

The variable-width strip packing problem

Attila Bódis · János Csirik

Received: date / Accepted: date

Abstract In this paper, we focus on a generalized version of the strip packing problem: we have several open-end strips with different widths, and we wish to pack rectangular items into these strips without overlapping such that we have to minimize either the makespan (i.e. the top of the topmost item), or the total area used. We investigate the online variant of the problem, where the items are arriving one-by-one, and we have to make irrevocable decisions on their packing.

A similar framework was proposed by Ye and Mei in [12] for scheduling models, and they studied the absolute competitive ratio of their algorithm.

Our contribution is to define a new objective function and several algorithms by combining so-called shelf algorithms with techniques taken from the areas of the variable-sized bin packing problem and scheduling. We analyzed the asymptotic competitive ratio of our algorithms.

Keywords Strip packing · variable widths · online algorithms · competitive ratio

Mathematics Subject Classification (2010) 68Q25 · 68W25 · 68W40

1 Introduction

In the area of packing, there are different kinds of problems that need to be tackled. One of them is the strip packing problem. It was first introduced by

A. Bódis
University of Szeged, Faculty of Science and Informatics
Szeged, Hungary
Tel.: +36-30-6479688
E-mail: abodis@inf.u-szeged.hu

J. Csirik
University of Szeged, Faculty of Science and Informatics
Szeged, Hungary
E-mail: csirik@inf.u-szeged.hu

Baker et al. in [1], and it is one of the best-known two-dimensional packing problems of computer science. Here, we have a list of items and a unit-width strip with infinite height, and our goal is to pack the items into this strip without overlaps such that the top of the uppermost item is as low as possible.

In this paper, we concentrate on a generalized version of this problem, where we have several given strips, but their widths are different (each strip has a unique width). To the best of our knowledge, there is only one existing paper [12] that considers a similar framework. There, the problem was formulated as a scheduling problem, where the items are jobs, their height is the time requirement of the job, their width is the amount of processors needed for the job, and the strips are clusters with different amount of processors (i.e. the width of the strip). The objective function had to minimize the maximum height achieved in any of the strips. We will also investigate this model, and we will define another interesting objective function based on the used area of the strips.

In this study, we consider only the online variant of the problem, where the algorithms know the strips and their widths beforehand, but they have no idea about the input list of items; and even the number of the items is unspecified. The items appear one-by-one and the algorithms have to pack them immediately, and they are not allowed to repack the items later. As usual, we will measure the performance of online algorithms with their competitive ratios. For a problem instance I , let us denote the objective value of algorithm ALG applied to I by $ALG(I)$, and let $OPT(I)$ stand for the offline optimal value for this instance. Then ALG is called asymptotically C -competitive if there exists a constant D such that $ALG(I) \leq C \cdot OPT(I) + D$ for every instance I . The smallest C , which satisfies this inequality is the asymptotic competitive ratio of ALG . When D equals zero, ALG is called absolutely C -competitive, and the smallest such C is the absolute competitive ratio of ALG .

In their paper, Ye and Mei [12] proposed an online algorithm with absolute competitive ratio of 14.2915. Here, we will analyze the asymptotic competitive ratio of the algorithms we introduce.

The first investigation of the online strip packing problem was conducted by Baker and Schwarz [2]. They proposed the so-called *shelf* algorithms: the NextFit Shelf algorithm and the FirstFit Shelf algorithm. The former is asymptotically $2 \cdot r$ -competitive, while the latter is asymptotically $1.7 \cdot r$ -competitive, where $r \in \{x \in \mathbb{R} | x > 1\}$ is a parameter of the algorithm. Csirik and Woeginger [5] introduced the Harmonic Shelf algorithm that has an asymptotic competitive ratio arbitrarily close to $1.69103 \cdot r$.

We will define several algorithms and analyze their asymptotic competitiveness. Our algorithms apply ideas taken from the field of variable-sized bin packing problem, which was first presented by Friesen and Langston [6]. In this model, we have given a fixed collection of bin sizes and we have an infinite number of bins of every size. The objective is to minimize the total size of the bins, i.e. we have to minimize the wasted space. Kinnersley and Langston [8] described two ways to adapt NextFit and FirstFit to the variable-sized bin

packing problem: 'only with Largest bin' and 'using Smallest possible bin'. NextFit was adapted to the former approach, while FirstFit was adapted to both of them. It was shown that all of these three algorithms have an asymptotic competitive ratio of 2. Csirik [3] defined the Variable Harmonic algorithm with an asymptotic competitive ratio of 1.691... Han et al. [7] proposed an algorithmic framework that can apply any Harmonic-based bin packing algorithm to the online variant of the strip packing problem such that their asymptotic competitive ratio remains the same. For a detailed overview of the results for this problem, we refer the reader to [4].

2 Problem definition

Now, we give a formal definition of the variable-width strip packing problem (VWSPP) and introduce the necessary notations.

For an instance of the VWSPP, we will use the notation $I = (\sigma, \mathbf{W})$, where σ is the list of n input items, and \mathbf{W} is the vector of the widths of the m strips.

Every item i is represented by its width w_i and its height h_i . So σ is the list of these pairs; that is, $\sigma = [(w_1, h_1), (w_2, h_2), \dots, (w_n, h_n)]$. The area of item i is denoted by A_i . The height of the highest item will be h_{max} .

The vector \mathbf{W} defines the widths of the strips such that $\mathbf{W} = \langle W_1, W_2, \dots, W_m \rangle$, where W_j is the width of strip j . The height of the strip j , namely the top of the uppermost item in the strip (also called 'makespan'), is denoted by H_j .

Without loss of generality, we can assume, that $0 < h_{max} \leq 1$, and $0 < w_i \leq 1$ holds for every item i , and $1 = W_1 > W_2 > \dots > W_m > 0$ holds for the strips.

Using these notations, in the VWSPP, one has to pack the items of σ without rotations such that they never exceed the width of the given strips, and the items never overlap each other (they can touch the boundaries of each other). In the online variant being investigated here, the items appear one-by-one, and repacking them is not allowed.

We will consider two objective functions for this problem. The first one, which is based on the area being used from the strips, can be formulated as $\min \sum_{j=1}^m (H_j \cdot W_j)$. We will denote this area-based model as VWSPP-A. The other one is a makespan-based objective formulated as $\min \max_{j=1}^m H_j$, and denoted by VWSPP-M.

The reader might think that these two objectives are equivalent, but this is not the case, which can be demonstrated by the following example. Let $\mathbf{W} = \langle 1, 0.3 \rangle$, that is, we have a strip with a width of 1 and another with a width of 0.3, and σ contains n items of size $(0.3, 1)$ – that is, every item has a width of 0.3 and height of 1. Then, for the area-based objective function, the optimal solution is to pack every item into the 0.3-width strip, so we use an area of $0.3 \cdot n$, and the makespan is n . However, for the makespan-based objective function, the optimal solution would be to pack the items in a balanced way, meaning we have to pack $\frac{n}{4}$ items into the 0.3-width strip, and $\frac{3n}{4}$ items into the 1-width strip, so the height of each strip (i.e. the makespan) is only $\frac{n}{4}$.

3 Dimension reducing algorithms

Next, we will define a class of the dimension reducing algorithms. They use a partitioning to adapt the standard one dimensional bin packing algorithms to the VWSPP. We will use this definition and the lemma below to make the proofs in this paper simpler.

Definition 1 A variable-width strip packing algorithm is a dimension reducing algorithm, if it partitions the strips vertically into blocks of (possibly) different height such that the height of the items packed into a block is at least α times the height of the block (where $0 < \alpha \leq 1$), and it packs the items into these blocks horizontally using a one dimensional variable-sized bin packing algorithm.

Let us remark that these blocks are used in a quite similar way as the bins in the variable-sized bin packing problem, because the items are packed next to each other such that they cannot exceed the width of the strip.

Lemma 1 *Suppose we have a dimension reducing algorithm ALG for the VWSPP, which uses the variable-sized bin packing algorithm ALG1 for the horizontal packing inside the blocks, and ALG1 is asymptotically C-competitive for the variable-sized bin packing problem. Then ALG is asymptotically $\frac{1}{\alpha} \cdot C$ -competitive for the VWSPP-A.*

Proof Let us consider separately the groups of items packed into blocks with the same height. The packing of such a group is equivalent to solving the variable-sized bin packing problem where the sizes of the items are the widths of the original items and the possible bin sizes are the widths of the strips.

Since we know that ALG1 is asymptotically C-competitive for the variable-sized bin packing problem, we can be sure, that asymptotically OPT cannot be more than C times better than ALG in the horizontal direction. Furthermore, ALG uses at least α fraction of every block vertically by definition. Since, even OPT cannot do better than completely fill every strip, thus OPT – even by mixing items with small and large height – cannot be more than $\frac{1}{\alpha}$ times better than ALG vertically.

These two results lead us to confirm the lemma. □

4 NextFit algorithm

The NextFit algorithm is one of the most widely known online algorithms of the bin packing problem. It keeps only one bin open at a time, and it tries to pack the upcoming item into this bin. When the item does not fit into the bin, NextFit closes this bin and opens a new one. A closed bin is never used again for a later item.

Now, we will introduce a version of the NextFit algorithm for the VWSPP. We will start with the definition of the NextFit Shelf algorithm, which is a well-known variant of NextFit for the strip packing problem. Then, we will modify

this algorithm such that it is able to handle variable-widths. This modification will be similar to the one made in [8] for adapting NextFit to the variable-sized bin packing problem.

The Shelf algorithms pack the items of a strip packing problem such that every item is assigned to a shelf based on its height, and a standard bin packing algorithm is applied to choose among the shelves with the same height. More formally, let $h(p) = \frac{1}{r^k}$ denote the height of the shelf p , where $k \in \mathbb{Z}^+ \cup \{0\}$ and $r \in \{x \in \mathbb{R} \mid x > 1\}$ is a parameter of the algorithm. Then the item i with size (w_i, h_i) is assigned to shelf p , iff $h(p) = \frac{1}{r^k}$ and $\frac{1}{r^{k+1}} < h_i \leq \frac{1}{r^k}$.

The NextFit Shelf (NFS) algorithm is a Shelf algorithm, which uses NextFit to choose among the shelves with the same height; that is, NFS checks to see if there is any shelf with the desired height. If there is not, or the current item does not fit into this shelf, then a new shelf is opened with this height, the item is packed into this and the earlier shelf with this height is closed (if there is any).

In [8], NextFit was adapted to the variable-sized bin packing problem such that it uses only the largest bins, and it never opens a smaller one. This algorithm was called NFL (Next Fit, only with Largest bins). It was proven that NFL is asymptotically 2-competitive for the variable-sized bin packing problem.

Applying the idea of NFL to the NextFit Shelf algorithm, we can define the NFSL (NEXT FIT SHELF, ONLY WITH LARGEST STRIP) algorithm for the VWSPP. This algorithm packs every item into the largest strip (which has the width of 1), and uses the NextFit Shelf algorithm to pack the items inside this strip.

In the next lemma for the VWSPP-A, we will state a result generalised to a whole class of algorithms. Then, we will present our result for the NFSL algorithm in the VWSPP-A.

Lemma 2 *For the VWSPP-A, the asymptotic competitive ratio is at least $2 \cdot r$ for every (online and offline) Shelf algorithm using just the largest strip.*

Proof The proof is based on the following problem instance. Let $\mathbf{W} = \langle 1, \frac{1}{2} + \varepsilon \rangle$ and let σ contain items only with size $(\frac{1}{2} + \varepsilon, \frac{1}{r} + \varepsilon)$. Then these algorithms pack every item into the 1-width strip on a 1-height shelf, and only one item goes into each shelf. OPT can pack every item into the other strip exactly onto each other, so OPT can pack without any wasted area. This immediately gives us the lower-bound. □

Theorem 1 *The NFSL algorithm is asymptotically $2 \cdot r$ -competitive for the VWSPP-A, and this ratio is tight.*

Proof For the lower-bound, we can apply Lemma 2 for NFSL by definition.

We know that NFL is asymptotically 2-competitive for the variable-sized bin packing problem [8]. Since NFSL is a dimension reducing algorithm using NFL with $\alpha = \frac{1}{r}$, we can apply Lemma 1 to get the upper-bound. □

In the next lemma, we will state a general result for the VWSPP-M.

Lemma 3 *Suppose we have a Shelf algorithm ALG which is asymptotically C -competitive for the standard strip packing problem. Let us define ALGL as ALG, adapted to the VWSPP, such that only the largest strip is used. For the VWSPP-M, ALGL is asymptotically $C \cdot m$ -competitive.*

Proof Let us define the OPTL algorithm such that it uses only the 1-width strip, but it packs the items inside this strip in an optimal way.

Then, we get the following inequality from the competitive ratio of ALG in the standard strip packing problem: $\frac{ALGL}{OPTL} \leq C$.

Furthermore, OPT can use the other strips as well, and it can pack the items into the same height in every strip (if this is possible). This means that $\frac{OPTL}{m} \leq OPT$ holds.

When we combine these two inequalities, we get the upper-bound of the theorem:

$$\frac{ALGL}{OPT} \leq \frac{ALGL}{\frac{OPTL}{m}} = \frac{ALGL}{OPTL} \cdot m \leq C \cdot m$$

□

Theorem 2 *The NFSL algorithm is asymptotically $2 \cdot r \cdot m$ -competitive for the VWSPP-M, and this ratio is tight.*

Proof For the upper-bound, we can apply Lemma 3 for NFSL by definition, because NFS is asymptotically $2 \cdot r$ competitive for the standard strip packing problem. [2]

To prove the tightness of this bound, let us examine the following problem instance. Let $\mathbf{W} = \langle 1, 1 - \varepsilon, 1 - 2 \cdot \varepsilon, \dots, 1 - (m - 1) \cdot \varepsilon \rangle$, and let σ contain $n = 2 \cdot k$ items such that each height is $\frac{1}{r} + \varepsilon$ and each width comes from the following list:

$$\left[\frac{1}{2} - \frac{(m-1) \cdot \varepsilon}{2}, \frac{(m-1) \cdot \varepsilon}{2} + \varepsilon \right]^k$$

NFSL packs these items with a total height of k into the 1-width strip, because, after each pair, a new shelf is opened with a height of 1.

Meanwhile, OPT packs the first items of each iteration in an equally balanced way among all the strips, because two such items fit next to each other in any strip. If ε is small enough, then all the narrow items can be packed next to each other into the 1-width strip. So, OPT can pack these items into a height of $\frac{k}{2 \cdot m} \cdot \left(\frac{1}{r} + \varepsilon \right) + \frac{1}{r} + \varepsilon$.

Comparing the performance of NFSL and OPT for this input, we get the following ratio:

$$\lim_{\substack{k \rightarrow \infty \\ \varepsilon \rightarrow 0}} \frac{k}{\frac{k}{2 \cdot m} \cdot \left(\frac{1}{r} + \varepsilon \right) + \frac{1}{r} + \varepsilon} = \lim_{k \rightarrow \infty} \frac{k}{\frac{k}{2 \cdot m} \cdot \frac{1}{r} + \frac{1}{r}} = \frac{1}{\frac{1}{2 \cdot m \cdot r}} = 2 \cdot m \cdot r$$

□

5 FirstFit algorithm

The FirstFit (FF) algorithm is also a well-known online bin packing algorithm. The key difference to NextFit is that FirstFit keeps all the bins open at the same time. If an item fits into any of the bins, it will choose the earliest opened one. For the strip packing problem, FirstFit was adapted to FirstFit Shelf (FFshelf), which uses FirstFit to choose the appropriate shelf to pack the item. This means that it maintains a list of shelves and it packs the item into the first opened shelf with the desired height. [2]

FirstFit was adapted in two ways for the variable-sized bin packing problem in the paper by Kinnersley et al. [8]: FFL (FirstFit, only with Largest bin) and FFS (FirstFit, using Smallest possible bin). In this section, we will describe these algorithms and adapt them to the VWSPP as we did for the NFS case.

5.1 FFSL

The FFSL (FIRSTFIT SHELF, ONLY WITH LARGEST STRIP) algorithm differs from NFSL only by using FirstFit instead of NextFit; that is, we pack every item into the largest strip, and use the FirstFit Shelf algorithm inside this strip.

For the VWSPP-A, FFSL has exactly the same competitive ratio as NFSL has.

Theorem 3 *The FFSL algorithm is asymptotically $2 \cdot r$ -competitive for the VWSPP-A, and this ratio is tight.*

Proof For the lower-bound, we can apply Lemma 2 to FFSL by using the definition.

We know that FFL is asymptotically 2-competitive for the variable-sized bin packing problem [8]. Since FFSL is a dimension reducing algorithm using FFL with $\alpha = \frac{1}{r}$, we can apply Lemma 1 for the upper-bound. \square

Theorem 4 *The FFSL algorithm is asymptotically $1.7 \cdot r \cdot m$ -competitive for the VWSPP-M.*

Proof For this upper-bound, we can apply Lemma 3 for FFSL by definition, because FirstFit Shelf is asymptotically $1.7 \cdot r$ competitive for the standard strip packing problem. [2] \square

Theorem 5 *The FFSL algorithm is asymptotically not better than $\frac{10}{6} \cdot r \cdot m$ -competitive for the VWSPP-M.*

Proof For this lower-bound, let us examine the following problem instance. Let $\mathbf{W} = \langle 1, 1 - \varepsilon, 1 - 2 \cdot \varepsilon, \dots, 1 - (m - 1) \cdot \varepsilon \rangle$, and let σ contain $n = 18 \cdot k$ items such that each height is $\frac{1}{r} + \varepsilon$ and each width comes from the following

list:

$$\begin{aligned} & \left[\frac{1 - (m-1) \cdot \varepsilon}{6} - 2 \cdot \varepsilon_2 \right]^{6k}, \\ & \left[\frac{1 - (m-1) \cdot \varepsilon}{3} + \varepsilon_2 \right]^{6k}, \\ & \left[\frac{1 - (m-1) \cdot \varepsilon}{2} + \varepsilon_2 \right]^{6k}, \end{aligned}$$

where $\varepsilon_2 > \frac{(m-1) \cdot \varepsilon}{2}$.

These items are packed by FFSL into the largest strip in the following way. From the first block, 6 items are packed into the same shelf. The second block is packed pairwise. The items of the third block are packed into separate shelves one-by-one. So, $k + 3k + 6k$ shelves are opened with the height of 1. This means that FFSL packs these items into a height of $10 \cdot k$.

However, OPT can pack one item from every block next to each other, and it can pack these $(1 - (m-1) \cdot \varepsilon, \frac{1}{r} + \varepsilon)$ size items in a balanced way among the strips. So, OPT packs with a height of $\frac{6k}{m} \cdot (\frac{1}{r} + \varepsilon)$.

Consolidating these results, we get the following:

$$\lim_{n \rightarrow \infty} \frac{FFSL}{OPT} = \lim_{k \rightarrow \infty} \frac{10k}{\frac{6k}{m} \cdot (\frac{1}{r} + \varepsilon)} = \frac{10}{\frac{6}{m \cdot r}} = \frac{10}{6} \cdot m \cdot r$$

□

5.2 FFSS

In Kinnersley's and Langston's paper [8], the other adaptation of FirstFit to the variable-sized bin packing was FFS (FirstFit, using Smallest possible bin), which opens the new bins with the smallest size the current item fits into. Using the basic idea of this algorithm, we will define the FFSS (FIRST FIT SHELF, USING SMALLEST POSSIBLE STRIP) algorithm such that, if an item does not fit into any of the open shelves, it will open a new shelf in the smallest strip the item fits into.

Theorem 6 *The FFSS algorithm is asymptotically $2 \cdot r$ -competitive for the VWSPP-A, and this ratio is tight.*

Proof We know that FFS is asymptotically 2-competitive for the variable-sized bin packing problem [8]. Since FFSS is a dimension reducing algorithm using FFS with $\alpha = \frac{1}{r}$, we can apply Lemma 1 to get the upper-bound.

To prove the tightness of this upper-bound, let us look at the following problem instance. Let $\mathbf{W} = \langle 1, 1 - \varepsilon \rangle$, and let σ contain items with a width of $\frac{1}{2}$ and a height of $\frac{1}{r} + \varepsilon$. FFSS packs these items one-by-one into the smaller strip, while OPT fills the space completely by packing them into the larger strip pairwise. □

Theorem 7 *The FFSS algorithm is asymptotically not constant competitive for the VWSPP- M .*

Proof Let $\mathbf{W} = \langle 1, \frac{1}{n} \rangle$, and let σ contain n items with a width of $\frac{1}{n}$ and a height of 1.

FFSS packs these items into the smaller strip, and produces a makespan of n , while OPT packs every item into the 1-width strip and its makespan is just 1. \square

6 Variable Harmonic Shelf algorithm

The Harmonic algorithm was defined for the standard bin packing problem by Lee and Lee [9]. The basic idea is to classify the items based on their sizes, and pack the classes separately into bins. The classification is implemented by partitioning the interval $(0, 1]$ into subintervals $I_k (k = 1 \dots M)$, where $I_k = \left(\frac{1}{k+1}, \frac{1}{k} \right]$ for $1 \leq k < M$, and $I_M = \left(0, \frac{1}{M} \right]$. M is a positive integer parameter of the algorithm.

The asymptotic competitive ratio of this algorithm can be described with the harmonic sequence of numbers. In [9], the harmonic sequence has a recursive definition: $\pi_1 = 1, \pi_i = \pi_{i-1} \cdot (\pi_{i-1} + 1)$. Then the competitive ratio of the Harmonic algorithm is the limit of the sum of the reciprocals of the first n values; that is, $H_\infty = \lim_{n \rightarrow \infty} H_n = \lim_{n \rightarrow \infty} \sum_{i=1}^n \frac{1}{\pi_i} = 1.691 \dots$

Csirik and Woeginger [5] defined the Harmonic Shelf algorithm for the strip packing problem, which uses the partitioning of the Harmonic algorithm to classify the items based on their widths. It packs the items into the shelves such that each shelf only contains items with the same width-class. It was proven that the asymptotic competitive ratio of this algorithm is $H_\infty \cdot r$.

For the variable-sized bin packing problem, Csirik introduced the Variable Harmonic (VH) algorithm [3] with the asymptotic competitive ratio of H_∞ . It was proven by Seiden [10] to be an optimal online algorithm for the bounded space variant of the problem.

Here, we propose an adapted version of VH for the VWSPP: the VARIABLE HARMONIC SHELF (VHS) algorithm.

The height-classification of the items is the usual classification of the shelf algorithms.

The width-classification of the items is based on the definition given in [10]. First, we have to define the set T :

$$T_i = \left\{ \frac{W_i}{j} \mid j \in \mathbb{N}^+, W_i > j \cdot \varepsilon \right\}, \quad T = \bigcup_{i=1}^m T_i,$$

where $\varepsilon \in (0, 1]$ is a parameter of the algorithm and \mathbb{N}^+ is the set of the positive natural numbers. The numbers in T are denoted by t_1, t_2, \dots, t_v such that $1 = t_1 > t_2 > \dots > t_v$. Furthermore, $t_{v+1} = \varepsilon$ and $t_{v+2} = 0$.

The type of the items is defined by the intervals $I_k = (t_{k+1}, t_k]$, where $k = 1, 2, \dots, v+1$. Then, an item i is of type k if $w_i \in I_k$. The interval I_k is of class γ and order j , if $t_k = \frac{W_\gamma}{j}$ (we can break ties arbitrarily).

The VHS algorithm packs items into the same shelf only if they are of the same type. When the item i arrives, we specify its type and the proper height of its shelf. From the type, we also get the class; that is, the strip to be packed into, and the order; that is, the number of items packed next to each other. If there is a shelf for the current type, where item i fits into, then item i is packed into it. If there is no proper open shelf, then the new shelf is opened in the strip corresponding to the class of this type. The items of type $v+1$, that is, the ones with a width falling into $(0, \varepsilon]$, are packed using NextFit Shelf into the 1-width strip.

Theorem 8 *VHS is asymptotically $r \cdot \Pi_\infty = r \cdot 1.691\dots$ -competitive for the VWSPP-A.*

Proof We know that VH is Π_∞ -competitive for the variable-sized bin packing problem [3]. Since VHS is a dimension reducing algorithm using VH with $\alpha = \frac{1}{r}$, we can apply Lemma 1 to get the upper-bound. \square

Theorem 9 *VHS is asymptotically not competitive for the VWSPP-M.*

Proof Let $\mathbf{W} = \langle 1, \frac{1}{n} - \varepsilon \rangle$, and let σ contain n items with a width of $\frac{1}{n} - \varepsilon$ and a height of 1.

VHS packs all the items into the smaller strip, so the makespan is n . In the meantime, OPT can pack each item into the larger strip next to each other, so the makespan is only 1. \square

7 Greedy Variable Harmonic Shelf algorithm

Now, we will combine the VHS algorithm with the well-known greedy scheduling algorithm to get a constant competitive ratio for the makespan model. This is called the GREEDY VARIABLE HARMONIC SHELF (GVHS) algorithm.

GVHS uses the same types as VHS, and works in a similar way. The only difference is that when we need to open a new shelf, GVHS (as opposed to VHS) can do this for any strip, where the item fits into. GVHS chooses the strip based on a greedy decision: the new shelf is opened in the lowest strip (i.e. where the top of the uppermost shelf is the lowest), where i fits into.

Theorem 10 *GVHS is asymptotically $2 \cdot r$ -competitive for the VWSPP-A.*

Proof The definition of the shelves implies that GVHS fills the shelves to a level of at least $\frac{1}{r}$ vertically.

Let us look at the horizontal direction. GVHS opens a new shelf for an item only, if this item does not fit into any open shelf with appropriate height and type. This can happen iff, each such shelf is filled half horizontally. We can prove this statement indirectly. Let us suppose that there is a shelf with

appropriate height and type, which is not filled half horizontally. Since the current item does not fit into this shelf, it must be wider than the half of the strip of this shelf. This means that the type has both wider and thinner items, than the half of the strip. However, this is not possible from the definition of the types. So, this is a contradiction. Therefore, almost every shelf is filled half horizontally (except possibly the last shelves for each height and type). \square

Theorem 11 *GVHS is asymptotically $4 \cdot r$ -competitive for the VWSPP-M.*

Proof Let us denote the total area of the shelves of GVHS by $A(GVHS(\sigma))$. Based on Theorem 10, the following inequality holds:

$$A(GVHS(\sigma)) \leq 2 \cdot r \cdot \sum_{i \in \sigma} A_i.$$

For every type k , let $S_{\geq k}$ and $S_{< k}$ denote the set of strips where an item of type k does and does not fit into, respectively. With a similar notation, $\sigma_{\geq k}$ and $\sigma_{< k}$ will be the set of items fitting into (at least) one of the strips in $S_{\geq k}$ and $S_{< k}$, respectively. We should remark that $\sigma_{< k} \subseteq \sigma_{\geq k} \equiv \sigma$.

Now, we can define a lower-bound α_k for the top of the topmost item of type k in any solution, because we know that in the strips of $S_{\geq k}$ one cannot do better than pack those items, that do not fit into any other strip, in a fully balanced way. This gives the following inequality:

$$\alpha_k \geq \frac{\sum_{i \in \sigma} A_i - \sum_{i \in \sigma_{< k}} A_i}{\sum_{j \in S_{\geq k}} W_j} = \frac{\sum_{i \in \sigma \setminus \sigma_{< k}} A_i}{\sum_{j \in S_{\geq k}} W_j}$$

Then, the following holds for OPT:

$$OPT(\sigma) \geq \max_{1 \leq k \leq v+1} \alpha_k \geq \max_{1 \leq k \leq v+1} \frac{\sum_{i \in \sigma \setminus \sigma_{< k}} A_i}{\sum_{j \in S_{\geq k}} W_j}$$

Furthermore, OPT cannot do better than pack the items completely balanced among the strips. This is stated in the next inequality:

$$OPT(\sigma) \geq \frac{\sum_{i \in \sigma} A_i}{\sum_{j=1}^m W_j}$$

Now, let us investigate GVHS. Let us denote by g the item defining the makespan; that is, g has the highest top of all the items. Let k and p be the type and the shelf of g , and B the bottom of p . The area of p will be denoted by $A(p)$. Then $GVHS(\sigma) = B + h_{max} \leq B + 1$.

We will consider the items of the sets $\sigma \setminus \sigma_{< k}$ and $\sigma_{< k}$ separately in the result of GVHS. The former items are packed into the strips of $S_{\geq k}$ as much balanced as GVHS is able to. The latter items can be packed into the strips of $S_{< k}$, so the total height of the shelves with such items in the strips of $S_{\geq k}$ cannot be more than the height of any strip in $S_{< k}$, because otherwise GVHS would pack those items into this lower strip of $S_{< k}$.

Putting these two observations together, using Theorem 10, and noticing that p was opened on the lowest strip at the moment, we get the following:

$$\begin{aligned}
B &\leq \frac{A(GVHS(\sigma \setminus \sigma_{<k})) - A(p)}{\sum_{j \in S_{\geq k}} W_j} + \frac{A(GVHS(\sigma_{<k}))}{\sum_{j=1}^m W_j} \\
&\leq \frac{2 \cdot r \cdot \sum_{i \in \sigma \setminus \sigma_{<k}} A_i - A(p)}{\sum_{j \in S_{\geq k}} W_j} + \frac{2 \cdot r \cdot \sum_{i \in \sigma_{<k}} A_i}{\sum_{j=1}^m W_j} \\
&\leq 2 \cdot r \cdot \left(\frac{\sum_{i \in \sigma \setminus \sigma_{<k}} A_i}{\sum_{j \in S_{\geq k}} W_j} + \frac{\sum_{i \in \sigma_{<k}} A_i}{\sum_{j=1}^m W_j} \right) \\
&\leq 2 \cdot r \cdot \left(\frac{\sum_{i \in \sigma \setminus \sigma_{<k}} A_i}{\sum_{j \in S_{\geq k}} W_j} + \frac{\sum_{i \in \sigma} A_i}{\sum_{j=1}^m W_j} \right) \\
&\leq 2 \cdot r \cdot (OPT + OPT) \\
&\leq 4 \cdot r \cdot OPT
\end{aligned}$$

Now, we apply this to the objective value of GVHS

$$\begin{aligned}
GVHS(\sigma) &\leq B + 1 \\
&\leq 4 \cdot r \cdot OPT + 1
\end{aligned}$$

□

8 Improved Variable Harmonic Strip algorithm

In this section, we will introduce the IMPROVED VARIABLE HARMONIC STRIP (IVHS) algorithm based on the results of Han et al. [7]. They proposed a grouping and packing framework ($G\&P_A$) for strip packing using any so-called, SUPER HARMONIC algorithm A, and they proved that the asymptotic competitive ratio of the original algorithm does not change with this method. Here, we will define the IVHS algorithm by applying this framework for the VHS algorithm.

The SUPER HARMONIC algorithms were introduced by Seiden in [11]. This class includes the bin packing algorithms based on classifying the items into intervals by their sizes, such as: HARMONIC, REFINED HARMONIC, MODIFIED HARMONIC, MODIFIED HARMONIC 2, HARMONIC+1 and HARMONIC++.

One can readily see that the Variable Harmonic algorithm defined by Csirik [3] is also a SUPER HARMONIC algorithm, which uses the types described as width-classification in Section 6, and it does not pack multiple types into the same bin.

Now, we can define the IVHS algorithm with the $G\&P_A$ framework.

Grouping The items are grouped based on their widths according to the width-classification described in Section 6. This means that this grouping is equivalent to the classification by the types in VHS. The items of type $v + 1$ (i.e. if their width is less than ε) are called *narrow* items, and all the other ones are called *wide* items.

Packing The *narrow* items are packed using the VHS algorithm. We pack the *wide* items with a two-level packing: the first level is packing these items into so-called *slips*, the second level is packing these slips into so-called *bins*.

Items are packed into the same slip only if they are of the same type. If a slip contains the items of type i , then its size is considered to be (t_i, c) , where c is a constant parameter such that $c = o(OPT(\sigma)) > 1$.

Slips are packed into the same bin only if they contain items of the same type, that is, every item of any bin has the same type. If the type of the items in a bin is of the class j , then this bin is packed into the W_j -width strip, and its size is (W_j, c) .

When an item with type i arrives then we check to see whether there is a slip of type i with a total height of items at most $c - 1$. If there is such a slip, then the item is packed into it; otherwise a new slip is opened with size (t_i, c) . When a new slip is created we pack it into a bin. If there is a bin for the type of this slip, then it is packed into that bin; otherwise a new bin is opened for this type on the strip corresponding to the class of the type.

Theorem 12 *IVHS is asymptotically $H_\infty = 1.691\dots$ -competitive for the VWSPP-A.*

Proof We can treat IVHS as a dimension reduction algorithm packing items of at least $\frac{c-1}{c}$ height (i.e. the slips); that is, $\alpha = \frac{c-1}{c}$, into c -height blocks using VH. So we can apply Lemma 1 for an upper-bound of $\frac{c}{c-1} \cdot H_\infty$. Since c is a sufficiently large constant by definition, this leads us immediately to the statement of the theorem. \square

Theorem 13 *IVHS is asymptotically not competitive for the VWSPP-M.*

Proof We can use the same input as in the proof of Theorem 9. IVHS will also use just the smaller strip for the items, while OPT can pack everything into the 1-width strip. \square

9 Greedy Improved Variable Harmonic Strip algorithm

Now, we will define the GREEDY IMPROVED VARIABLE HARMONIC STRIP (GIVHS) algorithm, which is a combination of IVHS and the greedy scheduling algorithm.

GIVHS works in a very similar way to IVHS. The only difference is how one chooses the strips for the new bins. While IVHS always opens the new

bins in the strip defined by the class of the item-type, GIVHS chooses from any strip, where the item fits into, in a greedy way: the lowest strip is chosen (i.e. the one where the top of the uppermost bin is the lowest).

Theorem 14 *GIVHS is asymptotically 2-competitive for the VWSPP-A.*

Proof Since GIVHS, like IVHS, packs items of at least $\frac{c-1}{c}$ in height (i.e. the slips) into c -height bins, GIVHS has an asymptotic vertical area-usage of 1.

Looking at the horizontal direction, we can make a similar assertion as that for GVHS in the proof of Theorem 10. GIVHS opens a new bin for a slip only, if it does not fit into any bin with appropriate type. This can happen iff, each such bin is filled at least half horizontally. The proof of this statement is also similar to that in the proof of Theorem 10. If there were a bin which is not filled half horizontally, then the corresponding type would have both wider and thinner items than the strip of the bin. Thus, almost every bin is filled half horizontally (except possibly the last bin for each type). \square

Theorem 15 *GIVHS is asymptotically 4-competitive for the VWSPP-M.*

Proof We notice that the bins of GIVHS are similar containers inside the strips as the shelves of GVHS. So, this theorem can be proven similarly to the one for Theorem 11. We will present only the differing parts here.

Let us denote the total area of the bins of GIVHS by $A(\text{GIVHS}(\sigma))$. From Theorem 14, the following inequality holds:

$$A(\text{GIVHS}(\sigma)) \leq 2 \cdot \sum_{i \in \sigma} A_i.$$

Let us denote by g the item defining the makespan, that is, g has the highest top of all the items. Let b be the bin of g , and B the bottom of b . The area of b will be denoted by $A(b)$. Then $\text{GIVHS}(\sigma) \leq B + c$.

With the same considerations as in the proof of Theorem 11 and the fact that b was opened on the lowest strip at the moment, we get the following:

$$B \leq \frac{A(\text{GIVHS}(\sigma \setminus \sigma_{<k})) - A(b)}{\sum_{j \in S_{\geq k}} W_j} + \frac{A(\text{GIVHS}(\sigma_{<k}))}{\sum_{j=1}^m W_j} \leq 4 \cdot \text{OPT}(\sigma)$$

Now, we apply this to the objective value of GIVHS:

$$\text{GIVHS}(\sigma) \leq B + c \leq 4 \cdot \text{OPT}(\sigma) + c.$$

Next, we can compare the objective value of GIVHS and OPT, and we can use the fact that $c = o(\text{OPT}(\sigma))$:

$$\lim_{\text{OPT} \rightarrow \infty} \frac{\text{GIVHS}(\sigma)}{\text{OPT}(\sigma)} \leq \lim_{\text{OPT} \rightarrow \infty} \frac{4 \cdot \text{OPT}(\sigma) + c}{\text{OPT}(\sigma)} = 4$$

\square

10 Conclusion

In this paper, we investigated online algorithms for a variant of the strip packing problem, where we have multiple strips with different widths. This model was investigated earlier by Ye and Mei in [12], where the makespan-based objective model and the absolute competitive ratios of algorithms were studied.

Here, we introduced a new objective function based on the area used from the strips. For both objective functions, we examined the asymptotic competitive ratios of several classical algorithms adapted for this problem.

First, we adapted NextFit and FirstFit such that they use only the largest strip all the time. Then we presented an adaptation of FirstFit using the smallest possible strip. We defined the Variable Harmonic Strip algorithm, and we also combined this with greedy scheduling. Finally, we introduced the improved variants of these latter two algorithms. The results obtained are summarized in Table 1.

	Area-based model (VWSPP-A)		Makespan-based model (VWSPP-M)	
	Lower-bound	Upper-bound	Lower-bound	Upper-bound
NFSL	$2 \cdot r$		$2 \cdot r \cdot m$	
FFSL	$2 \cdot r$		$\frac{10}{6} \cdot r \cdot m$	$1.7 \cdot r \cdot m$
FFSS	$2 \cdot r$		∞	-
VHS	?	$r \cdot 1.691 \dots$	∞	-
GVHS	?	$2 \cdot r$?	$4 \cdot r$
IVHS	?	$1.691 \dots$	∞	-
GIVHS	?	2	?	4

Table 1 The asymptotic competitive ratios of the online algorithms investigated. Recall that m denotes the number of the strips and $r \in \{x \in \mathbb{R} | x > 1\}$ is a parameter of the Shelf algorithms.

We have several open questions in connection with the VWSPP and the above-listed algorithms. One can probably prove the tightness of the $1.7 \cdot r \cdot m$ upper-bound for FFSL with a more sophisticated list of items. And it would be nice to find appropriate lower-bounds for the harmonic algorithms described here. Since the Variable Harmonic algorithm was proved to be optimal bounded-space algorithm for the variable-size bin packing problem [10], we believe IVHS is also an optimal bounded-space algorithm for the area-based model, but this conjecture has yet to be proved. The off-line version of the problem is also exciting and worth thorough analysis.

References

1. Baker, B.S., Coffman Jr., E.G., Rivest, R.L.: Orthogonal Packings in Two Dimensions. *SIAM Journal on Computing* **9**(4), 846–855 (1980). DOI 10.1137/0209064. URL <http://epubs.siam.org/doi/10.1137/0209064>

2. Baker, B.S., Schwarz, J.S.: Shelf Algorithms for Two-Dimensional Packing Problems. *SIAM Journal on Computing* **12**(3), 508–525 (1983). DOI 10.1137/0212033. URL <http://epubs.siam.org/doi/10.1137/0212033>
3. Csirik, J.: An on-line algorithm for variable-sized bin packing. *Acta Informatica* **26**(8), 697–709 (1989). DOI 10.1007/BF00289157. URL <http://link.springer.com/10.1007/BF00289157>
4. Csirik, J.: Variable Sized Bin Packing and Bin Covering. In: T.F. Gonzalez (ed.) *Handbook of Approximation Algorithms and Metaheuristics, Second Edition, Volume 1: Methodologies and Traditional Applications*, pp. 539–552. Chapman and Hall/CRC (2018). DOI 10.1201/9781351236423-30. URL <https://doi.org/10.1201/9781351236423-30>
5. Csirik, J., Woeginger, G.J.: Shelf algorithms for on-line strip packing. *Information Processing Letters* **63**(4), 171–175 (1997). DOI 10.1016/S0020-0190(97)00120-8. URL <https://www.sciencedirect.com/science/article/pii/S0020019097001208>
6. Friesen, D.K., Langston, M.A.: Variable sized bin packing. *SIAM Journal on Computing* **15**(1), 222–230. (1986). URL <https://doi.org/10.1137/0215016>
7. Han, X., Iwama, K., Ye, D., Zhang, G.: Approximate strip packing: Revisited. *Information and Computation* **249**, 110–120 (2016). DOI 10.1016/J.IC.2016.03.010. URL <https://www.sciencedirect.com/science/article/pii/S0890540116300013>
8. Kinnersley, N.G., Langston, M.A.: Online variable-sized bin packing. *Discrete Applied Mathematics* **22**, 143–148 (1989). URL <https://core.ac.uk/download/pdf/82055472.pdf>
9. Lee, C.C., Lee, D.T.: A simple on-line bin-packing algorithm. *Journal of the ACM* **32**(3), 562–572 (1985). DOI 10.1145/3828.3833
10. Seiden, S.S.: An Optimal Online Algorithm for Bounded Space Variable-Sized Bin Packing. *SIAM Journal on Discrete Mathematics* **14**(4), 458–470 (2001). DOI 10.1137/S0895480100369948. URL <http://epubs.siam.org/doi/10.1137/S0895480100369948>
11. Seiden, S.S.: On the Online Bin Packing Problem. *Automata, Languages and Programming*, pp. 237–248. Springer, Berlin, Heidelberg (2001). DOI 10.1007/3-540-48224-5_20
12. Ye, D., Mei, L.: On-Line Scheduling of Parallel Jobs in Heterogeneous Multiple Clusters. *Frontiers in Algorithmics and Algorithmic Aspects in Information and Management*, pp. 139–148. Springer, Berlin, Heidelberg (2012). DOI 10.1007/978-3-642-29700-7_13