Batched Bin Packing

Gregory Gutin, Tommy Jensen and Anders Yeo*
Department of Computer Science
Royal Holloway, University of London
Egham, Surrey, TW20 0EX, UK
gutin(tommy,anders)@cs.rhul.ac.uk

Abstract

We introduce and study the batched bin packing problem (BBPP), a bin packing problem in which items become available for packing incrementally, one batch at a time. A batched algorithm must pack a batch before the next batch becomes known. A batch may contain several items; the special case when each batch consists of merely one item is the well-studied on-line bin packing problem. We obtain lower bounds for the asymptotic competitive ratio of any algorithm for the BBPP with two batches. We believe that our main lower bound is optimal and provide some support to this conjecture. We suggest studying BBPP and other batched problems.

Keywords: On-line algorithm, lower bounds, bin packing, competitive ratio.

1 Introduction, Terminology and Notation

In this paper, we study a variation of the classical bin packing problem (BPP). In BPP, we are given a set B of items $a_1, a_2, \dots, a_n$ and a sequence of their sizes $(s_1, s_2, \dots, s_n)$ (each size $s_i \in (0, 1]$) and are required to pack the items into a minimum number of unit-capacity bins. In other words, we need to partition B into a minimum number m of subsets $B_1, B_2, \dots, B_m$ such that $\sum_{a_i \in B_j} s_i \leq 1$ for each $j = 1, 2, \dots, m$. For recent surveys of BPP, see [3, 4].

We introduce the *batched bin packing problem (BBPP)*, a bin packing problem in which items become available for packing incrementally, one batch at a time. A *batched algorithm* must pack a batch before the next batch becomes known. A batch may contain several items; the special case when each batch consists of merely one item is the well-studied on-line bin packing problem. In the case of just one batch, we have the classical off-line BPP. In BBPP, an input sequence L is a *batched sequence*, namely, $L = (B_1, B_2, \dots, B_k)$, where every B_j is a set of items and $B_i \cap B_j = \emptyset$ whenever $1 \leq i < j \leq k$.

BBPP may be of interest when, for example, items are delivered to a packing site by trucks, each truck containing several items. To the best of our knowledge, despite being a very natural generalization of BPP, BBPP has not been studied before; repacking and lookahead BPPs (see Section 2.2.6 in [3]) are different problems.

We believe that batched generalizations of other on-line problems have not been investigated yet, and that such generalizations are of definite interest. Being extensions of the corresponding on-line problems, batched problems may prove to be very difficult to investigate in their general setting. One way around this is to fix the number of batches in possible inputs; an assumption that may be not too restrictive for some batched problems.

*This research of all authors was partially supported by a Leverhulme Trust grant. Correspondence to: GG.

All batched sequences with exactly k batches (some of which may be empty) comprise a set, which we denote by $\mathcal{B}(k)$. For a given batched sequence L and batched algorithm A, let $A(L)$ be the number of bins required for L by algorithm A; let $\mathrm{OPT}(L)$ be the minimum number of bins needed to pack the items of L when they are all available at once (as in BBP). The *asymptotic competitive ratio* $R^{\infty}_{A,k}$ of A on $\mathcal{B}(k)$ is

$$\mathrm{limsup}_{N\to\infty} \max\{\frac{A(L)}{\mathrm{OPT}(L)} : \ L \in \mathcal{B}(k), \ \mathrm{OPT}(L) = N\}.$$

The asymptotic competitive ratio of a batched algorithm is defined similarly to the asymptotic competitive ratio of an on-line algorithm.

In this paper, we study lower bounds of $R^{\infty}_{A,2}$ for any batched algorithm A with inputs from $\mathcal{B}(2)$. We note that any additional assumptions, such as polynomiality, are not made about the algorithms that we study. In Section 2, we prove such a bound r in Theorem 1. We conjecture that the bound r is optimal. To formally support this conjecture we prove in Section 4 that the bound r is optimal for a wide family of batched sequences. In Section 3 we obtain lower bounds of $R^{\infty}_{A,2}$ for the restriction of $\mathcal{B}(2)$ to instances in which the number of item sizes is bounded (a natural constraint). Section 5 is devoted to open problems and suggestions for further research.

Yao [9] was the first to study lower bounds for the asymptotic competitive ratio of an on-line algorithm for BBP. He showed that such a bound is not smaller than 1.5. Brown [1] and Liang [6] independently improved Yao's result to 1.53635. This was further improved by van Vliet [8] to 1.54014. Chandra [2] showed that the preceding lower bounds also apply to randomized algorithms. Seiden, van Stee and Epstein [7] studied lower bounds for an extension of the on-line BPP.

2 Lower Bounds

Let 2-BBPP denote the restriction of BBPP to inputs with two batches, i.e., sequences from $\mathcal{B}(2)$.

Define σ as the solution in the interval $(3/2, 2)$ to $2\sigma - 3 = \ln\sigma$, with $\sigma = 1.7915\ldots$, and let $r = 2\sigma/(2\sigma - 1) = 1.3871\ldots$. In other words r is a solution to $\frac{r}{r-1} - 3 = \ln\frac{r}{2r-2}$.

Theorem 1 *If A is a batched algorithm for 2-BBPP, then $R^{\infty}_{A,2} \geq r$.*

Proof: Consider instances $L = (B_1, B_2)$ of 2-BBPP, the first batch B_1 consisting of n items all of size equal to s, where $0 < s < 1$. Let $t = \lfloor\frac{1}{s}\rfloor$; then t is the maximal number of items from B_1 which can be packed into one bin.

The second batch B_2 will be either equal to the empty batch B^0, or to B^j consisting of n/j items each of size $1 - js$, with $j = 1, 2, \ldots, m := \lceil\frac{1}{2s}\rceil - 1$. (We assume throughout, without loss of generality, that n is divisible by every integer $1, 2, \ldots, m$ and also by t.) Hence no two items from B^j fit into one bin together, and an item from B^j leaves room for j items from B_1, and no more ($1 \leq j \leq m$). Since this bin packing problem is easy to analyze for the range $s \geq 1/2$, we further assume $s < 1/2$, and hence $m \geq 1$.

Assume that an algorithm A for 2-BBPP packs the items of B_1 so that the number of bins containing exactly i items is $y_i = nx_i$, for each $i = 1, 2, \ldots, t$. Hence $y_1 + 2y_2 + \cdots + ty_t = n$ and

$$\sum_{i=1}^{t} ix_i = 1 \tag{1}$$

Assume that the number of bins used by A when packing any two consecutive batches never exceeds a factor of z times the optimum number of bins that may be used in packing (off-line) the items contained in the union of the same two batches. Applying this assumption to $B_2 = B^0$, it follows that

$$\sum_{i=1}^{t} x_i \leq \frac{1}{t} z, \tag{2}$$

since the items from B_1 can be packed into n/t bins.

For $B_2 = B^m$ there exists a packing of $B_1 \cup B_2$ into n/m bins, whereas A uses at least $n \sum_{i=m+1}^{t} x_i$ bins each of which contains no items from B_2 (as this many bins are already packed too full), together with an additional n/m bins each of which contains precisely one item from B_2. We then have

$$\sum_{i=m+1}^{t} x_i + \frac{1}{m} \leq \frac{1}{m} z. \tag{3}$$

Similarly, for $B_2 = B^j$, $1 \leq j \leq m-1$,

$$\sum_{i=j+1}^{t} x_i + \frac{1}{j} \leq \frac{1}{j} z. \tag{4}$$

Let $\ell \in \{1, 2, \ldots, m\}$. We consider the sum of (4) over the values $j = \ell, \ell+1, \ldots, m-1$:

$$\sum_{j=\ell}^{m-1} \sum_{i=j+1}^{t} x_i + \lambda_m(\ell) \leq \lambda_m(\ell) z, \tag{5}$$

where

$$\lambda_m(\ell) := \sum_{j=\ell}^{m-1} \frac{1}{j}.$$

Noting, by reordering of sums, that

$$\sum_{j=\ell}^{m-1} \sum_{i=j+1}^{t} x_i = \sum_{i=\ell+1}^{m} (i-\ell) x_i + \sum_{i=m+1}^{t} (m-\ell) x_i,$$

we get by adding ℓ times (2) to $t-m$ times (3) to (5),

$$\ell \sum_{i=1}^{t} x_i + (t-m) \sum_{i=m+1}^{t} x_i + \frac{t-m}{m} + \sum_{i=\ell+1}^{m} (i-\ell) x_i +$$

$$+ \sum_{i=m+1}^{t} (m-\ell) x_i + \lambda_m(\ell) \leq \left(\frac{\ell}{t} + \frac{t-m}{m} + \lambda_m(\ell)\right) z. \tag{6}$$

Collecting first in (6) the terms involving x_i, $1 \leq i \leq t$, and finally applying (1) yields

$$
\begin{aligned}
\ell\sum_{i=1}^{t} x_i + (t-m)\sum_{i=m+1}^{t} x_i + \sum_{i=\ell+1}^{m}(i-\ell)x_i + \sum_{i=m+1}^{t}(m-\ell)x_i &= \\
\sum_{i=1}^{\ell}\ell x_i + \sum_{i=\ell+1}^{m} i x_i + \sum_{i=m+1}^{t} t x_i &\geq \\
\sum_{i=1}^{t} i x_i &= 1.
\end{aligned}
$$

We conclude by (6) that

$$\frac{t}{m} + \lambda_m(\ell) \leq (\frac{\ell}{t} + \frac{t}{m} - 1 + \lambda_m(\ell))z,$$

and therefore

$$z \geq \frac{\frac{t}{m} + \lambda_m(\ell)}{\frac{t}{m} + \lambda_m(\ell) - 1 + \frac{\ell}{t}} = 1 + \left(\frac{\frac{t}{m} + \lambda_m(\ell)}{1 - \frac{\ell}{t}} - 1\right)^{-1}. \tag{7}$$

We will now assume that ℓ and m satisfy $m \geq \ell > 1$ and $(m-1)/(\ell-1) \leq \sigma \leq m/(\ell-1)$. (For any $\ell > 1$, it is clearly possible to achieve this by choosing an appropriate value of s.) It follows that

$$\lambda_m(\ell) = \sum_{j=\ell}^{m-1} \frac{1}{j} \leq \int_{\ell-1}^{m-1} \frac{1}{y} dy = \ln \frac{m-1}{\ell-1} \leq \ln \sigma = 2\sigma - 3.$$

Moreover, from $\sigma \leq m/(\ell-1)$, we have

$$\frac{\ell}{t} \leq \frac{1}{t}(\frac{m}{\sigma} + 1) = \frac{m}{\sigma t} + \frac{1}{t}.$$

It follows from (7) that

$$z \;\; \geq \;\; 1 + \left(\frac{\frac{t}{m} + 2\sigma - 3}{1 - \frac{m}{\sigma t} - \frac{1}{t}} - 1\right)^{-1},$$

which, using that t/m converges to 2, approaches the value $2\sigma/(2\sigma-1)$ as t grows to infinity. Hence, the lower bound

$$z \geq 2\sigma/(2\sigma - 1) = r$$

is proved. □

We believe that the bound r in the above theorem is optimal:

Conjecture 1 *There exists an algorithm A for 2-BBPP with $R^{\infty}_{A,2} = r$.*

We remark that (7) does not contradict this conjecture. The following lemma, which will be used in the proof of Theorem 2 shows that the value of the right hand side of (7) is indeed bounded from above by r.

Lemma 1 *Let* $0 < s < 1/2$, $t = \lfloor 1/s \rfloor$, $m = \lceil \frac{1}{2s} \rceil - 1$, $1 \leq \ell \leq m$, *and* $\lambda_m(\ell) = \sum_{j=\ell}^{m-1} 1/j$. *Then*

$$1 + \left(\frac{\frac{t}{m} + \lambda_m(\ell)}{1 - \frac{\ell}{t}} - 1 \right)^{-1} \leq r.$$

Proof: The estimate

$$\lambda_m(\ell) \geq \int_\ell^m \frac{1}{y} dy = \ln \frac{m}{\ell}$$

implies

$$\begin{aligned} 1 + \left(\frac{\frac{t}{m} + \lambda_m(\ell)}{1 - \frac{\ell}{t}} - 1 \right)^{-1} &\leq 1 + \left(\frac{\frac{t}{m} - \ln \frac{t}{m} - \ln \frac{\ell}{t}}{1 - \frac{\ell}{t}} - 1 \right)^{-1} \\ &\leq 1 + \left(\frac{2 - \ln 2 - \ln \frac{\ell}{t}}{1 - \frac{\ell}{t}} - 1 \right)^{-1}, \end{aligned}$$

where the last inequality follows from $t/m \geq 2$ and the fact that $x \mapsto x - \ln x$ defines an increasing function on $(1, \infty)$. Finally it is straightforward to verify that the function defined for $0 < x < 1$ by

$$x \mapsto \frac{2 - \ln 2 - \ln x}{1 - x}$$

assumes a unique minimum value of 2σ at the point $x_0 = 1/(2\sigma)$. This proves the lemma, since $r = 1 + 1/(2\sigma - 1)$. $\square$

Further formal and informal support to Conjecture 1 is provided in Section 4.

3 Lower Bounds for a Variation of 2-BBPP

The bound (7) may be thought of as derived from instances L of 2-BBPP in which the $m - \ell + 2$ distinct values $s, 1 - \ell s, 1 - (\ell + 1)s, \ldots, 1 - ms$ are the only item sizes which can occur (as the batches $B^1, B^2, \ldots, B^{\ell-1}$ are not considered when deriving (5)).

Consider now only instances of 2-BBPP in which the number of different item sizes is at most $p(\geq 2)$. Suppose a batched algorithm is given the possible item sizes at the same time as it gets the first batch of an instance $L \in \mathcal{B}(2)$. Then the lower bound (7) applies to the competitive ratio of any such empowered algorithm, with $\ell = m - p + 2$ and for any suitable choice of s. For each p, we have chosen $m = \lceil (\sigma(p-1) - 1)/(\sigma - 1) \rceil$, $t = 2m$, and $s = 2/(2t + 1)$.

This provides the values of $r(p)$, a lower bound for the asymptotic competitive ratio of such an empowered algorithm, given in the following table. It is in general not clear whether these might be the best possible bounds $r(p)$. It seems that $r(2)$ is best possible (we have a paper in preparation showing the existence of an algorithm for on-line bin packing which reaches an asymptotic competitive ratio of 4/3 when only two item sizes are allowed and known in advance).

p	s	t	m	ℓ	$\lambda_m(\ell)$	$r(p)$
2	2/5	2	1	1	0	1.3333...
3	2/17	8	4	3	1/3	1.3658...
4	2/25	12	6	4	9/20	1.3738...
5	2/33	16	8	5	107/210	1.3773...
6	2/45	22	11	7	1207/2520	1.3793...

4 Possible Optimality of r

For every fixed $s < 1/2$ we have exhibited a lower bound (7) for the asymptotic competitive ratio of any algorithm for the restriction of 2-BBPP to the special subclass of instances $L = (B_1, B_2)$ for which all items of the initial batch B_1 have the same size s. The bound is given as a function of $t = \lfloor 1/s \rfloor$ and $m = \lceil 1/(2s) \rceil - 1$, and choosing the value of ℓ suitably, $1 \leq \ell \leq m$.

Let $\mathcal{B}'$ denote the set of instances $L = (B_1, B_2) \in \mathcal{B}(2)$ for which all items of the initial batch B_1 are of the same size $s = 1/t$ for some integer $t > 2$. We now confirm Conjecture 1 for this subclass $\mathcal{B}'$ of $\mathcal{B}(2)$.

Theorem 2 *There exists a batched algorithm A with the property*

$$\text{limsup}_{N \rightarrow \infty} \max\{\frac{A(L)}{\text{OPT}(L)} : \; L \in \mathcal{B}', \; \text{OPT}(L) = N\} = r.$$

Proof: We will describe an algorithm A, such that for every $\varepsilon > 0$ the ratio $A(L)/OPT(L)$ exceeds the right hand side of (7) by at most ε for all instances $L \in \mathcal{B}'$ for which $OPT(L)$ is sufficiently large. Using Lemma 1 this implies

$$\text{limsup}_{N \rightarrow \infty} \max\{\frac{A(L)}{\text{OPT}(L)} : \; L \in \mathcal{B}', \; \text{OPT}(L) = N\} \leq r.$$

By the proof of Theorem 1, r is also a lower bound, so the Theorem follows.

Let $L = (B_1, B_2)$ be an instance in $\mathcal{B}'$ having n items in B_1 all of size $s = 1/t$, where $t > 2$ is an integer, and let $m = \lceil 1/(2s) \rceil - 1 = \lceil t/2 \rceil - 1$. Choose as ℓ the smallest positive number satisfying $\lambda_m(\ell) \geq \frac{t}{\ell} - \frac{t}{m} - 1$ (this inequality holds for $\ell = m$, so the choice is indeed possible), and let

$$z = \frac{\frac{t}{m} + \lambda_m(\ell)}{\frac{t}{m} + \lambda_m(\ell) - 1 + \frac{\ell}{t}}.$$

We observe that $z \geq 1$ holds. Now define values $w_1, w_2, \ldots, w_t, w_{t+1}$ by

$$w_i = \begin{cases} 0 & \text{if} \quad i = t+1, \\ \frac{1}{m}(z-1) & \text{if} \quad m+1 \leq i \leq t, \\ \frac{1}{i-1}(z-1) & \text{if} \quad \ell+1 \leq i \leq m, \\ \frac{1}{t} z & \text{if} \quad 1 \leq i \leq \ell. \end{cases}$$

The inequality $w_\ell \geq w_{\ell+1}$ follows from the choice of ℓ, which implies

$$\frac{z-1}{z} = \frac{1 - \frac{\ell}{t}}{\frac{t}{m} + \lambda_m(\ell)} \leq \frac{1 - \frac{\ell}{t}}{\frac{t}{\ell} - 1} = \frac{\ell}{t},$$

and hence $w_\ell = \frac{1}{t}z \geq \frac{1}{\ell}(z-1) = w_{\ell+1}$. This and the fact $z \geq 1$ imply $w_1 \geq w_2 \geq \cdots \geq w_{t+1}$.

We will consider differences of the form $\lfloor nw_i \rfloor - \lfloor nw_{i+1} \rfloor$ for $i = 1, 2, \ldots, t$. These are non-negative integers having the property

$$\sum_{i=1}^{t} i(\lfloor nw_i \rfloor - \lfloor nw_{i+1} \rfloor) = \sum_{i=1}^{t} \lfloor nw_i \rfloor.$$

Using the equality

$$\begin{aligned} \sum_{i=1}^{t} nw_i &= n((t-m)\frac{1}{m}(z-1) + \sum_{i=\ell+1}^{m} \frac{1}{i-1}(z-1) + \frac{\ell}{t}z) \\ &= n((\frac{t}{m} - 1 + \lambda_m(\ell) + \frac{\ell}{t})(z-1) + \frac{\ell}{t}) \\ &= n, \end{aligned}$$

we deduce that

$$0 \leq n - \sum_{i=1}^{t} i(\lfloor nw_i \rfloor - \lfloor nw_{i+1} \rfloor) \leq t.$$

We will let A pack B_1 by first distributing a subset of the items in such a way that the number of bins containing i items is precisely $\lfloor nw_i \rfloor - \lfloor nw_{i+1} \rfloor$, followed by packing any remaining items into one additional bin (if needed), which is possible by the preceding inequality. When receiving B_2, the existing packing of the items from B_1 is completed in an optimal way by A to a final packing of $B_1 \cup B_2$. Again we remark that the running time efficiency of A is not an issue.

Let y_i denote the number of bins which contain i items when A has finished the packing of B_1, for $i = 1, 2, \ldots, t$. Then with $i' = n - \sum_{i=1}^{t} i(\lfloor nw_i \rfloor - \lfloor nw_{i+1} \rfloor)$ we have

$$y_i = \begin{cases} \lfloor nw_i \rfloor - \lfloor nw_{i+1} \rfloor + 1 & \text{if } i = i', \\ \lfloor nw_i \rfloor - \lfloor nw_{i+1} \rfloor & \text{otherwise.} \end{cases}$$

Let $\varepsilon > 0$. To prove the theorem it is sufficient to show, that there exists a number $N(\varepsilon)$ such that $OPT(L) \geq N(\varepsilon)$ implies

$$\frac{A(L)}{OPT(L)} \leq z + \varepsilon.$$

Indeed we will show that this holds with $N(\varepsilon) = 1/\varepsilon$.

Let B_2 consist of k items of sizes $(s_1, s_2, \ldots, s_k)$. We begin by making a few simplifying assumptions.

(A1) *In any optimal solution to L each bin contains at most one item from B_2.*

Otherwise, if the items of sizes s_1, s_2, say, are placed in the same bin in some optimal solution, then we consider $L' = (B_1, B_2')$ where B_2' contains $k-1$ items of sizes $(s_1+s_2, s_3, \ldots, s_k)$. Then $OPT(L') = OPT(L)$ and $A(L') \geq A(L)$ are satisfied. We may now replace L by L', since $A(L)/OPT(L) \leq z + \varepsilon$ would follow from $A(L')/OPT(L') \leq z + \varepsilon$.

(A2) *s_j/s is an integer for every $j = 1, 2, \ldots, k$.*

Indeed, if we replace B_2 by B_2' having item sizes $(\lceil s_1/s \rceil s, \lceil s_2/s \rceil s, \ldots, \lceil s_k/s \rceil s)$ and let $L' = (B_1, B_2')$, then $OPT(L') = OPT(L)$ follows from (A1) and the integrality of $1/s$. As before, $A(L') \geq A(L)$ holds.

(A3) *Any pair of items in B_2 have combined size strictly greater than 1.*

Otherwise, say if $s_1 + s_2 \leq 1$, we consider the two bins of an optimal packing of L that contain the items of sizes s_1 and s_2. Using (A2) and the integrality of $1/s$, we may rearrange the items between these two bins to obtain a new packing using equally many bins, but having the items of sizes s_1, s_2 in the same bin. This packing would contradict (A1).

From these assumptions it follows that any bin can contain at most one item from B_2. Thus the solution $A(L)$ is given by a largest matching between the items from B_2 and the partially packed bins containing the items from B_1. Here an item may be matched to a bin, only if its size is no larger than the space which remains in the bin. We will apply the theorem of König on maximum matchings in bipartite graphs (e.g. see [5], Theorem 2.1.1): The maximal size of a matching in a bipartite graph equals the minimal number of vertices which cover all edges (where a vertex is said to 'cover' its incident edges). For simplicity assume $s_1 \leq s_2 \leq \cdots \leq s_k$, and let $b_1, b_2, \ldots, b_{n'}$ with $0 < b_1 \leq b_2 \leq \cdots \leq b_{n'}$ denote the contents of the bins which have been partially packed by A with items from B_1. Applying König's theorem, the largest size of a matching between items and bins is equal to

$$M = \min\{i_0 + j_0 \mid b_i + s_j \leq 1 \Rightarrow i \leq i_0 \vee j \leq j_0\},$$

where the minimum is taken over all i_0, j_0 with $0 \leq i_0 \leq n'$ and $0 \leq j_0 \leq k$. Suppose that the minimum is achieved as $M = i_0 + j_0$, so that $b_i + s_j \leq 1 \Rightarrow i \leq i_0 \vee j \leq j_0$. Then $A(L) = n' + k - i_0 - j_0$ follows. Let $k' = k - j_0$. If $k' = 0$, then we let B_2' be empty. Otherwise we let B_2' be a batch consisting of k' items each of size s_{j_0+1}. Applying König's theorem now for $L' = (B_1, B_2')$, and noting that $b_i + s_{j_0+1} \leq 1 \Rightarrow i \leq i_0$, it follows that $A(L') \geq n' + k' - i_0 = A(L)$. Moreover, $OPT(L') \leq OPT(L)$ is trivial. If strict inequality $OPT(L') < OPT(L)$ holds, then we consider $L" = (B_1, B_2")$ instead, with $B_2"$ obtained from B_2' by adding more items of size s_{j_0+1} until $OPT(L") = OPT(L)$ is satisfied. So we may assume that all items of B_2 have the same size.

For each $j = 0, 1, \ldots, m$ let $B(j,k)$ be a batch of k items all of size $1 - js$, and let $L(j,k)$ denote the instance $(B_1, B_2) = (B_1, B(j,k))$ of $\mathcal{B}'$. Then by the above argument, using (A2), (A3), and the integrality of $1/s$, we may assume the following.

(A4) *$L = L(j,k)$ for some $j = 0, 1, \ldots, m$.*

This simplification allows us to directly calculate the values of $A(L)$ and $OPT(L)$. The value for $OPT(L)$ is easily deduced.

$$OPT(L) = \begin{cases} k & \text{if } n < jk, \\ k + \lceil \frac{n-jk}{t} \rceil & \text{if } n \geq jk. \end{cases}$$

So in any case the bound $OPT(L) \geq k + (n - jk)/t$ follows.

For $A(L)$ the precise value is depending on whether all items from B_2 can be accomodated by the already partially packed bins. For each that cannot, A must open an additional bin. Thus we have

$$A(L) = \begin{cases} \sum_{i=1}^{t} y_i & \text{if } k < \sum_{i=1}^{j} y_i, \\ \sum_{i=j+1}^{t} y_i + k & \text{if } k \geq \sum_{i=1}^{j} y_i. \end{cases}$$

Case 1. $k < \sum_{i=1}^{j} y_i$.

Then $A(L) \leq \lfloor nz/t \rfloor + 1 \leq nz/t + 1$, hence,

$$\frac{A(L)}{OPT(L)} \leq \frac{nz/t}{k + (n-jk)/t} + \frac{1}{OPT(L)}$$

$$\begin{aligned} &= \frac{z}{1+k(t-j)/n} + \frac{1}{OPT(L)} \\ &\le z + \frac{1}{OPT(L)}, \end{aligned}$$

which concludes this case.

Case 2. $k \ge \sum_{i=1}^{j} y_i$.
It follows in particular that

$$A(L) \le \lfloor n w_{j+1} \rfloor + k + 1 = \begin{cases} \lfloor n(z-1)/j \rfloor + k + 1 & \text{if } \ell \le j \le m, \\ \lfloor nz/t \rfloor + k + 1 & \text{if } 0 \le j < \ell. \end{cases}$$

We distinguish two subcases.
Case 2.1. $j < \ell$.
The choice of ℓ yields

$$\lambda_m(\ell) = \lambda_m(\ell-1) - \frac{1}{\ell-1} < \frac{t-1}{\ell-1} - \frac{t}{m} - 1,$$

and therefore

$$\frac{z-1}{z} = \frac{1-\frac{\ell}{t}}{\frac{t}{m}+\lambda_m(\ell)} > \frac{1-\frac{\ell}{t}}{\frac{t-1}{\ell-1}-1} = \frac{\ell-1}{t} \ge \frac{j}{t}.$$

Now we obtain

$$\begin{aligned} \frac{A(L)}{OPT(L)} &\le \frac{nz/t+k}{k+n/t-jk/t} + \frac{1}{OPT(L)} \\ &\le \frac{nz/t+k}{k+n/t-k(z-1)/z} + \frac{1}{OPT(L)} \\ &= z + \frac{1}{OPT(L)}, \end{aligned}$$

achieving the desired bound.

Case 2.2. $j \ge \ell$.
Then we first observe that

$$OPT(L) \ge k + \frac{n-jk}{t} = k + (n-jk)\frac{w_\ell}{z} \ge k + (n-jk)\frac{w_{j+1}}{z} = k + (\frac{n}{j} - k)\frac{z-1}{z}.$$

Hence, with $A(L) \le \lfloor n(z-1)/j \rfloor + k + 1$,

$$\begin{aligned} \frac{A(L)}{OPT(L)} &\le \frac{n(z-1)/j+k}{k+(n/j-k)(z-1)/z} + \frac{1}{OPT(L)} \\ &= z + \frac{1}{OPT(L)}. \end{aligned}$$

This finishes the final case of the proof. □

We remark that the assumption of integrality of $1/s$, which has been used throughout this section, can be relaxed at the cost of increasing the upper bound on asymptotic competitive ratio by a factor of at most $1 + 1/t$, where $t = \lfloor 1/s \rfloor$. This can be seen by considering an algorithm behaving like A, except that it treats the items from the batch B_1 as if they were all of size $1/t$ instead of s.

Considering again briefly the general version of 2-BBPP, in which the initial batch may contain items of different sizes, it seems that the competitive ratio increases as the item sizes within the initial batch decrease (cf. the table in Section 3). However, if more than one item size occurs in the initial batch, and all item sizes are small, then it is intuitively reasonable to attempt to achieve a good competitive ratio by approximating all sizes by a single size, say, by their average. Thus altogether we feel that we have reasonable support for Conjecture 1.

5 Further Research

The introduction of BBPP raises several natural problems. It would be very interesting to obtain algorithms for 2-BBPP whose asymptotic competitive ratios are lower than those of on-line algorithms. This problem can be extended to k-BBPP, BBPP restricted to sequences with exactly k batches, for fixed $k \geq 2$.

It would also be interesting to obtain lower bounds for the asymptotic competitive ratio of algorithms for k-BBPP for fixed $k \geq 3$ and prove (or disprove) the optimality of our lower bound r.

We believe that batched generalizations of other on-line problems are of definite interest and deserve investigation.

References

[1] D.J. Brown, A lower bound for on-line one-dimensional bin packing algorithms. Tech. report R-864, Coordinated Science Laboratory, Urbana, IL, 1979.

[2] B. Chandra, Does randomization help in on-line bin packing? Inform. Process. Lett. 43 (1992) 15-19.

[3] E. G. Coffman, Jr., M. R. Garey, and D. S. Johnson, Approximation Algorithms for Bin Packing: A Survey. Approximation Algorithms for NP-Hard Problems, D. Hochbaum (editor), PWS Publishing, Boston (1996), 46-93.

[4] J. Csirik and G. Woeginger, On-line packing and covering problems. On-line Algorithms - the State of the Art, Lect. Notes In Comput. Sci. 1442, A. Fiat and G. Woeginger (editors), Springer, New York (1998), 147-177.

[5] R. Diestel, Graph Theory, 2nd edition, Springer, Berlin, 2000.

[6] F.M. Liang, A lower bound for online bin packing. Inform. Process. Lett. 10 (1980) 76-79.

[7] S. Seiden, R. van Stee and L. Epstein, New bounds for variable-sized online bin packing. SIAM J. Comput. 32 (2003) 455-469.

[8] A. van Vliet, An improved lower bound for online bin packing algorithms. Inform. Process. Lett. 43 (1992) 277-284.

[9] A.C.C. Yao, New algorithms for bin packing. J. ACM 27 (1980) 207-227.