Improved Lower Bound for Online Strip Packing

(Extended Abstract)

Rolf Harren[1] and Walter Kern[2]

[1] Max-Planck-Institut für Informatik (MPII)
Campus E1 4, 66123 Saarbrücken, Germany
rharren@mpi-inf.mpg.de

[2] University of Twente, Department of Applied Mathematics
P.O. Box 217, 7500 AE Enschede, The Netherlands
w.kern@utwente.nl

1 Introduction

In the two-dimensional strip packing problem a number of rectangles have to be packed without rotation or overlap into a strip such that the height of the strip used is minimal. The width of the rectangles is bounded by 1 and the strip has width 1 and infinite height.

We study the online version of this packing problem. In the online version the rectangles are given to the online algorithm one by one from a list, and the next rectangle is given as soon as the current rectangle is irrevocably placed into the strip. To evaluate the performance of an online algorithm we employ competitive analysis. For a list of rectangles L, the height of a strip used by online algorithm ALG and by the optimal solution is denoted by $\mathrm{ALG}(L)$ and $\mathrm{OPT}(L)$, respectively. The optimal solution is not restricted in any way by the ordering of the rectangles in the list. Competitive analysis measures the absolute worst-case performance of online algorithm ALG by its competitive ratio

$$\rho_{\mathrm{ALG}} = \sup_L \left\{ \frac{\mathrm{ALG}(L)}{\mathrm{OPT}(L)} \right\} .$$

Known Results. Regarding the upper bound on the competitive ratio for online strip packing, recent advances have been made by Ye, Han & Zhang[6] and Hurink & Paulus[3]. Independently they showed that a modification of the well-known shelf algorithm yields an online algorithm with competitive ratio $7/2 + \sqrt{10} \approx 6.6623$. We refer to these two papers for a more extensive overview of the literature.

In the early 80s, Brown, Baker & Katseff[1] derived a lower bound $\rho \geq 2$ on the competitive ratio of any online algorithm by constructing certain (adversary) sequences in a fairly straightforward way. These sequences, that we call BBK sequences in the sequel, were further studied by Johannes[4] and Hurink & Paulus[2], who derived improved lower bounds of 2.25 and 2.43, respectively. (Both results are computer aided and presented in terms of online parallel machine scheduling, a closely related problem.) The paper of Hurink & Paulus[2] also presents an upper bound of $\rho \leq 2.5$ for packing BBK sequences. Kern & Paulus[5] finally settled the question how well the BBK sequences can be packed by providing a matching upper and lower bound of $\rho_{\mathrm{BBK}} = 3/2 + \sqrt{33}/6 \approx 2.457$.

R. Solis-Oba and G. Persiano (Eds.): WAOA 2011, LNCS 7164, pp. 211–218, 2012.

Our Contribution. Using modified BBK sequences we show an improved lower bound of $2.589\ldots$ on the absolute competitive ratio of this problem. The modified sequences that we use consist solely of two types of items, namely, *thin* items that have negligible width (and thus can all be packed in parallel) and *blocking* items that have width 1. The advantage of these sequences is that the structure of the optimal packing is simple, i.e., the optimal packing height is the sum of the heights of the blocking items plus the maximal height of the thin items. Therefore, we call such sequences *primitive*.

On the positive side, we present an online algorithm for packing primitive sequences with competitive ratio $(3+\sqrt{5})/2 = 2.618\ldots$. This upper bound is especially interesting as it not only applies to the concrete adversary instances that we use to show our lower bound. Thus to show a new lower bound for strip packing that is greater than $2.618\ldots$ (and thus reduce the gap to the general upper bound of 6.6623), new techniques are required that take instances with more complex optimal solutions into consideration.

Organization. We start our presentation with a description of the Brown-Baker-Katseff sequences and their modification. Afterwards we present our lower bound based on these modifications, and finally we describe our algorithm for packing primitive sequences.

2 Sequence Construction

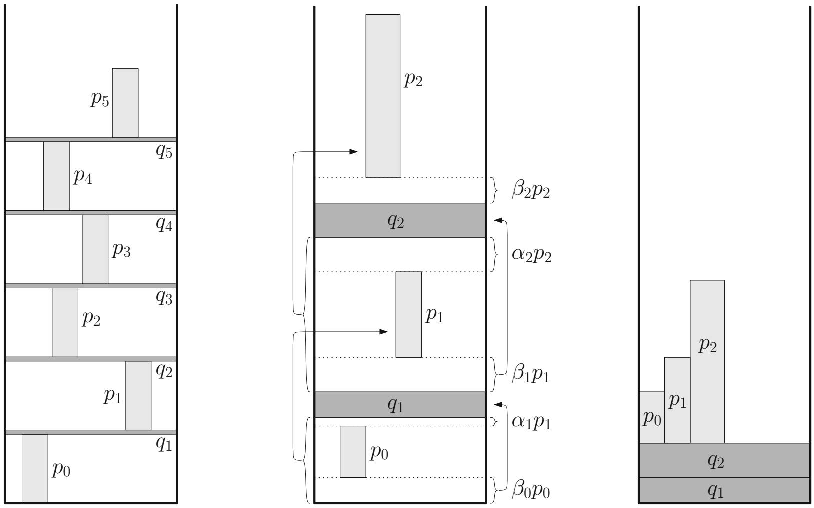
In this paper we denote the thin items by p_i and the blocking items by q_i (adopting the notation from [5]). As already mentioned in the introduction, we assume that the width of the thin items is negligible and thus all thin items can be packed next to each other. Moreover, the width of the blocking items q_i is always 1, so that no item can be packed next to any blocking item in parallel. Therefore, all items are characterized by their heights and we refer to their heights by p_i and q_i as well. By definition, for any list $L = q_1, q_2, \ldots, q_k, p_1, p_1, \ldots, p_\ell$ consisting of thin and blocking items we have

$$\mathrm{OPT}(L) = \sum_{i=1}^{k} q_i + \max_{i=1,\ldots,\ell} p_i.$$

To prove the desired lower bound we assume the existence of a ρ-competitive algorithm ALG for some $\rho < 2.589\ldots$ (the exact value of this bound is specified later) and construct an adversary sequence depending on the packing that ALG generates.

To motivate the construction, let us first consider the GREEDY algorithm for online strip packing, which packs every item as low as possible—see Figure 1a. This algorithm is not competitive (i.e., has unbounded competitive ratio): Indeed, consider the list $L_n = p_0, q_1, p_1, q_2, p_2, \ldots, q_n, p_n$ of items with

$$\begin{aligned} p_0 &:= 1, & \\ q_i &:= \varepsilon & \text{for } 1 \le i \le n, \\ p_i &:= p_{i-1} + \varepsilon & \text{for } 1 \le i \le n \end{aligned}$$



(a) A greedy packing (b) A packing of a BBK sequence (c) An optimal packing of a BBK sequence

Fig. 1. Online and optimal packings

for some $\varepsilon > 0$. GREEDY would pack each item on top of the preceding ones and thus generate a packing of height $\text{GREEDY}(L_n) = \sum_{i=0}^{n} p_i + \sum_{i=1}^{n} q_i = n + 1 + \Omega(n^2\varepsilon)$, whereas the optimum clearly has height $1 + 2n\varepsilon$.

The GREEDY algorithm illustrates that any competitive online algorithm needs to create gaps in the packing. These gaps work as a buffer to accommodate small blocking items—or, viewed another way, force the adversary to release larger blocking items.

BBK sequences. The idea of Brown, Baker & Katseff[1] was to try to cheat an arbitrary (non-greedy) online packing algorithm ALG in a similar way by constructing an alternating sequence $p_0, q_1, p_1, \ldots$ of thin and blocking items. The heights p_i respectively q_i are determined so as to force the online algorithm ALG to put each item above the previous ones—see Figure 1b for an illustration. To describe the heights of the items formally, we consider the gaps that ALG creates between the items. We distinguish two types of gaps, namely gaps below and gaps above a blocking item, and refer to theses gaps as α- and β-gaps, respectively. These gaps also play an important role in our analysis of the modified BBK sequences. We describe the height of the gaps around the blocking item q_i relative to the thin item p_i. Thus, we denote the height of the α-gap below q_i by $\alpha_i p_i$ and the height of the β-gap above q_i by $\beta_i p_i$. Using this notation, we are ready to formally describe the BBK sequences $L = p_0, q_1, p_1, q_2, \ldots$ with

$$\begin{aligned}
p_0 &:= 1, \\
q_1 &:= \beta_0 p_0 + \varepsilon, \\
p_i &:= \beta_{i-1} p_{i-1} + p_{i-1} + \alpha_i p_i + \varepsilon && \text{for } i \geq 1, \\
q_i &:= \max\left(\alpha_{i-1} p_{i-1}, \beta_{i-1} p_{i-1}, q_{i-1}\right) + \varepsilon && \text{for } i \geq 2.
\end{aligned}$$

As mentioned in the introduction, Brown, Baker & Katseff[1] used these sequences to derive a lower bound of 2 before Kern & Paulus[5] recently showed that the competitive ratio for packing them is $\rho_{\text{BBK}} = 3/2 + \sqrt{33}/6 \approx 2.457$.

The optimal online algorithm for BBK sequences that Kern & Paulus[5] describe generates packings with striking properties: No gaps are created except the first possible gap $\beta_0 = \rho_{\text{BBK}} - 1$ and the second α-gap $\alpha_2 = 1/(\rho_{\text{BBK}} - 1)$, which are chosen as large as possible while remaining ρ_{BKK}-competitive. Observing this behavior of the optimal algorithm led us to the modification of the BBK sequences.

Modified BBK sequences. When packing BBK sequences, a good online algorithm should be eager to enforce blocking items of relatively large size (as each blocking item of size q increases the optimal packing by q as well). These blocking items are enforced by generating corresponding gaps.

Modified BBK sequences are designed to counter this strategy: Each time the online algorithm places a blocking item q_i, the adversary, rather than immediately releasing a thin item p_{i+1} (of height defined as in standard BBK sequences) that does not fit in between the last two blocking items, generates a whole sequence of slowly growing thin items, which "continuously" grow from p_i to p_{i+1}. Packing this subsequence causes additional problems for the online algorithm: If the algorithm fits the whole subsequence into the last interval between q_{i-1} and q_i, it would fill out the whole interval and create an α-gap of 0. On the other extreme, if ALG would pack a thin item of height roughly p_i above q_i, then the (relative) β-gap it can generate is much less compared to what it could have achieved with a thin item of larger height p_{i+1}. The next blocking item q_{i+1} will be released as soon as the sequence of thin items has grown from p_i to p_{i+1}.

This general concept of modified BBK sequences applies after the first blocking item q_1 is released. Since subsequences of thin items and single blocking items are released alternately, we refer to this phase as the *alternating* phase. Before that, we have a *starting* phase which ends with the release of the first blocking item q_1. This starting phase needs special attention as we have no preceding interval height as a reference.

The optimal online algorithm by Kern & Paulus[5] generates an initial gap $\beta_0 = \rho_{\text{BBK}} - 1$ of maximal size to enforce a large first blocking item q_1. In the starting phase, we seek to prevent the algorithm from creating a large β_0-gap in the following way. Assume that the online algorithm places p_0 "too high" (i.e., β_0 is "too large"). Then the adversary, instead of releasing q_1, would continue generating higher and higher thin items and observe how the algorithm places them. As long as the algorithm places these thin items next to each other (overlapping in their packing height), the size of the gap below these items decreases monotonically relative to the height where items are packed. Eventually, β_0 has become sufficiently small—in which case the starting phase comes to an end with the release of q_1—or the online algorithm decides to "jump" in the sense that one of the items in this sequence of increasing height thin items is put strictly above all previously packed thin items, creating a new gap (distance between the last two items) and a significantly increased new packing height. Once a jump has occurred, the adversary continues generating thin items of slowly growing height until a next jump occurs or until the ratio of the largest current gap to the current packing height (the modified analogue to the standard β_0-gap) is sufficiently small and the starting phase comes to an end.

Summarizing, a *modified BBK sequence* simply consists of a sequence of thin items, continuously growing in height, interleaved with blocking items which (by definition of their height) must be packed above all preceding items, and are released as described above, *i.e.*, when the thin item size has grown up to the largest gap between two blocking items, *c.f.* the full paper for more details.

In the next section we use these modified BBK sequences to show the following theorem.

Theorem 1. *There exists no algorithm for online strip packing with competitive ratio*

$$\rho < \hat{\rho} = \frac{17}{12} + \frac{1}{48}\sqrt[3]{22\,976 - 768\sqrt{78}} + \frac{1}{12}\sqrt[3]{359 + 12\sqrt{78}} \approx 2.589\ldots.$$

3 Lower Bound

For the sake of contradiction, we assume that ALG is a ρ-competitive algorithm for online strip packing with $\rho < \hat{\rho}$. Let $\delta = \hat{\rho} - \rho > 0$. W.l.o.g. we assume that δ is sufficiently small.

We distinguish between the thin items p_i (whose height matches the height of the previous interval plus an arbitrarily small excess) and the subsequences of gradually growing thin items by denoting the whole sequence of thin items by $r_1, r_2, \ldots$ and designating certain thin items as p_i.

Our analysis (cf section 5) distinguishes two phases. In the first phase, the *starting* phase, we consider the following problem that the online algorithm faces. Given an input that consists only of thin items $r_1, r_2, \ldots$ (in this phase no blocking items are released), minimize the competitive ratio while retaining a free gap of maximal size (relative to the current packing height). More specifically, let

$$\frac{h(\mathrm{maxgap}_{\mathrm{ALG}}(r_i))}{\mathrm{ALG}(r_i)}$$

be the *max-gap-to-height* ratio after packing r_i where $h(\mathrm{maxgap}_{\mathrm{ALG}}(r_i))$ denotes the height of the maximal gap that algorithm ALG created up to item r_i and $\mathrm{ALG}(r_i)$ denotes the height algorithm ALG consumed up to item r_i. We say ALG is (ρ, c)-competitive in the starting phase if ALG is ρ-competitive (i.e., $\mathrm{ALG}(r_i) \leq \rho\mathrm{OPT}(r_i)$) and retains a max-gap-to-height ratio of c (i.e., $h(\mathrm{maxgap}_{\mathrm{ALG}}(r_i))/\mathrm{ALG}(r_i) \geq c$ for $i \geq 1$) for all lists $L = r_1, r_2, \ldots$ of thin items.

In the analysis of the starting phase we show that our modified BBK sequences force any ρ-competitive algorithm to reach a state with max-gap-to-height ratio less than

$$\hat{c} = \frac{\hat{\rho} - 2\sqrt{\hat{\rho} - 1}}{\hat{\rho} - 1}.$$

Thus no $(\rho, \hat{c})$-competitive algorithm exists for $\rho < \hat{\rho}$. In the moment ALG packs an item r_i and hereby reaches a max-gap-to-height ratio of less than $\hat{c}$, the starting phase ends with the release of the first blocking item q_1 of height $\hat{c} \cdot \mathrm{ALG}(r_i)$.

In the analysis of the *alternating phase* we show that no ρ-competitive algorithm can exist if the first blocking item after the starting phase has height $\hat{c}$ times the current packing height for

$$\hat{c} = \frac{1 - \sqrt{4\hat{\rho}^2 - 12\hat{\rho} + 5}}{2(\hat{\rho} - 1)}.$$

Thus our two phases fit together for

$$\hat{c} = \frac{\hat{\rho} - 2\sqrt{\hat{\rho} - 1}}{\hat{\rho} - 1} = \frac{1 - \sqrt{4\hat{\rho}^2 - 12\hat{\rho} + 5}}{2(\hat{\rho} - 1)},$$

which is satisfied for

$$\hat{\rho} = \frac{17}{12} + \frac{1}{48}\sqrt[3]{22\,976 - 768\sqrt{78}} + \frac{1}{12}\sqrt[3]{359 + 12\sqrt{78}} \approx 2.589\ldots.$$

The correseponding value of $\hat{c}$ is $\hat{c} \approx 0.04275\ldots$. We skip the proof of Theorem 1.

Algorithm 1. Online Algorithm for Restricted Instances

Initially the packing is considered to be blocked
whenever a rectangle r_j is released **do**
  if r_j is a blocking item **then**
    Pack r_j at the lowest possible height
  else if r_j is a thin item **then**
    if the packing is open **then**
      Pack r_j bottom-aligned with the top thin item
    else if the packing is blocked **then**
      Try to pack r_j below the top item
      If this is not possible, pack r_j at distance $(\rho - 2)r_j$ above the packing

4 Upper Bound

In this section we present the online algorithm ONL for packing instances that consist solely of thin and blocking items. We prove that the competitive ratio of ONL is $\rho = (3 + \sqrt{5})/2$. We distinguish two kinds of packings according to the item on top: If the item on top of the packing is a blocking item, we have a *blocked* packing, otherwise we have an *open* packing. Initially, we have a blocked packing by considering the bottom of the strip as a blocking item of height 0.

The general idea of the algorithm ONL is pretty straight-forward: Generate a β-gap of relative height $\rho - 2$ whenever a jump is unavoidable and pack arriving blocking items as low as possible. Since we neglect the starting phase, $\beta = \rho - 2$ is the maximal β-gap that we can ensure. This leads to the following algorithm—see also Algorithm 1.

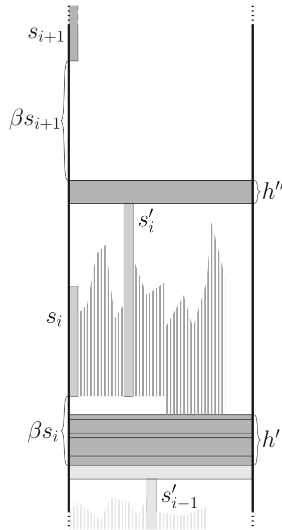


Fig. 2. Packing after the $(i+1)$-th jump. The blocking items that arrived after s_i are shown in darker shade. By definition, s_i is the first item that does not fit into the previous interval. Thus we have $s_{i+1} > s'_i + \beta\, s_i - h'$.

– If a blocking item r_j arrives, we pack r_j at the lowest possible height. This can be inside the packing, if a sufficiently large gap is available, or directly on top of the packing. In the latter case, the packing is blocked afterwards.
– If a thin item r_j arrives at an open packing, we bottom-align r_j with the top item.
– If, finally, a thin item r_j arrives at a blocked packing, we try to pack r_j below the blocking item on top. If this is not possible, i.e., r_j exceeds the height of all intervals for thin items, we pack r_j at distance $\beta\, r_j = (\rho - 2)r_j$ above the top of the packing. This changes the packing to an open packing again.

We show that ONL is ρ-competitive for $\rho = (3+\sqrt{5})/2$. Actually, this is only questionable in one case, namely, when we pack a thin item r_j with distance $(\rho - 2)r_j$ above the packing. All other cases are trivial since if the packing height increases, then the optimal height increases by the same value (for thin items the packing height only increases if r_j is the new maximal item).

We denote the thin items that are packed when generating a new gap by s_i for the i-th jump. Let s'_{i-1} be the highest thin item that is bottom-aligned with s_{i-1}. Note that the blocking item that blocks the packing after the i-th jump is packed directly above s'_{i-1}. See Figure 2 for an illustration.

It is obvious that the first jump item s_1, that is actually the first thin item that arrives, can be packed.

For the induction step we assume $\mathrm{ONL}(s_i) \le \rho\, \mathrm{OPT}(s_i)$. Before a jump can become unavoidable, new blocking items of total height greater than $\beta\, s_i$ need to arrive as otherwise the gap below s_i could accommodate all of them. Let h' be the height of the blocking items that are packed into the β-gap below s_i and let h'' be the total height of blocking items that arrive between s_i and s_{i+1} and are packed above s_i. We have $h' \leq (\rho-2)s_i$ and $h' + h'' > (\rho-2)s_i$ as otherwise no blocking item would be packed on top. As further blocking items could be packed even below s'_{i-1} we get

$$\begin{aligned}\mathrm{OPT}(s_{i+1}) &\geq \mathrm{OPT}(s_i) + h' + h'' + s_{i+1} - s_i\\ \mathrm{ONL}(s_{i+1}) &= \mathrm{ONL}(s_i) + s'_i - s_i + h'' + \beta s_{i+1} + s_{i+1}.\end{aligned}$$

And thus we have

$$\begin{aligned} & \mathrm{ONL}(s_{i+1}) \leq \rho\, \mathrm{OPT}(s_{i+1})\\ \Leftarrow\quad & \mathrm{ONL}(s_i) + s'_i - s_i + h'' + \beta s_{i+1} + s_{i+1} \leq \rho\big(\mathrm{OPT}(s_i) + h' + h'' + s_{i+1} - s_i\big)\\ \Leftarrow\quad & (\rho-1)s_i + s'_i - \rho h' - (\rho-1)h'' \leq (\rho-1-\beta)s_{i+1}.\end{aligned}$$

As $\rho - 1 - \beta = 1$ and $s_{i+1} > s'_i + (\rho-2)s_i - h'$ this is satisfied if

$$\begin{aligned} & (\rho-1)s_i + s'_i - \rho h' - (\rho-1)h'' \leq s'_i + (\rho-2)s_i - h'\\ \Leftrightarrow\quad & s_i \leq (\rho-1)(h' + h'')\\ \Leftarrow\quad & s_i \leq (\rho-1)(\rho-2)s_i = s_i.\end{aligned}$$

The last equality holds since $\rho = (3+\sqrt{5})/2$ and thus $(\rho-1)(\rho-2) = 1$. Summarizing, we arrive at

Theorem 2. ONL *is a ρ-competitive algorithm for packing primitive sequences with*

$$\rho = \frac{3+\sqrt{5}}{2} \approx 2.618.$$

So the true best possible competitive ratio for packing primitive sequences is somewhere in between the two values specified by Theorems 1 and 2. We have reasons to believe that it is strictly in between these two. But perhaps an even more challenging question is whether or not (or to what extent) primitive sequences provide worst case instances for online packing in general.

References

1. Brown, D.J., Baker, B.S., Katseff, H.P.: Lower bounds for online two-dimensional packing algorithms. Acta Informatica 18, 207–225 (1982)
2. Hurink, J., Paulus, J.: Online scheduling of parallel jobs on two machines is 2-competitive. Operations Research Letters 36(1), 51–56 (2008)
3. Hurink, J.L., Paulus, J.J.: Online Algorithm for Parallel Job Scheduling and Strip Packing. In: Kaklamanis, C., Skutella, M. (eds.) WAOA 2007. LNCS, vol. 4927, pp. 67–74. Springer, Heidelberg (2008)
4. Johannes, B.: Scheduling parallel jobs to minimize the makespan. Journal of Scheduling 9(5), 433–452 (2006)
5. Kern, W., Paulus, J.: A tight analysis of Brown-Baker-Katseff sequences for online strip packing. Submitted (J. Combinatorial Opt.)
6. Ye, D., Han, X., Zhang, G.: A note on online strip packing. Journal of Combinatorial Opt. 17(4), 417–423 (2009)