12}$ *waste subtracted from the total area assigned to the whole container.*

Proof. Most of this proof follows from previous lemmas. By Lemma 4, if we subtract $\frac{0.235}{12}$ when calculating the total area packed into the container, we can treat all vertical shelves as if they are small squares.

As in Lemma 1, let S be a shelf packed with a set P of some combination of small squares and vertical shelves. Let Q be an additional square or vertical shelf that does not fit into S. Notice that P must still cover an area of $hr(\ell - h_Q) = \ell hr - h_Q hr$ since the vertical shelves have a density of 1/2

If Q is a small square, the proof in Lemma 1 applies. Otherwise, if Q is a vertical shelf, recall that the largest vertical shelf has the height $h_1 = 1/12$. If such a shelf causes S to close, then $\ell - h_Q > 11/12$ and P must cover an area of at least $hr(11/12) = \ell hr - (hr)^2$. □

Lemma 6. *The waste due to unevenly packed small shelves is at most* $1/72$*.*

Proof. Note that the biggest small square has a height of 1/6 and we always pack into the shelf with the shortest used length. Then, the difference in length between the shelves is at most 1/6. Since this space should have been packed at a density of 1/2, the total waste is at most

$$\text{Difference in length} \times \text{Height of shelf} \times \text{Density} \leq \frac{1}{6} \cdot \frac{1}{6} \cdot \frac{1}{2} \leq \frac{1}{72}$$

□

Lemma 7. *The waste due to empty length* E *is at most* $E/6$*.*

Proof. Similar to the previous proof.

$$\text{Empty length} \times \text{Height of both shelves} \times \text{Density} = E \cdot \frac{1}{3} \cdot \frac{1}{2} = \frac{E}{6}$$

□

Lemma 8. *After receiving two small squares which do not fit in the initial region, the waste in the initial region is at most 1/72.*

Proof. We can say that each of these small squares has closed one of the two initial shelves and apply Lemma 1 and Corollary 2. We can do this even with medium or large squares in the initial region, since the length occupied by such squares will still have the required density of at least 1/2. This shows that the waste in each shelf is at most $(h_0 r_0)^2 = 1/144$. Thus, the waste of both shelves is at most 1/72. □

Lemma 9. *Let B be a rectangular section in the main packing region containing closed variable height shelves. Let B also include the open variable shelf if that shelf contains one large square or at least two medium squares. The density of B is at least $1/2$.*

Proof. Recall that each of our variable shelves has a length of $2/3$ and height equal the height of the biggest square in the shelf. Medium squares can have height in $(1/6, 1/3]$ and large squares have height greater than $1/3$.

We first show that a shelf S containing one large square or at least two medium squares has a density of $1/2$. If S contains a large square, this is true simply because the height of any large square is greater than half the length of S. If S contains at least two medium squares, Let T denote the tallest square in S and t denote some other square also in S. Since the height h_t of t is greater than $1/6$ and the height of S is equal to the height h_T of T, it follows that the covered area of S is

$$\frac{\text{Area of } T + \text{Area of } t}{\text{Area of } S} > \frac{{h_T}^2 + (1/6)^2}{h_T \cdot 2/3} \geq \frac{1}{2}$$

For the closed shelves, we consider only closed shelves with exactly one medium square as other closed shelves satisfy the density requirement by the previous arguments. Note that such a shelf must have been closed by a large square since two medium squares cannot exceed the length of the shelf.

Let S be the shelf with one medium square which has been closed by a large square and S' be the shelf the large square is packed into. Let M with height h_M be the medium square in S and let L with height h_L be the large square in S'. Then the combined density of S and S' is

$$\frac{\text{Area of } L + \text{Area of } M}{\text{Area of } S \text{ and } S'} = \frac{{h_L}^2 + {h_M}^2}{(h_L + h_M) \cdot 2/3} > \frac{(2/3 - h_M)^2 + {h_M}^2}{(2/3 - h_M + h_M) \cdot 2/3} \geq \frac{1}{2}$$

□

Lemma 10. *Let C be the rectangular section in the main packing region containing all small squares. If the small shelf with the longest used length is one of the top two shelves (M_2 or M_3) the waste in C is at most $5/144$. Otherwise, the waste in C is at most $9/144$. The remaining portion of C has a density of $1/2$.*

Proof. Let $\ell_u^{\max}$ be the length of the longest used length. By lemmas 3 and 4, the used length of each small shelf has density of at least $1/2$. So the waste is due to the differences in used lengths. If a shelf S has used length ℓ_u, the waste in S is $(l_u^{\max} - \ell_u) \cdot 1/6 \cdot 1/2$.

We can apply Corollary 2 to shelves in the main region in the following way. Let M_i be the current shelf and T_j be the current threshold which is at a distance $j/6$ from the left side of the container. According to our algorithm, when we receive a small square Q that would cause the used length of M_i to exceed threshold T_j, we pack Q into the shelf $M_{(i+1) \bmod 4}$. Using Corollary 2, we can assign M_i an area of $\ell hr - (hr)^2 = j/6 \cdot 1/12 - 1/144$

If the shelf $S^{\max}$ with the longest used length has been packed up to some point between thresholds T_j and T_{j+1}, the following must be true. $\ell_u^{\max}$ is at most $(j+1)/6$. Shelves below $S^{\max}$ have been assigned an area of at least $(j+1)/6 \cdot 1/12 - 1/144$, which represents a waste of at most $1/144$. Shelves above it have been assigned an area of at least $j/6 \cdot 1/12 - 1/144$, which represents a waste of at most $1/144 + 1/72 = 3/144$.

The extra 1/72 is due to the wasted space of length1/6 in those shelves between T_j and T_{j+1}.

If the small shelf with the longest used length is one of the top two shelves (M_2 or M_3), the worst case is when there is one shelf above and two shelves below. In this case the waste in B is at most $3/144 + 1/144 + 1/144 = 5/144$. Otherwise, the worst case is when the bottom shelf M_0 has the longest used length with three shelves above it. In this case, the waste in B is at most $3/144 + 3/144 + 3/144 = 9/144$. □

5.1 Rectangles with Rotation

As stated previously, the results in Sections 4.1 and 4.2 can be extended to rectangles with rotation at the expense of some additional wasted space and we provide a brief sketch of this for the interested reader. The *with rotation* setting allows each rectangle to be rotated 90 degrees before being packed. Note that we do not believe our results using variable height shelves can be extended to rectangles with rotation and consider only the results using fixed height shelves. With the exception of case 1 in the proof of Lemma 3, the results in Sections 4.1 and 4.2 can be directly extended to this setting. To see this, apply the proof of Lemma 1 to some rectangle Q with height h_Q such that $h \geq h_Q > hr$ and length ℓ_Q such that $\ell_Q \leq h_Q$. The only remaining challenge is case 1 in the proof of Lemma 3, which does not rely on Lemma 1.

Note that this case deals only with vertical shelves for class c_1 squares with height exactly half the length of the shelf. We first make the following minor adjustment to the algorithm. We define rectangles as having height h and length ℓ such that $\ell \leq h$. We split the height class c_1 into c_1^A and c_1^B. Both classes have maximum height $h_1 = h_0 r_0$ and minimum height $h_1 r_1$. However, we now set $r_1 = 3/4$. We also define lengths $\ell_1^A > (2/3)h_1$ for c_1^A and $\ell_1^B \leq (2/3)h_1$ for c_1^B.

Each class will its own vertical shelves with height h_1, length $h_0 = 2h_1$, and area $2(h_1)^2$. Therefore we must show that closed shelves of both classes will have covered an area of at least $(h_1)^2$. Notice that closed shelves for c_1^A must contain exactly two rectangles with an area at least $2(h_1 \ell_1^A) > 2((h_1 \frac{3}{4})(h_1 \frac{2}{3})) = (h_1)^2$. For the class c_1^B, we can use an argument similar to that in Lemma 1. For a c_1^B shelf to close we must have already covered a section as long as its length minus $(2/3)h_1$ and as tall as $h_1 r_1 = (3/4)h_1$. This is $(2h_1 - (2/3)h_1) \cdot (3/4)h_1 = (h_1)^2$.

The additional wasted space results from two changes made to the algorithm. First, maintaining two separate types of shelves for c_1 doubles the space wasted by that size class. Second, increasing r_1 to 0.75 (as opposed to 0.71 in our square packing algorithm) means that every smaller size class will have slightly greater height and their open shelves will cause a little more wasted space. The exact amount of wasted space is determined by the choice of h_0.