

# Packing Resizable Items with Application to Video Delivery over Wireless Networks\*

Sivan Albagli-Kim<sup>†</sup>   Leah Epstein<sup>‡</sup>   Hadas Shachnai<sup>§</sup>   Tami Tamir<sup>¶</sup>

## Abstract

Motivated by fundamental optimization problems in video delivery over wireless networks, we consider the following problem of *packing resizable items (PRI)*. Given is a bin of capacity  $B > 0$ , and a set  $I$  of items. Each item  $j \in I$  is of size  $s_j > 0$ . A packed item must stay in the bin for a fixed time interval. To accommodate more items in the bin, each item  $j$  can be *compressed* to a size  $p_j \in [0, s_j)$  for at most a fraction  $q_j \in [0, 1)$  of the packing interval. The goal is to pack in the bin, for the given time interval, a subset of items of maximum cardinality. PRI is strongly NP-hard already for highly restricted instances.

Our main result is an approximation algorithm that packs, for any instance  $I$  of PRI, at least  $\frac{2}{3}OPT(I) - 3$  items, where  $OPT(I)$  is the number of items packed in an optimal solution. Our algorithm yields better ratio for instances in which the maximum compression time of an item is  $q_{max} \in (0, \frac{1}{2})$ . For subclasses of instances arising in realistic scenarios, we give an algorithm that packs at least  $OPT(I) - 2$  items. Finally, we show that a non-trivial subclass of instances admits an *asymptotic fully polynomial time approximation scheme (AFPTAS)*.

## 1 Introduction

Video content delivery over wireless networks is expected to grow exponentially in the coming years. It is driven by applications including streaming TV content to mobile devices, internet video, video on demand, personal video streaming, video sharing applications (from mobile to mobile), video conferencing, and live video broadcasting (cloud to mobile as well as mobile to cloud). In fact, a recent study (Cisco Visual Networking Index [2]) predicts that the mobile video traffic will be approximately two-thirds of the global mobile data traffic by 2015. Improvements in video compression and wireless spectral efficiency will not be sufficient to accommodate this potential demand. This establishes the need for solutions on the intersection of theory and practice.

A common approach taken by companies, to better utilize the available bandwidth, is to deliver video content to the clients using different encodings (see, e.g., [22, 8]). This enables the system to support mobile users, who tend to change location (or viewing devices) throughout

---

\*Work partially supported by the Technion V.P.R. Fund, by Smoler Research Fund, and by the Ministry of Trade and Industry MAGNET program through the NEGEV Consortium ([www.negev-initiative.org](http://www.negev-initiative.org)).

<sup>†</sup>Computer Science Department, Technion, Haifa 32000, Israel. E-mail: [sivana1@cs.technion.ac.il](mailto:sivana1@cs.technion.ac.il)

<sup>‡</sup>Department of Mathematics, University of Haifa, Haifa, Israel. E-mail: [lea@math.haifa.ac.il](mailto:lea@math.haifa.ac.il)

<sup>§</sup>Computer Science Department, Technion, Haifa 32000, Israel. E-mail: [hadas@cs.technion.ac.il](mailto:hadas@cs.technion.ac.il)

<sup>¶</sup>School of Computer science, The Interdisciplinary Center, Herzliya, Israel. E-mail: [tami@idc.ac.il](mailto:tami@idc.ac.il)

the show. It also allows the system to degrade quality-of-service for bounded time intervals, while increasing the number of serviced clients. In wireless services, the available network bandwidth, shared by all users that are covered by an access point, is typically no more than 54Mbps. Therefore, no more than 36 MPEG-1 video streams can be delivered simultaneously to a local area [21]. This places strict limitation on the available bandwidth for simultaneous delivery of video content.

Consider a set of clients requesting to view video content over a wireless network. Suppose that each client is willing to tolerate a lower QoS level for some continuous time interval throughout the delivery.<sup>1</sup> The goal is to select a subset of the clients to be serviced and the QoS level for each client throughout the service, such that the total bandwidth allocated at any time does not exceed the available bandwidth, and the number of satisfied requests is maximized.

More specifically, given is a large database of video files, and a set of  $n$  clients. Suppose that, for some  $q_j \in [0, 1]$ , client  $j$  is willing to view a fraction,  $q_j$ , of her requested video content in low QoS.<sup>2</sup> Each file is stored in the system in several encodings – corresponding to several levels of QoS. Assume that high QoS requires  $s_j$  bandwidth units, while a lower QoS level requires  $p_j$  bandwidth units, for some  $0 < p_j < s_j$ . Let  $B$  denote the total bandwidth available for file transmissions to the clients. The goal is to service the maximal number of clients, such that each client  $j$  receives high-QoS transmission, except maybe for the pre-agreed fraction  $q_j$  of the video show, in which the client may receive a lower QoS. The degradation in QoS transmission may occur at most once throughout the transmission of the video content to certain client, (i.e., along a *contiguous* segment of the transmitted content).

We model this fundamental optimization problem as the following problem of *packing resizable items (PRI)*. Given is a set  $I$  of  $n$  items and a bin of capacity  $B > 0$ . Each packed item must stay in the bin for a given time interval. Each item  $j$  is associated with a size  $0 < s_j \leq B$  (also called expanded or non-compressed size), a compressed size  $0 \leq p_j < s_j$ , and a compression time,  $0 \leq q_j < 1$ , specifying the maximal fraction of the packing interval the item can be stored in its compressed size. The time interval in which item  $j$  is compressed must be contiguous. The goal is to pack in the bin a feasible subset of the items of maximum cardinality for the given time interval. W.l.o.g., we assume that the packing interval is  $(0, 1]$ . Thus, a solution for *PRI* specifies the subset of packed items  $I' \subseteq I$  and, for any  $j \in I'$ , the interval  $(c_j, e_j]$  in which  $j$  is compressed, such that the total size of expanded and compressed items at any time  $t \in (0, 1]$  is at most  $B$ .

The above application of video delivery over wireless network yields a general instance of *PRI*. When all clients share the same low-QoS encoding, we get the special case of *PRI* with uniform compressed size (see Section 4). When all clients share the same pre-agreed low-QoS fraction, we get an instance of *PRI* with uniform compression time (see Section 3.1).

By our definition of *PRI*, in any feasible packing, item  $j$  is expanded during time interval  $(0, c_j]$ , compressed during the interval  $(c_j, e_j]$ , and expanded again during the interval  $(e_j, 1]$  for some  $0 \leq c_j \leq e_j \leq 1$ , where  $0 \leq e_j - c_j \leq q_j$ . When expanded, the item consumes capacity  $s_j$  in the bin, while in its compressed form it consumes capacity  $p_j$ . W.l.o.g., we assume that

<sup>1</sup>The continuity requirement comes from the fact that repeated changes in encoding of the transmitted content may cause the client unpleasant interruptions.

<sup>2</sup>Allowing such degradation in QoS reduces the rates for the clients.

$B < \sum_j s_j$ , since otherwise, all items can be packed with no compression for the whole time interval.

In Appendix A we show that *PRI* is strongly NP-hard already for highly restricted instances, where all items have the same expanded and compressed sizes, or the same compression times. Indeed, solving *PRI* involves the selection of a subset of items to be packed, as well as finding a feasible placement for these items in the bin. This makes *PRI* harder than other single bin packing problems.

Figure 1 presents some examples of *PRI* instances, and their packings. Figure 1(a) presents an optimal packing of 6 items with uniform expanded size, where  $s_j = 1$  for all  $j$ , and uniform compression time, i.e.,  $q_j = 1/3$  for all  $j$ . The compressed sizes are 0.2, 0.4, 0.5, 0.5, 0.6, and 0.8, and the bin capacity is  $B = 5$ . Figure 1(b<sub>1</sub>) presents an optimal packing of 6 items with uniform expanded size, i.e.  $s_j = 1$  for  $j$ , and uniform compressed size  $p_j = 1/3$  for all  $j$ . The compression times are 0.2, 0.4, 0.5, 0.5, 0.6, and 0.8. As shown in the figures, parts of the same item can be stored at different heights in the bin, as long as the total capacity allocated to item  $j$  along its expanded and compressed intervals are  $s_j$  and  $p_j$ , respectively. These characteristics of the packings follow from the nature of our applications, in which the allocation of resource capacity (such as bandwidth, or servers on a cloud) to an element is not required to be contiguous.

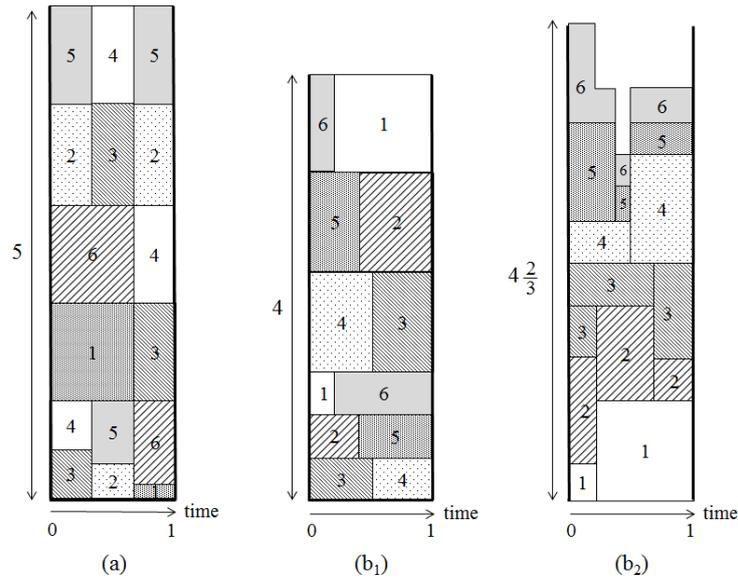


Figure 1: (a) An optimal packing of an instance with uniform  $q$ . An optimal (b<sub>1</sub>) and greedy (b<sub>2</sub>) packing of an instance with uniform  $p$ .

Figure 1(b<sub>2</sub>) demonstrates the challenge of finding a feasible placement for the selected items. The instance is the same as the one packed in Figure 1(b<sub>1</sub>). A natural greedy approach is to pack the items one after the other, while balancing the load along the packing interval. Thus, each item is packed as compressed in a ‘more loaded’ sub-interval, and as expanded in other parts of the packing interval. As shown in Figure 1(b<sub>2</sub>), packing the items using this approach requires a bin of capacity  $B = 4\frac{2}{3}$ . While the greedy approach yields efficient

approximation in some cases, it is not well defined for arbitrary instances, since the more loaded sub-interval may not be contiguous.

Another difficulty in solving *PRI* arises from the fact that there is no natural *ordering* for the packed items. For example, in the Knapsack problem, it is well known that an optimal fractional solution consists of items with the highest profit/size ratios. Moreover, with unit profits, Knapsack can be trivially solved by packing the smallest items. In *PRI*, items have unit profits and can be ordered by their total demand for capacity (given by  $s_j(1 - q_j) + p_j q_j$ ), however, an optimal solution does not necessarily pack the items having the smallest demands. This is valid even for a fractional solution, in which items may be partially packed (in a fractional solution the expanded and compressed sizes are divided).

## 1.1 Related Work

Packing items in a single bin, or in multiple bins, have been widely studied in the computer science, discrete mathematics and operations research communities. Most of these problems are NP-hard. The 0/1-Knapsack problem admits a *fully polynomial time approximation scheme (FPTAS)*, based on a pseudo-polynomial time solution.<sup>3</sup> That is, for any  $\varepsilon > 0$ , a  $(1 - \varepsilon)$ -approximation to the optimal can be found in  $O(n/\varepsilon^2)$ , where  $n$  is the number of items [19]. In contrast, the *multiple knapsack problem (MKP)* is NP-hard in the strong sense, therefore it is unlikely to have an FPTAS [18]. A PTAS for MKP was presented by Chekuri and Khanna [4]. For a comprehensive survey of the Knapsack problem and its variants, see, e.g., [14].

Our problem of packing resizable items is closely related to the *bin covering (BC)* problem, which was widely studied (see, e.g., [1, 5, 6, 13]). The input for bin covering is a set of items  $\{a_1, \dots, a_n\}$ ; each item  $a_j$  has a size  $s(a_j) \in (0, 1)$ . The goal is to pack the items into unit sized bins so as to maximize the number of bins that receive items of total size at least 1. It is easy to see that no polynomial time algorithm can have approximation ratio better than  $\frac{1}{2}$  (indeed, applying such approximation algorithm to instances where the total size of items is 2 would solve the Partition problem). The first *asymptotic approximation scheme (APTAS)* for bin covering was introduced by Csirik et al. [6]. Subsequently, Jansen and Solis-Oba [13] presented an AFPTAS for the problem. We discuss the relation between the two problems in Section 4.

In computational geometry, the problems of *covering a region by rectangles* [3, 11, 20] and *tiling by rectangles* [15] are well studied. Interestingly, these problems also relate to *PRI* (see Section 4.1). However, in both problems, the rectangles can translate, but cannot move along the  $x$ - or  $y$ -axis. Therefore, the techniques used to solve these problems cannot be applied when solving *PRI*.

Other related work deal with efficient broadcasting over wireless networks. This combines technical aspects as well as theoretical ones, in particular, algorithms for efficient bandwidth allocation, streaming, and routing [10, 9].

---

<sup>3</sup>Recently, it was shown in [17] that Knapsack can be solved in pseudo-polynomial time and polynomial space.

## 1.2 Our Contribution

We give a comprehensive study of PRI. For some subclasses of instances, our results are almost the best possible. Let  $OPT(I)$  denote an optimal solution for an instance  $I$ . When clear from the context, we omit  $I$ . We use  $OPT$  also to denote the cardinality of  $OPT(I)$ . Our main result (in Section 2) is an approximation algorithm for general instances of PRI. The performance of the algorithm depends on  $q_{max}$ , the maximum compression time of any item. Specifically, let  $\gamma = \left\lceil \frac{1}{q_{max}} \right\rceil - 1$ . If the compression times may take any value in  $[0, 1)$ , in particular, if  $q_{max} \geq \frac{1}{2}$ , then  $\gamma = 1$ . In this case, the algorithm packs at least  $\frac{2}{3}OPT - 3$  items. If for all  $j$ ,  $0 < q_j < \frac{1}{2}$ , then  $\gamma \geq 2$ , and the algorithm packs at least  $\frac{2\gamma+2}{2\gamma+3}OPT - 3$  items. For the case of uniform expanded size inputs, where  $s_j = 1$  for all  $j$ , the algorithm packs at least  $\frac{3}{4}OPT - 3$  items. We note that in the application motivating our work, all items tend to have small compression times, resulting in good approximation ratio. In particular, for  $\gamma = 6$ , a typical value in our application of video services, we obtain an asymptotic 14/15-approximation algorithm.

For other subclasses of instances arising in realistic scenarios, where compression times are drawn from a divisible sequence, we give (in Section 3) an algorithm that packs  $OPT(I) - 2$  items. Furthermore, we show (in Section 4) that a non-trivial subclass of instances, of items with uniform compressed and expanded size, admits an *asymptotic fully polynomial time approximation scheme (AFPTAS)*.

**Techniques:** In deriving our results for general PRI instances (in Section 2), we make non-standard use of a rounding technique applied in the Harmonic algorithm for Bin packing [16]. In particular, our algorithm initially selects the subset of packed items; then, the compression times of the items are rounded down to one of three values, where each value is a unit fraction that depends on  $\gamma$ . We show how each of these subsets, which has uniform compression time, can be packed.

In developing (in Section 4) an AFPTAS for instances with uniform size, we use a transformation of PRI to the problem of covering a region by sliceable rectangles, which finds applications also in computational geometry. Our *covering with holes* technique (Section 4.1) enables to draw a non-trivial connection between PRI and the bin covering problem. Due to space constraints some of the proofs are given in the Appendix.

## 2 Approximation Algorithm for General Instances

In this section we present an approximation algorithm, denoted  $Alg_{arb}$ , for instances of PRI with arbitrary expanded sizes,  $s_j$ , and arbitrary compressed sizes  $0 < p_j < s_j$ . For an item  $j$ , let  $r_j = s_j - p_j$ , and let the *weight* of an item be  $w_j = s_j(1 - q_j) + p_jq_j = s_j - r_jq_j$ , that is,  $w_j$  is the total capacity required for item  $j$  along the packing interval. For a set of items  $S$ , let  $w(S) = \sum_{j \in S} w_j$ ,  $s(S) = \sum_{j \in S} s_j$ ,  $p(S) = \sum_{j \in S} p_j$ , and  $r(S) = \sum_{j \in S} r_j$ .

The idea of  $Alg_{arb}$  is to select first the subset of packed items, then round down the compression time of each item into one of three values, and then pack each of the resulting sets (having uniform compression times) separately. Let  $I_{uni-q}$  be an instance with uniform compression time, such that for all  $j \in I_{uni-q}$ ,  $q_j = \frac{1}{\alpha}$ . In Section 3.1, we present an almost

optimal algorithm for such instances and show that it is possible to pack  $|I_{uni-q}| - 1$  items in a bin of capacity  $w(I_{uni-q})$ . This algorithm is used as a subroutine by  $Alg_{arb}$ .

The next simple observation considers instances in which items cannot be compressed at all.

**Observation 2.1** *If  $q_j = 0$  for all  $j$ ,  $I$  can be packed in a bin of capacity  $w(I)$ .*

Algorithm  $Alg_{arb}$  partitions the items in  $I$  into three sets.

$$X = \{j \in I \mid \frac{1}{\gamma+1} \leq q_j < \frac{1}{\gamma}\}.$$

$$Y = \{j \in I \mid \frac{1}{2(\gamma+1)} \leq q_j < \frac{1}{\gamma+1}\}.$$

$$Z = \{j \in I \mid 0 \leq q_j < \frac{1}{2(\gamma+1)}\}.$$

Let  $OPT$  be the set of items packed in an optimal solution, and let  $X_{opt}, Y_{opt}, Z_{opt}$  be the subsets of  $X, Y, Z$  respectively, in  $OPT$ .

**Lemma 2.2** *If  $\gamma \geq 2$  then  $[(2\gamma)s(X_{opt}) + 3p(X_{opt})] + [(2\gamma + 1)s(Y_{opt}) + 2p(Y_{opt})] + [(2\gamma + 2)s(Z_{opt}) + p(Z_{opt})] \leq (2\gamma + 3)B$ , and if  $\gamma = 1$  then  $[s(X_{opt}) + 2p(X_{opt})] + [2s(Y_{opt} \cup Z_{opt}) + p(Y_{opt} \cup Z_{opt})] \leq 3B$ .*

**Proof:** Consider the  $2\gamma + 3$  vertical lines  $0, \frac{1}{2(\gamma+1)}, \frac{2}{2(\gamma+1)}, \dots, 1$  in an optimal packing. Since the bin has height  $B$ , the total length of these lines is  $(2\gamma + 3)B$ . Consider the packing of an item  $j \in X_{opt}$ . The compressed part of  $j$  intersects with at most three vertical lines since  $\frac{2}{2(\gamma+1)} = \frac{1}{\gamma+1} \leq q_j < \frac{1}{\gamma} \leq \frac{3}{2(\gamma+1)}$  (which holds for any  $\gamma \geq 2$ ), and the distance between every two adjacent vertical lines is  $\frac{1}{2(\gamma+1)}$ . The expanded part of  $j$  intersects with the rest of the vertical lines. Thus, the total intersection length of all items in  $X_{opt}$  with all the vertical lines is at least  $(2\gamma)s(X_{opt}) + 3p(X_{opt})$ . Similarly, the total intersection length of all items in  $Y_{opt}$  with all the vertical lines is at least  $(2\gamma + 1)s(Y_{opt}) + 2p(Y_{opt})$ , and the total intersection length of the items in  $Z_{opt}$  is at least  $(2\gamma + 2)s(Z_{opt}) + p(Z_{opt})$ . Summing up, and combining with the fact that items in the packing do not overlap, we get the statement of the lemma. The proof for  $\gamma = 1$  is similar, using the three vertical lines  $0, \frac{1}{2}, 1$ .  $\blacksquare$

Note that this lemma cannot be proved using arguments which are simply based on the area of each item. For example, if  $\gamma = 1$ , then an item  $j \in X$  may have (in some solution) an area which is only slightly larger than  $p_j$ , and if an optimal solution consists only of such items, we only find  $p(X_{opt}) < B$ , which is a much weaker statement (since  $s(X_{opt})$  may be much larger than  $p(X_{opt})$ ). Define the *rounded-weight* of an item  $j \in X$  to be  $w_j^r = s_j \frac{\gamma}{\gamma+1} + p_j \frac{1}{\gamma+1}$ . Similarly, for an item  $j \in Y$  let  $w_j^r = s_j \frac{2\gamma+1}{2\gamma+2} + p_j \frac{1}{2\gamma+2}$ , and for an item  $j \in Z$  let  $w_j^r = s_j$ . For  $\gamma \geq 2$ , consider the set of items consisting of the  $\frac{2\gamma+2}{2\gamma+3}|X_{opt}|$  items with the smallest rounded weight from  $X_{opt}$ , the  $\frac{2\gamma+2}{2\gamma+3}|Y_{opt}|$  items with the smallest rounded weight from  $Y_{opt}$ , and the  $\frac{2\gamma+2}{2\gamma+3}|Z_{opt}|$  items with the smallest rounded weight from  $Z_{opt}$ . Denote this set by  $J^*$ . Note that the last item from each set might be fractional. For  $\gamma = 1$ , the set  $J^*$  is defined in the same way, using  $\frac{2}{3}$ -fractions rather than  $\frac{2\gamma+2}{2\gamma+3}$ .

**Lemma 2.3** *The total rounded weight of  $J^*$  is at most  $B$ .*

**Proof:** We give the proof for  $\gamma \geq 2$ . The proof for  $\gamma = 1$  is similar. The total rounded weight of the items in  $J^* \cap X_{opt}$  is at most  $\frac{2\gamma+2}{2\gamma+3}(\frac{\gamma}{\gamma+1}s(X_{opt}) + \frac{1}{\gamma+1}p(X_{opt}))$ . Similarly, the total rounded weight of the items in  $J^* \cap Y_{opt}$  is at most  $\frac{2\gamma+2}{2\gamma+3}(\frac{2\gamma+1}{2\gamma+2}s(Y_{opt}) + \frac{1}{2\gamma+2}p(Y_{opt}))$ , and the

total rounded weight of the items in  $J^* \cap Z_{opt}$  is at most  $\frac{2\gamma+2}{2\gamma+3}s(Z_{opt})$ . We need to prove that  $\frac{2\gamma+2}{2\gamma+3}(\frac{\gamma}{\gamma+1}s(X_{opt}) + \frac{1}{\gamma+1}p(X_{opt})) + \frac{2\gamma+2}{2\gamma+3}(\frac{2\gamma+1}{2\gamma+2}s(Y_{opt}) + \frac{1}{2\gamma+2}p(Y_{opt})) + \frac{2\gamma+2}{2\gamma+3}s(Z_{opt}) \leq B$ . This is equivalent to showing that  $\frac{2}{2\gamma+3}(\gamma s(X_{opt}) + p(X_{opt})) + \frac{1}{2\gamma+3}((2\gamma+1)s(Y_{opt}) + p(Y_{opt})) + \frac{2\gamma+2}{2\gamma+3}s(Z_{opt}) \leq B$ , which is true by Lemma 2.2.  $\blacksquare$

$Alg_{arb}(I, B)$ :

1. For each  $j \in I$ , round down  $q_j$ :
  - a. If  $j \in X$ , let  $q_j = \frac{1}{\gamma+1}$ .
  - b. If  $j \in Y$ , let  $q_j = \frac{1}{2(\gamma+1)}$ .
  - c. If  $j \in Z$ , let  $q_j = 0$ .
2. Sort the items in  $I$  in a non-decreasing order according to their rounded weights  $w_j^r$ .
3. Let  $I'$  be the longest prefix in the sorted list having total rounded weight at most  $B$ . Let  $X' = I' \cap X$ ,  $Y' = I' \cap Y$ , and  $Z' = I' \cap Z$ .
4. Pack  $|X'| - 1$  items of  $X'$  in the bin using capacity  $w^r(X')$ .
5. Pack  $|Y'| - 1$  items of  $Y'$  in the bin using capacity  $w^r(Y')$ .
6. Pack  $Z'$  in the bin using capacity  $w^r(Z')$ .

Algorithm  $Alg_{arb}$  receives an arbitrary PRI instance and proceeds as follows. In Steps 4 and 5, we use an almost optimal algorithm for instances with uniform compression times, given in Section 3.1.

**Lemma 2.4** *The algorithm outputs a feasible packing.*

**Theorem 2.5** *If  $\gamma \geq 2$ , then  $Alg_{arb}$  returns a packing of at least  $\frac{2\gamma+2}{2\gamma+3}OPT - 3$  items. If  $\gamma = 1$ , then  $Alg_{arb}$  returns a packing of at least  $\frac{2}{3}OPT - 3$  items.*

**Proof:** We present the proof for  $\gamma \geq 2$ . The proof for  $\gamma = 1$  is similar. In steps 4-6, the algorithm packs  $|X'| - 1 + |Y'| - 1 + |Z'| = |I'| - 2$  items. We show that  $|I'| \geq \frac{2\gamma+2}{2\gamma+3}OPT - 1$ . Recall that  $I'$  form a prefix of the sorted list. Thus, the total rounded weight of any subset of at least  $|I'| + 1$  items is larger than  $B$ . This implies that any set of items having total rounded weight at most  $B$  includes less than  $|I'| + 1$  items. In particular, by Lemma 2.3, as the total rounded weight of  $J^*$  is at most  $B$ , we have that  $|I'| + 1 \geq |J^*|$  (note that  $J^*$  might include fractions and we consider here their fractional size). By definition of  $J^*$ , it includes  $\frac{2\gamma+2}{2\gamma+3}OPT$  items. We conclude that  $|I'| \geq \frac{2\gamma+2}{2\gamma+3}OPT - 1$ .  $\blacksquare$

**Uniform Expanded Sizes:** In the case  $\gamma = 1$ , if for all items  $s_j = 1$ ,  $Alg_{arb}$  returns a packing of at least  $\frac{3}{4}OPT - 3$  items. To prove this, we can show that it is either the case that the set of  $\frac{3}{4}|X|$ ,  $\frac{3}{4}|Y|$  and  $\frac{3}{4}|Z|$  items of minimum weight of the sets  $X, Y, Z$ , respectively, have total weight at most 1, or that the set of all items of  $X$  together with  $\lfloor B - w(X) \rfloor$  additional items (that can always be packed), has a sufficiently large number of items. The bound  $\frac{3}{4}$  is tight for this case, while the bound  $\frac{2}{3}$  is tight for the general case (though the additive constant can be reduced to 2 by uniting  $Y$  and  $Z$ ).

### 3 Almost Optimal Algorithm for Divisible Compression Times

In this section we present an almost optimal algorithm for instances in which the compression times form a divisible sequence.

**Definition 3.1** A sequence  $\frac{1}{d_1} > \frac{1}{d_2} > \dots > \frac{1}{d_z}$  is divisible if for all  $1 \leq i \leq z$ ,  $d_i$  is an integer, and for all  $1 \leq i \leq z-1$ ,  $d_{i+1}$  divides  $d_i$ .

For example,  $\frac{1}{2}, \frac{1}{4}, \frac{1}{8}, \frac{1}{16}, \frac{1}{32}$  and  $\frac{1}{3}, \frac{1}{9}, \frac{1}{63}, \frac{1}{126}, \frac{1}{504}$  are divisible sequences. Let  $I$  be a PRI instance, in which item  $j$  has an arbitrary expanded size,  $s_j$ , an arbitrary compressed size,  $p_j$ , and a compression time  $q_j$ , such that  $q_j = \frac{1}{d_i}$  for some  $1 \leq i \leq z$ , where  $\{\frac{1}{d_i}\}$  is a divisible sequence. As we show in Appendix A, even the more restricted case of  $PRI$ , with unit expanded size and uniform compression time  $1/m$ , is strongly NP-hard. Clearly, an approximation algorithm with an additive error of 1 is the best one can expect.

Algorithm  $Alg_{div}(I, B)$  packs either  $OPT - 1$  or  $OPT - 2$  items, depending on several parameters of the instance (see below).

**Property 3.1** For any item  $j$ , before  $j$  is placed, the packing consists of strips whose widths are multiples of  $q_j$ , such that the load along each strip is uniform.

**Lemma 3.2** Throughout Step (d), the gap between the loads on any two time-points in  $(0, 1]$  is at most  $r_j$ .

$Alg_{div}(I, B)$ :

1. Sort the items of  $I$  in non-decreasing order by weights, i.e.,  $w_1 \leq w_2 \leq \dots$
2. Let  $I'$  be the longest prefix of the sorted list of total weight at most  $B$ .
3. Pack  $|I'| - 1$  or  $|I'| - 2$  items in the bin:
  - (a) Remove from  $I'$  the item  $j'$  for which  $r_{j'} = \max_{j \in I'} r_j$ .
  - (b) If  $w_{j'} < r_{j'}$  and  $q_{j'} > \min_{j \in I'} q_j$ , remove from  $I'$  also an item with maximal weight.
  - (c) Sort the remaining items in non-increasing order by compression times, i.e.,  $q_1 \geq q_2 \geq \dots$  denote the sorted list by  $L$ .
  - (d) while  $L \neq \emptyset$ , place the next item in the bin as follows:
    - i. Let  $\ell$  be the maximum load (=height) along  $(0, 1]$  in the bin.
    - ii. Let  $(x, \ell)$  be the leftmost (with the minimal  $x$ -coordinate) point in the bin having load  $\ell$ .
    - iii. Pack  $j$  as compressed in time interval  $(x, x + q_j]$ , and as expanded in  $(0, x]$ , and  $(x + q_j, 1]$ .

**Lemma 3.3** The packing generated in Step (d) does not exceed the height  $B$ .

**Proof:** Assume by contradiction that placing item  $j$  causes an overflow. Thus, a strip of width at least  $q_j$  has load more than  $B$ . By Lemma 3.2, the load along the rest of the bin is at least  $B - r_j$ . Therefore, the total weight of the packed items is more than  $q_j \cdot B + (1 -$

$q_j)(B - r_{j'}) = B - r_{j'} + q_j r_{j'}$ . On the other hand, the total weight of the packed items is at most  $B - w_{j'} = B - s_{j'} + q_{j'} r_{j'}$ . If  $w_{j'} \geq r_{j'}$  then  $p_{j'} \geq r_{j'} q_{j'}$ , and we get that  $r_{j'} > r_{j'}$ , which is a contradiction. Similarly, if  $q_{j'} \geq q_j$ , we get that  $r_{j'} > s_{j'}$ , which is also a contradiction. Otherwise, the condition in Step (b) holds and the total weight of the packed items is at most  $B - w_{j'} - w_{max}$ , where  $w_{max}$  is the maximal weight of an item in  $I' \setminus \{j'\}$ . We distinguish between two cases:

- (i) If  $w_{max} \geq w_{j'}$  then we use the fact that  $q_{j'} \leq \frac{1}{2}$  to conclude that  $w_{j'} > \frac{1}{2} r_{j'}$  and thus  $B - w_{j'} - w_{max} \leq B - 2w_{j'} \leq B - r_{j'}$ . Thus the total weight of packed item is less than the lower bound of  $B - r_{j'} + q_j r_{j'}$  on this value. A contradiction.
- (ii) If  $w_{max} < w_{j'}$  then all packed items have weight at most  $w_{max}$ . This implies that among the packed items  $r_{max} < 2w_{max}$  (again, since all compression times are at most  $\frac{1}{2}$ ). Lemma 3.2 is now valid with a smaller maximal gap. Specifically, along Step (d), the gap between the loads on any two time-points along  $(0, 1]$  is less than  $2w_{max}$ . Having an overflow implies that the total load of placed items is more than  $B - 2w_{max}$ , i.e., higher than the upper bound of  $B - w_{max} - w_{j'}$  on this value. A contradiction.

■

**Theorem 3.4** *If  $w_{j'} \leq r_{j'}$  or  $q_{j'} = \min_{j \in I'} q_j$ , then the algorithm packs  $OPT - 1$  items. Otherwise, it packs  $OPT - 2$  items.*

**Proof:** Since the total weight of items in  $OPT$  is at most  $B$ , and  $I'$  is the longest prefix of the sorted list having total weight at most  $B$ , it must be that  $|I'| \geq OPT$ . By Lemma 3.3, the algorithm packs all but one or two items from  $I'$ , depending on the stated condition. ■

**Remark:** For a given constant-size set  $U$  of items, it is possible to test in constant time whether all items of  $U$  can be packed. This can be done (also for arbitrary instances) by enumerating over all permutations of  $U$  and applying a greedy rule for each permutation (we omit the details). This implies that any algorithm which finds a solution in which  $OPT/r - C$  are packed, can be converted to an  $(r + \varepsilon)$ -approximation algorithm. In particular, an algorithm that packs  $OPT - \theta(1)$  items can be converted to a PTAS.

### 3.1 Uniform Compression Time

The above algorithm can be applied also if all items share the same compression time. That is, for all  $j$ ,  $q_j = q$  for any  $0 < q < 1$ . Note that we do not require  $q$  to be a unit fraction of the form  $1/\lceil \frac{1}{q} \rceil$ . Given  $q$ , let  $\gamma = \lceil \frac{1}{q} \rceil$ . We show that it is sufficient to consider only packings with a specific structure, in which the actual compression time of all items is exactly  $1/\gamma$ .

**Lemma 3.5** *There exists an optimal packing in which the items are divided into  $\gamma$  groups, such that the items of group  $1 \leq i \leq \gamma$  are all compressed during the interval  $(\frac{i-1}{\gamma}, \frac{i}{\gamma}]$ .*

Thus, for any instance with uniform  $q$ , it is possible to round down the compression times of all items to  $1/\lceil \frac{1}{q} \rceil$ , and apply  $Alg_{div}(I, B)$  on the resulting instance - the compression times for a divisible sequence with a single element. Note that in this case, the condition in Step (b) does not hold (as  $q_{j'} = \min_{j \in I'} q_j$ ), and thus, the algorithm packs  $OPT - 1$  item. Therefore,

**Theorem 3.6** *Let  $I$  be an instance with uniform compression time. It is possible to pack  $OPT(I) - 1$  items in polynomial time.*

Let  $I$  be an instance with uniform compression time  $q = \frac{1}{\alpha}$ , for some integer  $\alpha$ . By applying  $Alg_{div}(I, B)$  with a bin of capacity  $B = w(I)$ , all items except for one are packed. The following result is used in our algorithm for arbitrary instances (see Section 2).

**Theorem 3.7** *Let  $I$  be an instance with uniform compression time  $q = \frac{1}{\alpha}$ . It is possible to pack  $|I| - 1$  items in a bin of capacity  $w(I)$ .*

## 4 An AFPTAS for Instances with Uniform Size

In this section we present an improved approximation algorithm for instances with uniform expanded size. For such instances we assume, w.l.o.g., that for all items  $s_j = 1$ , and that the compression times of all items are positive (items with  $q_j = 0$  can be added to the bin if it is not fully utilized by compressible items). We first describe the *Covering with Holes* technique that we use for deriving our result.

### 4.1 Technique: Covering with Holes

Our approach is to utilize in the best way the *holes* created while items are packed as compressed. More formally, each item  $j$  defines a hole  $h_j$ , which is associated with a width  $0 < q_j < 1$  and a height  $0 < p'_j < 1$ , where  $p'_j = 1 - p_j$ . We describe formally the *Covering with Holes* problem, and its relation to *PRI*.

In the *Covering with Holes (CwH)* problem, we are given a number  $B$ , and a set  $H_I$  of  $n$  holes, such that each hole  $h_j$  is associated with a width  $0 < q_j < 1$  and a height  $0 < p'_j < 1$ . The goal is to find the maximal  $h$  such that it is possible to cover an  $h \times 1$  rectangle using at most  $B + h$  holes. A solution for *CwH* is given by a set of holes  $H = \{h_1, \dots, h_\ell\}$  where  $\ell \leq B + h$ . For each hole  $h_j$ , the solution specifies what is the  $x$ -interval  $X_j = (x_{1j}, x_{2j}]$  in which  $h_j$  is spanned, such that  $x_{2j} - x_{1j} = q_j$ . A solution covers an  $h \times 1$  rectangle, if for every  $0 \leq t \leq 1$  it holds that the total height of holes whose  $x$ -interval includes  $t$  is at least  $h$ .

Figure 2(a) presents a cover of a  $1 \times 1$  rectangle with 7 holes. Note that the holes need not be placed in the covered area as rectangles. For example, hole  $h_4$  spans along  $(0.6, 1]$  and its height is 0.5. This hole corresponds to an item in the *PRI* instance having  $q_j = 0.4$  and  $p_j = 0.5$ . Similarly, hole  $h_2$  spans along  $(0, 0.6]$  and its height is 0.3. This hole corresponds to a *PRI* item having  $q_j = 0.6$  and  $p_j = 0.7$ . Note also that it is possible to have overlapping holes as well as holes whose interval spans beyond the covered area.

In the next result we indicate the relation between the problems *CwH* and *PRI*. The relation is illustrated in Figure 2(b).

**Theorem 4.1** *Let  $I$  be an instance of *PRI* with uniform expanded size, and let  $H_I$  be the associated set of holes. It is possible to cover an  $h \times 1$  rectangle using at most  $B + h$  holes from  $H_I$  if and only if it is possible to pack  $B + h$  items of  $I$  in a bin of capacity  $B$ .*

Theorem 4.1 is valid for the maximal  $h$  such that at most  $B + h$  holes cover an  $h \times 1$  rectangle. Therefore, a solution for *CwH*, induces a solution for *PRI*.

Note that the above correspondence between the two problems holds for any instance of *PRI* with uniform expanded sizes.

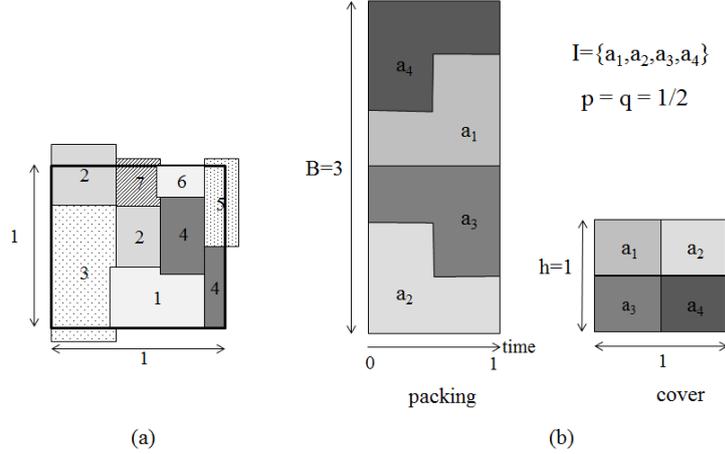


Figure 2: (a) Covering a  $1 \times 1$ -rectangle with 7 holes. (b) A packing in a bin with capacity 3 and the corresponding cover of a  $1 \times 1$  rectangle. The items define four holes of dimensions  $\frac{1}{2} \times \frac{1}{2}$ .

## 4.2 Approximation Scheme

In the *uniform size* case, the expanded and the compressed size are uniform for all items and equal to 1 and  $p$ , respectively (for some  $0 < p < 1 \leq B$ ). The compression times of the items (i.e.,  $q_j$ ), may be arbitrary.

**Observation 4.2** *W.l.o.g., in any feasible packing, the number of compressed items is uniform along the interval  $(0, 1]$ .*

As shown in Appendix A, PRI is strongly NP-hard already for uniform size instances. In the following, we use our technique of *covering with holes* to obtain an AFPTAS for such instances.

Assume the items are sorted such that  $q_1 \geq q_2 \geq \dots \geq q_n$ . Clearly, if for two items  $j_1, j_2$ , it holds that  $q_{j_1} > q_{j_2}$  then item  $j_1$  can be accommodated in the space allocated for item  $j_2$ . Therefore, simple exchange argument implies that w.l.o.g., an optimal solution packs the *first* items in the instance. In covering terms, this means that the covering is performed using the widest holes.

Since the compressed size is uniform, all the holes determining the covering have height  $(1 - p)$ . By Observation 4.2, in the *CwH* problem, there exists a cover in which the number of holes is uniform along the interval  $(0, 1]$ . In other words, the holes are divided into  $g = \left\lceil \frac{h}{1-p} \right\rceil$  groups (horizontal strips), such that group  $1 \leq i \leq g$  covers the  $i$ -th strip of height  $(1 - p)$ , of the rectangle. In other words, the covering problem in this case can be seen as a problem of maximizing the number of horizontal strips (each of height  $(1 - p)$ ), covered by rectangles of height  $1 - p$  and widths  $q_1, q_2, \dots$

In order to obtain an approximate solution for *CwH*, we use an AFPTAS for the *bin covering* problem. In the *bin covering* problem, we are given a set  $\{a_1, \dots, a_n\}$  of items, each item  $a_j$  has a size  $s(a_j) \in (0, 1)$ . The goal is to pack the items into bins in a way that maximizes the number of bins that receive items of total size at least 1.

Let  $A_{bc}$  be such an AFPTAS, and let  $c$  be its asymptotic constant (such an AFPTAS with  $c = 4$  is given in [13]). That is, given  $n$  items of sizes  $s(1), \dots, s(n)$ , such that  $\forall i, s(i) < 1$ ,  $A_{bc}$  uses the items to cover  $b$  bins of size 1, where  $b \geq (1 - \varepsilon)b^* - c$ , and  $b^*$  is the number of bins covered by an optimal solution. Given an instance  $I$  for  $PRI$ , let  $H_I$  be the corresponding instance for  $CwH$ . Define the following instance  $C_I$  for *bin covering*: for every hole  $j \in H_I$  of size  $(1 - p) \times q_j$ , include in  $C_I$  an item of size  $q_j$ .

Given  $H_I, h$ , we want to answer the following question: “Is it possible to cover an  $h \times 1$  rectangle with at most  $B+h$  holes from  $H_I$ ?” Since, unless  $P = NP$ , this question is unlikely to be decided in polynomial-time, our algorithm answers a slightly different question. Specifically, *Decision Algorithm*( $H_I, h$ ) receives  $H_I$  and  $h$  as an input, and returns true if it is possible to cover a  $((1 - \varepsilon)h - c) \times 1$  rectangle with at most  $(1 - \varepsilon)(B + h) - c$  holes from  $H_I$ . The algorithm uses  $A_{bc}$  as a subroutine.

Decision Algorithm ( $H_I, h$ ):

1. Let  $C_I$  be the input for *bin covering* corresponding to  $H_I$ .
2. Run  $A_{bc}$  on the first  $(1 - \varepsilon)(B + h) - c$  items in  $C_I$  (corresponding to the widest holes in  $H_I$ ).
3. Let  $b$  be the number of bins covered by  $A_{bc}$ .
4. If  $b \geq \frac{(1 - \varepsilon)(h + B) - B - c}{1 - p}$ , return true else return false.

**Lemma 4.3** *Let  $OPT_c$  be an optimal solution for  $CwH$  of the instance  $H_I$ . Assume that  $OPT_c$  covers a  $(n^* - B) \times 1$  rectangle with at most  $n^*$  holes. Then, a  $((1 - \varepsilon)n^* - B - c) \times 1$  rectangle can be covered with at most  $(1 - \varepsilon)n^* - c$  holes and Decision Algorithm( $h$ ) returns true for  $h = n^* - B$ .*

To obtain an AFPTAS for  $PRI$ , we can use binary search to find the maximal  $1 \leq h \leq n - B$  such that it is possible to cover a  $((1 - \varepsilon)h - c) \times 1$  rectangle with at most  $(1 - \varepsilon)(B + h) - c$  holes. We summarize in the next result.

**Theorem 4.4** *Let  $n^*$  be the maximal number of items that can be packed in a bin of capacity  $B$ . Then the above scheme returns a packing of  $(1 - \varepsilon)n^* - c$  items, in a bin of capacity  $B$ .*

**Proof:** Let  $n^*$  be the maximal number of items that can be packed in a bin of capacity  $B$ . By Theorem 4.1,  $OPT$  covers  $(n^* - B) \times 1$  with at most  $n^*$  items. When  $n^* - B$  is examined by the approximation scheme, it returns true (by Lemma 4.3), and a cover of a  $(1 - \varepsilon)h^* - c \times 1$  rectangle with at most  $(1 - \varepsilon)(B + h^*) - c$  holes is returned. By Theorem 4.1, this cover induces a packing of  $(1 - \varepsilon)n^* - c$  items in a bin of capacity  $B$ . ■

## References

- [1] S.B. Assman, D. S. Johnson, D.J. Kleitman and J.Y.T Leung. On a dual version of the one dimensional bin packing problem. In *J. Algorithms*, 5(4):502-525, 1984.
- [2] The Cisco Visual Networking Index (VNI). <http://www.cisco.com/en/US/netsol>
- [3] S. Chaiken, D. J. Kleitman, M. Saks, and J. Shearer. Covering regions by rectangles. In *SIAM. J. on Algebraic and Discrete Methods*, 2(4):394-410, 1981.
- [4] C. Chekuri and S. Khanna. A PTAS for the multiple knapsack problem. In *Proc. of SODA*, 213–222, 2000.
- [5] J. Csirik, J.B. Frenk, G. Galambos and A.H.G. Rinnooy Kan. Probabilistic analysis of algorithms for dual bin packing problems. In *J. Algorithms*, 12:189-203, 1991.
- [6] J. Csirik, D. S.Johnson and C. Kenyon. Better approximation algorithms for bin covering. In *In Proceedings of SODA*, 557–566, 2001.
- [7] DiviTrackIP Distributed Closed-Loop Statistical Multiplexing (IP-based), Harmonic. In [http://www.harmonicinc.com/view\\_product.cfm?id=273](http://www.harmonicinc.com/view_product.cfm?id=273).
- [8] eyeIO Unveils Breakthrough Video Encoding Technology, Netflix First to Deploy, Press release. <http://www.engadget.com/2012/02/03/netflix-teams-with-eyeio-to-lower-bandwidth-usage/>
- [9] F. H. P. Fitzek, S. Hendrata, P. Seeling, and M. Reisslein, Video Streaming in Wireless Internet. *CRC Press*, 2004.
- [10] B. Furht and S.A. Ahson (Editors). Handbook of Mobile Broadcasting: DVB-H, DMB, ISDB-T, AND MEDIAFLO. Auerbach Publications, 2008.
- [11] D. .S Franzblau and D. J. Kleitman . An algorithm for covering polygons with rectangles. In *Information and Control archive*, 1986.
- [12] M. R. Garey, David S. Johnson. Computers and Intractability: A Guide to the Theory of NP-Completeness.W. H. Freeman& Co. 1979.
- [13] K. Jansen and R. Solis-Oba. An asymptotic fully polynomial time approximation scheme for bin covering. *Theoretical Computer Science*, 306, 543–551, 2003.
- [14] H. Kellerer, U. Pferschy and D. Pisinger. Knapsack Problems. 2004.
- [15] C. Kenyon and R. Kenyon. Tiling a Polygon with Rectangles. In *Proc. of FOCS*,1992.
- [16] C. C. Lee and D. T. Lee. A simple on-line packing algorithm. *Journal of the ACM*. vol. 32(3), 1985.
- [17] D. Lokshtanov and J. Nederlof. Saving space by algebraization. In *Proc. of STOC*, 2010.
- [18] D. Pisinger Algorithms for Knapsack problems, *Ph.D. Thesis, Dept. of Computer Science, University of Copenhagen, Denmark*, February 1995.
- [19] S. Sahni. Approximate algorithms for the 0/1 knapsack problem, *J. of the ACM*, 22, 115–124, 1975.
- [20] Y. Stoyan, T. Romanova, G. Scheithauer and A. Krivulya. Covering a polygonal region by rectangles. In *Computational Optimization and Applications*, 2009.
- [21] Duc. A. Tran and T. Nguyen, Broadcasting techniques for video on demand in wireless networks, *Handbook of Mobile Broadcasting* (CRC Press), 2008.
- [22] Video encoding, CyberTech Media Group. <http://www.cybertechmedia.com/videoencoding.html>

## A Hardness of PRI

**Theorem A.1** *For any  $0 < p < s$ , PRI with uniform expanded size  $s$  and uniform compressed size  $p$  is strongly NP-hard.*

**Proof:** We show a reduction from 3-Partition. Given a set of  $3m$  numbers  $S = \{a_1, \dots, a_{3m}\}$  such that  $\forall i, a_i \in (\frac{1}{4}, \frac{1}{2})$  and  $\sum_{i=1}^n a_i = m$ , the goal is to divide  $S$  into  $m$  subsets  $S_1, \dots, S_m \subseteq S$  such that for all  $1 \leq j \leq m$   $\sum_{a \in S_j} a = 1$ . The 3-Partition problem is known to be strongly NP-hard [12].

Given  $S$  and any  $0 < p < s$ , we assume w.l.o.g that  $s = 1$  (scale  $p$  accordingly). Construct the following instance of PRI with uniform expanded size  $s = 1$  and uniform compressed size  $p$ : The set of items is  $A = \{a_1, \dots, a_{3m}\}$ , where  $\forall i, q_i = a_i$  and  $p_i = p$ . The bin has capacity  $B = (2 + p)m$ . We show that  $S$  has a 3-partition if and only if  $A$  can be packed in the bin.

Given a 3-partition of  $S$ , let  $S_1, \dots, S_m \subseteq S$  be the required partition, i.e., for all  $1 \leq j \leq m$   $\sum_{i \in S_j} a_i = 1$ . We pack the items of  $A$  as follows: for every subset  $S_j$ , assume that  $S_j = \{a_{j_1}, a_{j_2}, a_{j_3}\}$ , then item  $j_1$  is compressed during  $(0, a_{j_1}]$ , item  $j_2$  is compressed during  $(a_{j_1}, 1 - a_{j_3}]$ , and item  $j_3$  is compressed during  $(1 - a_{j_3}, 1]$ . Since  $a_{j_1} + a_{j_2} + a_{j_3} = 1$ , the compression time of item  $j_2$  is  $a_{j_2}$  as required. In other words, for every  $S_j$ , exactly one item from  $S_j$  is compressed in any time point  $t \in (0, 1]$ . Clearly, each item is packed as compressed in a contiguous interval. We get, that in every time point, every subset  $S_j$ , requires total height  $2 + p$ . Therefore, all  $m$  subsets require  $(2 + p)m$  which is exactly the capacity of the bin.

For the other direction, assume that the set  $A$  is packed in a bin of capacity  $B = (2 + p)m$ . By Observation 4.2, the number of compressed items is uniform during the packing. Since there are  $3m$  items and the compressed size equals  $p$ , exactly  $m$  items are packed as compressed at every  $t \in [0, 1]$ . Therefore, a total volume of  $mp$  out of the bin capacity is allocated to compressed items. Since  $p$  is uniform, this volume can be viewed as a collection of  $m$  bins of size 1 such that each bin accommodates items having total compression time 1. The packing into the bins induces a partition of  $\{a_1, \dots, a_{3m}\}$  into  $m$  disjoint sets such that sum of items in each set is 1, This partition is the required 3-partition of  $S$ . ■

**Theorem A.2** *PRI with uniform expanded size and uniform compression time is strongly NP-hard.*

**Proof:** Once again, we use a reduction from 3-Partition. Given  $S$ , construct the following instance of PRI with uniform compression time  $q = 1/m$ : The set of items is  $A = \{a_1, \dots, a_{3m}\}$ , where  $\forall i, p_i = a_i$  and  $q_i = 1/m$ . The bin has capacity  $B = 3m - 2$ . We show that  $S$  has a 3-partition if and only if  $A$  can be packed in the bin.

Given a 3-partition of  $S$ , let  $S_1, \dots, S_m \subseteq S$  be the required partition, i.e., for all  $1 \leq j \leq m$   $\sum_{i \in S_j} a_i = 1$ . We pack the items of  $A$  as follows: Let  $S_j = \{a_{j_1}, a_{j_2}, a_{j_3}\}$ . The items originated from  $S_j$  are packed as compressed during  $(\frac{j-1}{m}, \frac{j}{m}]$ . Clearly, each item is packed as compressed in a contiguous interval of length  $q$ . Moreover, since  $a_i \in (\frac{1}{4}, \frac{1}{2})$ , every set  $S_j$  consists of exactly 3 items, thus  $3m - 3$  items are expanded while  $S_j$  is compressed. Thus, for every  $1 \leq j \leq m$ , the total size of compressed and expanded items along  $(\frac{j-1}{m}, \frac{j}{m}]$  is  $3m - 3 + a_{j_1} + a_{j_2} + a_{j_3} = 3m - 2 = B$ , which is bin's capacity.

For the other direction, assume that the set  $A$  is packed in a bin of size  $B = 3m - 2$ . By Lemma 3.5, the interval  $(0, 1]$  can be partitioned into strips, such that exactly 3 items are compressed along every strip. Consider the  $j$ -th strip. Since  $B = 3m - 2$ , and  $3m - 3$  items are expanded along the strip, the 3 item compressed along the strip must have total compressed size 1. In other words, these items can form one subset of the partition. Since each item is compressed in a single strip, the whole packing induced a 3-partition of  $S$ . ■

## B Some Proofs

**Proof of Lemma 2.4:** Since the compression time of each item is rounded down, the packing of each item is feasible. Also, by Theorem 3.7 (see Section 3.1) and Observation 2.1, in steps 4-6, the algorithm uses total capacity  $w^r(X') + w^r(Y') + w^r(Z') = w^r(I')$ . By the choice of  $I'$  in step 3,  $w^r(I') \leq B$ . ■

**Proof of Property 3.1:** The proof is by induction on the number of placed items. Before the first item is placed, the whole  $(0, 1]$  interval is a single strip with load 0. Since  $q_1 = 1/d_i$ , the length of this strip is a multiple of  $q_1$ . In step (d)iii item  $j$  is placed such that its compressed interval is along the highest strip. By the induction hypothesis, the highest strip has width  $aq_j$  for some integer  $a \geq 1$ . If  $a = 1$  the width of this strip remains the same. If  $a > 1$  then the highest strip is divided by item  $j$  into two strips. The left one has width  $q_j$  and load  $\ell + p_j$ , and the right one has width  $(a - 1)q_j$  and load  $\ell + s_j$ . Since the items are sorted by their compression time, it holds that  $q_{j+1} = q_j/z$  for some integer  $z \geq 1$ . Therefore, both  $q_j$  and  $(a - 1)q_j$  divide  $q_{j+1}$ . Also, the width of all other strips do not change, thus the width of any other strips also divides  $q_{j+1}$ . ■

**Proof of Lemma 3.2:** The proof is by induction on the number of placed items. Before the first item is placed, the gap is clearly 0. The placement of the first item might create a gap of at most  $r_{j'}$ . Assume that the claim holds before item  $j$  is placed. In step (d)iii item  $j$  is placed such that its compressed interval is along the highest strip. By property 3.1, this strip has width at least  $q_j$ , thus, the whole compressed interval of item  $j$  is placed along the highest strip, and the load on this interval is increased by  $p_j$ . The added load along the rest of the bin is  $s_j$ . If the load on the highest strip remains the highest, the gap does not increase. If another strip became the most loaded, then the gap could increase to at most  $s_j - p_j = r_j \leq r_{j'}$ . ■

**Proof of Lemma 3.5:** Let  $P$  be a feasible packing of  $n$  items. We present an algorithm for modifying  $P$  to a feasible packing  $P'$  of  $n$  items, such that  $P'$  is organized in strips. Specifically, for each  $1 \leq i \leq \gamma$ : all the items such that their compressed packing starts between  $\frac{i-1}{\gamma}$  and  $\frac{i}{\gamma}$  in  $P$ , will be compressed during the interval  $(\frac{i-1}{\gamma}, \frac{i}{\gamma}]$  in  $P'$ . We show that  $P'$  is feasible by bounding the required volume along each of the strips. For every  $1 \leq i \leq \gamma$ , let  $Z_i$  be the set of items that are compressed in strip  $i$  in  $P'$ . By the modifying algorithm, there exists a time point in  $P$  in which all the items that are not in  $Z_i$  are expanded, therefore,  $B \geq n - |Z(i)| + \sum_{j \in Z(i)} p_j$ . This implies that  $P'$  is feasible along the  $i$ -th strip. all the items in  $Z_i$  are packed together and their packing is feasible. Thus,  $P'$  is a feasible packing of all the  $n$  items. ■

**Proof of Theorem 4.1:** Let  $H$  be a solution for  $CwH$ , in which an  $h \times 1$  rectangle is covered using  $B + h$  holes. We construct a solution for  $PR I$  in which  $B + h$  items are packed. For every hole  $h_j$  in  $H$ , let  $X_j = (x_{1j}, x_{2j}]$  be the interval in which hole  $h_j$  spans. We pack item  $j$  as compressed in  $(x_{1j}, x_{2j}]$ , and as expanded in  $(0, x_{1j}]$  and  $(x_{2j}, 1]$ . We show that for every  $j \in H$ , item  $j$  is legally packed, and the total size of packed items in every time point does not exceed the bin's capacity  $B$ . Note that each hole is placed along a single interval in  $(0, 1]$ , and therefore, each item is compressed for along a continuous time interval.

Let  $t$  be some time point in  $(0, 1]$ , and let  $k_t$  be the number of holes such their  $x$ -interval includes  $t$ , that is,  $k_t = |\{h_j \in H | x_{1j} \leq t < x_{2j}\}|$ . Let  $p_1, \dots, p_{k_t}$  be the compressed ratio of the corresponding items. Since  $H$  is a solution for  $CwH$ , it holds that the total height of the holes covering  $t$  is at least  $h$ . Thus,  $(1 - p_1) + (1 - p_2) + \dots + (1 - p_{k_t}) \geq h \rightarrow (p_1 + \dots + p_{k_t}) - k_t \leq -h$ . We show that there is enough capacity in time point  $t$  in the bin. In the corresponding packing, at time  $t$  there are exactly  $k_t$  compressed items and  $B + h - k_t$  expanded items. Therefore, the capacity needed is  $(p_1 + \dots + p_{k_t}) + B + h - k_t \leq -h + B + h \leq B$ , as required. Since this argument holds for every  $t \in (0, 1]$ , we conclude that  $B + h$  items can be packed.

For the other direction, let  $P$  be a solution for  $PR I$  such that  $|P| = B + h$ . We build a solution for  $CwH$  as follows: Go through all the items  $j$  in  $P$  and set the  $x$ -interval of hole  $h_j$  to be  $X_j = (c_j, e_j]$ .

Since each item is packed during  $(0, 1]$  and compressed continuously, each hole is placed at most once and thus the covering is legal. We show that a  $h \times 1$  rectangle is covered by at most  $B + h$  holes. Let  $k_t$  denote the number of items that are compressed on time  $t \in (0, 1]$ , that is,  $k_t = |\{j \in P | c_j \leq t < e_j\}|$ . Let  $p_1, \dots, p_{k_t}$  be the compressed ratio of the corresponding items. Since  $P$  is a solution for *PRR*, it holds that  $(p_1 + \dots + p_{k_t}) + B + h - k_t \leq B \rightarrow (p_1 + \dots + p_{k_t}) - k_t \leq -h \rightarrow k_t - (p_1 + \dots + p_{k_t}) \geq h$ . Therefore  $(1 - p_1) + (1 - p_2) + \dots + (1 - p_{k_t}) \geq h$ , and  $h$  is covered by at most  $B + h$  holes in time point  $t$ . Since this argument holds for every  $t \in (0, 1]$ , we conclude that  $h$  is covered by at most  $B + h$  holes. ■

**Proof of Observation 4.2:** Assume that  $\hat{n}$  items are packed. Let  $\hat{z}$  be the minimal number of compressed items on any time point along  $(0, 1]$ . At the time  $\hat{z}$  items are compressed,  $\hat{n} - \hat{z}$  items are expanded. Therefore  $B \geq \hat{n} - \hat{z}(1 - p)$ . This implies that if more than  $\hat{z}$  items are compressed at some other time, there is enough capacity to expand at least one of them. ■

**Proof of Lemma 4.3:** OPT packs  $n^*$  items in a bin of capacity  $B$ . Thus, by Theorem 4.1,  $OPT_{covering}$  covers an  $((n^* - B) \times 1)$  rectangle using at most  $n^*$  items. Since the rectangle is divided into horizontal strips of height  $(1 - p)$ ,  $OPT_{covering}$  actually covers  $\frac{n^* - B}{1 - p}$  bins using at most  $n^*$  holes. Averaging arguments imply that using at most  $(1 - \varepsilon)n^* - c$  holes,  $OPT_{covering}$  covers  $\frac{(1 - \varepsilon)(n^* - B) - c}{1 - p}$  bins.

Let  $X$  be the maximal number of bins that receive holes of total size at least 1 from the first  $(1 - \varepsilon)n^* - c$  holes in  $I$ . Then,  $X \geq \frac{(1 - \varepsilon)(n^* - B) - c}{1 - p}$ . In step 2 of the algorithm, the value  $b$  is returned by the AFPTAS for *Bin Covering*. Thus,  $b \geq (1 - \varepsilon)X - c'$ . We need to show that  $b \geq \frac{(1 - \varepsilon)(h + B) - B - c}{1 - p}$ , for  $h = n^* - B$ . In other words, we need to show that  $b \geq \frac{(1 - \varepsilon)n^* - B - c}{1 - p}$ .

Combining the upper and lower bounds for  $b$ , it is sufficient to show the following

$$(1 - \varepsilon)X - c' \geq \frac{(1 - \varepsilon)n^* - B - c}{1 - p}. \quad (1)$$

Since  $p > 0$ , at least two holes are required to cover every bin. Thus,

$$X \geq \frac{n^* - B}{1 - p} - \frac{\varepsilon n^* - c}{2} \quad (2)$$

Combining (1) and (2), we need to show that

$$(1 - \varepsilon)\frac{n^* - B}{1 - p} - \frac{\varepsilon n^* - c}{2} - c' \geq \frac{(1 - \varepsilon)n^* - B - c}{1 - p}.$$

Simple calculation show that this is equivalent to showing

$$n^*(\varepsilon(p - 1) + \varepsilon^2(1 - p)) + c(3 - \varepsilon + p(\varepsilon - 1)) + c'(2p - 2) + 2\varepsilon B \geq 0.$$

Taking  $c' = c$ , we get that  $c(3 - \varepsilon + p(\varepsilon - 1)) + c'(2p - 2) > 0$ , and our goal reduces to showing  $n^*(\varepsilon(p - 1) + \varepsilon^2(1 - p)) + 2\varepsilon B \geq 0$ .

Clearly,  $n^* \geq B$ , and therefore,  $n^*(\varepsilon(p - 1) + \varepsilon^2(1 - p)) + 2B \geq \varepsilon B((p - 1) + \varepsilon(1 - p) + 2)$ . Thus, it is sufficient to show that  $\varepsilon B((p - 1) + \varepsilon(1 - p) + 2) \geq 0$ . Indeed,  $(p - 1) + \varepsilon(1 - p) + 2 > p + 1 > 0$  for any  $p > 0$ . ■