# Near-Tight Approximation Algorithms for Bottleneck Multiple Knapsack Problems


**Lin Chen**
Zhejiang University, Hangzhou, China

**Tingwei Hu**
Zhejiang University, Hangzhou, China

**Yuchen Mao**
Zhejiang University, Hangzhou, China

**Yong Chen**
Hangzhou Dianzi University, China

**Lili Mei**
Hangzhou Dianzi University, China

**An Zhang**
Hangzhou Dianzi University, China

**Guangting Chen**
Zhejiang University of Water Resources and Electric Power, Hangzhou, China

**Guochuan Zhang**
Zhejiang University, Hangzhou, China



## Abstract

In the bottleneck multiple knapsack problem, we are given a set of items and a set of knapsacks, where each item has a profit and a weight, and each knapsack has a capacity. Our goal is to assign items to knapsacks so as to maximize the minimum profit received by any knapsack subject to the capacity constraint. When all knapsacks have identical capacity, we give a $(\frac{2}{3} - \varepsilon)$-approximation algorithm for any constant $\varepsilon > 0$. This result almost matches the $(\frac{2}{3} + \varepsilon)$ inapproximability bound for the bottleneck multiple subset sum problem (Caprara et al., 2000). When the knapsacks can have arbitrary capacities, we propose a $(\frac{1}{2} - \varepsilon)$-approximation algorithm for any constant $\varepsilon > 0$. We also prove a hardness bound of $(\frac{1}{2} + \varepsilon)$ for any constant $\varepsilon > 0$.



**2012 ACM Subject Classification** Theory of computation → Approximation algorithms analysis

**Keywords and phrases** Bottleneck multiple knapsack, approximation algorithms

**Digital Object Identifier** 10.4230/LIPIcs.ICALP.2026.60

**Category** Track A: Algorithms, Complexity and Games

**Related Version** *Full Version*: `http://arxiv.org/abs/2605.05233`

**Funding** *Lin Chen*: National Natural Science Foundation of China [Project No. 6257070197]
*Yuchen Mao*: National Natural Science Foundation of China [Project No. 62402436]
*Guochuan Zhang*: National Natural Science Foundation of China [Project No. 12271477]


## 1 Introduction

Maximizing the minimum profit received by multiple agents is a central objective in the study of fairness and equitable resource allocation. In this paper, we study this objective in the context of knapsack-type constraints.

We consider the *bottleneck multiple knapsack problem* (BMKP), which can be viewed as a max-min variant of the classical multiple knapsack problem (MKP). Given items with profits and weights and multiple knapsacks with capacities, the classical *multiple knapsack problem* (MKP) maximizes the sum of profits among all knapsacks. In BMKP, the objective changes to maximizing the minimum total profit received by any knapsack. Our results address two settings: identical capacities and arbitrary capacities.

The bottleneck (max–min) objective has been studied in a special case of MKP, namely the *multiple subset sum problem* (MSSP), where each item has profit equal to its weight and all knapsacks have identical capacities. Caprara, Kellerer, and Pferschy [5] presented a polynomial-time approximation scheme (PTAS) for the max-sum objective in MSSP and gave a $\frac{2}{3}$-approximation for bottleneck MSSP. They also showed that, for any $\varepsilon > 0$, achieving a $(\frac{2}{3} + \varepsilon)$-approximation is impossible unless P = NP. For the max-sum objective, PTASs for MKP have been studied extensively with successive improvements in running time (see [5, 6, 7, 21, 9, 19, 20]). In contrast, the bottleneck objective for MKP remains far less understood. It remains open whether the same $\frac{2}{3}$ ratio can be attained for the more general bottleneck multiple knapsack problem, or whether stronger inapproximability results hold.

The bottleneck (max-min) objective also arises naturally in the literature on fair allocation of indivisible goods. A central notion in this area is the *maximin share* (MMS), which measures the largest value an agent can guarantee for herself by partitioning the items into feasible bundles and then receiving the least valuable bundle. The MMS criterion has been extensively studied under a variety of feasibility constraints, including cardinality [4, 18], matroid [13, 14], and knapsack-type constraints [11, 17], as a fundamental benchmark for fairness.

Recently, Hummel [17] studied MMS guarantees in hereditary set system valuations, a generalization of the identical-capacity knapsack (budget) setting. His result implies a $\frac{2}{5}$-approximation algorithm for identical-capacity BMKP, which is the best previously known guarantee and still far from the $\frac{2}{3}$ hardness barrier. And it is unclear what approximation ratio can be achieved for arbitrary-capacity BMKP.

For max-sum objectives, one may lose an $\varepsilon$ fraction of the total profit by sacrificing the total profits of a few knapsacks. For instance, under the identical setting, one may apply the APTAS for bin packing [10] to pack guessed items into at most $(1 + \varepsilon)m + 1$ bins where $m$ is the number of knapsacks, and then discard the extra bins at an $\varepsilon$-fraction loss in the overall objective [6]. Such a strategy fails for the bottleneck objective, as the loss must be controlled for every knapsack, which calls for finer per-knapsack structure.

## 1.1 Our Contributions

We give nearly tight approximation guarantees for BMKP in both the identical-capacity and arbitrary-capacity settings. In the identical-capacity setting, we obtain a $(\frac{2}{3} - \varepsilon)$-approximation algorithm, which nearly matches the $(\frac{2}{3} + \varepsilon)$ inapproximability under the assumption that P $\neq$ NP [5]. Let $|I|$ denote the input size.

▶ **Theorem 1.** *In the identical-capacity setting, for any constant $\varepsilon > 0$, there is a polynomial-time $(\frac{2}{3} - \varepsilon)$-approximation algorithm for the bottleneck multiple knapsack problem with running time $|I|^{2^{O(1/\varepsilon \log(1/\varepsilon))}}$.*

In the arbitrary-capacity setting, we establish a nearly tight approximation guarantee at $\frac{1}{2}$. We give a $(\frac{1}{2} - \varepsilon)$-approximation algorithm for any constant $\varepsilon > 0$, and show that no polynomial-time $(\frac{1}{2} + \varepsilon)$-approximation is possible unless P = NP.

▶ **Theorem 2.** *For any constant $\varepsilon > 0$, there is a polynomial-time $(\frac{1}{2} - \varepsilon)$-approximation algorithm for the bottleneck multiple knapsack problem with running time $2^{2^{poly(1/\varepsilon)}} \cdot poly(|I|)$.*

▶ **Theorem 3.** *Unless* P = NP*, the bottleneck multiple knapsack problem has no polynomial-time $(\frac{1}{2} + \varepsilon)$-approximation algorithm for any constant $\varepsilon > 0$.*

The following table summarizes the results of approximation algorithms for multiple knapsack problems under different objectives and capacity settings.

| | MKP (max-sum objective) | | BMKP (max-min objective) | |
|---|---|---|---|---|
| | Negative | Positive | Negative | Positive |
| Identical capacity | NP-hard | $1 - \varepsilon$ [9, 19, 20] | $\frac{2}{3} + \varepsilon$ [5] | $\frac{2}{3} - \varepsilon$ (Theorem 1) |
| Arbitrary capacity | NP-hard | $1 - \varepsilon$ [9, 19, 20] | $\frac{1}{2} + \varepsilon$ (Theorem 3) | $\frac{1}{2} - \varepsilon$ (Theorem 2) |

## 1.2 Related Work

We briefly review approximation schemes for the multiple subset sum problem (MSSP) and the multiple knapsack problem (MKP) with the objective of maximizing the total profit. Caprara et al. [5] gave the first PTAS for MSSP, and also showed that MSSP admits no FPTAS even for two knapsacks unless P = NP. Subsequent work extended the PTAS to MKP under the arbitrary capacity setting as well as reduce its running time (see [5, 6, 7, 21, 9, 19, 20]). So far, the best-known result is due to Jansen [20], which is a PTAS for MKP where knapsacks can have arbitrary capacities, and has a running time of $poly(|I|) + 2^{O(1/\varepsilon(\log(1/\varepsilon))^4)}$. This PTAS is near-optimal in the sense that there is no $poly(|I|) + 2^{o(1/\varepsilon)}$ time PTAS assuming ETH (Exponential Time Hypothesis).

Machine scheduling problems can be viewed as a lower-dimensional analogue of multiple knapsack, where one primarily focuses on the profit dimension (i.e., the processing time of jobs). For identical parallel machines, Hochbaum and Shmoys [15] gave a PTAS for the objective of minimizing the maximum completion time (makespan). Woeginger [26] presented a PTAS for maximizing the minimum completion time. For related machines, PTASs are known for both the max-min and min-max objectives [1, 12]. For unrelated machines, Lenstra, Shmoys and Tardos [23] gave a 2-approximation algorithm for makespan minimization, and also proved that no polynomial-time algorithm can achieve an approximation ratio better than 3/2 unless P = NP.

The max–min variant of scheduling on unrelated machines is commonly known as the *Santa Claus* problem. Interpreting machines as agents and jobs as items with agent-dependent values, the goal is to allocate items so as to maximize the minimum total value received by any agent. The best approximation algorithm known for this problem is due to Chakrabarty et al. [8] achieving an approximation ratio of $\tilde{\Omega}(\frac{1}{n^\varepsilon})$, where $n$ is the number of items. And the best known hardness of approximation stands at $\frac{1}{2}$ [3, 8]. For more results of the Santa Claus problem, please refer to e.g. [2, 24, 22].

# 2 Technical Overview

**Our framework.** Our algorithms follow the standard framework of the PTASs for MKP [9, 19, 20]. Assume that the optimal objective value OPT is known. We classify items into two categories:

- *Expensive items*: items whose profit is large (say, at least $\varepsilon \cdot \mathrm{OPT}$);
- *Cheap items*: items whose profit is small (say, less than $\varepsilon \cdot \mathrm{OPT}$).

By this classification, we may assume that each knapsack contains at most $1/\varepsilon$ expensive items. If some knapsack contains more than $1/\varepsilon$ expensive items, then we can discard the extra ones and still maintain optimality.

Since there are constant number of expensive items per knapsack, we can apply the standard *round-and-guess* approach to them. We round profits and weights so that each expensive item belongs to one of constantly many types. We then show that there exists a feasible solution in which every expensive item is replaced by its (rounded) type. This allows us to guess, for each knapsack, how many items of each type it receives (i.e., a *configuration*), and to pack expensive items according to the guessed configuration while losing only a small amount of profit.

To handle cheap items, we formulate a linear programming. Since each cheap item has profit below $\varepsilon \cdot \mathrm{OPT}$, losing only $O(1)$ cheap items per knapsack decreases the objective by at most $O(\varepsilon) \cdot \mathrm{OPT}$. Thus, it suffices to show that we can round an LP solution to an integral assignment while discarding only a constant number of cheap items per knapsack.

For our max–min objective, the main technical challenge can be phrased as follows:

*Is it possible to round the items in such a way that there exists a solution in which, for every knapsack, feasibility is preserved and the loss in profit is small?*

We discuss the challenges and techniques for the identical capacity setting and the arbitrary capacity setting separately.

## 2.1 Identical Capacities

Without loss of generality, we assume the knapsack capacity $B = 1$ and $\mathrm{OPT} = 1$. Our goal is to find an assignment such that each knapsack receives total profit at least $\left(\frac{2}{3} - \varepsilon\right)$. Let $m$ be the number of knapsacks.

**Challenges.** Consider an optimal solution. The main difficulty is that weight rounding (in particular, rounding up) may force us to drop some items to restore feasibility, and the resulting profit loss in a single knapsack may exceed $(1/3 + \varepsilon)$. This can happen for items that are highly sensitive to weight rounding, including

- *Heavy items:* items with weight close to 1 (e.g., $1 - o(\varepsilon)$) and profit greater than $(1/3+\varepsilon)$;
- *Exact-fit pairs:* two items whose total weight is close to 1 and whose total profit is close to 1, where each item has profit larger than $(1/3 + \varepsilon)$.

Assume there are knapsacks containing such items in the optimal solution. If the weight of any item in the above classes is rounded up, one may have to discard at least one such item, which would incur a profit loss exceeding $(1/3 + \varepsilon)$ and violate the desired guarantee.

**Our algorithm.** We classify items into heavy/light and expensive/cheap categories.

- **Step 1 (Heavy items).** We round heavy items into a constant number of types and guess their assignment to knapsacks.
- **Step 2 (Light-expensive items).** Once we know that knapsack $j$ contains a heavy item $h$ with rounded weight $\widetilde{w}(h)$, we can round the weights of light items using the scaling factor $1 - \widetilde{w}(h)$, and then guess the assignment of light items in each knapsack.
- **Step 3 (Exact-fit pairs).** We determine the placement of exact-fit pairs by solving a maximum matching problem.
- **Step 4 (Cheap items).** We pack the remaining cheap items by solving a linear program and rounding its fractional solution.

**Main techniques.** The main technical component is to show that there exists a solution (of the rounded instance) in which weight rounding is feasible and incurs a profit loss of at most $(1/3 + \varepsilon)$ per knapsack. Starting from an optimal solution, we divide knapsacks into two types depending on whether they contain a heavy item.

**(a)** *Knapsacks with only light items:* We show that removing items of total profit at most $(1/3 + \varepsilon)$ is sufficient to make the weight rounding of the light items feasible.

**(b)** *Knapsacks with both heavy and light items:*

**(b.1)** *Heavy items:* We apply a *rounding* and *shifting* process as follows.

- *Rounding.* We adopt the *linear grouping scheme* from bin packing [10, 25]. Group the heavy items as follows: the first group contains the heaviest $k$ items, the second group contains the next heaviest $k$ items, and so on. Here $k$ is chosen

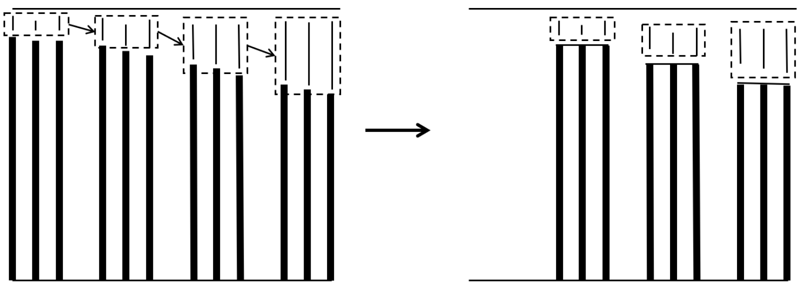

**Figure 1** Illustration of the rounding and shifting procedure. Thick (resp., thin) solid segments denote heavy (resp., light) items. Dashed boxes highlight the items being shifted, and arrows show their destinations.

so that the number of groups is constant. We discard the first group (to be compensated later), and round the weight of heavy items in each group to the heaviest item in that group.

- *Shifting.* To maintain feasibility, we shift light items groupwise from their groups to lighter ones (where heavy items are lighter).

This process guarantees that the weight rounding of heavy items is feasible and the total profit loss is bounded. See Figure 1 for an illustration.

**(b.2)** *Light items:* We show that removing light items of total profit at most $(1/3+\varepsilon)$ from each knapsack is sufficient to make the weight rounding feasible. Moreover, we prove that these removed items can be reassembled and packed into the knapsacks in the discarded group created by the heavy-item shifting step.

## 2.2 Arbitrary Capacities

For more details, see the full version of this paper.

# 3 Notation

In the bottleneck multiple knapsack problem, we are given a set of $m$ knapsacks with capacity $B_1, \ldots, B_m$, and a set $I$ of $n$ items, where each item $i \in I$ has weight $w(i)$ and profit $p(i)$. For any positive integer $k$, we write $[k] = \{1, 2, \ldots, k\}$. A feasible solution consists of $m$ disjoint subsets $I_1, \ldots, I_m \subseteq I$ with $\sum_{i \in I_j} w(i) \leq B_j$ for each $j \in [m]$. Our goal is to find a feasible solution $(I_1, \ldots, I_m)$ that maximizes the minimum profit received by any knapsack. More precisely, we want to maximize $\min_{j\in[m]} \sum_{i\in I_j} p(i)$.

For any subset $I' \subseteq I$ of items, we use $p(I')$ and $w(I')$ to denote the total profit and total weight of the items in $I'$, respectively.

We define the density of an item $i$ to be $\rho(i) = \frac{p(i)}{w(i)}$. If $w(i) = 0$, we interpret $i$ as a dummy (nonexistent) item, and set $p(i) = 0$ and $\rho(i) = 0$. For any subset $I' \subseteq I$ of items, define its density as $\rho(I') = p(I')/w(I')$.

By scaling, we assume that $\mathrm{OPT} = 1$. In the identical-capacity setting, we normalize the capacity and write $B = 1$. In the arbitrary-capacity setting, we assume $B_1 = 1$ and $1 = B_1 \leq B_2 \leq \cdots \leq B_m$.

# 4 An Algorithm for Identical Capacities

In this section, we prove Theorem 1.

## 4.1 Proof Roadmap

We introduce the key concepts of our proof and explain their roles at a high level.

1. **Item categories.** We will define heavy/light items and expensive/cheap items. Our main focus is on *heavy-expensive* items and *light-expensive* items.
2. **Critical items.** Given a solution, an item is *critical* (with respect to this solution) if it is either expensive or heavy, and packed in a knapsack containing at least three items. Critical items are exactly the items that we will round.
3. **Slack solution.** A solution is said to be *slack* if, in every knapsack, the unused capacity is large enough to allow the rounding of its critical items. We will define the precise meaning of "large enough".
4. **Profiles.** A *profile* (of a slack solution) is a subset of heavy items that serves as the rounded types of critical heavy items. More precisely, it is defined so that, after replacing each critical heavy item by one of the items in the profile (its rounded type), the resulting solution remains slack.

**Organization of the proof.** We guess the critical heavy items, critical light items, non-critical items successively. Our argument follows an "existence-guessing" pattern: before each guessing step, we first prove that there exists a slack solution (containing previously guessed structure); we then guess the assignment of items in this slack solution. The main technical component is the first existence result.

1. **Existence of a slack solution and its profile.** We show that there exists a slack solution whose minimum knapsack profit is at least $\frac{2}{3} - \varepsilon$, and whose profile has constant cardinality (Lemma 8). Since the profile has constant size, we can enumerate all possibilities of the profile in polynomial time (Lemma 10).
2. **Guessing critical heavy items.** There exists a slack solution with objective value at least $\frac{2}{3} - 2\varepsilon$ whose profile is the guessed one (Lemma 13). Moreover, for each rounded type in the profile, every critical heavy item in this solution is heavier than any non-critical heavy item of the same type (Lemma 13). We show that these critical heavy items can be guessed (Lemma 14).
3. **Guessing critical light items.** There exists a slack solution with objective value at least $\frac{2}{3} - 3\varepsilon$ whose critical heavy items are the guessed ones (Lemma 18). Moreover, for each rounded type of light items, every critical light item in this solution is heavier than any non-critical light item of the same type (Lemma 18). We show that these critical light items can be guessed (Lemma 19).
4. **Determining non-critical items.** Finally, we complete the packing by handling knapsacks with $|I_j| = 2$ (Lemma 21) and assigning the non-critical cheap items in knapsacks with $|I_j| \geq 3$ (Lemma 22).

## 4.2 Preliminaries

Recall that $\mathrm{OPT} = 1$ and $B = 1$. Our goal is to compute, in polynomial time, a feasible solution $(I_1, \ldots, I_m)$ such that $p(I_j) \geq \frac{2}{3} - 6\varepsilon$ for all $j \in [m]$. (The constant factor in front of $\varepsilon$ is for convenience; by rescaling $\varepsilon$, one can achieve $\frac{2}{3} - \varepsilon$.) For simplicity, we assume that $\varepsilon$ is a sufficiently small constant and $1/\varepsilon$ is an integer.

We assume that every item $i$ satisfies $p(i) < \frac{2}{3} - 6\varepsilon$. Otherwise, if some item has profit at least $\frac{2}{3} - 6\varepsilon$, we can assign it to an empty knapsack and remove both the item and that knapsack from the instance. The remaining instance still admits a solution with objective value at least 1. Consequently, we can assume that every knapsack in the optimal solution contains at least two items.

▶ **Definition 4** (Heavy, Light, Expensive, and Cheap Items). *An item $i$ is heavy if $w(i) \geq 1-\varepsilon^2$, and light otherwise. It is expensive if $p(i) \geq \varepsilon$, and cheap otherwise.*

We assume that each knapsack contains at most $1/\varepsilon$ expensive items. If a knapsack contains more, we keep an arbitrary subset of $1/\varepsilon$ expensive items, whose total profit is at least 1.

▶ **Definition 5** (Critical Items). *Let $\mathcal{S} = (I_1, \ldots, I_m)$ be a feasible solution. The critical items of $\mathcal{S}$ are defined to be the heavy items and the expensive items in any $I_j$ with $|I_j| \geq 3$. All the items in $\mathcal{S}$ that are not critical are called non-critical items.*

There are four types of items: *heavy-expensive*, *heavy-cheap*, *light-expensive*, and *light-cheap*. We assume w.l.o.g. that heavy-cheap items do not exist. By Definition 4, each knapsack contains at most one heavy item. If this heavy item is cheap, removing it decreases the profit of that knapsack by less than $\varepsilon$; by rescaling $\varepsilon$, we can still state the guarantee as at least $\frac{2}{3} - \varepsilon$. Therefore, every heavy item is expensive. Combined with Definition 5, it follows that every critical item is expensive, and thus each knapsack contains at most $1/\varepsilon$ critical items.

▶ **Definition 6** (Slack Solutions). *We say that a solution $(I_1, \ldots, I_m)$ is slack if for any $j \in [m]$ with $|I_j| \geq 3$, exactly one of the following conditions holds.*

**(i)** *All the items in $I_j$ are light, and $w(I_j) \leq 1 - \varepsilon^3$,*

**(ii)** *$I_j$ contains a heavy item $h$, and*

$$w(I_j) - w(h) \leq (1-\varepsilon)(1-w(h)), \quad \textit{or equivalently,} \quad 1 - w(I_j) \geq \varepsilon(1-w(h)).$$

The first case in Definition 6 implies that the unused capacity $1 - w(I_j)$ is at least $\varepsilon^3$. The second case implies that the total weight of light items $w(I_j) - w(h)$ occupy at most a $(1-\varepsilon)$-fraction of $(1 - w(h))$. In other words, the unused capacity $1 - w(I_j)$ is at least $\varepsilon(1 - w(h))$.

▶ **Definition 7** (Profile). *Let $T \subseteq I$ be a subset of heavy items. Let $\mathcal{S} = (I_1, \ldots, I_m)$ be a slack solution. We say that $T$ is a profile of $\mathcal{S}$ if, for any $I_j$ that contains a critical heavy item $h$, there exists $t \in T$ such that $\lfloor p(t)/\varepsilon \rfloor = \lfloor p(h)/\varepsilon \rfloor$ and*

$$w(I_j) - w(h) \leq (1-\varepsilon)(1-w(t)) \leq (1-\varepsilon)(1-w(h)). \tag{1}$$

Note that the inequality above guarantees that $w(t) \geq w(h)$. It implies that if we replace the heavy item $h$ by $t$, we have

$$w\big((I_j \setminus \{h\}) \cup \{t\}\big) - w(t) = w(I_j) - w(h) \leq (1-\varepsilon)\big(1 - w(t)\big).$$

Therefore, if we round $h$ to $t$, the resulting solution is still slack.

## 4.3 Existence of a Structured Slack Solution

We show that there is a slack solution that has a profile of constant size.

▶ **Lemma 8.** *There is a slack solution $\mathcal{S} = (I_1, \ldots, I_m)$ such that $p(I_j) \geq \frac{2}{3} - \varepsilon$ for any $j \in [m]$ and $\mathcal{S}$ has a profile $T \subseteq I$ of heavy items with $|T| \leq O((\frac{1}{\varepsilon})^2)$.*

**Proof.** We first show the existence of a slack solution $\mathcal{S}$ with $p(I_j) \geq \frac{2}{3} - \varepsilon$ for any $j \in [m]$. Then we show the existence of the profile.

**Step 1: Existence of a slack solution.** Given an optimal solution $(X_1^*, \ldots, X_m^*)$ with $p(X_j^*) \geq 1$ for all $j \in [m]$, we convert it into a slack soluton with $p(I_j) \geq \frac{2}{3} - \varepsilon$ for any $j \in [m]$. Without loss of generality, we assume every $X_j^*$ is minimal in the sense that deleting any item makes its total profit less than 1.

By assumption, each item has profit of at most $\frac{2}{3} - 6\varepsilon$ and $|X_j^*| \geq 2$ for any $j \in [m]$. If there exists $j \in [m]$ such that $X_j^*$ contains two items $i_1, i_2$ with $p(i_1) + p(i_2) \geq \frac{2}{3} - \varepsilon$, then we simply set $I_j = \{i_1, i_2\}$. By Definition 5, they are non-critical items and does not affect slackness or the profile.

We thus restrict attention to the case where, for all $i_1, i_2 \in X_j^*$,

$$p(i_1) + p(i_2) < \frac{2}{3} - \varepsilon. \tag{2}$$

We distinguish between the two cases below.

**(Case 1) All items in $X_j^*$ are light.** If $w(X_j^*) \leq 1 - \varepsilon^3$, then let $I_j = X_j^*$; by Definition 6, this knapsack is slack, and we are done. Therefore, we assume $w(X_j^*) > 1 - \varepsilon^3$ and partition the items in $X_j^*$ into three sets:

$$X_e = \{\, i \in X_j^* \mid p(i) > \tfrac{1}{3} + \varepsilon \text{ and } w(i) \leq 1 - \varepsilon^2 \,\},$$

where $w(i) \leq 1 - \varepsilon^2$ holds since all items are light,

$$X_c = \{\, i \in X_j^* \mid p(i) \leq \tfrac{1}{3} + \varepsilon \text{ and } w(i) \leq \varepsilon^3 \,\},$$

and

$$X_d = \{\, i \in X_j^* \mid p(i) \leq \tfrac{1}{3} + \varepsilon \text{ and } w(i) > \varepsilon^3 \,\}.$$

By construction, $X_j^* = X_e \cup X_c \cup X_d$. If $X_d \neq \emptyset$, then we can pick any item $i \in X_d$ and set $I_j = X_j^* \setminus \{i\}$; by Definition 6, we are done. Therefore, we assume that $X_d = \emptyset$.

Note that there is at most one item in $X_e$, since otherwise there would be two items whose total profit is at least $\frac{2}{3} - \varepsilon$, contradicting equation (2). This implies

$$w(X_c) = w(X_j^*) - w(X_e) \geq (1 - \varepsilon^3) - (1 - \varepsilon^2) = \varepsilon^2 - \varepsilon^3 \geq \varepsilon^3. \tag{3}$$

Let $Y \subseteq X_c$ be obtained by sorting the items in $X_c$ in increasing order of density and adding them until the total weight first reaches at least $\varepsilon^3$. The set $Y$ always exists due to equation (3), and it follows that

$$\varepsilon^3 \leq w(Y) \leq 2\varepsilon^3. \tag{4}$$

Since $Y$ consists of items with the smallest density, the density of $Y$ is at most the density of $X_c$. Recall that $X_j^*$ is minimal. Fix any $i \in X_c$. Then $p(X_c \setminus \{i\}) \leq p(X_j^* \setminus \{i\}) < 1$, and since $p(i) \leq \frac{1}{3} + \varepsilon$, we get $p(X_c) < 1 + \frac{1}{3} + \varepsilon$. Hence, by equation (3) and (4), we have

$$p(Y) \leq w(Y) \cdot \frac{p(X_c)}{w(X_c)} \leq 2\varepsilon^3 \cdot \frac{1 + \frac{1}{3} + \varepsilon}{\varepsilon^2 - \varepsilon^3} \leq \frac{1}{3},$$

where the last inequality holds for $0 < \varepsilon \leq \frac{-9+\sqrt{105}}{12}$. Recall that we assume $\varepsilon$ is a sufficiently small constant, so the above inequality holds.

Then we let $I_j = X_j^* \setminus Y$. It is clear that $p(I_j) = p(X_j^*) - p(Y) \geq \frac{2}{3} - \varepsilon$ and $w(I_j) = w(X_j^*) - w(Y) \leq 1 - \varepsilon^3$, which implies that $I_j$ is slack.

**(Case 2) $X_j^*$ contains a heavy item.** By Definition 4, each knapsack contains at most one heavy item. Denote it by $h$. If $p(h) \le \frac{1}{3} + \varepsilon$, let $I_j = X_j^* \setminus \{h\}$. Then $I_j$ contains only light items, and

$$p(I_j) = p(X_j^*) - p(h) \ge \frac{2}{3} - \varepsilon, \qquad w(I_j) = w(X_j^*) - w(h) \le 1 - \varepsilon^3.$$

By Definition 6, $I_j$ is slack. It remains to consider the case where $p(h) > \frac{1}{3} + \varepsilon$.

▷ Claim 9. If $p(h) > \frac{1}{3} + \varepsilon$, there is a subset $Y \subseteq X_j^* \setminus \{h\}$ with

$$\varepsilon \le p(Y) \le \frac{1}{3} \quad \text{and} \quad \varepsilon(w(X_j^*) - w(h)) \le w(Y) \le \varepsilon^2. \tag{5}$$

Proof of Claim 9. Let $X_{light} = X_j^* \setminus \{h\}$. Since $p(h) > \frac{1}{3} + \varepsilon$, every item $i \in X_{\text{light}}$ satisfies $p(i) \le \frac{1}{3} - \varepsilon$. Otherwise $p(h) + p(i) \ge \frac{2}{3}$, contradicting (2). Recall that $p(h) \le \frac{2}{3} - \varepsilon < \frac{2}{3}$ by assumption and $w(h) \ge 1 - \varepsilon^2$ by Definition 4. It follows that

$$p(X_{light}) = p(X_j^*) - p(h) \ge \frac{1}{3} \text{ and } w(X_{light}) = w(X_j^*) - w(h) \le 1 - (1 - \varepsilon^2) = \varepsilon^2. \tag{6}$$

Let $Y' \subseteq X_{light}$ be obtained by sorting the items in $X_{light}$ in increasing order of density and adding them until $p(Y') > 1/3$. Let the last item added to $Y'$ be $l$, and set $Y = Y' \setminus \{l\}$. Recall that every item in $X_{light}$ has a profit of at most $\frac{1}{3} - \varepsilon$. We have

$$\frac{1}{3} \ge p(Y) \ge \frac{1}{3} - p(l) \ge \frac{1}{3} - (\frac{1}{3} - \varepsilon) = \varepsilon. \tag{7}$$

Since $Y$ consists of items of the smallest density, its density is at most the density of $X_{light}$. Moreover, by minimality of $X_j^*$, $p(X_{\text{light}}) = p(X_j^* \setminus \{h\}) < 1$. Therefore,

$$w(Y) \ge p(Y) \cdot \frac{w(X_{light})}{p(X_{light})} > \varepsilon \cdot w(X_{light}) = \varepsilon(w(X_j^*) - w(h)).$$

The strict inequality follows from (7) and $p(X_{\text{light}}) < 1$. Finally, equation (6) implies $w(Y) \le w(X_{\text{light}}) \le \varepsilon^2$. Combining the above inequalities yields (5), completing the proof. ◁

We set $I_j = X_j^* \setminus Y$ where $Y$ is defined in Claim 9. It is clear that $p(I_j) = p(X_j^*) - p(Y) \ge \frac{2}{3} - \varepsilon$ and

$$w(I_j) - w(h) = w(X_j^*) - w(Y) - w(h) \le (1 - \varepsilon)(w(X_j^*) - w(h)) \le (1 - \varepsilon)(1 - w(h)),$$

which implies that $I_j$ is slack.

**Step 2: Existence of a profile of small cardinality.** We have constructed a slack solution $\mathcal{S} = (I_1, \ldots, I_m)$ with $p(I_j) \ge \frac{2}{3} - \varepsilon$ for all $j \in [m]$. Next, we convert $\mathcal{S}$ into another slack solution $\mathcal{S}' = (I'_1, \ldots, I'_m)$ whose profile has constant cardinality.

Consider the knapsacks that contain a heavy item in the slack solution $\mathcal{S}$. Let $\mathcal{J}_H \subseteq [m]$ be the set of indices $j$ such that $I_j$ contains a heavy item. Recall that each such $I_j$ contains exactly one heavy item. We denote it by $h_j$.

We partition $\mathcal{J}_H$ into groups according to the value of $\lfloor p(h_j)/\varepsilon \rfloor$: two indices $j, j' \in \mathcal{J}_H$ are in the same group if $\lfloor p(h_j)/\varepsilon \rfloor = \lfloor p(h_{j'})/\varepsilon \rfloor$. Note that there are at most $1/\varepsilon$ such groups since every heavy item has profit at most $\frac{2}{3} - 6\varepsilon$ and at least $\varepsilon$. We handle each group separately as follows.

Fix an integer $t$ and consider the group $\mathcal{K} = \{ j \in \mathcal{J}_H \mid \lfloor p(h_j)/\varepsilon \rfloor = t \}$. Write $\mathcal{K} = \{1, 2, \ldots, k\}$ after relabeling the corresponding knapsacks, and further relabel so that

$$w(h_1) \leq w(h_2) \leq \cdots \leq w(h_k). \tag{8}$$

For each $j \in [k]$, let $Y_j \subseteq X_j^* \setminus \{h_j\}$ be the set of light items guaranteed by Claim 9. Recall that $I_j = X_j^* \setminus Y_j$. We now construct new sets $I'_1, \ldots, I'_k$ as follows.

- For $1 \leq j \leq \lfloor \varepsilon k \rfloor$, let $\ell = \lceil \frac{2}{3\varepsilon} \rceil$ and define

$$I'_j \ = \ Y_{(j-1)\ell+1} \ \cup \ Y_{(j-1)\ell+2} \ \cup \ \cdots \ \cup \ Y_{j\ell}.$$

By Claim 9, each $Y_j$ consists only of light items and satisfies $p(Y_j) \geq \varepsilon$ and $w(Y_j) \leq \varepsilon^2$. Hence,

$$p(I'_j) \geq \ell\varepsilon \geq \frac{2}{3} \qquad \text{and} \qquad w(I'_j) \leq \ell\varepsilon^2 \leq \varepsilon \leq 1 - \varepsilon^3,$$

where $\ell\varepsilon^2 \leq \varepsilon \leq 1 - \varepsilon^3$ holds for $0 < \varepsilon \leq \frac{1}{3}$. Therefore, $I'_j$ is slack by Definition 6.

- For $\lfloor \varepsilon k \rfloor < j \leq k$, let $s = j - \lfloor \varepsilon k \rfloor$ and set

$$I'_j \ = \ \big(I_j \setminus \{h_j\}\big) \ \cup \ \{h_s\}.$$

That is, we replace the heavy item $h_j$ by $h_s$. Since $s < j$, by equation (8), we have $w(h_s) \leq w(h_j)$, and thus $w(I'_j) \leq w(I_j)$.

Note that $\lfloor p(h_j)/\varepsilon \rfloor = \lfloor p(h_s)/\varepsilon \rfloor$, which implies $p(h_s) \geq p(h_j) - \varepsilon$. By Claim 9 and $I_j = X_j^* \setminus Y_j$, the profit of $I_j$ is

$$p(I'_j) = p(I_j) + p(h_s) - p(h_j) \geq p(X_j^*) - p(Y_j) - \varepsilon \geq 1 - \frac{1}{3} - \varepsilon = \frac{2}{3} - \varepsilon.$$

Meanwhile, we have

$$\begin{aligned} w(I'_j) - w(h_s) &= w(I_j) - w(h_j) \\ &\leq (1-\varepsilon)(1 - w(h_j)) \\ &\leq (1-\varepsilon)(1 - w(h_s)). \end{aligned} \tag{9}$$

The first inequality follows from the slackness of $I_j$ in $\mathcal{S}$. The last inequality holds since $w(h_s) \leq w(h_j)$. By Definition 6, $I'_j$ is slack.

Above all, we obtained a slack solution $\mathcal{S}' = (I'_1, \ldots, I'_m)$. Next, we construct its profile $T$. Let $\delta = \max\{1, \lfloor \varepsilon k \rfloor\}$. Let $\{h_\delta, h_{2\delta}, \ldots, h_{r\delta}\}$ be the representative heavy items of this group, where $r = \lfloor \frac{k}{\delta} \rfloor = O(\frac{1}{\varepsilon})$.

For a group with $\lfloor p(h_j)/\varepsilon \rfloor = t$, we set $T_t = \{h_\delta, h_{2\delta}, \ldots, h_{r\delta}\}$ to be the set of representative heavy items defined above. Here $1 \leq t \leq 1/\varepsilon$ because by assumption, $\varepsilon \leq p(h_j) \leq \frac{2}{3} - 6\varepsilon$ for any heavy item $h_j$. We then define the profile set $T$ to be the union of $T_t$ over all groups, i.e., $T = \bigcup_{t=1}^{1/\varepsilon} T_t$. Since $r = O(\frac{1}{\varepsilon})$ for each group, we have $|T| = O\big((\frac{1}{\varepsilon})^2\big)$.

It remains to argue that $T$ is the profile of $\mathcal{S}' = (I'_1, \ldots, I'_m)$. Towards this, it suffices to show that equation (1) holds for each $T_t$ where $1 \leq t \leq 1/\varepsilon$. To see this, observe that for each heavy item $h_s \in I_j$ where $s = j - \lfloor \varepsilon k \rfloor$, there must exist some $h_{j'} \in T$ with $s \leq j' \leq j$, which implies $w(h_s) \leq w(h_{j'}) \leq w(h_j)$. Therefore, by equation (9), we have

$$\begin{aligned} w(I'_j) - w(h_s) &= w(I_j) - w(h_j) \\ &\leq (1-\varepsilon)(1 - w(h_j)) \\ &\leq (1-\varepsilon)(1 - w(h_{j'})) \\ &\leq (1-\varepsilon)(1 - w(h_s)). \end{aligned}$$

Combined with $\lfloor p(h_{j'})/\varepsilon \rfloor = \lfloor p(h_s)/\varepsilon \rfloor$, we conclude that $T$ is the profile of $\mathcal{S}'$ and $|T| = O\big((\frac{1}{\varepsilon})^2\big)$. The constructed solution $\mathcal{S}'$ is exactly the solution $\mathcal{S}$ claimed in the lemma. ◀

▶ **Lemma 10.** *After guessing for $n^{O\left((\frac{1}{\varepsilon})^2\right)}$ rounds, at least one of the guesses gives the $T$ that is the profile of the slack solution in Lemma 8.*

**Proof.** Since $|T| = O\big((\frac{1}{\varepsilon})^2\big)$, and according to the definition of profile, every element of $T$ must be one of the heavy items (i.e., at most $n$ possibilities), we can guess $T$ via $n^{O\left((\frac{1}{\varepsilon})^2\right)}$ enumerations. ◀

## 4.4 Guessing Critical Heavy Items

We have correctly guessed the profile $T \subseteq I$ satisfying Lemma 8. Our goal in this subsection is to guess the critical heavy items using $T$.

In particular, we associate each heavy item $h \in I$ with a unique item $\mathcal{T}(h) \in T$. We define the *label weight* and *label profit* of $h$ to be the weight and profit of $\mathcal{T}(h)$, respectively.

▶ **Definition 11** (Label weight and label profit). *Fix a profile $T \subseteq I$. For a heavy item $h$, let*

$$\mathcal{T}(h) = \left\{ t \in T \;\middle|\; \left\lfloor \frac{p(t)}{\varepsilon} \right\rfloor = \left\lfloor \frac{p(h)}{\varepsilon} \right\rfloor \text{ and } w(t) \geq w(h) \right\}.$$

*If $\mathcal{T}(h) \neq \emptyset$, choose any $t \in \arg\min_{x \in \mathcal{T}(h)} w(x)$ and define $\widetilde{w}(h) = w(t)$ and $\widetilde{p}(h) = p(t)$. Otherwise, define $\widetilde{w}(h) = +\infty$ and $\widetilde{p}(h) = 0$.*

▶ Remark. One can view item $t$ as the result of rounding $h$ to a nearby item in the profile $T$. Note that we neither alter the instance nor replace $h$ by $t$ here. Instead, we simply introduce the notations $\widetilde{w}(h)$ and $\widetilde{p}(h)$ to label $h$.

▶ **Definition 12** (Type and $\prec$-order). *Two heavy items $h_1, h_2$ are of the same* type *if $\widetilde{w}(h_1) = \widetilde{w}(h_2)$ and $\widetilde{p}(h_1) = \widetilde{p}(h_2)$. We define a partial order $\prec$ on heavy items as follows: for heavy items $h_1, h_2$, we write $h_1 \prec h_2$ iff $h_1$ and $h_2$ have the same type and $w(h_1) < w(h_2)$. Given a set $A$ of heavy items, an item $h \in A$ is* $\prec$-minimal *(resp.,* $\prec$-maximal*) if there is no $h' \in A$ with $h' \prec h$ (resp., $h \prec h'$).*

Note that we cannot directly guess which heavy items are critical within each type because it may take exponential time. Instead, we may guess the number of critical heavy items of each type and select them arbitrarily. To guarantee that such a selection is valid, we need to show that even if we select the heaviest items of each type as critical heavy items, the resulting solution is slack and have the desired objective value.

The following lemma shows that there exists a slack solution in which, within each type, every critical heavy item is heavier than every non-critical heavy item (Property iii). Moreover, the total profit of this solution decreases by at most $\varepsilon$ compared to the solution in Lemma 8 (Property i). And the property of profile is preserved (Property ii). Here we replace the notation $w(t)$ in equation (1) by $\widetilde{w}(h)$ .

▶ **Lemma 13.** *There exists a slack solution $\mathcal{S} = (I_1, \ldots, I_m)$ satisfying the following.*

**(i)** *$p(I_j) \geq \frac{2}{3} - 2\varepsilon$ for any $j \in [m]$.*

**(ii)** *For each $j \in [m]$, if $I_j$ contains a critical heavy item $h$, then*

$$w(I_j) - w(h) \leq (1 - \varepsilon)(1 - \widetilde{w}(h)).$$

**(iii)** *For any critical heavy item $h$, and for any non-critical heavy item $h'$, if $h$ and $h'$ are of the same type, then $h \succ h'$.*

**Proof.** (See the full version of this paper.) ◀

▶ **Lemma 14.** *After guessing $O(n^{(\frac{1}{\varepsilon})^2})$ rounds, at least one of the guesses gives $m$ disjoint set $H_1^*, \ldots, H_m^*$ of heavy items such that there exists a slack solution $\mathcal{S} = (I_1, \ldots, I_m)$ satisfying the following.*

**(i)** $p(I_j) \geq \frac{2}{3} - 2\varepsilon$ *for any* $j \in [m]$.

**(ii)** *For each $j \in [m]$, if $I_j$ contains a critical heavy item $h$, then*

$$w(I_j) - w(h) \leq (1 - \varepsilon)(1 - \widetilde{w}(h)).$$

**(iii)** *The set of critical heavy items of $\mathcal{S}$ in $I_j$ is exactly $H_j^*$.*

**Proof.** We first guess the number of critical heavy items of each type in the slack solution $\mathcal{S}$ satisfying Lemma 13. Note that the number of heavy-item types is at most $|T| = O((1/\varepsilon)^2)$. Index these types by $\tau_1, \ldots, \tau_{|T|}$. For each $\ell \in \big[|T|\big]$, let $n_\ell$ denote the number of critical heavy items of type $\tau_\ell$ in $\mathcal{S}$, and let $(n_1, \ldots, n_{|T|})$ be the count vector. We enumerate all possibilities for $(n_1, \ldots, n_{|T|})$. Since $0 \leq n_\ell \leq n$ for every $\ell$, this takes at most $n^{|T|} = n^{O((1/\varepsilon)^2)}$ rounds.

Fix the round that guesses the correct tuple. In this round, for every $\ell \in [|T|]$ we select the heaviest $n_\ell$ items of type $\tau_\ell$. Call the union of all selected items $H^*$. By Lemma 13, the set of critical heavy items used by $\mathcal{S}$ is exactly $H^*$.

We distribute the items of $H^*$ among $m$ sets $H_1^*, \ldots, H_m^*$, allowing empty sets, so that $|H_j^*| \leq 1$ for all $j \in [m]$ and $\bigcup_{j=1}^m H_j^* = H^*$. We can relabel the knapsacks in $\mathcal{S}$ so that property iii is satisfied. Property i and ii follow directly from Lemma 13. ◀

Note that $H_j^*$ either contains one heavy item or is empty, meaning that there is no critical heavy item in knapsack $j$.

## 4.5 Guessing Critical Light Items

Given $H_1^*, \ldots, H_m^*$ satisfying Lemma 14, our next goal is to guess the critical light items. Similarly to heavy items, we define label weight, label profit, type and $\prec$-order for light items.

Let $e$ be a light-expensive item. We define $W(e)$ to be the scaling factor for rounding the weight of $e$. To keep the solution feasible after rounding, the factor $W(e)$ should depend on the heavy item (if any) packed together with $e$, which is not known in advance. We therefore use a conservative choice: define $t$ to be the heaviest item in the profile $T$ that $w(e) \leq 1 - w(t)$, and then let $W(e)$ be $\varepsilon(1 - \widetilde{w}(t))$, where $\widetilde{w}(t)$ is given in Definition 11. If no such $t$ exists, we use a fixed scaling factor $\varepsilon^3$.

▶ **Definition 15** (Label weight and label profit for light items)**.** *Fix a profile $T \subseteq I$. Let $e$ be a light-expensive item and define $\mathcal{T}(e) = \{\, t \in T \mid w(e) \leq 1 - w(t) \,\}$. If $\mathcal{T}(e) \neq \emptyset$, let $t \in \arg\max_{x \in \mathcal{T}(e)} w(x)$ and set*

$$W(e) = \varepsilon\big(1 - \widetilde{w}(t)\big).$$

*Otherwise, set $W(e) = \varepsilon^3$. We define the* label weight *and* label profit *of $e$ by*

$$\widetilde{w}(e) = \left\lceil \frac{w(e)}{\varepsilon W(e)} \right\rceil \cdot \varepsilon W(e), \qquad \widetilde{p}(e) = \left\lfloor \frac{p(e)}{\varepsilon^2} \right\rfloor \cdot \varepsilon^2.$$

*For each light cheap item $i$, we set $\widetilde{w}(i) = w(i)$ and $\widetilde{p}(i) = p(i)$. For a set of items $X$, we define $\widetilde{w}(X) = \sum_{i \in X} \widetilde{w}(i)$ and $\widetilde{p}(X) = \sum_{i \in X} \widetilde{p}(i)$.*

▶ **Definition 16** (Type and $\prec$-order)**.** *Two light-expensive items $e_1, e_2$ are of the same* type *if $\widetilde{w}(e_1) = \widetilde{w}(e_2)$ and $\widetilde{p}(e_1) = \widetilde{p}(e_2)$. We define a partial order $\prec$ on light-expensive items by declaring $e_1 \prec e_2$ iff $e_1$ and $e_2$ have the same type and $w(e_1) < w(e_2)$.*

Next, we count the number of different types of light-expensive items. Note that if we choose $W(e) = \varepsilon(1 - \widetilde{w}(t))$, then we have $w(e) \le 1 - w(t)$, which implies $\frac{w(e)}{\varepsilon W(e)} \le \frac{1}{\varepsilon^2}$. If we choose $W(e) = \varepsilon^3$, then $\frac{w(e)}{\varepsilon W(e)} \le \frac{1}{\varepsilon^4}$. On the other hand, $\frac{p(e)}{\varepsilon^2}$ is at most $\frac{1}{\varepsilon^2}$. We have the following observation.

▶ **Observation 17** (Constant number of types). *There are at most $O\big((\frac{|T|}{\varepsilon^2} + \frac{1}{\varepsilon^4}) \cdot \frac{1}{\varepsilon^2}\big) = O((\frac{1}{\varepsilon})^6)$ different types.*

The following lemma shows that there exists a solution in which, within each type, every critical light item is heavier than every non-critical heavy item (Property iv). Moreover, the total profit decreases by at most $\varepsilon$ compared to the solution in Lemma 13 (Property i). And if we replace the weights of all critical items by their label weights, the feasibility is still preserved (Property ii), where we apply the property of slack.

▶ **Lemma 18.** *Given $H_1^*, \dots, H_m^*$, there exists a feasible solution $\mathcal{S} = (I_1, \dots, I_m)$ satisfying*

**(i)** *$p(I_j) \ge \frac{2}{3} - 3\varepsilon$ for any $j \in [m]$.*

**(ii)** *For $j \in [m]$, the set of critical heavy items in $I_j$ is exactly $H_j^*$.*

**(iii)** *For each $j \in [m]$, if $|I_j| \ge 3$, then*

$$w(I_j) - w(L_j^*) + \widetilde{w}(L_j^*) - w(H_j^*) + \widetilde{w}(H_j^*) \le 1,$$

*where $L_j^*$ is the set of critical light items in $I_j$.*

**(iv)** *For any critical light-expensive item $e_1$, and for any non-critical light-expensive item $e_2$, if $e_1$ and $e_2$ are of the same type, then $e_1 \succ e_2$.*

**Proof.** (See the full version of this paper.) ◀

▶ **Lemma 19.** *Given $H_1^*, \dots, H_m^*$, after guessing for $m^{2^{O(\frac{1}{\varepsilon})}}$ rounds, at least one of the guesses gives $m$ sets $L_1^*, \dots, L_m^*$ of light-expensive items such that there exists a feasible solution $\mathcal{S} = (I_1, \dots, I_m)$ satisfying*

**(i)** *$p(I_j) \ge \frac{2}{3} - 4\varepsilon$ for any $j \in [m]$.*

**(ii)** *For each $j \in [m]$, the set of critical heavy items of $\mathcal{S}$ in $I_j$ is exactly $H_j^*$.*

**(iii)** *For each $j \in [m]$, the set of critical light items of $\mathcal{S}$ in $I_j$ is exactly $L_j^*$.*

**Proof.** We first guess, for each type of light items, how many critical items of this type appear in the solution $\mathcal{S}'$ from Lemma 18. By Observation 17, this can be done in $O\big(n^{(1/\varepsilon)^6}\big)$ rounds. By Lemma 18 iv, selecting the heaviest items with the guessed number in each type exactly gives the set of critical light items in $\mathcal{S}'$. Denote the set by $L^*$.

We now guess the assignment of critical items in $\mathcal{S}'$ by guessing the assignment of their types. Let $\mathcal{R}$ be the set of indices of all distinct types of critical items (including both heavy and light items). By Observation 17 and $|T| = O\big((1/\varepsilon)^2\big)$, we have $|\mathcal{R}| = O\big((1/\varepsilon)^6\big)$. Recall that each knapsack contains at most $1/\varepsilon$ critical items. A *configuration* is a vector $\mathbf{r} = (r_1, \dots, r_{1/\varepsilon}) \in \mathcal{R}^{1/\varepsilon}$, where $r_k$ specifies which rounded type the $k$-th critical item in the knapsack uses (padding with a dummy type if there are fewer than $1/\varepsilon$ critical items). Thus, the number of possible configurations is $N \le |\mathcal{R}|^{1/\varepsilon} \le \big((1/\varepsilon)^6\big)^{1/\varepsilon} = 2^{O(\frac{1}{\varepsilon}\log\frac{1}{\varepsilon})}$.

Then we guess the multiplicities of these configurations. Let $(c_1, \dots, c_N) \in \mathbb{Z}_{\ge 0}^N$ be the count vector, where $c_i$ denotes the number of knapsacks using configuration $i$. We enumerate all possibilities for $(c_1, \dots, c_N)$ satisfying $\sum_{i=1}^N c_i = m$, which has $\binom{m+N-1}{N-1} \le m^{O(N)} = m^{2^{O(1/\varepsilon \log(1/\varepsilon))}}$ possibilities.

For convenience, an item is said to have a rounded type, determined by its label weight and label profit. Note that at least one of the guesses gives the count vector of $\mathcal{S}'$ in Lemma 18. Fixing this guess, we obtain $m$ configurations of $\mathcal{S}'$, say $\mathbf{r}_1, \dots, \mathbf{r}_m$.

We assign the configurations to knapsacks as follows. We compute a perfect matching between the sets $H_1^*, \ldots, H_m^*$ and $\mathbf{r}_1, \ldots, \mathbf{r}_m$. We add an edge between $H_j^*$ and $\mathbf{r}_i$ if either (1) the rounded type of $H_j^*$ appears in $\mathbf{r}_i$, or (2) $H_j^* = \emptyset$ and $\mathbf{r}_i$ contains no heavy-item type. Note that the perfect matching exists given a correct guess of $\mathcal{S}'$.

Now we construct $L_1^*, \ldots, L_m^*$. We relabel $\mathbf{r}_1, \ldots, \mathbf{r}_m$ such that $\mathbf{r}_j$ matches $H_j^*$. For each rounded type in $\mathbf{r}_j$, we select an arbitrary unused original item within this type from the guessed critical-item set $L^*$. This selection is valid, because for each type, the total number of occurrences across $\mathbf{r}_1, \ldots, \mathbf{r}_m$ equals the guessed number in $L^*$. Then let $L_j^*$ be the set of light items selected for $\mathbf{r}_j$.

It remains to show the existence of $\mathcal{S}$ claimed in the Lemma. Given $\mathcal{S}' = (I_1', \ldots, I_m')$, we replace all the critical items in $I_j'$ with the guessed set $L_j^*$ for all $j \in [m]$, while keeping other items unchanged. Let the resulting solution be $\mathcal{S} = (I_1, \ldots, I_m)$. The construction guarantees that Property iii is satisfied. By Lemma 18ii, Property ii holds.

Note that $p(I_j) \geq \widetilde{p}(I_j) = \widetilde{p}(I_j')$, since we construct $L_j^*$ by selecting light items of the same rounded types as in $\mathbf{r}_j$. It remains to bound $\widetilde{p}(I_j')$. For any light item $l$, we have $p(l) - \widetilde{p}(l) \leq \varepsilon^2$ by Definition 15. Lemma 18i guarantees that $p(I_j') \geq \frac{2}{3} - 3\varepsilon$ for all $j \in [m]$. Since each $I_j'$ contains at most $1/\varepsilon$ critical light items, we have $\widetilde{p}(I_j') \geq p(I_j') - \varepsilon^2 \cdot \frac{1}{\varepsilon} \geq \frac{2}{3} - 4\varepsilon$, which completes the proof. ◀

## 4.6 Determining Non-Critical Items

It remains to determine the non-critical items (of $\mathcal{S}$ in Lemma 19) in each knapsack. There are two kinds of non-critical items: the items in $I_j$ with $|I_j| = 2$ and the cheap items in $I_j$ with $|I_j| \geq 3$.

### 4.6.1 Determining the items in $I_j$ with $|I_j| = 2$

In solution $\mathcal{S}$ of Lemma 19, we have $p(I_j) \geq \frac{2}{3} - 4\varepsilon$ for all $j \in [m]$. Recall that the profit of each item does not exceed $\frac{2}{3} - 6\varepsilon$. Therefore, if $|I_j| = 2$, each item has a profit at least $(\frac{2}{3} - 4\varepsilon) - (\frac{2}{3} - 6\varepsilon) > \varepsilon$. We have the following simple observation.

▶ **Observation 20.** *Items in $I_j$ of $\mathcal{S}$ in Lemma 19 with $|I_j| = 2$ are expensive.*

The following lemma determines the expensive items in all knapsacks.

▶ **Lemma 21.** *Given $(H_1^*, \ldots, H_m^*)$ and $(L_1^*, \ldots, L_m^*)$, one can compute sets $(E_j^*)_{j \in [m]}$ in time $O(m^{2.5})$ such that there exists a feasible solution $\mathcal{S} = (I_1, \ldots, I_m)$ satisfying*

**(i)** *$p(I_j) \geq \frac{2}{3} - 4\varepsilon$ for all $j \in [m]$.*

**(ii)** *For each $j \in [m]$, the expensive items in $I_j$ are exactly $E_j^*$.*

**Proof.** We first compute the sets $(E_j^*)_{j \in [m]}$. Let $E$ denote the set of all expensive items and let $NE = I \setminus \bigcup_{j \in [m]} (H_j^* \cup L_j^*)$ denote the set of non-critical expensive items. If $H_j^*$ or $L_j^*$ is non-empty, then let $E_j^* = H_j^* \cup L_j^*$. It remains to determine $E_j^*$ for knapsacks $j$ with $H_j^* = L_j^* = \emptyset$.

We construct an undirected graph $G$ as follows. Each item in $NE$ corresponds to a vertex, and we add an edge $\{u, v\}$ iff

$$w(u) + w(v) \leq 1 \quad \text{and} \quad p(u) + p(v) \geq \tfrac{2}{3} - 4\varepsilon. \tag{10}$$

We then compute a maximum matching $M$ in $G$, which can be done in $O(|E|\sqrt{|NE|}) = O(m^{2.5})$ time [16]. Let $n^*$ denote the number of knapsacks $j$ with $H_j^* = L_j^* = \emptyset$. We select arbitrary $\min\{|M|, n^*\}$ knapsacks $j$ with $H_j^* = L_j^* = \emptyset$, and set $E_j^*$ to be the pair of items corresponding to an edge in $M$; for all other such knapsacks, we set $E_j^* = \emptyset$. It remains to show the existence of the desired solution $\mathcal{S}$.

Let $\mathcal{S}' = (I_1', \ldots, I_m')$ be the solution guaranteed by Lemma 19, and let $Z(\mathcal{S}') = \{j \in [m] : |I_j'| = 2\}$. For each $j \in Z(\mathcal{S}')$, writing $I_j' = \{u_j, v_j\}$, Lemma 19 implies that $p(u_j) + p(v_j) \geq \frac{2}{3} - 4\varepsilon$ and $w(u_j) + w(v_j) \leq 1$. By the definition of $G$, this means $\{u_j, v_j\}$ is an edge of $G$. Moreover, these edges are pairwise vertex-disjoint (since the $I_j$'s are disjoint), and hence they form a matching of size $|Z(\mathcal{S}')|$ in $G$. Therefore, a maximum matching $M$ satisfies $|M| \geq |Z(\mathcal{S}')|$.

We construct the desired solution $\mathcal{S} = (I_1, \ldots, I_m)$ as follows. Given $\mathcal{S}' = (I_1', \ldots, I_m')$, if $H_j^* = L_j^* = \emptyset$ and $E_j^* \neq \emptyset$, then let $I_j = E_j^*$; otherwise, let $I_j = I_j'$.

We verify that $\mathcal{S}$ satisfies Property i. If $I_j = E_j^*$, then by equation (10), $p(E_j^*) \geq \frac{2}{3} - 4\varepsilon$. Otherwise, $I_j = I_j'$ and by Lemma 19, $p(I_j') \geq \frac{2}{3} - 4\varepsilon$.

We verify that $\mathcal{S}$ satisfies Property ii. If $I_j = E_j^*$, then the property holds trivially. Otherwise, either at least one of $H_j^*$ and $L_j^*$ is non-empty, or $E_j^* = \emptyset$. For the former case, by Lemma 19, the expensive items in $I_j'$ are the critical items $H_j^* \cup L_j^*$, which is exactly $E_j^*$ by our construction. For the latter case, it implies that $H_j^* = L_j^* = \emptyset$. Thus, the critical (i.e., expensive) items in $I_j'$ are empty, which is exactly $E_j^*$. ◀

### 4.6.2 Determine the cheap items in $I_j$ with $|I_j| \geq 3$

Let $C = I \setminus \bigcup_{j\in[m]} E_j^*$ denote the set of cheap items. Let the variable $x_{ij} \in \{0, 1\}$ represent whether the cheap item $i$ is packed in knapsack $j$. Relaxing $x_{ij}$ to be fractional, we have the following LP.

$$\begin{aligned}
\max \quad & t \\
s.t. \quad & p(E_j^*) + \sum_{i\in C} p(i)x_{ij} \geq t, \quad \forall j = 1, \ldots, m \\
& w(E_j^*) + \sum_{i\in C} w(i)x_{ij} \leq B, \quad \forall j = 1, \ldots, m \\
& \sum_{j=1}^{m} x_{ij} = 1, \quad \forall i \in C \\
& 0 \leq x_{ij} \leq 1, \quad \forall i \in C, j = 1, \ldots, m.
\end{aligned} \tag{11}$$

Note that Lemma 21 guarantees a feasible solution to the above LP with objective value at least $\frac{2}{3} - 4\varepsilon$. Applying Lemma 22 below, we can obtain an integral feasible solution with objective value at least $\frac{2}{3} - 4\varepsilon - 2\max_{i\in C} p(i)$. Since items in $C$ are cheap, this objective value is at least $\frac{2}{3} - 6\varepsilon$, which completes the proof of Theorem 1.

▶ **Lemma 22.** *Given nonnegative $p(i), w(i), t_j, B_j$ and a fractional solution $x_{ij}^f$ to the following linear system:*

$$\begin{aligned}
& \sum_{i=1}^{n} p(i)x_{ij}^f \geq t_j, \quad \forall j = 1, \ldots, m \\
& \sum_{i=1}^{n} w(i)x_{ij}^f \leq B_j, \quad \forall j = 1, \ldots, m \\
& \sum_{j=1}^{m} x_{ij}^f = 1, \quad \forall i = 1, \cdots, n \\
& 0 \leq x_{ij}^f \leq 1, \quad \forall i = 1, \cdots, n, j = 1, \ldots, m
\end{aligned} \tag{12}$$

*we can compute in polynomial time an integral solution $x^*_{ij}$ to the following:*

$$\begin{aligned}
&\sum_{i=1}^{n} p(i)x^*_{ij} \geq t_j - 2p_{\max}, \quad \forall j = 1, \dots, m \\
&\sum_{i=1}^{n} w(i)x^*_{ij} \leq B_j, \quad \forall j = 1, \dots, m \\
&\sum_{j=1}^{m} x^*_{ij} = 1, \quad \forall i = 1, \cdots, n \\
&x^*_{ij} \in \{0, 1\}, \quad \forall i = 1, \cdots, n, j = 1, \dots, m
\end{aligned} \tag{13}$$

*where* $p_{\max} = \max_i p(i)$.

**Proof.** (See the full version of this paper.) ◀

## 5 An Algorithm for Arbitrary Capacities

See the full version of this paper.

## 6 Hardness Result

We prove Theorem 3 in this section. We will give a reduction from the restricted numerical 3-dimensional matching problem (RN3DM), which is known to be strongly NP-complete [27].

We first describe the numerical 3-dimensional matching problem (N3DM). Given are three multisets of integers $U, V, W$ where $|U| = |V| = |W| = n$ together with an integer $t$, the goal is to find a subset $M \subseteq U \times V \times W$ such that every element in $U \cup V \cup W$ occurs in $M$ exactly once, and for every $(u, v, w) \in M$, $u + v + w = t$.

The restricted numerical 3-dimensional matching problem further requires that $V = W = \{1, 2, \cdots, n\}$ and $U \subseteq [O(n)]$. The following is an equivalent statement.

▶ **Definition 23** (RN3DM). *Given a multiset $U = \{u_1, \dots, u_n\}$ of nonnegative integers and an integer $t$ such that $\sum_{j=1}^{n} u_j + n(n+1) = nt$, does there exist two permutations $v$ and $w$ of $\{1, \dots, n\}$ such that*

$$u_j + v(j) + w(j) = t, \text{ for } j = 1, \dots, n?$$

Now we are ready to prove Theorem 3.

**Proof of Theorem 3.** Given an instance of RN3DM with $U = \{u_1, \dots, u_n\}$ and $t$, we construct an instance of BMKP as follows. There are $2n$ items and $n$ knapsacks. Every item has a profit of 1. For $i = 1, 2, \cdots, n$, there are exactly two items of weight $i$. The capacity of knapsack $j \in [n]$ is $B_j = t - u_j$.

We first show that if the answer to the given RN3DM instance is "yes", then the constructed BMKP instance admits a feasible solution with objective value 2. This follows by observing that we can pack two items into each knapsack $j$, one of weight $v(j)$ and one of weight $w(j)$. The fact that $v$ and $w$ are permutations of $[n]$ guarantees that we have packed exactly two items of weight $i$ for every $i = 1, 2, \cdots, n$.

Next, we show that if the constructed BMKP instance admits a feasible solution of objective value 2, then the answer to the given RN3DM instance is "yes". Since there are $n$ knapsacks and $2n$ items, each of profit 1, it is clear that in this feasible solution all items are packed, and every knapsack contains exactly two items. Moreover, observe that the total weight of all items is exactly $2 \cdot (1 + 2 + \cdots + n) = n(n+1) = nt - \sum_{j=1}^{n} u_j = \sum_{j=1}^{n} B_j$, thus the total weight of the two items in knapsack $j$ is exactly $B_j = t - u_j$.

Now we construct the two permutations $v$ and $w$ for the RN3DM instance based on the assignment of items. Specifically, consider the weights of the two items in knapsack $j$. We will show that it is possible to set one weight as $v(j)$ and the other weight as $w(j)$ such that $v$ and $w$ are both permutations. As we showed above that the total weight of items in knapsack $j$ is exactly $B_j$, $v(j) + w(j) = t - u_j$ follows directly.

It remains to show which of the two item weights in knapsack $j$ should be set to $v(j)$ (and the other is $w(j)$) so that $v$ and $w$ are permutations. We present an algorithm for achieving this. For ease of description, we take a graphical view. We create a graph $G = (Q, E)$ as follows. Each item is represented as a node, so there are in total $2n$ nodes. When we set the weight of an item as $v(j)$ (or $w(j)$), we color the node corresponding to this item red (or blue). We may also abuse the notation by saying we color an item red or blue. The two items in the same knapsack are called partners. We create two types of edges: (i) an edge between the two nodes corresponding to partners, (ii) an edge between two nodes whose corresponding items have the same weight.

Consider graph $G$. The degree of every node is either 1 or 2. Consider an arbitrary node $q \in Q$ of degree 1, and let $(q, q')$ be the edge incident to it. Then the two items corresponding to $q$ and $q'$ are in the same knapsack, and have the same weight. We color $q$ red and $q'$ blue, and then remove the two nodes from graph $G$.

After removing all the degree 1 nodes, let $\bar{G}$ be the residual graph, and $\bar{G}_1, \bar{G}_2, \cdots, \bar{G}_\ell$ be its connected components. Since the degree of every node in each $\bar{G}_j$ is exactly 2, $\bar{G}_j$ is a cycle. Let the cycle be $(q_1, q_2, \cdots, q_k)$. We observe the following.

▶ **Observation 24.** *$k = |\bar{G}_j|$ is even for every $j$.*

**Proof.** Suppose, on the contrary, that $k$ is odd. Without loss of generality, let $q_1, q_2$ be two items of the same weight, then $q_2, q_3$ are partners, and $q_3, q_4$ are two items of the same weight, etc. We get that $q_{k-1}, q_k$ are partners, and $q_k, q_1$ have the same weight. Then $q_1, q_2, q_k$ all have the same weight. Since there are at most 2 items of the same weight, we get $k = 2$, which is a contradiction. ◀

Since $k$ is even, for each cycle $(q_1, q_2, \cdots, q_k)$, we color $q_1, q_3, \cdots, q_{k-1}$ red and $q_2, q_4, \cdots, q_k$ blue.

Now we have colored every node either red or blue; it remains to obtain the permutation $v$ and $w$ from this coloring. According to our coloring, every edge is between a red node and a blue node, which means: (i) Partners (items in the same knapsack) always have different colors. (ii) Items of the same weight always have different colors. Thus, the $n$ red nodes correspond to $n$ items on $n$ knapsacks, and their weights are exactly $1, 2, \cdots, n$. Hence, define $v(j)$ (or $w(j)$) as the weight of red (or blue) node in knapsack $j$, $v(j)$ and $w(j)$ are permutations.

The above argument shows that the answer to the given RN3DM instance is "yes" if and only if the constructed BMKP instance admits a feasible solution with objective value 2.

Observe that, for our constructed BMKP instance, the objective value is at most 2. Recall that the instance contains $2n$ items and $n$ knapsacks, and every item has profit 1. If some knapsack contains at least three items, then the remaining $n-1$ knapsacks together contain at most $2n-3$ items. By the pigeonhole principle, at least one knapsack contains at most one item, and thus the objective value is at most 1. Otherwise, every knapsack contains at most two items, and thus the objective value is at most 2. In either case, the objective value is at most 2.

Now suppose that BMKP admits a polynomial-time $(\frac{1}{2}+\varepsilon)$-approximation algorithm for some constant $\varepsilon > 0$. Given an RN3DM instance, we construct the corresponding BMKP instance and run this algorithm. If the answer to the RN3DM instance is "yes", then $\mathrm{OPT} = 2$ for the BMKP instance and the algorithm returns a solution of value at least $(\frac{1}{2}+\varepsilon)\cdot 2 > 1$, which exactly computes the optimal solution. If the answer is "no", then $\mathrm{OPT} = 1$ and the returned value is at most 1. Thus, the algorithm would allow us to decide RN3DM in polynomial time, completing the proof of Theorem 3. ◀

## 7 Conclusion and Discussion

We studied the bottleneck multiple knapsack problem (BMKP) under both identical-capacity and arbitrary-capacity settings. For identical capacities, we presented a polynomial-time $(\frac{2}{3}-\varepsilon)$-approximation algorithm, which nearly matches the known $(\frac{2}{3}+\varepsilon)$ inapproximability bound [5]. For arbitrary capacities, we established a nearly tight guarantee at $\frac{1}{2}$ by giving a $(\frac{1}{2}-\varepsilon)$-approximation algorithm together with a $(\frac{1}{2}+\varepsilon)$-inapproximability result.

Several intriguing questions remain open. First, although the $\frac{2}{3}$ approximation ratio is known to be tight for the bottleneck multiple subset sum problem with identical capacities, it remains unclear whether a $\frac{2}{3}$-approximation is achievable when the knapsack capacities of the subset sum problem are arbitrary.

Our hardness result in Theorem 3 shows that, for arbitrary capacities, no $(\frac{1}{2}+\varepsilon)$-approximation algorithm exists for any constant $\varepsilon > 0$, unless P = NP, even in the restricted case where all item profits are 1. This naturally raises the question of whether a $\frac{1}{2}$-approximation can be achieved in this unit-profit setting.

Another interesting direction is to study intermediate models between identical and arbitrary capacities. In particular, it is open whether a $(\frac{2}{3}-\varepsilon)$-approximation is achievable when the number of distinct knapsack capacities is bounded by a constant.

## References


1 Yossi Azar and Leah Epstein. Approximation schemes for covering and scheduling in related machines. In *Proceedings of the 1st International Workshop on Approximation Algorithms for Combinatorial Optimization (APPROX)*, volume 1444, pages 39–47. Springer, 1998. doi:10.1007/BFb0053962.

2 Nikhil Bansal and Maxim Sviridenko. The Santa Claus problem. In *Proceedings of the 38th Annual ACM Symposium on Theory of Computing (STOC)*, pages 31–40. ACM, 2006. doi:10.1145/1132516.1132522.

3 Ivona Bezáková and Varsha Dani. Allocating indivisible goods. *ACM SIGecom Exchanges*, 5(3):11–18, 2005. doi:10.1145/1120680.1120683.

4 Arpita Biswas and Siddharth Barman. Fair division under cardinality constraints. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence (IJCAI)*, pages 91–97, 2018. doi:10.24963/ijcai.2018/13.

5 Alberto Caprara, Hans Kellerer, and Ulrich Pferschy. The multiple subset sum problem. *SIAM Journal on Optimization*, 11(2):308–319, 2000. doi:10.1137/S1052623498348481.

6 Alberto Caprara, Hans Kellerer, and Ulrich Pferschy. A PTAS for the multiple subset sum problem with different knapsack capacities. *Information Processing Letters*, 73(3-4):111–118, 2000. doi:10.1016/S0020-0190(00)00010-7.

7 Alberto Caprara, Hans Kellerer, and Ulrich Pferschy. A 3/4-approximation algorithm for multiple subset sum. *Journal of Heuristics*, 9:99–111, 2003. doi:10.1023/A:1022584312032.

8 Deeparnab Chakrabarty, Julia Chuzhoy, and Sanjeev Khanna. On allocating goods to maximize fairness. In *Proceedings of the 50th Annual IEEE Symposium on Foundations of Computer Science (FOCS)*, pages 107–116, 2009. doi:10.1109/FOCS.2009.51.

9 Chandra Chekuri and Sanjeev Khanna. A polynomial time approximation scheme for the multiple knapsack problem. *SIAM Journal on Computing*, 35(3):713–728, 2005. doi:10.1137/S0097539700382820.

10 Wenceslas Fernandez de la Vega and George S. Lueker. Bin packing can be solved within $1+\varepsilon$ in linear time. *Combinatorica*, 1(4):349–355, 1981. doi:10.1007/BF02579456.

11 Bin Deng and Weidong Li. The budgeted maximin share allocation problem. *Optimization Letters*, 19(5):955–968, 2025. doi:10.1007/s11590-024-02145-6.

12 Leah Epstein and Jirí Sgall. Approximation schemes for scheduling on uniformly related and identical parallel machines. *Algorithmica*, 39:43–57, 2004. doi:10.1007/s00453-003-1077-7.

13 Laurent Gourvès and Jérôme Monnot. Approximate maximin share allocations in matroids. In *Proceedings of the International Conference on Algorithms and Complexity (CIAC)*, volume 10236, pages 310–321. Springer, 2017. doi:10.1007/978-3-319-57586-5_26.

14 Laurent Gourvès and Jérôme Monnot. On maximin share allocations in matroids. *Theoretical Computer Science*, 754:50–64, 2019. doi:10.1016/j.tcs.2018.05.018.

15 Dorit S. Hochbaum and David B. Shmoys. Using dual approximation algorithms for scheduling problems theoretical and practical results. *Journal of the ACM (JACM)*, 34(1):144–162, 1987. doi:10.1145/7531.7535.

16 John E. Hopcroft and Richard M. Karp. An $n^{5/2}$ algorithm for maximum matchings in bipartite graphs. *SIAM Journal on Computing*, 2(4):225–231, 1973. doi:10.1137/0202019.

17 Halvard Hummel. Maximin shares in hereditary set systems. *ACM Transactions on Economics and Computation*, 13(3):12:1–12:33, 2025. doi:10.1145/3727149.

18 Halvard Hummel and Magnus Lie Hetland. Maximin shares under cardinality constraints. In *Proceedings of the European Conference on Multi-Agent Systems (EUMAS)*, volume 13442, pages 188–206. Springer, Springer, 2022. doi:10.1007/978-3-031-20614-6_11.

19 Klaus Jansen. Parameterized approximation scheme for the multiple knapsack problem. *SIAM Journal on Computing*, 39(4):1392–1412, 2009. doi:10.1137/080731207.

20 Klaus Jansen. A fast approximation scheme for the multiple knapsack problem. In *Proceedings of the 38th International Conference on Current Trends in Theory and Practice of Computer Science (SOFSEM)*, volume 7147, pages 313–324. Springer, 2012. doi:10.1007/978-3-642-27660-6_26.

21 Hans Kellerer. A polynomial time approximation scheme for the multiple knapsack problem. In *Proceedings of the International Workshop on Randomization and Approximation Techniques in Computer Science (APPROX)*, volume 1671, pages 51–62. Springer, 1999. doi:10.1007/978-3-540-48413-4_6.

22 Sheng-Yen Ko, Ho-Lin Chen, Siu-Wing Cheng, Wing-Kai Hon, and Chung-Shou Liao. Polynomial-time combinatorial algorithm for general max–min fair allocation. *Algorithmica*, 86(2):485–504, 2024. doi:10.1007/s00453-023-01105-3.

23 Jan Karel Lenstra, David B. Shmoys, and Éva Tardos. Approximation algorithms for scheduling unrelated parallel machines. *Mathematical Programming*, 46:259–271, 1990. doi:10.1007/BF01585745.

24 Barna Saha and Aravind Srinivasan. A new approximation technique for resource-allocation problems. *Random Structures & Algorithms*, 52(4):680–715, 2018. doi:10.1002/rsa.20756.

25 David P. Williamson and David B. Shmoys. *The design of approximation algorithms*. Cambridge University Press, 2011. URL: http://www.cambridge.org/de/knowledge/isbn/item5759340/?site_locale=de_DE.

26 Gerhard J. Woeginger. A polynomial-time approximation scheme for maximizing the minimum machine completion time. *Operations Research Letters*, 20(4):149–154, 1997. doi:10.1016/S0167-6377(96)00055-7.

27 Wenci Yu, Han Hoogeveen, and Jan Karel Lenstra. Minimizing makespan in a two-machine flow shop with delays and unit-time operations is np-hard. *Journal of Scheduling*, 7(5):333–348, 2004. doi:10.1023/B:JOSH.0000036858.59787.c2.