

# On the configuration LP for maximum budgeted allocation

Christos Kalaitzis<sup>1</sup> · Aleksander Mądry<sup>3</sup> ·  
Alantha Newman<sup>4</sup> · Lukáš Poláček<sup>2</sup> ·  
Ola Svensson<sup>1</sup>

Received: 30 April 2014 / Accepted: 9 June 2015 / Published online: 4 July 2015  
© Springer-Verlag Berlin Heidelberg and Mathematical Optimization Society 2015

**Abstract** We study the maximum budgeted allocation problem, i.e., the problem of selling a set of  $m$  indivisible goods to  $n$  players, each with a separate budget, such that we maximize the collected revenue. Since the natural assignment LP is known to have an integrality gap of  $\frac{3}{4}$ , which matches the best known approximation algorithms, our main focus is to improve our understanding of the stronger configuration LP relaxation. In this direction, we prove that the integrality gap of the configuration LP is strictly better than  $\frac{3}{4}$ , and provide corresponding polynomial time roundings, in the following restrictions of the problem: (i) the restricted budgeted allocation problem, in which all the players have the same budget and every item has the same value for any player it can be sold to, and (ii) the graph MBA problem, in which an item can be assigned to

---

This research was partially supported by ERC Advanced Investigator Grant 226203 and ERC Starting Grant 335288-OptApprox.

---

✉ Christos Kalaitzis  
christos.kalaitzis@epfl.ch

Aleksander Mądry  
madry@mit.edu

Alantha Newman  
alantha.newman@grenoble-inp.fr

Lukáš Poláček  
polacek@csc.kth.se

Ola Svensson  
ola.svensson@epfl.ch

<sup>1</sup> EPFL, Lausanne, Switzerland

<sup>2</sup> KTH Royal Institute of Technology, Stockholm, Sweden

<sup>3</sup> MIT Cambridge, MA, USA

<sup>4</sup> CNRS-Université Grenoble, Alpes and G-SCOP, Grenoble, France

at most 2 players. Finally, we improve the best known upper bound on the integrality gap for the general case from  $\frac{5}{6}$  to  $2\sqrt{2} - 2 \approx 0.828$  and also prove hardness of approximation results for both cases.

**Mathematics Subject Classification** 68W25

## 1 Introduction

Consider the following scenario: there are multiple players that are willing to pay to gain access to some advertisement resources and each of these players has a specified budget that he cannot exceed. The owner of these resources wants to allocate them so as to maximize his total revenue, i.e., he wishes to maximize the total amount paid by the players. Since no player can pay more than his budget, the task of the owner is to find a revenue-maximizing assignment of resources to players where each player pays the minimum of his budget and his valuation of the items assigned to him.

The above problem is called maximum budgeted allocation (MBA) and arises often in the context of advertisement allocation systems. Formally, a problem instance  $\mathcal{I}$  can be defined as follows: there is a set of players  $\mathcal{A}$  and a set of items  $\mathcal{Q}$ . Each player  $i$  has a budget  $B_i$  and each player  $i \in \mathcal{A}$  has a certain valuation  $p_{ij}$  for every item  $j \in \mathcal{Q}$ . This valuation represents the amount of money player  $i$  would be willing to pay in order to acquire item  $j$ . Since our goal will be to maximize revenue,  $p_{ij}$  will also correspond to the price that we will charge player  $i$  for acquiring item  $j$  (the natural assumption that  $p_{ij} \leq B_i$  is without loss of generality, because no player can spend more money than his budget). Our goal is to find disjoint sets  $S_i \subseteq \mathcal{Q}$  for each player  $i$ , i.e., an indivisible assignment of items to players, so as to maximize

$$\sum_{i \in \mathcal{A}} \min \left\{ \sum_{j \in S_i} p_{ij}, B_i \right\}.$$

In this paper, we are interested in designing good algorithms for the MBA problem and we shall focus on understanding the power of a strong convex relaxation called the configuration LP. The general goal is to obtain a better understanding of basic allocation problems that have a wide range of applications. In particular, the study of the configuration LP is motivated by the belief that a deeper understanding of this type of relaxation can lead to better algorithms for many allocation problems, including MBA, the generalized assignment problem, unrelated machine scheduling and max-min fair allocation.

As the maximum budgeted allocation problem is known to be NP-hard [11, 14], we turn our attention to approximation algorithms. Recall that an  $r$ -approximation algorithm is an efficient (polynomial-time) algorithm that is guaranteed to return a solution within a factor  $r$  of the optimal value. The factor  $r$  is referred to as the approximation ratio or guarantee.

Garg et al. [11] obtained the first approximation algorithm for MBA with a guarantee of  $\frac{2}{1+\sqrt{5}} \approx 0.618$ . This was later improved to  $1 - \frac{1}{e}$  by Andelman and Mansour [1], who also showed that an approximation guarantee of 0.717 can be obtained in the case when

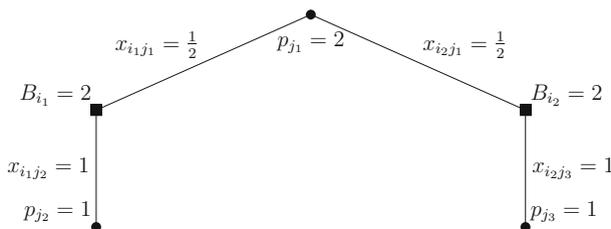
all the budgets are equal. Subsequently, Azar, Birnbaum, Karlin and Mathieu [3] gave a  $\frac{2}{3}$ -approximation algorithm, which Srinivasan [18] extended to give the best-known approximation guarantee of  $\frac{3}{4}$ . Concurrently, the same approximation guarantee was achieved by Chakrabarty and Goel [6], who also proved that it is NP-hard to achieve an approximation ratio better than  $\frac{15}{16}$ .

It is interesting to note that the progress on MBA has several points in common with other basic allocation problems. First, it is observed that when the prices are relatively small compared to the budgets, the problem becomes substantially easier (e.g. [6, 18]), analogous to how unrelated machine scheduling becomes easier when the largest processing time is small compared to the optimal makespan. Second, the above mentioned  $3/4$ -approximation algorithms give a tight analysis of a standard LP relaxation, called the assignment LP, which has been a successful tool for allocation problems ever since the breakthrough work by Lenstra et al. [15]. Indeed, we now have a complete understanding of the strength of the assignment LP for all of the above mentioned allocation problems. The strength of a relaxation is measured by its integrality gap, which is the maximum ratio between the solution quality of the exact integer programming formulation and of its relaxation.

A natural approach for obtaining better (approximation) algorithms for allocation problems is to use stronger relaxations than the assignment LP. Similarly to other allocation problems, there is a strong belief that a strong convex relaxation called the configuration LP relaxation could give strong guarantees for the MBA problem. Even though we only know that the integrality gap is no better than  $\frac{5}{6}$  [6], our current techniques fail to prove that the configuration LP gives even marginally better guarantees for MBA than the assignment LP. The goal of this paper is to increase our understanding of the configuration LP and to shed light on its strength.

*Our contributions.* To analyze the strength of the configuration LP compared to the assignment LP, it is instructive to consider the tight integrality gap instance for the assignment LP from [6] depicted in Fig. 1. This instance satisfies several structural properties: (i) at most two players have a positive price of an item, (ii) every player has the same budget (also known as *uniform budgets*), (iii) the price of an item  $j$  for a player is either  $p_j$  or 0, i.e.,  $p_{ij} \in \{0, p_j\}$ .

Motivated by these observations and previous work on allocation problems, we shall mainly concentrate on two special cases of MBA. The first case is obtained by



**Fig. 1** Here, circles correspond to items and boxes correspond to players. The budget of every player is 2. The solution  $x$  has value of 4. Any integral solution has a value of at most 3, since one player will be assigned only one item of value 1

enforcing (i) in which at most two players have a positive price of an item. We call this case *graph MBA*, as an instance can be represented by a graph where items are edges, players are vertices and assigning an item corresponds to orienting an edge. The same restriction, where it is often called graph balancing, has led to several nice results for unrelated machine scheduling [7] and max-min fair allocation [5, 22].

The second case is obtained by enforcing (ii) and (iii). That is, each item  $j$  has a non-zero price, denoted by  $p_j$ , for a subset of players, and the players have uniform budgets. We call this case *restricted MBA* or the restricted budgeted allocation Problem as it closely resembles the restricted assignment problem that has been a popular special case of both unrelated machine scheduling [19] and max-min fair allocation [2, 4, 8]. These two properties produce natural restrictions whose study helps increase our understanding of the general problem, as one can realize by going through [6, 18], and specifically, instances displaying property (ii) have been studied in [1].

Our main result proves that the configuration LP is indeed stronger than the assignment LP for the considered problems.

**Theorem 1** *Restricted MBA and graph MBA have  $(\frac{3+c}{4})$ -approximation algorithms that also bound the integrality gap of the configuration LP, for some constant  $c > 0$ .*

The result for graph MBA is inspired by the work by Feige and Vondrak [9] on the generalized assignment problem and is presented in Sect. 4. The main idea is to first recover a  $3/4$ -fraction of the configuration LP value by randomly (according to the LP solution) assigning items to the players. The improvement over  $3/4$  is then obtained by further assigning some of the items that were left unassigned to players whose budgets are not already exceeded. The difficulty in the above approach lies in analyzing the contribution of the items assigned in the second step over the random assignment in the first step (Lemma 10).

For restricted MBA, we need a different approach. Indeed, randomly assigning items according to the configuration LP only recovers a  $(1 - 1/e)$ -fraction of the LP value when an item can be assigned to any number of players. Current techniques only gain an additional small  $\epsilon$ -fraction by assigning unassigned items in the second step. This would lead to an approximation guarantee of  $(1 - 1/e + \epsilon)$  (matching the result in [9] for the generalized assignment Problem) which is strictly less than the best known approximation guarantee of  $3/4$  for MBA. We therefore take a different approach. We first observe that an existing algorithm, described in Sect. 3.1, already gives a guarantee better than  $3/4$  for configuration LP solutions that are not well-structured (see Definition 1). Informally, an LP solution is well-structured if half the budgets of most players are assigned to expensive items, which are defined as those items whose price is very close to the budget. For the rounding of well-structured solutions in Sect. 3.4, the main new idea is to first assign expensive/big items (of value close to the budgets) using random bipartite matching and then assign the cheap/small items in the space left after the assignment of expensive items. For this to work, it is not sufficient to assign the big items in any way that preserves the marginals from the LP relaxation. Instead, a key observation is that we can assign big items so that the probability that two players  $i, i'$  are both assigned big items is at most the probability that player  $i$  is assigned a big item times the probability that player  $i'$  is assigned a big item (i.e., the events are negatively correlated). Using this we can show that we

can assign many of the small items even after assigning the big items leading to the improved guarantee. We believe that this is an interesting use of bipartite matchings for allocation problems as we are using the fact that the events that vertices are matched can be made negatively correlated. Note that this is in contrast to the events that edges are part of a random matching which are not necessarily negatively correlated.

Finally, we complement our positive results by hardness results and integrality gaps. For restricted MBA, we prove hardness of approximation that matches the strongest results known for the general case. Specifically, we prove in Sect. 6 that it is NP-hard to approximate restricted MBA within a factor  $15/16$ . This shows that some of the hardest known instances for the general problem are the ones we study. We also improve the  $5/6$  integrality gap of the configuration LP for the general case: we prove that it is not better than  $2(\sqrt{2} - 1) \approx 0.828$  in Sect. 5.

## 2 Preliminaries

*Assignment LP.* The assignment LP for the MBA problem has a fractional “indicator” variable  $x_{ij}$  for each player  $i \in \mathcal{A}$  and each item  $j \in \mathcal{Q}$  that indicates whether item  $j$  is assigned to player  $i$ . Recall that the profit received from a player  $i$  is the minimum of his budget  $B_i$  and the total value  $\sum_{j \in \mathcal{Q}} x_{ij} p_{ij}$  of the items assigned to  $i$ . In order to avoid taking the minimum for each player, we impose that each player  $i$  is fractionally assigned items of total value at most his budget  $B_i$ . Note that this is not a valid constraint for an integral solution but it does not change the value of a fractional solution: we can always fractionally decrease the assignment of an item without changing the objective value if the total value of the fractional assignment exceeds the budget. To further simplify the relaxation, we enforce that all items are fully assigned by adding a dummy player  $\ell$  such that  $p_{\ell j} = 0$  for all  $j \in \mathcal{Q}$  and  $B_\ell = 0$ . The assignment LP for MBA is defined as follows:

$$\begin{aligned} & \max && \sum_{i \in \mathcal{A}} \sum_{j \in \mathcal{Q}} x_{ij} p_{ij} \\ \text{subject to} &&& \sum_{j \in \mathcal{Q}} x_{ij} p_{ij} \leq B_i \quad \forall i \in \mathcal{A} \\ &&& \sum_{i \in \mathcal{A}} x_{ij} = 1 \quad \forall j \in \mathcal{Q} \\ &&& 0 \leq x_{ij} \leq 1 \quad \forall i \in \mathcal{A}, \forall j \in \mathcal{Q} \end{aligned}$$

As discussed in the introduction, it is known that the integrality gap of the assignment LP is exactly  $\frac{3}{4}$ ; therefore, in order to achieve a better approximation, we employ a stronger relaxation called the configuration LP.

*Configuration LP.* The intuition behind the configuration LP comes from observing that, in an integral solution, the players are assigned disjoint sets, or configurations, of items. The configuration LP models this by having a fractional “indicator” variable  $y_{iC}$  for each player  $i$  and configuration  $C \subseteq \mathcal{Q}$ , which indicates whether or not  $C$  is the set of items assigned to player  $i$  in the solution. The constraints of the configuration LP require that each player is assigned at most one configuration and each item is assigned to at most one player. If we let  $w_i(C) = \min \left\{ \sum_{j \in C} p_{ij}, B_i \right\}$  denote the

total value of the set  $C$  of items when assigned to player  $i$ , the configuration LP can be formulated as follows:

$$\begin{aligned} & \max \quad \sum_{i \in \mathcal{A}} \sum_{C \subseteq \mathcal{Q}} w_i(C) y_{iC} \\ \text{subject to} \quad & \sum_{C \subseteq \mathcal{Q}} y_{iC} \leq 1 \quad \forall i \in \mathcal{A} \\ & \sum_{i \in \mathcal{A}, C \subseteq \mathcal{Q}: j \in C} y_{iC} \leq 1 \quad \forall j \in \mathcal{Q} \\ & y_{iC} \geq 0 \quad \forall i \in \mathcal{A}, \forall C \subseteq \mathcal{Q} \end{aligned}$$

We remark that even though the relaxation has exponentially many variables, it can be solved approximately in a fairly standard way by designing an efficient separation oracle for the dual which has polynomially many variables. We refer the reader to [6] for more details.

The configuration LP is stronger than the assignment LP as it enforces a stricter structure on the fractional solution. Indeed, every solution to the configuration LP can be transformed into a solution of the assignment LP of at least the same value (see e.g. Lemma 2). However, the converse is not true; configuration LP has value 3 for the example shown in Fig. 1 while assignment LP has value 4, the key difference being that the fractional assignments of the players cannot induce a fractional assignment of configurations of the same value. More convincingly, our results show that the configuration LP has a strictly better integrality gap than the assignment LP for large natural classes of the MBA problem.

For a solution  $y$  to the configuration LP, let  $\text{Val}_i(y) = \sum_C w_i(C) y_{iC}$  be the value of the fractional assignment to player  $i$  and let  $\text{Val}(y) = \sum_i \text{Val}_i(y)$ . Note that  $\text{Val}(y)$  is equal to the objective value of the solution  $y$ . Abusing notation, we also define  $\text{Val}_i(x) = \sum_j x_{ij} p_{ij}$  for a solution  $x$  to the assignment LP.

*Random bipartite matching.* As alluded to in the introduction, one of the key ideas of our algorithm for the restricted case is to first assign expensive/big items (of value close to the budgets) by picking a random bipartite matching so that the events that vertices are matched are negatively correlated. The following theorem is a minor modification of a theorem proved by Gandhi, Khuller, Parthasarathy and Srinivasan in their work on selecting random bipartite matchings with particular properties [10]; their proof directly gives the following:

**Theorem 2** ([10]) *Consider a bipartite graph  $G = ((A, B), E)$  and an assignment  $(x_e)_{e \in E}$  to edges so that the fractional degree  $\sum_{u:uv \in E} x_{uv}$  of each vertex  $v$  is at most 1. Then there is an efficient, randomized algorithm that generates a (random) matching satisfying:*

(P1): **Marginal Distribution** *For every vertex  $v \in A \cup B$ ,*

$$\Pr[v \text{ is matched}] = \sum_{u:uv \in E} x_{uv}$$

(P2): **Negative Correlation** *For any  $S \subseteq A$ ,*

$$\Pr \left[ \bigwedge_{v \in S} (v \text{ is matched}) \right] \leq \prod_{v \in S} \Pr[v \text{ is matched}].$$

One should note that the events {edge  $e$  is in the matching} and {edge  $e'$  is in the matching} are not necessarily negatively correlated (if we preserve the marginals). A crucial ingredient for our algorithm is therefore the idea that we can concentrate on the event that a player has been assigned a big item without regard to the specific item assigned.

### 3 Restricted budget allocation

In this section we consider the MBA problem with uniform budgets where the prices are restricted to be of the form  $p_{ij} \in \{p_j, 0\}$ . This is the so called *restricted maximum budgeted allocation*. Our main result is the following.

**Theorem 3** *There is a  $(3/4 + c)$ -approximation algorithm for restricted MBA for some constant  $c > 0$ .*

Since the budgets are uniform, by scaling we can assume that each player has a budget of 1. We refer to  $p_j$  as the price of item  $j$ . It is convenient to distinguish whether items have big or small prices, so we let  $\mathcal{B} = \{j : p_j \geq 1 - \beta\}$  be the big items for some  $\beta$ ,  $1/3 \geq \beta > 0$  to be determined. Let  $\mathcal{S}$  denote the set of the remaining items with small prices, which we also refer to as small items.

We first introduce a  $3/4$ -approximation algorithm for the general MBA problem; this algorithm is the starting point of our approach. In our analysis of that algorithm, we identify a family of LP solutions, for which our analysis is tight. These solutions, which we call *well-structured solutions*, play a key role in our approach. Based on their structure, we develop a new algorithm, which achieves a good guarantee exactly for them. Therefore, the distinction between two families of LP solutions, and the design of an algorithm that performs well for each one of them, will be the basis of our approach.

#### 3.1 General $3/4$ -approximation algorithm

We introduce an algorithm (inspired by [17]) to round assignment LP solutions and then we present an analysis showing that it is a  $3/4$ -approximation algorithm. In the rest of Sect. 3 we use this analysis to show that the algorithm has a better approximation ratio than  $3/4$  in some cases.

Each player has a set of buckets. Algorithm 1 first partitions  $x$  into the buckets, creating a new assignment  $x'$ , such that the sum of  $x'$  in each bucket is exactly 1 except possibly the last bucket of each player. Some items are split into two buckets. The process for one player is illustrated in Fig. 2.

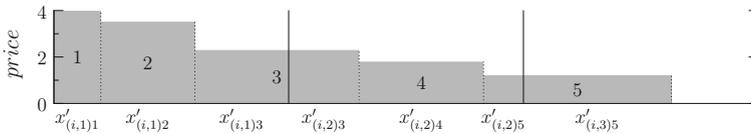
From the previous discussion, for every bucket  $b = (i, k)$  we have  $\sum_j x'_{bj} \leq 1$ . Also,  $\sum_{b \in U} x'_{bj} = 1$  for every item  $j$ , which is implied by  $\sum_{i \in \mathcal{A}} x_{ij} = 1$  for all  $j \in \mathcal{Q}$ . Hence  $x'$  is inside the bipartite matching polytope, where the two sides of the bipartite graph are the buckets  $U$  and the items  $\mathcal{Q}$ , and has the extra property that for every item  $j$ ,  $\sum_{b \in U} x'_{bj} = 1$ . Using an algorithmic version of Carathéodory's theorem (see e.g. Theorem 6.5.11 in [12]) we can in polynomial time decompose  $x'$

```

Input : Solution  $x$  to the assignment LP, ordering  $o_i$  of the items by prices for player  $i$ 
Output: Assignment  $x^*$  of items to the players

foreach  $i \in \mathcal{A}$  do
  // Create buckets for player  $i$ , see Fig. 2
   $c_i \leftarrow \lceil \sum_j x_{ij} \rceil$ 
  Create  $c_i$  buckets  $(i, 1), \dots, (i, c_i)$ 
   $k \leftarrow 1$ 
  Create  $x'_{(i,\cdot)}$  from  $x_i$  as follows:
  foreach  $j \in \mathcal{Q}$  do
    // The items are examined in non-increasing order of  $p_{ij}$ 
    if  $x_{ij} + \sum_{j' \neq j} x_{(i,k)j'} \leq 1$  then
       $x_{(i,k)j} \leftarrow x_{ij}$ 
    else
       $x_{(i,k)j} \leftarrow 1 - \sum_{j' \neq j} x_{(i,k)j'}$ 
       $k \leftarrow k + 1$ 
       $x_{(i,k)j} \leftarrow x_{ij} - x_{(i,k-1)j}$ 
    end if
  end foreach
  end foreach
   $U \leftarrow \{(i, k) \mid 1 \leq k \leq \lceil \sum_j x_{ij} \rceil\}$ 
   $V \leftarrow \mathcal{Q}$ 
  Express  $x'$  as a convex combination of matchings :  $x' = \sum_t \gamma_t m_t$ , where  $\gamma_t$  is the coefficient of
  matching  $m_t$  when expressing  $x'$  as a convex combination of matchings.
  Sample a matching  $m_t$  with probability  $\gamma_t$ , and return the corresponding assignment  $x^*$  of items, i.e.,
  assign to player  $i$  all items assigned to buckets  $(i, k)$  for all  $1 \leq k \leq c_i$ .
  
```

**Algorithm 1:** Bucket algorithm



**Fig. 2** Illustration of bucket creation by Algorithm 1 for player  $i$ . Buckets are marked by *solid lines*. The value  $x_{i3}$  is split into  $x'_{(i,1)3}$  and  $x'_{(i,2)3}$  and  $x_{i5}$  is split into  $x'_{(i,2)5}$  and  $x'_{(i,3)5}$ . For the other items we have  $x'_{(i,1)1} = x_{i1}$ ,  $x'_{(i,1)2} = x_{i2}$ ,  $x'_{(i,2)4} = x_{i4}$ . Items are ordered in non-decreasing order by their prices, in such a way that the length of an item  $j$  in the  $x$ -axis corresponds to  $x_{ij}$  and the  $y$ -axis corresponds to  $p_{ij}$

into a convex combination of polynomially many matchings for  $V$ , such that every such matching matches all items to a bucket.

In the algorithm we use an ordering  $o_i$  for player  $i$  such that  $p_{i o_{ih}} \geq p_{i o_{i,h+1}}$ , i.e., descending order of items according to their prices for player  $i$ . Thus, the algorithm does not depend on the actual prices, but rather the order they induce on the items. Also note that Algorithm 1 can be made deterministic: instead of picking a random matching we can pick the most valuable one.

Let  $\text{Alg}_i(x)$  be the expected price that player  $i$  pays. We know that  $\text{Alg}_i(x) \leq \text{Val}_i(x)$ , because the probability of assigning  $j$  to  $i$  is  $x_{ij}$ , but we do not have equality

in the expression, because some matchings might assign a price that is over the budget for some players.

Given a solution  $x$  to the assignment LP, we define  $\alpha_i = B_i/\text{Val}_i(x)$ . Let  $a_{(i,k)}$  be the average price of items in the bucket  $(i, k)$  of player  $i$ , i.e.,  $a_{(i,k)} = \sum_j x'_{(i,k)j} p_{ij}$ . Let  $r_{(i,k)}$  be the average price of items in  $(i, k)$  that have price more than  $\alpha_i a_{(i,k)}$ , that is  $r_{(i,k)} = \frac{\sum_{j:p_{ij} \geq \alpha_i a_{(i,k)}} x'_{(i,k)j} p_{ij}}{\sum_{j:p_{ij} \geq \alpha_i a_{(i,k)}} x'_{(i,k)j}}$ . In the following lemma we bound  $\text{Alg}_i(x)$  from below.

**Lemma 1** *Let  $x$  be a solution to the assignment LP. For any  $i \in \mathcal{A}$*

$$\text{Alg}_i(x) \geq \text{Val}_i(x) \left( 1 - \frac{r_{(i,1)}}{4B_i} \right).$$

*In particular, since  $r_{(i,1)} \leq B_i$ , this implies that Algorithm 1 gives a 3/4-approximation.*

*Proof* The expected total price of the items of the rounded integral solution  $x^*$  for player  $i$  is  $\mathbb{E}[\text{Val}_i(x^*)] = \text{Val}_i(x)$ , because the probability of assigning item  $j$  to player  $i$  is  $x_{ij}$ . The problem with the assignment  $x^*$  is that it may exceed the budget for some players, so we cannot make use of the full value  $\text{Val}_i(x^*)$ . We now prove that we only lose  $\frac{r_{(i,1)}}{4B_i}$ -fraction of  $\text{Val}_i(x)$  by going over the budget of player  $i$ .

Note that  $\alpha_i \geq 1$ , since, as addressed in Sect. 2, a solution to the assignment LP never exceeds a player’s budget  $B_i$ . Furthermore, the matching picks at most one item from each bucket. Let  $b = (i, k)$  be the  $k$ -th bucket of player  $i$ . Suppose the algorithm picks an item of price more than  $\alpha_i \cdot a_b$  from bucket  $b$ . Then since  $\alpha_i \sum_{\ell} a_{(i,\ell)} = B_i$ , the player could possibly be assigned more than her budget. However, if we assume that all items within each bucket  $(i, \ell)$  have price at most  $\alpha_i \cdot a_{(i,\ell)}$ , all matchings assign total price at most  $B_i$  to player  $i$ . We therefore define  $p'_{(i,\ell)j}$  to be  $\min\{p_{(i,\ell)j}, \alpha_i \cdot a_{(i,\ell)}\}$  for each  $i, \ell, j$  and we get a new instance  $\mathcal{J}$  in which the set of prices are denoted by  $p'$  and the set of players correspond to the buckets from the original instance. In the new instance, a player corresponding to bucket  $(i, \ell)$  has budget  $\alpha_i \cdot a_{(i,\ell)}$ . We abuse the notation and use  $\text{Val}_i^{\mathcal{J}}(\cdot)$  to refer to the total value of buckets belonging to player  $i$  in instance  $\mathcal{J}$ .

From the previous discussion we know no player goes over budget when using the fictional prices  $p'$ , so  $\text{Val}_i^{\mathcal{J}}(x^*) \leq B_i$ . Thus  $\text{Alg}_i(x)$  can be lower-bounded by  $\mathbb{E}[\text{Val}_i^{\mathcal{J}}(x^*)] = \text{Val}_i^{\mathcal{J}}(x')$ . We now prove that  $\text{Val}_i^{\mathcal{J}}(x') \geq \left(1 - \frac{r_{(i,1)}}{4B_i}\right) \text{Val}_i(x)$ .

Let  $r_b$  be the average price of items in bucket  $b = (i, k)$  with price above  $\alpha_i \cdot a_b$  and let  $q_b$  be the average price of the items in bucket  $b$  with price less than  $\alpha_i \cdot a_b$ . Note that  $q_b = q_{(i,k)} \geq r_{(i,k+1)}$ . We let  $x'_b$  be the probability corresponding to  $r_b$ , i.e. the sum of all  $x'_{bj}$  where  $p_{bj} > \alpha_i \cdot a_b$ . Since  $p_{ij}$  is changed to  $\alpha_i \cdot a_b$  in  $p'$  for these items, the difference in price for bucket  $b$  is  $(r_b - \alpha_i a_b)x'_b$ . The loss of value by going from  $p$  to  $p'$  is thus

$$\text{Val}_i(x) - \text{Val}_i^{\mathcal{J}}(x') = \sum_{\ell} (r_{(i,\ell)} - \alpha_i a_{(i,\ell)}) x'_{(i,\ell)}.$$

Since  $a_b$ , i.e., the average price of items in bucket  $b$ , is equal to  $(r_b - q_b)x'_b + q_b$ , we have  $(r_b - \alpha_i a_b)x'_b = (r_b - \alpha_i q_b)x'_b - (r_b - q_b)\alpha_i x'^2_b \leq (r_b - q_b)(x'_b - \alpha_i x'^2_b)$  where the last inequality follows from  $\alpha_i \geq 1$ . Let  $b' = (i, k + 1)$ , i.e. the next bucket after  $b$ . Since  $q_b \geq r_{b'}$ , we get  $(r_b - \alpha_i a_b)x'_b \leq (r_b - q_b)(x'_b - \alpha_i x'^2_b) \leq (r_b - r_{b'})(x'_b - \alpha_i x'^2_b)$ . Since  $b = (i, k)$  and  $b' = (i, k + 1)$ , it follows that

$$\text{Val}_i(x) - \text{Val}_i^{\mathcal{J}}(x') \leq \sum_{\ell} (r_{(i,\ell)} - r_{(i,\ell+1)}) \left( x'_{(i,\ell)} - \alpha_i x'^2_{(i,\ell)} \right). \tag{1}$$

The maximum of  $x'_{(i,\ell)} - \alpha_i x'^2_{(i,\ell)}$  is attained for  $x'_{(i,\ell)} = \frac{1}{2\alpha_i}$  and we get

$$\text{Val}_i(x) - \text{Val}_i^{\mathcal{J}}(x') \leq \sum_{\ell} \frac{r_{(i,\ell)} - r_{(i,\ell+1)}}{4\alpha_i} \leq \frac{r_{(i,1)}}{4\alpha_i}$$

Hence

$$\text{Alg}_i(x) \geq \text{Val}_i^{\mathcal{J}}(x') \geq \text{Val}_i(x) - \frac{r_{(i,1)}}{4\alpha_i} = \text{Val}_i(x) \left( 1 - \frac{r_{(i,1)}}{4B_i} \right) \tag{2}$$

Since  $r_{(i,1)} \leq B_i$ , Algorithm 1 is a 3/4-approximation algorithm. □

The lower bounding in the previous analysis has some slack, that we list in the following remarks:

*Remark 1* Whenever  $x'_{(i,1)}$  is bounded away from  $\frac{1}{2\alpha_i}$ , then from Inequality (1) we see that the algorithm achieves a ratio better than 3/4. This is the opposite of the notion of *well-structured solutions* that we introduce later, which is a central concept of our approach.

*Remark 2* If  $r_{(i,1)}$  is bounded away from  $B_i$ , which means that most items do not have a very high price for player  $i$ , then, as we deduce by examining Inequality (2), an approximation ratio greater than 3/4 is achieved.

These observations are the motivation behind the use of well-structured solutions, that we introduce in the next section.

### 3.2 Well-structured solutions

Let us describe the concept of well-structured solutions. In short, a solution  $y$  to the configuration LP is  $(\epsilon, \delta)$ -well-structured, if for at least  $(1 - \epsilon)$ -fraction of players roughly half of their configurations contain a big item.

**Definition 1** A solution  $y$  to the configuration LP is  $(\epsilon, \delta)$ -well-structured if

$$\Pr_i \left[ \sum_{\mathcal{C} \subseteq \mathcal{Q}} |\mathcal{B} \cap \mathcal{C}| \cdot y_{i\mathcal{C}} \notin \left[ \frac{1 - \delta}{2}, \frac{1 + \delta}{2} \right] \right] \leq \epsilon,$$

where the probability is taken over a weighted distribution of players such that player  $i$  is chosen with probability  $\text{Val}_i(y)/\text{Val}(y)$ .

It is instructive to notice that this definition addresses exactly the structural properties of the LP solutions for which Algorithm 1 might perform poorly, as discussed in Remarks 1 and 2. Specifically, the LP solutions that make the analysis of Algorithm 1 tight look very much like well-structured solutions; being tight with respect to Remark 2 means that the LP solution contains a fair amount of big items for player  $i$ , while being tight with respect to Remark 1 means that the total fractional assignment of big items should be around  $\frac{1}{2\alpha_i}$ .

We want to be able to switch from the configuration LP to the assignment LP without changing the well-structuredness of the solution. The following lemma shows that this is indeed possible.

**Lemma 2** *Let  $y$  be a well-structured solution to the configuration LP. Then there exists a solution  $x$  to the assignment LP such that  $\text{Val}_i(x) = \text{Val}_i(y)$  and*

$$\sum_{\mathcal{C} \subseteq \mathcal{Q}} |\mathcal{B} \cap \mathcal{C}| \cdot y_{i\mathcal{C}} \in \left[ \frac{1 - \delta}{2}, \frac{1 + \delta}{2} \right] \Leftrightarrow \sum_{j: j \in \mathcal{B}} x_{ij} \in \left[ \frac{1 - \delta}{2}, \frac{1 + \delta}{2} \right]$$

for all  $i \in \mathcal{A}$ . Furthermore,  $x$  can be produced from  $y$  in polynomial time.

*Proof* Note that we can assume that each configuration in  $y$  contains at most 2 big items. If a configuration contains more than 2 big items, all but 2 big items can be removed without decreasing the objective value. This follows from the fact that, in that case, the configuration has value at least  $2(1 - \beta)$ , which is greater than 1 since  $\beta \leq 1/3$ .

We first modify  $y$  to obtain a new solution  $y'$  where for each player, the following holds: the player does not have both a configuration containing 2 big items and a configuration with no big items. Then  $y'$  is projected to a solution  $x$  for the assignment LP with the desired properties.

Fix player  $i$  and two configurations  $\mathcal{C}$  and  $\mathcal{C}'$  such that  $\mathcal{C}$  contains two big items and  $\mathcal{C}'$  contains no big items. We can assume that  $y_{i\mathcal{C}} = y_{i\mathcal{C}'}$ , otherwise we can split the bigger fractional value into two and focus on just one of them. We want to move the second big item from  $\mathcal{C}$  into  $\mathcal{C}'$  without decreasing the objective value. We ensure that the objective value does not decrease by moving small items from  $\mathcal{C}'$  to  $\mathcal{C}$ .

Let us order the small items in  $\mathcal{C}'$  by value in decreasing order  $j_1, \dots, j_k$ . We move a big item from  $\mathcal{C}$  to  $\mathcal{C}'$  and then move  $j_1, j_2, \dots$  to  $\mathcal{C}$  until it has value at least 1 or we run out of items. Let  $F$  and  $F'$  be the transformed configurations which arise from  $\mathcal{C}$  and  $\mathcal{C}'$ , respectively.

If  $w(F) < 1$ , then we moved all items from  $\mathcal{C}'$  to  $F$ , so  $w(F) \geq 1 - \beta + w(\mathcal{C}')$  and  $\beta > w(\mathcal{C}')$ .  $F'$  contains one big item, so  $w(F') \geq 1 - \beta$ . Then  $w(F) + w(F') \geq 2 - 2\beta + w(\mathcal{C}') > w(\mathcal{C}) + w(\mathcal{C}')$ . This implies that the objective value improved by moving the items and  $y$  is not an optimal solution, a contradiction. Hence we can safely assume that  $w(F) \geq 1$ .

It remains to prove that  $w(F') \geq w(\mathcal{C}')$ , so that the objective value does not decrease with the move. In the process above we moved a big item of value at least  $1 - \beta$  to  $\mathcal{C}'$  and

we now show that we moved less in the other direction. Suppose towards contradiction that we moved items of value at least  $1 - \beta \geq 2\beta$  to  $\mathcal{C}$ . Since  $\mathcal{C}$  has one big item, it needs at most value of  $\beta$  from  $\mathcal{C}'$ . If we only moved one small item, then this cannot be true, since a small item is of value less than  $1 - \beta$ . Hence we moved  $s \geq 2$  small items to  $\mathcal{C}$ . The last item added was of value at least  $2\beta/s$ . By  $s \geq 2$ , moving only the  $s - 1$  most valuable items would suffice, since  $\frac{s-1}{s}2\beta \geq \beta$  and  $F$  would reach price 1 already with the  $s - 1$  most valuable items from  $\mathcal{C}'$ .

Applying this procedure whenever we can, we end up with a modified solution  $y'$  to the configuration LP in which there is no player at the same time having a configuration containing two big items and a configuration with no big items. Also,  $y'$  is such that

$$\sum_{\mathcal{C} \subseteq \mathcal{Q}} |\mathcal{B} \cap \mathcal{C}| \cdot y_{i\mathcal{C}} = \sum_{\mathcal{C} \subseteq \mathcal{Q}} |\mathcal{B} \cap \mathcal{C}| \cdot y'_{i\mathcal{C}},$$

because big items are only moved between configurations, so their contribution to the sums above is preserved.

The pairs of configurations  $\mathcal{C}, \mathcal{C}'$  can be chosen in such a way that we only create polynomially many new configurations in total. To see this, let  $T = \{\mathcal{C}_1, \dots, \mathcal{C}_k\}$  be the configurations in  $y$  that have two big items and let  $S = \{\mathcal{C}'_1, \dots, \mathcal{C}'_\ell\}$  be the configurations in  $y$  with no big items. We process  $T$  one by one. For each  $\mathcal{C}_j \in T$  we try to move the second big item to the configurations  $\mathcal{C}'_1, \dots, \mathcal{C}'_\ell$  one by one. In the end we try at most  $k \cdot \ell$  different pairs and each pair creates at most 2 new configurations. Since  $k \cdot \ell$  is polynomial in the size of the instance, also the number of new configurations is polynomial.

In the final step we project  $y'$  to a solution  $x$  to the assignment LP as follows: for every player  $i$  and configuration  $\mathcal{C}$ , consider the items in  $\mathcal{C}$  in non-increasing order according to  $p_{ij}$ . If the total value of  $\mathcal{C}$  is at most 1, the configuration  $\mathcal{C}$  contributes  $y'_{i\mathcal{C}}$  to the value  $x_{ij}$  for all items  $j \in \mathcal{C}$ . Otherwise, consider the items in  $\mathcal{C}$  in non-increasing order with respect to their price, and let  $t$  be the minimum number such that the sum of the values of the first  $t$  items is greater than the budget. Then, the first  $t - 1$  items  $j$  will have a value of  $y'_{i\mathcal{C}}$  contributed towards their assignment  $x_{ij}$ , the  $t$ -th item will have some fraction of  $y'_{i\mathcal{C}}$  contributed towards its assignment, and the rest of the items will not receive any contribution towards their assignment. In particular this means that big items get the full contribution  $y'_{i\mathcal{C}}$ .

Let us now formalize the intuition given above. Let  $j'$  be the minimum index such that  $\sum_{1 \leq j \leq j'} p_{ij} > 1$ , where the items are now ordered in decreasing order according to their prices for player  $i$ . For all  $j > j'$ , set  $z_{ij\mathcal{C}} = 0$ , for all  $j < j'$  set  $z_{ij\mathcal{C}} = y'_{i\mathcal{C}}$  and

$$z_{ij'\mathcal{C}} = \frac{1 - \sum_{1 \leq j < j'} p_{ij}}{p_{ij'}} y'_{i\mathcal{C}}.$$

Finally, we define

$$x_{ij} = \sum_{\mathcal{C} \subseteq \mathcal{Q}} z_{ij\mathcal{C}}.$$

We have  $\text{Val}(x) = \text{Val}(y') = \text{Val}(y)$ , since  $\sum_j z_{ij} p_{ij} = w_i(\mathcal{C})y'_{i\mathcal{C}}$ , i.e. the contribution of each configuration is preserved by the projection.

The projection from  $y'$  to  $x$  gives full contribution  $y'_{i\mathcal{C}}$  to the largest item in  $\mathcal{C}$ . If there are two big items in  $\mathcal{C}$ , the second big item does not get the full contribution. However, this happens only when  $\sum_{\mathcal{C} \subseteq \mathcal{Q}} |\mathcal{B} \cap \mathcal{C}| \cdot y'_{i\mathcal{C}} > 1$  and in this case all configurations in  $y'$  for player  $i$  have a big item, so  $\sum_{j:j \in \mathcal{B}} x_{ij} > 1$ . However, if the total weight of big items in  $y'$  for player  $i$  is less than 1, the same total weight is projected on  $x$ . We thus have

$$\sum_{j:j \in \mathcal{B}} x_{ij} = \sum_{\mathcal{C} \subseteq \mathcal{Q}} |\mathcal{B} \cap \mathcal{C}| \cdot y_{i\mathcal{C}}$$

if  $\sum_{\mathcal{C} \subseteq \mathcal{Q}} |\mathcal{B} \cap \mathcal{C}| \cdot y_{i\mathcal{C}} \leq 1$  and otherwise  $\sum_{j:j \in \mathcal{B}} x_{ij} > 1$ . This concludes the proof.  $\square$

In the next section (Lemma 5), we show that Algorithm 1 actually performs better than 3/4 if the solution  $x$  to the assignment LP is produced from a non-well-structured solution  $y$ . In Sect. 3.4 (Lemma 6), we show a new algorithm for well-structured solutions that also has an approximation guarantee strictly better than 3/4. Finally, Lemma 5 and Lemma 6 immediately imply the main result of this section, Theorem 3.

### 3.3 Better analysis for non-well-structured solutions

So far, we have pointed out where some slack is introduced in the analysis of Algorithm 1; as a complement to that discussion, we now show that Algorithm 1 performs well if not all players are fully assigned in the LP. The intuition is that if a player’s budget is not fully used, then the probability that the total price assigned to that player is more than the budget decreases and therefore we exceed the budget by a lower amount on average.

**Lemma 3** *Let  $\varepsilon' > 0$  be a small constant and consider player  $i$  such that  $\text{Val}_i(x) \leq 1 - \varepsilon'$ . Then  $\text{Alg}_i(x) \geq \frac{3+\varepsilon'/5}{4} \text{Val}_i(x)$ .*

*Proof* For player  $i$ , let  $f_i(z) = \sum_j x_{ij} \min\{p_{ij}, z\}$ . Note that  $f_i(z)$  is a continuous function, for which we have  $f_i(1) = \text{Val}_i(x) < 1$  and  $f_i(0) = 0$ . Let  $Q_i$  be the largest value  $z$  that satisfies  $f_i(z) = z$ .

If we substitute prices  $p_{ij}$  with  $q_{ij} = \min\{p_{ij}, Q_i\}$ , the ordering of the items for a player by the price stays the same. Thus Algorithm 1 produces the same result no matter which one of the two prices we use. Let  $D_i$  denote the difference  $\sum_j x_{ij}(p_{ij} - q_{ij}) = \text{Val}_i(x) - Q_i$ . We do case distinction based on the size of  $D_i$ .

In the first case the cut-off  $D_i$  is significant; in this case, we can ensure that if we add back the cut-off in the first bucket, we do not violate the budget and we get an improved approximation guarantee. In the second case we use the fact that  $Q_i$  is bounded away from 1, so the algorithm performs well and with negligible  $D_i$  this leads to an improvement.

Case  $D_i > \frac{\varepsilon'}{5} \text{Val}_i(x)$  :

By Lemma 1, if we run Algorithm 1 on  $x$  but use values  $q_{ij}$  and budget  $Q_i$  in the analysis, we get  $\text{Alg}_i(x) \geq \frac{3}{4} Q_i$ . In order to improve this we use the fact that

$q_{ij} < p_{ij}$  only for at most one total unit of the largest items. If already more than one unit of items is at least  $Q_i$ , then we have  $f_i(Q_i) > Q_i$ ; therefore,  $Q_i$  is not a solution to  $f_i(z) = z$ , a contradiction.

Moreover, since the assignment according to the analysis with respect to  $q$  does not violate the budget  $Q_i$  and the real budget is  $B_i$ , we can get an improved performance guarantee by restoring the original item prices. Specifically, the items in the first bucket had their prices possibly capped to  $Q_i$  from at most  $B_i$ . Hence, if such an item  $j$  is assigned to  $i$ , we can consider its price to be  $p_{ij}$ ; furthermore, since  $q_{ij} < p_{ij}$  can be true only for items in the first bucket, restoring the original prices will not cause us to exceed the budget. Therefore, from all items  $j$ , we gain an increase of  $p_{ij} - q_{ij}$  in the final algorithm performance. Since we are already guaranteed that our algorithm returns an expected value of at least  $3Q_i/4$ , we are not losing a quarter from the difference  $D_i = \text{Val}_i(x) - Q_i$ , so we have an advantage of  $D_i/4 > \frac{\epsilon'}{20} \text{Val}_i(x)$ . Formally,  $\text{Alg}_i(x)$  is at least

$$\frac{3}{4}Q_i + D_i = \frac{3}{4}(Q_i + D_i) + D_i/4 = \frac{3}{4}\text{Val}_i(x) + D_i/4,$$

which is at least  $(\frac{3}{4} + \frac{\epsilon'}{20}) \text{Val}_i(x)$ .

Case  $D_i \leq \frac{\epsilon'}{5} \text{Val}_i(x)$ :

We apply Lemma 1 with prices  $q_{ij}$  and all budgets equal to 1. Since  $Q_i$  is bounded away from 1,  $\text{Alg}_i(x)$  is more than  $3/4Q_i$ . Formally,

$$\begin{aligned} \text{Alg}_i(x) &\geq (1 - Q_i/4)Q_i \\ &= (1 - Q_i/4)\text{Val}_i(x) - (1 - Q_i/4)D_i \\ &\geq \frac{4 - Q_i}{4}\text{Val}_i(x) - D_i. \end{aligned}$$

As  $Q_i \leq \text{Val}_i(x) \leq 1 - \epsilon'$ ,

$$\begin{aligned} \text{Alg}_i(x) &\geq \left(1 - \frac{1 - \epsilon'}{4}\right)\text{Val}_i(x) - D_i \\ &\geq \left(1 - \frac{1 - \epsilon'/5}{4}\right)\text{Val}_i(x) \\ &= \frac{3 + \frac{\epsilon'}{5}}{4}\text{Val}_i(x). \end{aligned}$$

□

From the above, we see that the difficult players to round are those that have an almost full budget. Furthermore, we show in the following lemma that such players must have a special fractional assignment in order to be difficult to round; something we already mentioned in Remarks 1 and 2.

When all items that are fractionally assigned to the player are big, then there is only one bucket, and hence there is no loss in Algorithm 1. On the other hand, if all the items are small, then the probability that the assigned items have total value *much* higher than the budget is low, as suggested in Remark 2. The following analysis formalizes this intuition, showing that the worst-case structure is when the player’s LP value is coming both from small and big items to the same extent.

**Lemma 4** *Let  $\delta > 0$  be a small constant and  $\beta$  be such that  $\delta/4 \leq \beta$ . Consider a player  $i$  such that  $\text{Val}_i(x) \geq 1 - \delta^2/8$  and  $\sum_{j:j \in \mathcal{B}} x_{ij} \notin \left[ \frac{1-\delta}{2}, \frac{1+\delta}{2} \right]$ . Then  $\text{Alg}_i(x) \geq \frac{3+\delta^2/64}{4} \text{Val}_i(x)$ .*

*Proof* If the average in the first bucket is more than  $\frac{3+\delta^2/64}{4} \text{Val}_i(x)$  then we are done, since assigning a random item from that bucket gives sufficient profit. If  $r_{(i,1)} \leq 1 - \delta^2/16$ , Lemma 1 already implies the claim. Therefore assume from now on that  $r_{(i,1)} \geq 1 - \delta^2/16$  and  $r_{(i,2)} \leq r_{(i,1)} \leq \frac{3+\delta^2/16}{4}$ , so  $r_{(i,1)} - r_{(i,2)} \geq 1/8$ , since  $\delta$  is small.

In the proof of Lemma 1 we have that the expected decrease  $\text{Val}_i(x) - \text{Alg}_i(x)$  in our rounding is at most  $\sum_{(i,k)} (r_{(i,k)} - r_{(i,k+1)}) (x'_{(i,k)} - \alpha_i x'_{(i,k)})^2$ .

This can be rewritten as

$$\begin{aligned} & (r_{(i,1)} - r_{(i,2)}) (x'_{(i,1)} - \alpha_i x'_{(i,1)})^2 \\ & + \sum_{(i,\ell):\ell \geq 2} (r_{(i,\ell)} - r_{(i,\ell+1)}) (x'_{(i,\ell)} - \alpha_i x'_{(i,\ell)})^2 \\ & \leq (r_{(i,1)} - r_{(i,2)}) (x'_{(i,1)} - \alpha_i x'_{(i,1)})^2 + r_{(i,2)} \cdot \frac{1}{4\alpha_i} \\ & \leq 1/8 (x'_{(i,1)} - \alpha_i x'_{(i,1)})^2 + 7/8 \cdot \frac{1}{4\alpha_i}. \end{aligned} \tag{3}$$

The last inequality follows from  $(x'_{(i,1)} - \alpha_i x'_{(i,1)})^2 \leq \frac{1}{4\alpha_i}$  and  $r_{(i,1)} - r_{(i,2)} \geq 1/8$ .

In the rest of the proof we prove that  $x'_{(i,1)} - \alpha_i x'_{(i,1)}$  is smaller than  $\frac{1}{4\alpha_i}$  which follows from that  $x'_{(i,1)}$  can not be close to  $1/2$ . This fact will in turn follow from the premises of the lemma; intuitively,  $\text{Val}_i(x) \geq 1 - \delta^2/8$  greatly restricts how the structure of the first bucket will look like, and the premise  $\sum_{j:j \in \mathcal{B}} x_{ij} \notin \left[ \frac{1-\delta}{2}, \frac{1+\delta}{2} \right]$  will then enforce that  $x'_{(i,1)}$  can not be close to  $1/2$ .

Formally, the probability  $x'_{(i,1)}$  corresponds to items in the first bucket that have value at least  $\alpha_i a_{(i,1)}$ . Suppose towards contradiction that more than  $\delta^2/(16\beta)$ -fraction of these items are small items, so they have value less than  $1 - \beta$ , while the rest of the items are big. Then we lose  $\beta$  in the value for a  $\delta^2/(16\beta)$ -fraction of items, so  $r_{(i,1)} < 1 - \frac{\delta^2}{16\beta} \cdot \beta = 1 - \delta^2/16$ , a contradiction. This means that  $x'_{(i,1)} = (\sum_{j \in \mathcal{B}} x_{ij}) + \gamma$ , where  $\gamma \in [0, \delta^2/(16\beta)]$ . By  $\beta \geq \delta/4$ ,  $\gamma \in [0, \delta/4]$ . Since  $\sum_{j \in \mathcal{B}} x_{ij} \notin \left[ \frac{1-\delta}{2}, \frac{1+\delta}{2} \right]$ , we have  $x'_{(i,1)} \notin \left[ \frac{1-\delta/2}{2}, \frac{1+\delta/2}{2} \right]$ , thus  $x'_{(i,1)}$  is bounded away from  $1/2$ .

Observe that the loss in performance in (3) is a multiple of  $x'_{(i,1)} - \alpha_i x'_{(i,1)}{}^2$ , which is an expression of the form  $z - \alpha_i z^2$ . For a function  $z - z^2$  the maximum is attained for  $z = 1/2$ , so  $z$  bounded away from  $1/2$  gives values bounded away from the maximum which is  $1/4$ . For a function  $z - \alpha_i z^2$  the maximum is attained close to  $1/2$  provided that  $\alpha_i$  is close to 1. Again,  $z$  bounded away from  $1/2$  gives values bounded away from the maximum. In the rest of the proof we formalize this intuition.

The maximum for the function  $z(1 - \alpha_i z)$  is attained for  $z = \frac{1}{2\alpha_i}$  and we can prove that  $\frac{1}{2\alpha_i} \in \left[\frac{1-\delta/2}{2}, \frac{1+\delta/2}{2}\right]$ . Since  $\alpha_i \geq 1$ , it only remains to prove that  $\frac{1}{2\alpha_i} \geq \frac{1-\delta/2}{2}$ . By  $1/\alpha_i \geq 1 - \delta^2/8$ ,

$$\frac{1}{2\alpha_i} \geq \frac{1 - \delta^2/8}{2} > \frac{1 - \delta/2}{2}.$$

The function  $z - \alpha_i z^2$  is symmetric around  $\frac{1}{2\alpha_i}$  and this value is closer to the beginning of the interval  $\left[\frac{1-\delta/2}{2}, \frac{1+\delta/2}{2}\right]$ , so the maximum of  $x'_{(i,1)} - \alpha_i x'_{(i,1)}{}^2$  is attained when  $x'_{(i,1)} = \frac{1-\delta/2}{2}$ .

For such  $x'_{(i,1)}$ ,  $x'_{(i,1)} - \alpha_i x'_{(i,1)}{}^2 \leq x'_{(i,1)} - x'_{(i,1)}{}^2 = (1 - \delta^2/4)/4 \leq (1 - \delta^2/8)^2/4$ . Since  $1 - \delta^2/8 \leq \frac{1}{\alpha_i}$ ,

$$x'_{(i,1)} - \alpha_i x'_{(i,1)}{}^2 \leq \frac{1 - \delta^2/8}{4\alpha_i}.$$

We can finally bound the decrease in our rounding  $\text{Val}_i(x) - \text{Alg}_i(x)$ :

$$\text{Val}_i(x) - \text{Alg}_i(x) \leq r_{(i,1)} \left( \frac{1}{8} \cdot \frac{1 - \delta^2/8}{4\alpha_i} + \frac{7}{8} \cdot \frac{1}{4\alpha_i} \right) = \frac{r_{(i,1)}(1 - \delta^2/64)}{4\alpha_i}.$$

The claim follows from the fact that  $r_{(i,1)} \leq 1$  and  $\alpha_i \geq 1$ . □

From Lemmas 3 and 4 we have that as soon as a weighted  $\epsilon$ -fraction (weight of player  $i$  is  $\text{Val}_i(y)$ ) of the players satisfies the conditions of either lemma, we get a better approximation guarantee than  $3/4$ . Therefore, when a solution  $y$  to the configuration LP is not  $(\epsilon, \delta)$ -well-structured, we use Lemma 2 to produce a solution  $x$  to the assignment LP for which  $\epsilon$ -fraction of players satisfies either conditions of Lemmas 3 or 4. Hence we have the following lemma:

**Lemma 5** *Given a solution  $y$  to the configuration LP which is not  $(\epsilon, \delta)$ -well-structured and  $\beta \geq \delta/4$ , we can in polynomial time find a solution with expected value at least  $\frac{3+\epsilon\delta^2/64}{4} \text{Val}(y)$ .*

*Proof* Let  $x$  be a solution to the assignment LP produced from  $y$  as in Lemma 2. Then more than weighted  $\epsilon$ -fraction of players have  $\sum_{j:j \in \mathcal{B}} x_{ij} \notin \left[\frac{1-\delta}{2}, \frac{1+\delta}{2}\right]$ .

According to Lemma 3 using  $\epsilon' = \delta^2/8$ , we have  $\text{Alg}_i(x) \geq \frac{3+\delta^2/40}{4} \text{Val}_i(x)$  if  $\text{Val}_i(x) \leq 1 - \delta^2/8$ . By Lemma 4, using  $\beta \geq \delta/4$  implies  $\text{Alg}_i(x) \geq \frac{3+\delta^2/64}{4} \text{Val}_i(x)$

if  $\text{Val}_i(x) \geq 1 - \delta^2/8$  and  $\sum_{j:j \in \mathcal{B}} x_{ij} \notin \left[ \frac{1-\delta}{2}, \frac{1+\delta}{2} \right]$ . Hence for weighted  $\epsilon$ -fraction of players we get  $\text{Alg}_i(x) \geq \frac{3+\delta^2/64}{4} \text{Val}_i(x)$ , so the total gain is at least  $\frac{3+\epsilon\delta^2/64}{4} \text{Val}(y)$ .  $\square$

### 3.4 Algorithm for well-structured solutions

We devise a novel algorithm that gives an improved approximation guarantee for  $(\epsilon, \delta)$ -well-structured instances when  $\epsilon$  and  $\delta$  are small constants.

**Lemma 6** *Let  $1 - \beta$  be the threshold for the big items. Given a solution  $y$  to the configuration LP that is  $(\epsilon, \delta)$ -well-structured, we can in polynomial time find a solution with expected value at least  $(1 - \delta)^2(1 - \beta - \epsilon) \cdot \frac{25}{32} \text{Val}(y)$ .*

To prove the above lemma we first give the algorithm and then its analysis.

*Algorithm.* The algorithm constructs a slightly modified version  $y'$  of the solution  $y$  to the configuration LP. Solution  $y'$  is obtained from  $y$  in three steps and  $y'$  is then rounded in two additional steps; in the fourth step we assign big items and in the fifth step we assign small items.

*Step 1* Remove all players  $i$  with  $\sum_{\mathcal{C} \subseteq \mathcal{Q}} |\mathcal{B} \cap \mathcal{C}| y_{i\mathcal{C}} \notin \left[ \frac{(1-\delta)}{2}, \frac{(1+\delta)}{2} \right]$ .

*Step 2* Change  $y$  as in the proof of Lemma 2 by getting rid of configurations with 2 big items without losing any objective value. Then remove all small items from the configurations containing big items. After this step, we have the property that big items are alone in each configuration. We call such configurations big and the remaining ones small.

*Step 3* We scale down the fractional assignment of small configurations (if necessary), so as to ensure that  $\sum_{\mathcal{C}:\mathcal{C} \cap \mathcal{B} = \emptyset} y'_{i\mathcal{C}} \leq 1/2$  for each player  $i \in \mathcal{A}$ .

*Step 4* Let  $x'$  be the solution to the assignment LP from Lemma 2 applied on  $y'$  and note that  $\text{Val}(x') = \text{Val}(y')$ . Consider the bipartite graph where we have a vertex  $a_i$  for each player  $i \in \mathcal{A}$ ; a vertex  $b_j$  for each big item  $j \in \mathcal{B}$ ; and an edge of weight  $x'_{ij}$  between vertices  $a_i$  and  $b_j$ . Note that a matching in this graph naturally corresponds to an assignment of big items to players. We shall find our matching/assignment of big items by using Theorem 2. Note that by Theorem 2 we have that (i) each big item  $j$  is assigned with probability  $\sum_i x'_{ij}$  and (ii) the probability that two players  $i$  and  $i'$  are assigned big items is negatively correlated, i.e., it is at most  $\left( \sum_{j \in \mathcal{B}} x'_{ij} \right) \cdot \left( \sum_{j \in \mathcal{B}} x'_{i'j} \right)$ . These two properties are crucial in the analysis of our algorithm. It is therefore important that we assign the big items according to a distribution that satisfies the properties of Theorem 2.

*Step 5* After assigning big items, we assign the small items as follows. First, obtain an optimal solution  $x^{(2)}$  to the assignment LP for the small items together with the players that were *not* assigned a big item in the previous step; these items are the ones that are not assigned during the previous step. Then we obtain an integral assignment (of the small items) of value at least  $\frac{3}{4} \text{Val}(x^{(2)})$  by using Algorithm 1.

*Analysis.* As solution  $y$  is  $(\epsilon, \delta)$ -well-structured, Step 1 decreases the value of the solution by at most  $\epsilon \text{Val}(y)$ . Step 2 decreases the LP value of  $y$  by at most  $\beta \text{Val}(y)$  because each big item has value at least  $1 - \beta$  and each configuration has value at most 1. Regarding the loss incurred by Step 3, since remaining players are assigned big configurations with a total fraction at least  $(1 - \delta)/2$  and therefore small configurations with a total fraction at most  $(1 + \delta)/2$ , it may decrease the value by a factor  $1/(1 + \delta) > 1 - \delta$ .

From now on we work with solution  $y'$ , for which we have  $\text{Val}(y') \geq (1 - \beta - \epsilon)(1 - \delta)\text{Val}(y)$ . Let us reiterate the properties of our tentative solution  $y'$ :

- No big configuration in the support of  $y'$  contains a small item.
- For all players  $i \in \mathcal{A}$  we have  $\sum_{C:C \cap \mathcal{B} = \emptyset} y'_{i,C} \leq 1/2$  and  $\sum_{C:C \cap \mathcal{B} \neq \emptyset} y'_{i,C} \in [\frac{1-\delta}{2}, \frac{1+\delta}{2}]$ .

It remains to argue about the value that is returned from Steps 4 and 5. Let  $x^*$  denote the integral assignment found by the algorithm, and let  $T$  be the players that were *not* assigned a big item in Step 4; since the assignment output by the procedure of Theorem 2 satisfies the marginals, the expected value of  $x^*$  due to Step 4 (i.e., over the randomly chosen assignment of big items) is expressed as:

$$\begin{aligned} \mathbb{E}[\text{Val}(x^*)] &= \mathbb{E}_T \left[ \sum_{j \in \mathcal{B}} \sum_{i \in \mathcal{A} \cap T} x'_{ij} p_j + \frac{3}{4} \text{Val}(x^{(2)}) \right] \\ &= \sum_{j \in \mathcal{B}} \sum_{i \in \mathcal{A}} x'_{ij} p_j + \frac{3}{4} \mathbb{E}_T [\text{Val}(x^{(2)})], \end{aligned}$$

where the expectation is taken over the random choice of the set  $T$  of the players that were not assigned a big item from the sampled matching.

Essentially, this means we recover all of the fractional value of big items in  $x'$  and about 3/4 of the fractional value of small items in  $x^{(2)}$ . We now analyze  $\mathbb{E}[\text{Val}(x^{(2)})]$ , i.e., the expected optimal value of the assignment LP when only considering small items and the set of players  $T \subseteq \mathcal{A}$  that were not assigned big items in Step 4. Then a solution  $z$  to the assignment LP can be obtained by scaling up the fractional assignments of the small items assigned to players in  $T$  according to  $x'$  by up to a factor of 2 while maintaining that an item is assigned at most once. In other words,  $\sum_{i \in \mathcal{A} \cap T} z_{ij} = \min \left\{ 1, \sum_{i \in \mathcal{A} \cap T} 2x'_{ij} \right\}$  for each  $j$  and  $z$  is a feasible solution to the assignment LP, because we have  $\sum_{j \in \mathcal{S}} x'_{ij} p_j \leq 1/2$  for every  $i$ .

Thus we have that the expected value of the optimal solution to the assignment LP is by linearity of expectation at least  $\mathbb{E}_T[\text{Val}(x^{(2)})] \geq \sum_{j \in \mathcal{S}} p_j \cdot \mathbb{E}_T \left[ \min \left\{ 1, \sum_{i \in \mathcal{A} \cap T} 2x'_{ij} \right\} \right]$ .

We continue by analyzing the expected fraction of a small item. In this lemma we use that the randomly selected matching of big items has negative correlation. To see why this is necessary, consider a small item  $j \in \mathcal{S}$  and suppose that  $j$  is assigned to two players  $u$  and  $v$  both with a fraction 1/2, i.e.,  $x'_{uj} = x'_{vj} = 1/2$ . As the instance is well-structured and each configuration contains at most one big item, both  $u$  and  $v$  are

roughly assigned half a fraction of big items; for simplicity assume it to be exactly  $1/2$ . Note that in this case we have that  $\min \left\{ 1, \sum_{i \in \mathcal{A} \cap T} 2x'_{ij} \right\}$  is equal to 1 if either  $u$  or  $v$  are not assigned a big item and 0 otherwise. Therefore, on the one hand, if the event that  $u$  is assigned a big item and the event that  $v$  is assigned a big item were perfectly correlated then we would have  $\mathbb{E}_T \left[ \min \left\{ 1, \sum_{i \in \mathcal{A} \cap T} 2x'_{ij} \right\} \right] = 1/2$ . On the other hand, if those events are negatively correlated then  $\mathbb{E}_T \left[ \min \left\{ 1, \sum_{i \in \mathcal{A} \cap T} 2x'_{ij} \right\} \right] \geq 3/4$ , as in this case the probability that both  $u$  and  $v$  are assigned big items is at most  $1/4$ .

**Lemma 7** For every  $j \in \mathcal{S}$ ,  $\mathbb{E}_T \left[ \min \left\{ 1, \sum_{i \in \mathcal{A} \cap T} 2x'_{ij} \right\} \right] \geq (1 - \delta)^{\frac{3}{4}} \sum_{i \in \mathcal{A}} x'_{ij}$ .

*Proof* We slightly abuse notation and also denote by  $T$  the event that the players in  $T \subseteq \mathcal{A}$  were those that were not assigned big items. Let also  $x(T) = \min \left\{ 1, \sum_{k \in \mathcal{A} \cap T} 2x'_{kj} \right\}$  for the considered small item  $j$ . With this notation,

$$\mathbb{E}_T \left[ \min \left\{ 1, \sum_{i \in \mathcal{A} \cap T} 2x'_{ij} \right\} \right] = \sum_{T \subseteq \mathcal{A}} \Pr[T] \cdot x(T).$$

We shall show that we can lower bound this quantity by assuming that  $j$  is only fractionally assigned to two players. Indeed, suppose  $j$  is fractionally assigned to more than two players. Then there must exist two players, say  $i$  and  $i'$ , so that  $0 < x'_{ij} < 1/2$  and  $0 < x'_{i'j} < 1/2$ ; the fractional assignment of a small item to some player never exceeds  $1/2$  by construction of  $y'$  and  $x'$ . We can write  $\sum_{T \subseteq \mathcal{A}} \Pr[T] \cdot x(T)$  as

$$\begin{aligned} & \sum_{T \subseteq \mathcal{A} \setminus \{i, i'\}} (\Pr[T] \cdot x(T) + \Pr[T \cup \{i\}] \cdot x(T \cup \{i\}) \\ & + \Pr[T \cup \{i'\}] \cdot x(T \cup \{i'\}) + \Pr[T \cup \{i, i'\}] \cdot x(T \cup \{i, i'\})) \end{aligned} \tag{4}$$

Note that if we shift some amount of fractional assignment from  $x'_{ij}$  to  $x'_{i'j}$  (or vice-versa) then  $x(T)$  and  $x(T \cup \{i, i'\})$  do not change. We now analyze the effect such a shift has on the sums  $\sum_{T \subseteq \mathcal{A} \setminus \{i, i'\}} \Pr[T \cup \{i\}] \cdot x(T \cup \{i\})$  and  $\sum_{T \subseteq \mathcal{A} \setminus \{i, i'\}} \Pr[T \cup \{i'\}] \cdot x(T \cup \{i'\})$ . After this shift  $x'$  might not be a valid solution to the assignment LP, namely we might go over the budget for some players. However, our goal is only to prove a lower-bound on  $\mathbb{E}[\text{Val}(x^{(2)})]$ .

For this purpose let  $F_i$  denote the probability that the set  $T$  is selected such that the value of  $x(T \cup \{i\})$  is strictly less than 1, i.e.,

$$F_i := \sum_{T \subseteq \mathcal{A} \setminus \{i, i'\}: x(T \cup \{i\}) < 1} \Pr[T \cup \{i\}].$$

The definition of  $F_{i'}$  is analogous. Note that if we, on the one hand, decrease  $x'_{ij}$  by a small  $\eta$  and increase  $x'_{i'j}$  by  $\eta$ , this changes (4) by at most  $2\eta(-F_i + F_{i'})$ . On the other hand, if we increase  $x'_{ij}$  and decrease  $x'_{i'j}$  by  $\eta$ , then (4) changes by at most  $2\eta(F_i - F_{i'})$ . Necessarily, one of  $2\eta(F_i - F_{i'})$  and  $2\eta(-F_i + F_{i'})$  is non-positive.

Let us assume  $(-F_i + F_{i'})$  is non-positive (the complementary case is treated analogously), in which case we will shift a small fraction  $\eta$  from  $x'_{ij}$  to  $x'_{i'j}$ . This implies that the change in (4) is non-positive. After the small change by  $\eta$ ,  $F_i$  can not decrease. To see this, let  $\mathcal{T}_i = \{T \subseteq \mathcal{A} \setminus \{i, i'\} : x(T \cup \{i\}) < 1\}$  with respect to solution  $x'$  before the  $\eta$ -shift, and let  $\mathcal{T}'_i = \{T \subseteq \mathcal{A} \setminus \{i, i'\} : x(T \cup \{i\}) < 1\}$  with respect to solution  $x'$  after the  $\eta$ -shift. Then,  $\mathcal{T}_i \subseteq \mathcal{T}'_i$ , which by the definition of  $F_i$  implies that  $F_i$  will not decrease after the  $\eta$ -shift. Similarly, let  $\mathcal{T}_{i'} = \{T \subseteq \mathcal{A} \setminus \{i, i'\} : x(T \cup \{i'\}) < 1\}$  with respect to solution  $x'$  before the  $\eta$ -shift, and let  $\mathcal{T}'_{i'} = \{T \subseteq \mathcal{A} \setminus \{i, i'\} : x(T \cup \{i'\}) < 1\}$  with respect to solution  $x'$  after the  $\eta$ -shift. Then,  $\mathcal{T}'_{i'} \subseteq \mathcal{T}_{i'}$ , which by the definition of  $F_{i'}$  implies that  $F_{i'}$  will not increase after the  $\eta$ -shift. Therefore, if we make any shift in the same direction (i.e., from  $x'_{ij}$  to  $x'_{i'j}$ ), the change in (4) will always be non-positive. We can therefore shift a fraction of  $x'_{ij}$  to  $x'_{i'j}$  without increasing (4) until one of the variables becomes 0 or 1/2. If it becomes 0 then we repeat the process with one less fractionally assigned small item and if it becomes 1/2 we consider two other players where  $j$  is fractionally assigned strictly between 0 and 1/2. After repeating the above process we may assume that  $j$  is fractionally assigned to at most two players  $u$  and  $v$  and  $x'_{uj}, x'_{vj} \leq 1/2$ . Therefore (4) is at least

$$\begin{aligned} & \Pr[u, v \notin T] \cdot 0 + \Pr[u \in T, v \notin T] \cdot 2x'_{uj} \\ & + \Pr[u \notin T, v \in T] \cdot 2x'_{vj} + \Pr[u, v \in T] \cdot \min\{1, 2x'_{uj} + 2x'_{vj}\} \\ & \geq \Pr[u \in T] \cdot 2x'_{uj} + \Pr[v \in T] \cdot 2x'_{vj} + \Pr[u, v \in T] \left( \min\left\{1, 2x'_{uj} + 2x'_{vj}\right\} \right. \\ & \left. - 2x'_{uj} - 2x'_{vj} \right). \end{aligned} \tag{5}$$

We have  $\min\{1, 2x'_{uj} + 2x'_{vj}\} - 2x'_{uj} - 2x'_{vj} \geq -x'_{uj} - x'_{vj}$ , since  $\min\{1, 2x'_{uj} + 2x'_{vj}\} \geq x'_{uj} + x'_{vj}$ . Let us fix  $\Pr[u \in T]$  and  $\Pr[v \in T]$ , then (5) is minimized when  $\Pr[u, v \in T]$  is maximized. By Theorem 2 the assignments of big items are negatively correlated, so  $\Pr[u, v \in T] \leq \Pr[u \in T] \Pr[v \in T]$ . Thus (5) is at least

$$\Pr[u \in T] \cdot 2x'_{uj} + \Pr[v \in T] \cdot 2x'_{vj} - \Pr[u \in T] \Pr[v \in T] (x'_{uj} + x'_{vj}). \tag{6}$$

The distribution from Theorem 2 preserves the marginals, so  $\Pr[u \in T]$  and  $\Pr[v \in T]$  are in the interval  $[\frac{1-\delta}{2}, \frac{1+\delta}{2}]$ . Since (6) is non-decreasing in  $\Pr[u \in T]$  and  $\Pr[v \in T]$ , the worst case is when  $\Pr[u \in T] = \Pr[v \in T] = \frac{1-\delta}{2}$ , which implies  $\Pr[u, v \in T] \leq \frac{(1-\delta)^2}{4}$ . Hence (6) is at least

$$\begin{aligned} & (1 - \delta)x'_{uj} + (1 - \delta)x'_{vj} - \frac{(1 - \delta)^2}{4}(x'_{uj} + x'_{vj}) \\ & \geq (1 - \delta) \left( 1 - \frac{1 - \delta}{4} \right) (x'_{uj} + x'_{vj}) \geq (1 - \delta) \frac{3}{4} (x'_{uj} + x'_{vj}). \end{aligned}$$

□

Let us now see how it implies Lemma 6. We have that  $\mathbb{E}[\text{Val}(x^*)]$  is equal to

$$\sum_{j \in \mathcal{B}} \sum_{i \in \mathcal{A}} x'_{ij} p_j + \frac{3}{4} \mathbb{E} [\text{Val} (x^{(2)})] \geq \sum_{j \in \mathcal{B}} \sum_{i \in \mathcal{A}} x'_{ij} p_j + (1 - \delta) \left(\frac{3}{4}\right)^2 \sum_{j \in \mathcal{S}} \sum_{i \in \mathcal{A}} x'_{ij} p_j.$$

As  $\sum_{j \in \mathcal{B}} x'_{ij} \geq \frac{1-\delta}{2}$  for every remaining player, we have

$$\frac{\mathbb{E}[\text{Val}(x^*)]}{\text{Val}(x')} \geq (1 - \delta) \left(\frac{1}{2} + \frac{1}{2} \frac{9}{16}\right) = (1 - \delta) \frac{25}{32}.$$

Lemma 6 now follows from that  $\text{Val}(x') \geq (1 - \beta - \epsilon)(1 - \delta)\text{Val}(y)$ . We have proved Lemmas 5 and 6, which in turn imply Theorem 3 and our analysis is concluded.

### 4 An algorithm for graph MBA

In this section, we consider the graph MBA problem. Specifically, every player  $i \in \mathcal{A}$  has a (possibly different) budget  $B_i$  and every item  $j \in \mathcal{Q}$  has a non-zero valuation for at most two players. This can be viewed as a graph problem where items are edges, players are vertices and assigning an item means directing an edge towards a vertex.

We already know that the integrality gap of the assignment LP for this variant is exactly  $3/4$  [1], and that of the configuration LP is no better than  $5/6$  [6]. We prove that using the configuration LP, we can recover a fraction of the LP value that is bounded away from  $3/4$  by a constant, implying the following theorem.

**Theorem 4** *There is a polynomial time algorithm which returns a  $(\frac{3}{4} + c)$ -approximate solution to the graph MBA problem, for some constant  $c > 0$ .*

We remark that our algorithm for Graph MBA is slightly different from the one in the conference version of this paper due to a technical error.

Let us define some notation first. Let  $y$  be a solution to the configuration LP. We abuse notation and use  $x$  to denote a fractional assignment such that for all  $i \in \mathcal{A}$  and  $j \in \mathcal{Q}$ ,  $x_{ij} = \sum_{\mathcal{C} \subseteq \mathcal{Q}: j \in \mathcal{C}} y_{i\mathcal{C}}$ , i.e., it corresponds to the sum of all values  $y_{i\mathcal{C}}$  for which  $j$  is a member of  $\mathcal{C}$ . Note that we can always maintain that for all  $j \in \mathcal{Q}$ ,  $\sum_{i \in \mathcal{A}} x_{ij} = 1$ , by assigning item  $j$  to some arbitrary configuration if needed, even though that configuration exceeds the budget of the player to which it is assigned. Similarly, we can always guarantee for all players  $i \in \mathcal{A}$  that  $\sum_{\mathcal{C} \subseteq \mathcal{Q}} y_{i\mathcal{C}} = 1$ , by setting  $y_{i\emptyset}$  to whatever value that is needed.

We denote by  $p_{ij\mathcal{C}}$  the contribution of item  $j$  to  $w_i(\mathcal{C})$ , i.e., the part of the value of configuration  $\mathcal{C}$  that is attributed to item  $j$ . More specifically, if  $w_i(\mathcal{C}) < B_i$ , then the contribution is the same as the valuation, i.e.,  $p_{ij\mathcal{C}} = p_{ij}$ . Otherwise, we take

$$p_{ij\mathcal{C}} = p_{ij} \frac{B_i}{\sum_{j' \in \mathcal{C}} p_{ij'}},$$

i.e., we scale down the valuation, so that each item gets a part of  $B_i$  that is proportional to its valuation. Note that  $\sum_{j \in \mathcal{C}} p_{ij\mathcal{C}} = w_i(\mathcal{C})$ , so  $p_{ij\mathcal{C}}$  is a partition of  $w_i(\mathcal{C})$  among all items in  $\mathcal{C}$  proportional to  $p_{ij}$ .

Let  $\text{Val}_{ij}(y) = \sum_{\mathcal{C} \subseteq \mathcal{Q}: j \in \mathcal{C}} y_{i\mathcal{C}} p_{ij\mathcal{C}}$  be the contribution of item  $j$  to the LP objective value which comes from its assignment to player  $i$ . Note that  $\text{Val}_{ij}(y)$  corresponds to the contribution that player  $i$  expects from item  $j$  when configuration  $\mathcal{C}$  is assigned to  $i$  with probability  $y_{i\mathcal{C}}$ . Also note that the definition of  $\text{Val}$  here is slightly different than the one in Sect. 3. Since  $p_{ij\mathcal{C}}$  depends on the configuration  $\mathcal{C}$ ,  $\text{Val}_{ij}(y)$  depends heavily on the specific structure of the solution  $y$  to the configuration LP, and the configurations it assigns to player  $i$ . Furthermore, let  $\text{Val}_j(y) = \sum_{i \in \mathcal{A}} \text{Val}_{ij}(y)$  be the contribution of item  $j$  to the objective value of the LP. We have that  $\text{Val}(y) = \sum_{j \in \mathcal{Q}} \text{Val}_j(y)$  is the objective value of  $y$ . Let  $\text{Avg}_{ij}(y) = \text{Val}_{ij}(y)/x_{ij}$ , i.e., the average contribution of  $j$  conditioned on the event that  $j$  is in a configuration chosen by  $i$  (this happens with probability  $x_{ij}$ ).

### 4.1 Preprocessing

We first preprocess the solution  $y$  to the configuration LP. The idea is to find two players  $i$  and  $i'$  with two configurations  $\mathcal{C}$  and  $\mathcal{C}'$  (for  $i$  and  $i'$  respectively) that could be used to gain at least a  $(3/4+c)$ -fraction of the total budget  $B_i + B_{i'}$  without affecting the rest of the players.

For two players  $i, i' \in \mathcal{A}$  let  $\mathcal{Q}_{i,i'}$  be items that can only be assigned to these players. The preprocessing is as follows. If there are configurations  $\mathcal{C}$  and  $\mathcal{C}'$  with  $y_{i\mathcal{C}} > 0$  and  $y_{i'\mathcal{C}'} > 0$  such that

$$\sum_{j \in (\mathcal{C} \setminus \mathcal{C}') \cap \mathcal{Q}_{i,i'}} p_{ij\mathcal{C}} + \sum_{j \in (\mathcal{C}' \setminus \mathcal{C}) \cap \mathcal{Q}_{i,i'}} p_{ij\mathcal{C}'} \geq (3/4 + c)(B_i + B_{i'}),$$

then assign the items in  $\mathcal{Q}_{i,i'}$  according to  $\mathcal{C}$  and  $\mathcal{C}'$ . If an item is assigned to both then assign it arbitrarily. We then remove players  $i, i'$  from the instance as well as the items in  $\mathcal{Q}_{i,i'}$ . In each preprocessing step we make progress towards a  $(3/4+c)$ -approximation, since we assign items of value at least  $(3/4 + c)(B_i + B_{i'})$  to  $i$  and  $i'$ . Note that we only use items from  $\mathcal{Q}_{i,i'}$ , so all other players are unaffected by the removal.

We continue until there are no two players with such two configurations left. Since there is only a polynomial number of configurations in  $y$  with  $y_{i\mathcal{C}} > 0$ , the preprocessing takes polynomial time.

### 4.2 First algorithm

To prove Theorem 4, we employ two algorithms that perform well in two different cases. Algorithm 2 recovers more than a  $3/4$ -fraction of  $\text{Val}_j(y)$  whenever  $\text{Avg}_{ij}(y)$  and  $\text{Avg}_{i'j}(y)$  differ a lot or when  $x_{ij}$  is bounded away from  $1/2$ .

Let  $\max(j)$  be the player that maximizes  $\text{Avg}_{ij}(y)$  for item  $j$  and let  $\min(j)$  be the player that minimizes  $\text{Avg}_{ij}(y)$  for item  $j$ .

```

Input : Solution  $y$  to the configuration LP
Output: Assignment  $y^*$  of items to the players
foreach  $i \in \mathcal{A}$  do
    // Configuration assignment
    Pick a configuration  $\mathcal{C}$  with probability equal to  $y_{i\mathcal{C}}$ 
end foreach
foreach  $j \in \mathcal{Q}$  do
    // Conflict resolution
    if  $j$  is chosen by both  $i$  and  $i'$  then
        Assign  $j$  to  $i^* \in \{i, i'\}$  which maximizes  $\text{Avg}_{i^*j}(x)$ 
    end if
end foreach
Return assignment of items  $y^*$ 
    
```

**Algorithm 2:** Sampling algorithm using averages

**Lemma 8** For all  $j \in \mathcal{Q}$ , the expected contribution of  $j$  to the objective value is at least

$$\text{Val}_j(y) - \rho(1 - \rho)\text{Avg}_{i'j}(y)$$

where  $i' = \min(j)$ ,  $i = \max(j)$  and  $\rho = x_{ij}$ .

*Proof* Item  $j$  is assigned to  $i$  whenever  $i$  picks a configuration containing  $j$ , because  $i$  has priority over  $i'$ . Therefore, the expected contribution of  $j$  due to its assignment to  $i$  is  $\text{Val}_{ij}(y)$ . On the other hand,  $j$  is taken away from  $i'$  whenever  $i$  also picks  $j$  and this happens with probability  $\rho = x_{ij}$ . Hence, the expected contribution of  $j$  due to its assignment to  $i'$  is  $(1 - \rho)\text{Val}_{i'j}(y)$ .

We can write the total expected contribution as

$$\text{Val}_{ij}(y) + (1 - \rho)\text{Val}_{i'j}(y) = \text{Val}_j(y) - \rho\text{Val}_{i'j}(y) = \text{Val}_j(y) - \rho(1 - \rho)\text{Avg}_{i'j}(y),$$

since  $\text{Val}_j(y) = \text{Val}_{ij}(y) + \text{Val}_{i'j}(y)$  and  $(1 - \rho)\text{Avg}_{i'j}(y) = \text{Val}_{i'j}(y)$ . □

**Corollary 1** If  $\text{Avg}_{ij}(y) - \text{Avg}_{i'j}(y) \geq \gamma\text{Avg}_{i'j}(y)$ , the expected contribution of item  $j$  is at least

$$\left(1 - \frac{\rho(1 - \rho)}{1 + \rho\gamma}\right) \text{Val}_j(y),$$

where  $\rho = x_{ij}$ .

*Proof* We can write

$$\begin{aligned} \text{Val}_j(y) &= \rho\text{Avg}_{ij}(y) + (1 - \rho)\text{Avg}_{i'j}(y) \\ &= \text{Avg}_{i'j}(y) + \rho(\text{Avg}_{ij}(y) - \text{Avg}_{i'j}(y)) \\ &\geq (1 + \rho\gamma)\text{Avg}_{i'j}(y). \end{aligned}$$

We thus have that

$$\frac{\rho(1 - \rho)\text{Avg}_{i'j}(y)}{\text{Val}_j(y)} \leq \frac{\rho(1 - \rho)\text{Avg}_{i'j}(y)}{(1 + \rho\gamma)\text{Avg}_{i'j}(y)} = \frac{\rho(1 - \rho)}{1 + \rho\gamma}.$$

By Lemma 8, the expected contribution of  $j$  to the objective value is at least

$$\text{Val}_j(y) - \rho(1 - \rho)\text{Avg}_{i'j}(y) \geq \left(1 - \frac{\rho(1 - \rho)}{1 + \rho\gamma}\right) \text{Val}_j(y).$$

□

Suppose a weighted  $\varepsilon$ -fraction of items has  $\frac{\rho(1-\rho)}{1+\rho\gamma} \leq 1/4 - c'$ , such that  $\varepsilon c' = c$ . Then Algorithm 2 returns a solution of value that is at least  $3/4 + c'\varepsilon = 3/4 + c$  fraction of  $\text{Val}(y)$ .

If  $\rho \in [1/2 - \beta, 1/2 + \beta]$  for some small  $\beta$ , then the loss in Corollary 1 is at most  $\frac{1}{4(1+\gamma/3)}$ , since  $\rho(1 - \rho) \leq 1/4$  and  $\rho\gamma \geq \gamma/3$ . Thus for  $\gamma > 0$  we get a better than  $3/4$ -approximation. On the other hand, if  $\rho \notin [1/2 - \beta, 1/2 + \beta]$ , the loss in the corollary is at most  $1/4 - \beta^2 < 1/4$  and we also get a better than  $3/4$ -approximation. We can therefore assume that each item is assigned to both players with probability roughly  $1/2$  and that it has roughly the same average value for both players, because otherwise we already do better than  $3/4$ .

### 4.3 Second algorithm

We now present Algorithm 3. It works in two phases, called the primary assignment phase and the secondary assignment phase. Note that the secondary assignment is similar to the techniques used to tackle GAP in [9].

During the primary assignment phase, player  $i$  picks configuration  $\mathcal{C}$  with probability equal to  $y_{i\mathcal{C}}$ . An item  $j$  might be picked by two players  $i$  and  $i'$  and these conflicts are resolved as follows: we assign item  $j$  to  $i$  with probability  $x_{i'j}$  and to  $i'$  with probability  $x_{ij}$ . It is important that we use the value for player  $i'$  to assign an item to player  $i$  and vice versa.

During the secondary assignment phase, we want to assign items that were not picked by any player. We assign an item  $j$  to players  $i$  and  $i'$  uniformly at random.

The first important property of the algorithm is that the primary assignment already recovers a  $3/4$ -fraction of the LP value. The second important property is that when many values in  $x$  are close to  $1/2$  or  $\text{Avg}_{\max(j)j}(y)$  and  $\text{Avg}_{\min(j)j}(y)$  are close for many items, in expectation a constant fraction of the budget is left empty after the primary assignment and thus it can be used by the yet unassigned items to allocate a constant fraction of the LP value. Let us now formalize this intuition.

### 4.4 Analysis of the primary assignment

Let us first prove a lemma regarding the value recovered by the primary assignment, i.e., the phase of the algorithm that involves configuration assignment and conflict

```

Input : Solution  $y$  to the configuration LP
Output: Assignment  $y^*$  of items to the players
Let  $x_{ij} = \sum_{\mathcal{C} \subseteq \mathcal{Q}: j \in \mathcal{C}} y_{i\mathcal{C}}$ , for all  $i \in \mathcal{A}$  and  $j \in \mathcal{Q}$ 
foreach  $i \in \mathcal{A}$  do
    // Configuration assignment, part of primary assignment
    Pick a configuration  $\mathcal{C}$  with probability equal to  $y_{i\mathcal{C}}$ 
end foreach
foreach  $j \in \mathcal{Q}$  do
    // Conflict resolution, part of primary assignment
    if  $j$  is chosen by both  $i$  and  $i'$  then
        Assign  $j$  to  $i$  with probability  $x_{i'j}$ , otherwise assign  $j$  to  $i'$ 
    end if
    // Secondary assignment
    if  $j$  is unassigned then
        Assign  $j$  to  $i \in \{k \mid p_{kj} > 0\}$  chosen uniformly at random
    end if
end foreach
Return assignment of items  $y^*$ 
    
```

**Algorithm 3:** Configuration sampling algorithm

resolution. Let  $y^p$  be the solution to the configuration LP that corresponds to the primary assignment. We have the following lemma.

**Lemma 9** *Let  $i \in \mathcal{A}$ ,  $j \in \mathcal{Q}$  and  $\rho = x_{ij}$ . The expected contribution of  $j$  to  $i$  due to primary assignment is at least*

$$\mathbb{E} [\text{Val}_{ij}(y^p)] \geq (1 - \rho + \rho^2) \text{Val}_{ij}(y).$$

*Proof* An item  $j$  is expected to contribute with  $\text{Val}_{ij}(y)$  to the value of  $i$  when configurations are randomly picked according to  $y$ . However,  $j$  might be assigned to another player in the conflict resolution, so we cannot make use of the full value  $\text{Val}_{ij}(y)$ .

Let  $i$  and  $i'$  be the players for which  $p_{ij} > 0$  and  $p_{i'j} > 0$ . If  $j$  is in a configuration that was picked by  $i$ , the probability that  $j$  is not picked by  $i'$  is  $\rho$ , while the probability that  $j$  is also picked by  $i'$  but then assigned in conflict resolution to  $i$  is  $(1 - \rho)^2$ . So the total probability of  $j$  being assigned to  $i$ , when  $j$  is picked by  $i$ , is  $\rho + (1 - \rho)^2 = 1 - \rho + \rho^2$ .

Therefore, the expected contribution of  $j$  to  $i$  due to primary assignment to  $i$  is

$$\mathbb{E} [\text{Val}_{ij}(y^p)] \geq (1 - \rho + \rho^2) \text{Val}_{ij}(y).$$

□

The above lemma implies the following corollary.

**Corollary 2** *Let  $j \in \mathcal{Q}$  be such that it can be assigned to players  $i$  and  $i'$ . The expected contribution of  $j$  to the objective value due to primary assignment is*

$$\mathbb{E} [\text{Val}_j(y^p)] \geq (1 - \rho + \rho^2) (\text{Val}_{ij}(y) + \text{Val}_{i'j}(y)).$$

where  $\rho = x_{ij}$ . Moreover,

$$\mathbb{E} [\text{Val}_j(y^p)] \geq \frac{3}{4} \text{Val}_j(y).$$

*Proof* We have  $1 - x_{ij} + x_{ij}^2 = 1 - x_{i'j} + x_{i'j}^2 = 1 - \rho + \rho^2$ , since  $x_{ij} + x_{i'j} = 1$ . The expected contribution of  $j$  to the objective value of the LP due to primary assignment is

$$\begin{aligned} \mathbb{E} [\text{Val}_j(y^p)] &= \mathbb{E} [\text{Val}_{ij}(y^p) + \text{Val}_{i'j}(y^p)] \\ &= (1 - \rho + \rho^2) (\text{Val}_{ij}(y) + \text{Val}_{i'j}(y)) \\ &\geq \frac{3}{4} \text{Val}_j(y), \end{aligned}$$

since  $1 - \rho + \rho^2 \geq 3/4$  for all  $\rho$ . □

As we can see, the primary assignment already recovers a 3/4-fraction of the objective value. It thus remains to prove that the secondary assignment recovers an additional constant fraction of the objective value.

### 4.5 Analysis of the secondary assignment

Let  $\delta \in (0, \frac{1}{2})$  and  $\gamma > 0$  be parameters to be defined later. Let

$$\begin{aligned} I &= \{j \in \mathcal{Q} : \max\{x_{ij} : i \in \mathcal{A}\} \geq 1 - \delta \\ &\quad \vee \text{Avg}_{\max(j)}(y) \geq (1 + \gamma) \text{Avg}_{\min(j)}(y)\} \end{aligned}$$

be the set of almost-integral items together with items that have big difference between  $\text{Avg}_{\max(j)}(y)$  and  $\text{Avg}_{\min(j)}(y)$ . Let  $H = \mathcal{Q} \setminus I$  be the remaining items.

By Corollary 1, Algorithm 2 recovers more than a 3/4-fraction of the value of each item in  $I$ . Therefore, when the fraction of the objective value which corresponds to items in  $I$  is non-negligible, we can improve over the approximation ratio 3/4. Hence, our troubles begin when the contribution of items in  $I$  to the objective value is tiny.

More formally, if for some  $\varepsilon > 0$ , at least an  $\varepsilon$ -fraction of the LP value corresponds to items in  $I$ , i.e.,

$$\sum_{j \in I} \text{Val}_j(y) \geq \varepsilon \text{Val}(y),$$

then Corollary 1 implies that

$$\sum_{j \in \mathcal{Q}} \mathbb{E} [\text{Val}_j(y^p)] \geq \left(\frac{3}{4} + \varepsilon'\right) \text{Val}(y).$$

for some  $\varepsilon' > 0$  which depends on  $\varepsilon, \delta$  and  $\gamma$ .

Since we know how to achieve a better than  $3/4$ -approximation in this case, we can make the assumption that less than an  $\varepsilon$ -fraction of the LP value comes from items in  $I$ . We remove items in  $I$  and work under the assumption that all the items belong to  $H$ . The removal causes a decrease in the LP value that is at most an  $\varepsilon$ -fraction of the original LP value. For simplicity, we ignore this term for the rest of our analysis, up to the point where we finalize the proof of Theorem 4.

From now on, assume that all items belong to  $H$ . Let  $y^*$  be the rounded integral solution after the secondary assignment step. First, we prove a lemma concerning the value we gain from Algorithm 3 for each item in  $H$ , and then conclude the analysis by proving that the approximation ratio of Algorithm 2 and Algorithm 3 is greater than  $3/4$ . Let  $\text{Alg}$  be the value of the rounded integral solution  $y^*$ . Then we have the following lemma.

**Lemma 10** *If all items are in  $H$ ,*

$$\mathbb{E}[\text{Alg}] \geq \left( \frac{3}{4} + \frac{\delta^2(1 - \delta)^2}{15} \right) \text{Val}(y),$$

where the expectation is taken over the random choices of Algorithm 3.

*Proof* Fix item  $j \in H$  and let  $i$  and  $i'$  be the players that  $j$  can be assigned to. The idea of the proof is that the preprocessing step leaves a fraction of the total budget  $B_i + B_{i'}$  either unassigned or assigned to items that are removed with constant probability in the conflict resolution. This constant fraction can be used by  $j$  in the secondary assignment.

We already analyzed the expected contribution of  $j$  due to the primary assignment in Lemma 9, so we focus mostly on the secondary assignment in this proof. Item  $j$  gets secondarily assigned if  $j$  was not assigned in the primary assignment to any of the players  $i$  and  $i'$ . This means that player  $i$  picked configuration  $\mathcal{C}$  and player  $i'$  picked  $\mathcal{C}'$  such that  $j \notin \mathcal{C} \cup \mathcal{C}'$ . This happens with probability at least  $\delta(1 - \delta)$ , since  $j \in H$ . The preprocessing makes sure that at most a  $(3/4 + c)$ -fraction of the total budget  $B_i + B_{i'}$  comes from items that are assigned only to  $i$  or only to  $i'$ . Hence, when  $i$  and  $i'$  pick configurations  $\mathcal{C}$  and  $\mathcal{C}'$ , a value of at least  $(1/4 - c)(B_i + B_{i'})$  is completely unassigned or corresponds to items not in  $\mathcal{Q}_{i,i'}$ , i.e., items that are assignable to players other than  $i$  and  $i'$ .

With probability of  $1 - x_{ij'}$  item  $j' \in \mathcal{C} \setminus \mathcal{Q}_{i,i'}$  is also picked by another player  $\ell$  and assigned to her with probability  $x_{ij'}$  in the conflict resolution, and this is true even if we condition on the event that players  $i$  and  $i'$  picked  $\mathcal{C}$  and  $\mathcal{C}'$  respectively. Since all items are in  $H$ ,  $x_{ij'} \in [\delta, 1 - \delta]$  and therefore each item  $j' \in \mathcal{C} \setminus \mathcal{Q}_{i,i'}$  has a probability of at least  $x_{ij'}(1 - x_{ij'}) \geq \delta(1 - \delta)$  to not be assigned to  $i$  after conflict resolution. The same analysis applies to player  $i'$  and items in  $\mathcal{C}' \setminus \mathcal{Q}_{i,i'}$ .

For the rest of the proof we make the assumption that items in  $\mathcal{C}$  and  $\mathcal{C}'$  do not exceed the budgets, i.e.,  $\sum_{j' \in \mathcal{C}} p_{ij'} \leq B_i$  and  $\sum_{j' \in \mathcal{C}'} p_{i'j'} \leq B_{i'}$ . This is without loss of generality, because the opposite situation is the easier case. Suppose towards contradiction that the lemma is false when  $\sum_{j' \in \mathcal{C}} p_{ij'} > B_i$  or  $\sum_{j' \in \mathcal{C}'} p_{i'j'} > B_{i'}$ , but is true otherwise. After  $i$  and  $i'$  pick  $\mathcal{C}$  and  $\mathcal{C}'$ , scale down the valuations of items in  $\mathcal{C}$  and  $\mathcal{C}'$  so that their total valuations match exactly  $B_i$  and  $B_{i'}$ , i.e., we use  $p_{ij\mathcal{C}}$  and

$p_{i'j\mathcal{C}'}$  as valuations. The expected contribution of each item did not decrease with these new valuations, since we scale down  $p_{ij}$  to  $p_{ij\mathcal{C}}$  when calculating  $\text{Val}_{ij}(y)$  anyway. Therefore the lemma would be true in this case as well, a contradiction.

Since  $\sum_{j' \in \mathcal{C}} p_{ij'} \leq B_i$  and  $\sum_{j' \in \mathcal{C}'} p_{i'j'} \leq B_{i'}$ , in expectation at least a  $\delta(1 - \delta)$ -fraction of  $(1/4 - c)(B_i + B_{i'})$  is free after the conflict resolution, i.e. the total free value is at least

$$\delta(1 - \delta) \left( \frac{1}{4} - c \right) (B_i + B_{i'}) \geq \frac{\delta(1 - \delta)}{5} (B_i + B_{i'}),$$

since we assume that  $c$  is small. This part of the budget is used in the secondary assignment by  $j$  and other secondarily assigned items.

Suppose a  $b$ -fraction of  $i$ 's budget  $B_i$  is free after the primary assignment and let  $U_i$  be the set of items that is secondarily assigned to  $i$ . By the definition of  $\text{Val}$ ,  $\sum_j \text{Val}_{ij}(y) \leq B_i$  and  $\text{Val}_{ij}(y) \leq p_{ij}$ . We use this to partition the freed-up budget  $bB_i$  among all items in  $U_i$  such that item  $j' \in U_i$  is attributed an expected contribution of  $b\text{Val}_{ij'}(y)$ . Since  $\sum_{j' \in U_i} \text{Val}_{ij'}(y) \leq \sum_{j'} \text{Val}_{ij'}(y) \leq B_i$ , the total expected contribution of  $U_i$  is at most  $bB_i$ . Furthermore, since  $\text{Val}_{ij'}(y) \leq p_{ij'}$ , no item has expected contribution that is bigger than its valuation.

Suppose that after the primary assignment an  $a$ -fraction of  $B_i$  is free and a  $b$ -fraction of  $B_{i'}$  is free. As  $j$  is secondarily assigned to players  $i$  and  $i'$  uniformly at random, the expected contribution of  $j$  due to the secondary assignment is

$$\frac{a\text{Val}_{ij}(y) + b\text{Val}_{i'j}(y)}{2}.$$

Without loss of generality, we can assume that  $\text{Val}_{ij}(y) \geq \text{Val}_{i'j}(y)$ , so

$$\frac{a\text{Val}_{ij}(y) + b\text{Val}_{i'j}(y)}{2} \geq \frac{a + b}{2} \text{Val}_{i'j}(y).$$

In expectation, at least a  $\frac{\delta(1-\delta)}{5}$ -fraction of the total budget  $B_i + B_{i'}$  is freed up, so  $\mathbb{E}[a + b] \geq \frac{2\delta(1-\delta)}{5}$ . Hence, the expected contribution of  $j$  during the secondary assignment is at least

$$\frac{\delta(1 - \delta)}{5} \text{Val}_{i'j}(y).$$

For every pair of configurations  $\mathcal{C}$  and  $\mathcal{C}'$  such that  $j \notin \mathcal{C} \cup \mathcal{C}'$ , we have a lower bound on the expected contribution of  $j$  during the secondary assignment. Since  $j \in H$ , players  $i$  and  $i'$  pick such  $\mathcal{C}$  and  $\mathcal{C}'$  with probability at least  $\delta(1 - \delta)$ , so the expected contribution of  $j$  due to secondary assignment is at least

$$\frac{\delta^2(1 - \delta)^2}{5} \text{Val}_{i'j}(y).$$

Let  $\rho = x_{ij}$ . We have  $\text{Val}_j(y) = \rho \text{Avg}_{ij}(y) + (1 - \rho) \text{Avg}_{i'j}(y)$  and  $\text{Avg}_{ij}(y) \leq (1 + \gamma) \text{Avg}_{i'j}(y)$ , where  $i' = \min(j)$ , since  $j \in H$ . Therefore,

$$\text{Val}_j(y) \leq \rho(1 + \gamma) \text{Avg}_{i'j}(y) + (1 - \rho) \text{Avg}_{i'j}(y) \leq (1 + \rho\gamma) \text{Avg}_{i'j}(y).$$

It follows that  $\text{Val}_j(y)/(1 + \rho\gamma) \leq \text{Avg}_{i'j}(y)$ . Since  $\text{Val}_{i'j}(y) = (1 - \rho) \text{Avg}_{i'j}(y)$ ,

$$\text{Val}_{i'j}(y) \geq \frac{\text{Val}_j(y)(1 - \rho)}{1 + \rho\gamma}.$$

We have  $\rho \approx 1/2$  and  $\gamma$  is a small constant, so  $\text{Val}_{i'j}(y) \geq \text{Val}_j(y)/3$ . Therefore, the expected contribution of  $j$  due to secondary assignment is at least

$$\frac{\delta^2(1 - \delta)^2}{5} \text{Val}_{i'j}(y) \geq \frac{\delta^2(1 - \delta)^2}{15} \text{Val}_j(y).$$

By Corollary 2, the contribution due to the primary assignment is at least  $(3/4) \text{Val}_j(y)$ , so the total expected contribution of  $j$  is at least

$$\left( \frac{3}{4} + \frac{\delta^2(1 - \delta)^2}{15} \right) \text{Val}_j(y).$$

The lemma follows from

$$\mathbb{E}[\text{Alg}] \geq \sum_j \left( \frac{3}{4} + \frac{\delta^2(1 - \delta)^2}{15} \right) \text{Val}_j(y) \geq \left( \frac{3}{4} + \frac{\delta^2(1 - \delta)^2}{15} \right) \text{Val}(y).$$

□

We are now ready to prove Theorem 4.

*Proof of Theorem 4* Let us first assume that at least an  $\varepsilon$ -fraction of the LP value of the original solution  $y$  comes from items in  $I$ , i.e.,

$$\sum_{j \in I} \text{Val}_j(y) \geq \varepsilon \text{Val}(y).$$

Pick  $\delta < 1/2$  and by Corollary 1, the expected contribution of item  $j \in I$  is at least  $(3/4 + \varepsilon') \text{Val}_{ij}(y)$  for some  $\varepsilon' > 0$ . Corollary 1 implies that items in  $H$  have expected contribution at least  $(3/4) \text{Val}_{ij}(y)$ , since  $\rho(1 - \rho) \leq 1/4$ .

Therefore, the expected value of the solution returned by Algorithm 2 is at least

$$\begin{aligned} \sum_{j \in I} \left(\frac{3}{4} + \varepsilon'\right) \text{Val}_j(y) + \sum_{j \in H} \frac{3}{4} \text{Val}_j(y) &= \sum_{j \in Q} \frac{3}{4} \text{Val}_j(y) + \sum_{j \in I} \varepsilon' \text{Val}_j(y) \\ &\geq \sum_{j \in Q} \left(\frac{3}{4} + \varepsilon\varepsilon'\right) \text{Val}_j(y) \\ &= \left(\frac{3}{4} + \varepsilon\varepsilon'\right) \text{Val}(y) \end{aligned}$$

which proves the theorem in the case when at least an  $\varepsilon$ -fraction of  $\text{Val}(y)$  comes from items in  $I$ .

Let us then consider the case where less than an  $\varepsilon$ -fraction of the LP value comes from items in  $I$ . We can preprocess the solution  $y$  into a solution  $y'$  that does not contain items in  $I$  by removing all such items from any configuration in the support of  $y$  as well as from the original instance. It follows that the LP value of  $y'$  is at least an  $(1 - \varepsilon)$ -fraction of the LP value of  $y$ ,

$$\text{Val}(y') \geq (1 - \varepsilon)\text{Val}(y).$$

We then run Algorithm 3 on  $y'$ . By Lemma 10,  $\mathbb{E}[\text{Alg}] \geq \left(\frac{3}{4} + \frac{\delta^2(1-\delta)^2}{15}\right) \text{Val}(y')$ , so Algorithm 3 recovers at least a value of

$$\left(\frac{3}{4} + \frac{\delta^2(1-\delta)^2}{15}\right) \text{Val}(y') \geq (1 - \varepsilon) \left(\frac{3}{4} + \frac{\delta^2(1-\delta)^2}{15}\right) \text{Val}(y),$$

where the last inequality follows from  $\text{Val}(y') \geq (1 - \varepsilon)\text{Val}(y)$ .

The theorem follows by choosing appropriate  $\delta$  and  $\varepsilon$ : fix any  $\delta > 0$ , then  $\left(\frac{3}{4} + \frac{\delta^2(1-\delta)^2}{15}\right) = 3/4 + \eta$ , where  $1/100 > \eta > 0$ . Let  $\varepsilon = \eta/2$ , then  $(1 - \varepsilon)(3/4 + \eta) \geq 3/4 + \eta/2 - \eta^2/2 > 3/4$ , since  $\eta(1 - \eta)/2 > 0$ . □

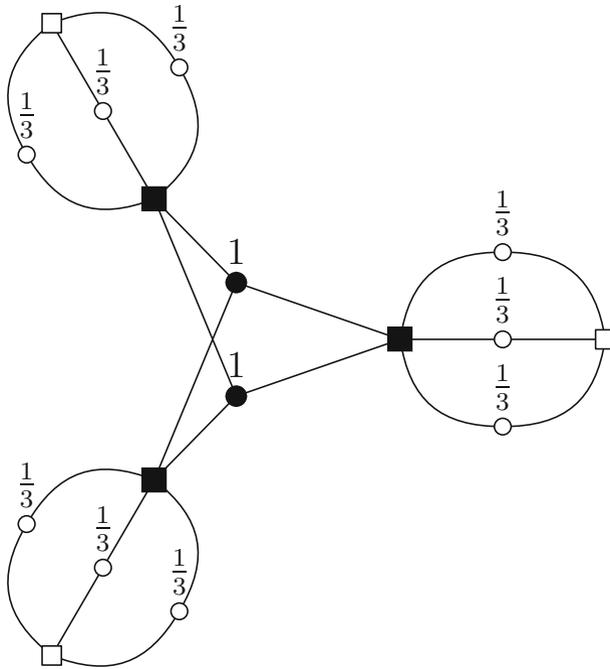
### 5 An improved integrality gap for the unrestricted case

The previously best known upper bound on integrality gap of the configuration LP was  $5/6 = 0.833$  proved by Chakrabarty and Goel in [6]. We improve this to approximately 0.828. Unlike the previous result, our gap instance is not a gap instance.

**Theorem 5** *The integrality gap of the configuration LP is at most  $2(\sqrt{2}-1) \approx 0.828$ .*

*Proof* For  $p, q \in \mathbb{N}$  such that  $p < q$ , consider the following MBA instance: there are  $q$  players  $l_i \in L$  for  $1 \leq i \leq q$  with budget 1 and  $q$  players  $s_i \in S$  for  $1 \leq i \leq q$  with budget  $\frac{p}{q}$ . Additionally, there are  $p$  items  $c_j \in R$  for  $1 \leq j \leq p$ , each of which can be assigned to players in  $L$  with a value of 1. Finally, for each player  $l_i \in L$ , there are  $q$  items  $o_{ij} \in O_i$  for  $1 \leq j \leq q$ , which can be assigned to  $l_i$  and  $s_i$  with a value of  $\frac{1}{q}$ .

An example with  $p = 2$  and  $q = 3$  is drawn in Fig. 3.



**Fig. 3** An instance with  $p = 2$  and  $q = 3$ . Black squares are players in  $L$ , white squares players in  $S$ . They have budgets 1 and  $2/3$  respectively. Items in  $R$  are black dots and items in  $O_i$  are white dots with their values written next to them. An edge between a player and an item denotes that the player is interested in that item

The optimal integral solution assigns  $p$  items from  $R$  to  $p$  distinct players in  $L$ ; for each player  $i$  that is assigned an item from  $R$ , we assign  $p$  items from  $O_i$  to  $s_i$ . Let  $i'$  be one of the  $q - p$  players which do not get an item from  $R$ , the optimal integral solution assigns the  $q$  items from  $O_{i'}$  to  $i'$ . The total value of the solution is  $p(1 + \frac{p}{q}) + q - p = \frac{p^2}{q} + q$ .

Consider the following fractional solution to the configuration LP. Every item in  $R$  is shared by the  $q$  players in  $L$ , each with a fraction of  $\frac{1}{q}$ . Furthermore, every player  $i$  in  $L$  is assigned a fraction  $\frac{q-p}{q}$  of every item in  $O_i$ . More precisely, the configuration  $C = O_i$  has  $y_{iC} = \frac{q-p}{q}$ , so the budget of  $i$  is completely filled.

Finally, every player  $i$  in  $S$  uses the unassigned fraction  $\frac{p}{q}$  of every item in  $O_i$  to form  $\binom{q}{p}$  configurations of size  $p$ , which fill up the budget of  $i$  completely. Hence, the value of the fractional solution is  $q + q\frac{p}{q} = q + p$ . Note that the total value of items is  $p + q$ , so there can not be a better assignment.

Hence, the integrality gap  $I(p, q)$  is

$$I(p, q) = \frac{\frac{p^2}{q} + q}{p + q} = \frac{\frac{p^2}{q^2} + 1}{\frac{p}{q} + 1}.$$

For  $x \in \mathbb{R}$ , the expression  $f(x) = \frac{x^2+1}{x+1}$  is minimized at  $x^* = \sqrt{2} - 1$  and has  $f(x^*) = 2(\sqrt{2} - 1)$ . Hence, choosing  $p, q$  such that  $\frac{p}{q}$  is arbitrarily close to  $\sqrt{2} - 1$ , we can achieve an integrality gap arbitrarily close to  $2(\sqrt{2} - 1)$ .  $\square$

## 6 Hardness of approximation

In this section we strengthen the known hardness results. First we prove hardness 59/60 in the graph case and then we prove that the known 15/16-hardness holds also for restricted MBA where players have the same budget.

Previously Chakrabarty and Goel in [6] proved hardness of approximation to within a factor of 15/16 by a reduction from MAX-3-LIN(2). Remember that MAX-3-LIN(2) is the following problem: given  $m$  linear equations modulo 2 over  $n$  variables, each equation containing 3 variables, find an assignment of variables to maximize the number of satisfied constraints.

We modify this construction to use MAX-2-LIN(2) which is the same problem, with the difference being that each equation contains 2 variables. The important change is that we create items for assignments that do not satisfy an equation, while the previous construction used satisfying assignments. The use of equations of size 2 implies hardness for the graph case, i.e., where each item can only be assigned to two players.

**Theorem 6** *For every  $\epsilon > 0$ , it is NP-hard to approximate graph MBA within a factor of  $59/60 + \epsilon$ . Furthermore, this is true for the restriction of the problem where all players have the same budget.*

*Proof* We reduce from an instance  $\phi$  of MAX-2-LIN(2). Let  $x$  be a variable occurring  $\deg(x)$  times in  $\phi$ . We have two players  $\langle x : 1 \rangle$  and  $\langle x : 0 \rangle$  both with budgets  $\deg(x)$  and an item of value  $\deg(x)$  that can only be assigned to these two players. The meaning of this item is that if it is assigned to the player  $\langle x : a \rangle$ , then a truth assignment  $\alpha$  has  $\alpha(x) = a$ .

We use the notation  $\langle x : a_1, y : a_2 \rangle$  to denote items of our graph MBA instance, where  $x$  and  $y$  corresponds to variables of the MAX-2-LIN(2) instance we reduce from, and  $a_1, a_2 \in \{0, 1\}$ . Specifically, for each equation  $x + y = b$ , there are two items  $\langle x : a_1, y : a_2 \rangle$  of value 1. Each such item corresponds to an assignment  $\alpha$  for which  $\alpha(x) = a_1, \alpha(y) = a_2$  and  $a_1 + a_2 \neq b$ . An item  $\langle x : a_1, y : a_2 \rangle$  can be assigned to  $\langle x : c \rangle$  only if  $c = a_1$  and to  $\langle y : d \rangle$  only if  $d = a_2$ .

Every item can only be assigned to two players, so this is a graph instance. Furthermore, the valuation for both players is the same, so it is the restricted case.

The analysis is now very similar to the one in [6]. We can prove that an optimal assignment of items always assigns items that have weight  $\deg(x)$ , for some  $x$ , and this can be translated into a truth assignment  $\alpha$  to variables. We have  $\alpha(x) = a$  if an item of value  $\deg(x)$  is assigned to  $\langle x : a \rangle$ . If  $\alpha$  satisfies  $x + y = b$ , i.e.,  $\alpha(x) + \alpha(y) = b$ , we can assign both items  $\langle x : a_1, y : a_2 \rangle$ . Otherwise we can only assign one of them, since the budgets of  $\langle x : \alpha(x) \rangle$  and  $\langle y : \alpha(y) \rangle$  are fully assigned. So if  $\phi$  is  $\delta$ -satisfiable (i.e., there is a truth assignment that satisfies a  $\delta$ -fraction of the equations) with  $m$  equations, the MBA instance has objective value  $\sum_x \deg(x) + m(2\delta + (1 - \delta)) = 3m + \delta m$ .

Håstad [13], using techniques from Trevisan et al. [21], proved that it is NP-hard to distinguish instances of MAX-2-LIN(2) that are at least  $(3/4 - \epsilon)$ -satisfiable and those that are at most  $(11/16 + \epsilon)$ -satisfiable. Hence it is hard to distinguish between an instance of MBA with objective value at least  $3m + \frac{3}{4}m - \epsilon m = m(60/16 - \epsilon)$  and at most  $3m + \frac{11}{16}m + \epsilon m = m(59/16 + \epsilon)$ , where  $m$  is the number of equations in  $\phi$ .

Therefore graph MBA is NP-hard to approximate to within a factor of  $59/60 + \epsilon$ . The instance from [21] can also be modified to be regular, i.e., with all degrees  $\text{deg}(x)$  being the same, thus producing an instance with equal budgets. In order to achieve regularity, one has to take a closer look at the reduction behind the APX-hardness result of MAX-2-LIN(2). This reduction can be split into two steps: the first one is the PCP reduction from the label cover problem to MAX-3-LIN(2), and the second one is the gadget reduction from MAX-3-LIN(2) to MAX-2-LIN(2). Formally, instances of the label cover problem are described as follows: we are given a bipartite graph  $G = (U \cup V, E)$ , alphabets  $[W_U]$  and  $[W_V]$ , and a projection  $f_e : [W_U] \rightarrow [W_V]$  associated with every edge  $e$  of the graph. The label cover problem is the problem of choosing a labeling  $(l_U, l_V)$  of the vertices of  $G$ , where  $l_U$  and  $l_V$  are functions such that  $l_U : U \rightarrow [W_U]$  and  $l_V : V \rightarrow [W_V]$ , such that the number of satisfied edges (i.e., edges  $e = (u, v) \in U \times V$  for which it holds that  $f_e(l(u)) = l(v)$ ) is maximized. For any constant  $\epsilon > 0$ , it is known [16] that it is NP-hard to distinguish, given an instance of label cover, whether there is a labeling that satisfies all the edges of the graph, or there is no labeling that satisfies more than an  $\epsilon$ -fraction of the edges; furthermore, Raz’s parallel repetition [16] can produce instances that are left and right regular, i.e., all vertices in  $U$  have degree  $d_l$  and all vertices in  $V$  have degree  $d_r$  (we can achieve this by using a regular NP-hard constraint satisfaction problem (CSP) as the starting problem of parallel repetition, i.e., a CSP where every variable appears a fixed and constant amount of times).

Now, Håstad’s  $(1/2 + \epsilon)$ -hardness result for MAX-3-LIN(2) [13] can be seen as a PCP reduction from the label cover problem to MAX-3-LIN(2). One nice property of this reduction the following: if we take a look at the instances of MAX-3-LIN(2) this reduction produces, then we can partition the variables of any such instance into two sets  $L$  and  $R$ , such that:

- every clause contains one variable from  $L$  and two variables from  $R$ .
- all the variables in  $L(R)$  appear in the same number  $D_L(D_R)$  of clauses.

Due to this property, we can actually produce instances of MAX-3-LIN(2) that are regular, i.e., all variables appear in the same number of clauses, using a technique from [20]. What we do is essentially to create a modified instance  $I'$  of MAX-3-LIN(2), by splitting each variable of the original instance  $I$  into more variables.  $I'$  consists of  $D_L D_R$  copies of  $I$ , where for each variable  $x$  of  $L$ , we introduce  $D_L$  copies  $x_1 \dots x_{D_L}$ , each of which is substituted for  $x$  in  $D_R$  copies of  $I$ . Similarly, for each variable  $y$  of  $R$ , we introduce  $D_R$  copies  $y_1 \dots y_{D_R}$ , each of which is substituted for  $y$  in  $D_L$  copies of  $I$ . It is easy to see that in  $I'$ , every variable appears in exactly  $D_L D_R$  clauses; furthermore, since  $I'$  only contains copies of  $I$ , it is easy to deduce that, for any  $\epsilon > 0$ , the following is true:

- if there is an assignment of truth values to the variables of  $I'$  that satisfies at least an  $(1 - \epsilon)$ -fraction of the clauses, the same is true for  $I$ .
- if there is no assignment of truth values to the variables of  $I'$  that satisfies more than an  $(1/2 + \epsilon)$ -fraction of the clauses, the same is true for  $I$ .

In order to produce regular instances for MAX-2-LIN(2), we apply the gadget reduction of Håstad [13] to an instance of MAX-3-LIN(2) constructed as above. Due to the construction of the gadget, any instance  $I$  with  $n$  variables and  $m$  clauses that is output will have the following property: all the variables of  $I$  can be split into two sets  $L$  and  $R$ , such that:

- every clause contains one variable from  $L$  and one variable from  $R$ .
- all the variables in  $R$  appear in the same number  $D_R$  of clauses.
- there is one variable in  $L$  that appears in  $m/4$  clauses; the rest of the variables in  $L$  appear in the same number  $D_L$  of clauses.

Hence, for any  $\epsilon > 0$ , and using an approach similar to the one we took for MAX-3-LIN(2), we can transform  $I$  into a regular instance  $I'$ , such that:

- if there is an assignment of truth values to the variables of  $I'$  that satisfies at least a  $(3/4 - \epsilon)$ -fraction of the clauses, the same is true for  $I$ .
- if there is no assignment of truth values to the variables of  $I'$  that satisfies more than an  $(11/16 + \epsilon)$ -fraction of the clauses, the same is true for  $I$ .

Combined with our own reduction from MAX-2-LIN(2) to graph MBA, our hardness result extends to the restricted version of graph MBA in which all players have the same budget.  $\square$

Let us now give a proof of hardness for the restricted MBA. The construction is the same as the one of Chakrabarty and Goel in [6] but we reduce from a MAX-3-LIN(2) instance with stronger properties.

**Theorem 7** *For every  $\epsilon > 0$ , it is NP-hard to approximate restricted MBA within a factor of  $15/16 + \epsilon$ . Furthermore, this is true for instances where all items can be assigned to at most 3 players.*

*Proof* Chakrabarty and Goel in [6] prove the  $(15/16 + \epsilon)$ -hardness for MBA instances where all items can be assigned to at most 3 players, and they also have uniform prices, i.e., for all items  $j$  there is a  $p_j$  such that for all players  $i$ ,  $p_{ij} \in \{p_j, 0\}$ . They achieve this by reducing MAX-3-LIN(2) problem to MBA. The MAX-3-LIN(2) problem was proved to be NP-hard to approximate within a factor of  $1/2 + \epsilon$  by Håstad in [13].

Their reduction is the following: given an instance  $\phi$  of MAX-3-LIN(2) with  $n$  variables and  $m$  equations, we construct an instance of restricted MBA.

- For each variable  $x_i$  of the original instance, we introduce players  $\langle x_i, 0 \rangle$  and  $\langle x_i, 1 \rangle$ . The budget of these players is  $4 \deg(x_i)$ .
- For each variable  $x_i$  of the original instance, we introduce item  $s_i$ ;  $s_i$  can be assigned to players  $\langle x_i, 0 \rangle$  and  $\langle x_i, 1 \rangle$ , and its price for both of them is  $4 \deg(x_i)$ .
- For every equation  $e : x_i + x_j + x_k = a$  ( $a \in \{0, 1\}$ ) of the original instance, we introduce 3 copies of 4 items, where each of the 4 items corresponds to one of the

satisfying assignments of  $e$ . For each one of these items, let  $a_1, a_2$  and  $a_3$  be the truth values of the corresponding satisfying assignment of  $e$ : then, this item can be assigned to players  $\langle x_i, a_i \rangle, \langle x_j, a_j \rangle$  and  $\langle x_k, a_k \rangle$ , for a price of 1.

Chakrabarty and Goel prove that if the original MAX-3-LIN(2) instance is at least  $(1 - \epsilon)$ -satisfiable, then there exists an assignment of items to players in the reduced instance of value at least  $24m - 12m\epsilon$ ; conversely, they prove that if the original instance is at most  $1/2 + \epsilon$ -satisfiable, then any assignment of items to players in the reduced instance has value at most  $22.5m + 3m\epsilon$ .

We use the same proof but use a different starting point. By re-using the techniques we used in the proof of hardness for graph MBA, the result of Håstad can be modified so that each variable in the MAX-3-LIN(2) instance has the same degree.

The construction of Chakrabarty and Goel gives budget  $4 \deg(x_i)$  to the 2 players corresponding to variable  $x_i$ . Hence, if all variables have the same degree, all players have the same budget, and the result of Chakrabarty and Goel carries over.  $\square$

## 7 Conclusion and future directions

We showed that the integrality gap of the configuration LP is strictly better than  $\frac{3}{4}$  for two interesting and natural restrictions of maximum budgeted allocation: restricted and graph MBA. These results imply that the configuration LP is strictly better than the natural assignment LP and pose promising research directions. Specifically, our results on restricted MBA suggest that our limitations in rounding configuration LP solutions do not necessarily stem from the items being fractionally assigned to many players, while our results on graph MBA suggest that they do not necessarily stem from the items having non-uniform prices. Whether these limitations can simultaneously be overcome is left as an interesting open problem.

Finally, it would be interesting to see whether the techniques presented, and especially the exploitation of the big items structure, can be applied to other allocation problems with similar structural features as MBA (e.g. GAP).

**Acknowledgments** Many thanks go to Johan Håstad for finding an error in the proof of Theorem 4 in an earlier version of this paper. The authors would also like to thank the anonymous referees for many comments on improving the readability of this paper.

## References

1. Andelman, N., Mansour, Y.: Auctions with budget constraints. In: SWAT, pp. 26–38 (2004)
2. Asadpour, A., Feige, U., Saberi, A.: Santa claus meets hypergraph matchings. In: APPROX-RANDOM, pp. 10–20. Springer (2008)
3. Azar, Y., Birnbaum, B.E., Karlin, A.R., Mathieu, C., Nguyen, C.T.: Improved approximation algorithms for budgeted allocations. IICALP 1, 186–197 (2008)
4. Bansal, N., Sviridenko, M.: The santa claus problem. In: Kleinberg, J.M. (ed.) STOC, pp. 31–40. ACM (2006)
5. Chakrabarty, D., Chuzhoy, J., Khanna, S.: On allocating goods to maximize fairness. In: 50th Annual IEEE Symposium on Foundations of Computer Science, FOCS 2009, October 25–27, 2009, Atlanta, Georgia, USA, pp. 107–116 (2009)

6. Chakrabarty, D., Goel, G.: On the approximability of budgeted allocations and improved lower bounds for submodular welfare maximization and gap. *SIAM J. Comput.* **39**(6), 2189–2211 (2010)
7. Ebenlendr, T., Krčál, M., Sgall, J.: Graph balancing: a special case of scheduling unrelated parallel machines. In: *SODA*, pp. 483–490. Society for Industrial and Applied Mathematics (2008)
8. Feige, U.: On allocations that maximize fairness. In: *SODA*, pp. 287–293. SIAM (2008)
9. Feige, U., Vondrák, J.: Approximation algorithms for allocation problems: improving the factor of  $1-1/e$ . In: *FOCS*, pp. 667–676 (2006)
10. Gandhi, R., Khuller, S., Parthasarathy, S., Srinivasan, A.: Dependent rounding and its applications to approximation algorithms. *J. ACM* **53**(3), 324–360 (2006)
11. Garg, R., Kumar, V., Pandit, V.: Approximation algorithms for budget-constrained auctions. In: *APPROX-RANDOM*, pp. 102–113. Springer (2001)
12. Grötschel, M., Lovász, L., Schrijver, A.: *Geometric Algorithms and Combinatorial Optimization, Volume 2 of Algorithms and Combinatorics*. Springer, Berlin (1993)
13. Håstad, J.: Some optimal inapproximability results. *J. ACM* **48**(4), 798–859 (2001)
14. Lehmann, B., Lehmann, D.J., Nisan, N.: Combinatorial auctions with decreasing marginal utilities. *Games Econ. Behav.* **55**(2), 270–296 (2006)
15. Lenstra, J.K., Shmoys, D.B., Tardos, É.: Approximation algorithms for scheduling unrelated parallel machines. *Math. Program.* **46**, 259–271 (1990)
16. Raz, R.: A parallel repetition theorem. *SIAM J. Comput.* **27**(3), 763–803 (1998)
17. Shmoys, D.B., Tardos, É.: An approximation algorithm for the generalized assignment problem. *Math. Program.* **62**, 461–474 (1993)
18. Srinivasan, A.: Budgeted allocations in the full-information setting. In: *APPROX-RANDOM*, pp. 247–253 (2008)
19. Svensson, O.: Santa claus schedules jobs on unrelated machines. *SIAM J. Comput.* **41**(5), 1318–1341 (2012)
20. Trevisan, L.: Non-approximability results for optimization problems on bounded degree instances. In: *STOC*, pp. 453–461 (2001)
21. Trevisan, L., Sorkin, G.B., Sudan, M., Williamson, D.P.: Gadgets, approximation, and linear programming. *SIAM J. Comput.* **29**(6), 2074–2097 (2000)
22. Verschae, J., Wiese, A.: On the configuration-LP for scheduling on unrelated machines. In: *Algorithms-ESA*, pp. 530–542. Springer (2011)