

Perpetually Fair Assignments Via Balanced Sequences of Permutations

Terrence Adams¹ and Erel Segal-Halevi²

¹ University at Albany, State University of New York, 1400 Washington Avenue,
Albany, NY 12222. terry@ieee.org

² Ariel University, Ariel 40700, Israel. erelsgl@gmail.com

Abstract. There is a set of n indivisible items (goods or chores), and a set of n players. Each day, a single item should be assigned to each player. Assignments based on latin squares guarantee fairness after every n days; our goal is to ensure fairness after every single day.

We present two 'balance' conditions on latin squares. Informally, a latin square is balanced if its top rows and leftmost columns contain all n labels; this ensures that all n players receive one of the top items in one of the early days. One such condition can always be satisfied, but is arguably too weak; a second condition is strong, and can be satisfied for all $n \leq 12$, but cannot be satisfied for some larger values of n , including all $n > 108$.

We show that the second balance condition guarantees that the cumulative assignment is always *proportional up to one item (PROP1)*, where proportionality holds in a strong ordinal sense — for every valuations that are consistent with the item ranking.

Finally, we present a weaker balance condition on a sequence, that guarantees ordinal proportionality up to two items (PROP2). Whether or not this condition can be satisfied for all n remains an open question.

This is the full version of a paper accepted to SAGT 2026 conference.

Keywords: Fair division · Repeated assignment · latin squares · Proportionality · Balanced sequences

1 Introduction

Some n house-mates need to divide the n daily house-chores among them. Each housemate must do exactly one chore per day. What would be a fair way to assign the chores? Ideally, a fair allocation would give each housemate the same amount of work. But this might be impossible, as some chores are easier and some are harder; in every single day, some housemates must do the hard chores. However, if the same housemates do the same unwanted chores day after day, they will rightfully complain that this *repeated assignment* is unfair.

A similar problem may arise in different contexts. For example, suppose grandpa wants to divide sweets to his grandchildren each weekend. He has to decide who gets the sweets first, who is second, and so on. Each week, the children

who are last to get their sweets might feel somewhat unhappy; but if the same children are last in the line week after week, they will naturally feel that the allocation is unfair. Similar considerations are applicable when assigning seats in classroom to pupils — a task which may be repeated in different terms.

In all these settings, the same assignment problem repeats many times. Whereas some inequality can be tolerated in every single assignment, we expect it to eventually “cancel out” as the same process repeats long enough. We call this problem *Fair Repeated Assignment*.

In Section 2 below we define the problem formally. In the following sections 3–7 we present our main results on existence and non-existence of fair repeated assignments. In Section 8 we discuss various other fair allocation settings studied in the literature and compare them to our setting.

2 Setting

There is a set N of n players, $N := \{A_1, \dots, A_n\}$. There is a set M of n indivisible items, $M := [n] = \{1, \dots, n\}$. We assume that 1 is the best item (e.g. the easiest chore), 2 is the second-best item, etc. This does not lose much generality, as instead of dividing the actual items, we can divide positions in a sequence by which the players pick items, so that the player who receives “1” is the first to pick an item, the player who receives “2” picks second, etc. Clearly, an earlier position in the picking sequence is better for each player.³

An *assignment* is a bijection between M and N ; equivalently, a permutation of M . A *repeated assignment* is a finite sequence of assignments. We say that each assignment is implemented in a different *day*. The multiset of items that a particular player i has after a repeated assignment is called a *bundle*. The bundle of i after day t is denoted Z_i^t . The multiset of all items allocated up to day t is denoted $Z_N^t := \uplus_{i \in N} Z_i^t$ = the items $1, \dots, n$, each with multiplicity t .

We often present a repeated assignment in a table in which each column represents a day and each row represents an item. The number in column t and row j represents the player who receives item j in day t ; see Table 1.

For any bundle Z and any $k \leq |Z|$, we denote by $Z[k]$ the k -th highest valued item in Z , so that $Z[1]$ is the best item in Z , $Z[2]$ the second-best, etc. The matching between the players and their bundles is called an *allocation*.

3 Fairness After n Days: latin square

We are looking for a formal condition that captures the intuitive notion of “fairness”. First, suppose that we are now at the end of day n , and we consider the aggregate bundle of n items that is held by each player. We define the allocation at that point as “fair” if and only if each player has exactly one copy of each

³ This is essentially the “Identical Order Transformation” introduced by Bouveret and Lemaître [12]. Cookson et al. [23] prove that this transformation preserves the approximate proportionality condition PROP1.

1	6	5	4	3	2
2	1	6	5	4	3
3	2	1	6	5	4
4	3	2	1	6	5
5	4	3	2	1	6
6	5	4	3	2	1

1	8	5	4	3	6	7	2
2	7	6	3	4	5	8	1
3	6	7	2	1	8	5	4
4	5	8	1	2	7	6	3
5	4	1	8	7	2	3	6
6	3	2	7	8	1	4	5
7	2	3	6	5	4	1	8
8	1	4	5	6	3	2	7

Table 1. Examples of permutation sequences (repeated assignments) of 6 items (left) and 8 items (right). Each column represents an assignment in a single day; each row represents the players who get a single item throughout all days.

item. It is easy to agree that this condition is sufficient for fairness, as it implies that all players have exactly the same bundle. We claim that it is also necessary. This is because we make no assumptions on how much exactly item 1 is better than item 2, 2 is better than 3, etc. Hence, we cannot assume e.g. that a player who received no copy of item 1, can be compensated by receiving two copies of item 2. Only allocations that give each item to each player exactly once are guaranteed to be fair.

This condition is satisfied in both examples in Table 1, where each row is a permutation of $[n]$. As each column is also a permutation of $[n]$, the entire repeated assignment table is a *latin square*. There are many ways to construct latin squares, so this requirement is easy to satisfy.

Clearly, in fewer than n days, it is impossible to give all players exactly the same bundle. Suppose item 1 is very valuable (a diamond) and the other items are much less valuable (stones). Then at any day before the n -th day, some players (those who did not get the diamond so far) will naturally feel that they did not get their fair share. We could tell these players to wait patiently until day n ; but n days may be too much to wait. We would like the allocation to be “as fair as possible” after each day — not only after day n .

Below, we explore different ways to define what this fairness means.

4 Fairness After Every Day: Top-Balance

A natural fairness requirement is to require that the top items are allocated among the players in a “balanced” way: in the first two days, each player should get at least one item from the top half (the highest-valued $\lceil n/2 \rceil$ items); in the first three days, each player should get at least one item from the top third (the highest-valued $\lceil n/3 \rceil$ items); and so on.⁴ We now state this succinctly.

⁴ The top-balance condition is asymmetric with respect to time, as it assesses fairness based only on the first t days, and not on the last t days or any other t consecutive days. This asymmetry makes sense when the allocated items are physical items that the players keep, so after t days, each player owns a bundle of t items, which should be

Definition 1. A permutation sequence is called *top-balanced* if after every day $t \in [n]$, every player i has at least one item from the top $\lceil n/t \rceil$. Formally,

$$Z_i^t[1] \leq \lceil n/t \rceil.$$

In terms of the latin square, top-balance means that, for all $t \in [n]$, the top-left rectangle with t columns and $\lceil n/t \rceil$ rows must contain all integers $1, \dots, n$. The condition is violated by the sequences of Table 1: for $n = 6$ and $t = 2$, the top-left 3×2 rectangle does not contain 5; for $n = 8$ and $t = 3$, the top-left 3×3 rectangle does not contain 4. Are there other sequences that satisfy it? The answer is yes, and moreover, the condition can be satisfied in combination with the latin square requirement.

Theorem 1. For every positive integer n , there exists an $n \times n$ latin square that is also top-balanced.

The proof uses two known results about completing incomplete latin squares. A *latin rectangle* is a rectangle with n rows and $t < n$ columns, such that each integer in $\{1, \dots, n\}$ appears exactly once in each column, and at most once in each row. Marshall Hall [28], using Philip Hall’s marriage theorem, proved that any latin rectangle can be extended, one column at a time, to produce a latin square. In our setting, columns correspond to days; we will construct a top-balanced latin-square sequence one day at a time. We will also use Theorem 4 from [26] to iteratively extend columns from a partial latin rectangle to a full latin rectangle. For convenience, we restate it here as Lemma 1.

Lemma 1 (Column completion lemma, Theorem 4 in [26]). Let L be an $n \times t$ latin rectangle where $t < n$.

Suppose that some ℓ distinct labels are entered into some ℓ distinct rows in column $t+1$. If we can insert the $n - \ell$ remaining labels into the $n - \ell$ remaining rows such that the outcome is an $n \times (t+1)$ latin rectangle, then we say that column $t+1$ is *completable*.

The column is *completable* if and only if any subset I of the remaining labels is contained in at most $(n - \ell - |I|)$ of the remaining rows.

Equivalently: For any subset I of remaining labels, there are at least $|I|$ remaining rows that do not contain I .

Remark 1. Since L is a latin rectangle with t columns, any subset I of labels can be contained in at most t rows of L (by the pigeonhole principle). Hence, a sufficient condition for completability is that $n - \ell - |I| \geq t$. Moreover, the condition holds trivially when $|I| > t$, so it is sufficient to check the condition for $|I| \leq t$. Hence, it is sufficient to prove that $n - \ell - t \geq t$.

Proof (of Theorem 1). First, we show how to define the assignment for the first half of the days, namely for columns $t \in \{1, \dots, \lceil \frac{n}{2} \rceil\}$, and for the top portion

sufficiently valuable. A stronger variant of top-balance, that considers simultaneously every sequence of t consecutive days, is briefly discussed at the end of Section 7.

of each of these columns (representing the assignment of the top items). After the top-balance condition is satisfied for all these columns, we show how to append the remaining columns such that the entire sequence is a latin square and top-balanced.

Step 1: Constructing a top-balanced partial latin rectangle. We build the top-balanced sequence and partial latin square inductively in the columns t . We want to establish the following two claims inductively in t .

1. The set of labels in the top- $\lceil \frac{n}{t} \rceil$ rows in columns $1, \dots, t$ contains $[n]$, and
2. The labels in the top- $(\lceil \frac{n}{t} \rceil - 1)$ rows in columns $1, \dots, t$ are all unique.

For the induction base, column $t = 1$ is filled by $1, \dots, n$. Column $t = 2$ is filled by $\lceil \frac{n}{2} \rceil + i$ for $1 \leq i \leq \lfloor \frac{n}{2} \rfloor$, and then by i for $1 \leq i \leq \lceil \frac{n}{2} \rceil$. This construction clearly satisfies both claims.

Suppose we have filled columns $1, \dots, t$ for $2 \leq t < n$ such that claims (1) and (2) hold. We show how to progress to the next column $t + 1$. There are two cases to consider.

Case #1: $\lceil \frac{n}{t+1} \rceil \leq \lceil \frac{n}{t} \rceil - 1$. Then by the induction hypothesis the labels in the top- $\lceil \frac{n}{t+1} \rceil$ rows and first t columns are unique. In column $t + 1$, fill the top $\lceil \frac{n}{t+1} \rceil - 1$ cells by labels not in this collection; This is possible since $(t + 1)(\lceil \frac{n}{t+1} \rceil - 1) < n$. The labels in the top- $(\lceil \frac{n}{t+1} \rceil - 1)$ rows in the first $t + 1$ columns are still unique, so claim (2) is satisfied.

Since $(t + 1)\lceil \frac{n}{t+1} \rceil \geq n$, at most one label does not appear in a top- $\lceil \frac{n}{t+1} \rceil$ row yet; put that label in cell $\lceil \frac{n}{t+1} \rceil$. Now, all n labels appear in a top- $\lceil \frac{n}{t+1} \rceil$ row in the first $t + 1$ columns, so claim (1) is satisfied too.

Case #2: $\lceil \frac{n}{t+1} \rceil = \lceil \frac{n}{t} \rceil$. Then by the induction assumption all labels already appear in a top- $\lceil \frac{n}{t+1} \rceil$ row in columns $1, \dots, t$, so claim (1) is already satisfied. We only have to ensure uniqueness. Take any labels that did not appear in a top- $(\lceil \frac{n}{t+1} \rceil - 1)$ row yet, and put them in the top cells of column $t + 1$. Since $(t + 1)(\lceil \frac{n}{t+1} \rceil - 1) < n$, all labels in one of the top $\lceil \frac{n}{t+1} \rceil - 1$ rows in the first $t + 1$ columns are unique, and claim (2) is satisfied too.

Step 2: Completing the columns to a full latin rectangle. For each $t \in \{1, \dots, \lceil \frac{n}{2} \rceil - 1\}$, we will show how to complete column $t + 1$, such that the rectangle consisting of days $1, \dots, t + 1$ is a latin rectangle. Step 1 already handles $t = 1$, so we can assume $t \geq 2$. In Section 5, we will present a fairness notion stronger than both latin square and top-balance; in Theorem 2, we will show that it can be satisfied for all $n \leq 11$. Hence, we can assume here that $n \geq 12$.

To complete column $t + 1$, we apply Lemma 1 with ℓ being the number of cells filled in Step 1. Note, we have that $\lceil \frac{n}{t+1} \rceil - 1 \leq \ell \leq \lceil \frac{n}{t+1} \rceil$. If $t = 2$,

$$n - \ell - t \geq n - \left\lceil \frac{n}{t+1} \right\rceil - t > n - \left(\frac{n}{3} + 1 \right) - 2 = \frac{2n}{3} - 3 \geq 5 > t,$$

which is sufficient by Remark 1. Similarly, if $t = 3$, then

$$n - \ell - t \geq n - \left\lceil \frac{n}{t+1} \right\rceil - t > n - \left(\frac{n}{4} + 1 \right) - 3 = \frac{3n}{4} - 4 \geq 5 > t.$$

Now, for $n \geq 12$, $t \geq 4$ and $t \leq \left\lceil \frac{n}{t+1} \right\rceil$,

$$n - \ell - t \geq n - 2 \left\lceil \frac{n}{t+1} \right\rceil > \frac{3n}{5} - 2 > \frac{n}{5} + 1 \geq \left\lceil \frac{n}{t+1} \right\rceil \geq t.$$

Again, by Remark 1, Lemma 1 is satisfied and we are able to complete column $t + 1$. Finally, we handle $t > \left\lceil \frac{n}{t+1} \right\rceil$. We consider two cases.

Case #1: n is even (and $n \geq 12$). Recall $\left\lceil \frac{n}{t+1} \right\rceil - 1 \leq \ell \leq \left\lceil \frac{n}{t+1} \right\rceil$. The construction of Step 1 ensures that all n labels appear at least once in the top $\left\lceil \frac{n}{t+1} \right\rceil$ rows of the first $t + 1$ columns. Hence, any set I of labels that do not already appear in the top ℓ rows of column $t + 1$, must appear in the top $\left\lceil \frac{n}{t+1} \right\rceil$ rows of the first t columns at least once. If $|I| = t$, since $t > \left\lceil \frac{n}{t+1} \right\rceil$, there exists an element $a \in I$ such that a appears in the top ℓ rows and the first t columns. Thus, the element a can only appear at most $t - 1$ times in the bottom $n - \ell$ rows. Hence, to apply Lemma 1, it is sufficient to show a slightly weaker condition than that of Remark 1: $n - \ell - |I| \geq t - 1$. As $|I| \leq t$, it is sufficient to show that $n - \ell - t \geq t - 1$.

On the other hand, if $|I| \leq t - 1$, then it is sufficient to show $n - \ell - (t - 1) \geq t$ which also implies it is sufficient to show $n - \ell - t \geq t - 1$.

Substituting an upper bound for ℓ and rearranging terms leads to the following sufficient condition:

$$n - \left(\frac{n}{t+1} + 1 \right) - t - (t - 1) = n - \frac{n}{t+1} - 2t \geq 0.$$

We know for $n \geq 12$ and $t = 1$, $n - \frac{n}{t+1} - 2t > 0$. Solving $n - \frac{n}{t+1} - 2t = 0$ gives $t = \frac{n}{2} - 1$ or $t = 0$ (recall that n is even). By Bolzano's Theorem [9], this implies the Lemma 1 condition holds for $0 \leq t \leq \frac{n}{2} - 1$. Hence, we can complete all columns up to column $\frac{n}{2}$.

Case #2: n is odd (and $n \geq 13$). By the argument of Case #1, we can complete column $t + 1$ for all $t \leq \frac{n}{2} - 1$, which for odd n means $t \leq \frac{n-3}{2}$. Thus, the first $\frac{n-1}{2}$ columns can be completed.

It remains to complete column $\frac{n+1}{2} = \left\lceil \frac{n}{2} \right\rceil$. As $n - 1$ labels have already appeared in the top two rows and the first $\frac{n-1}{2}$ columns, the construction of Step 1 has to put only the single remaining label at the top row of column $\frac{n+1}{2}$, so we can take $\ell = 1$. As $t = \frac{n-1}{2}$, we get

$$n - \ell - t = n - 1 - \frac{n-1}{2} = \frac{n-1}{2} = t.$$

The condition in Remark 1 is satisfied, so column $\frac{n+1}{2}$ can be completed.

Step 3: Completing the rectangle to a full latin square. At this point, we have a top-balanced latin rectangle with $\lceil \frac{n}{2} \rceil$ columns. Apply Hall's theorem [28] to complete the final $\lfloor \frac{n}{2} \rfloor$ columns and obtain a latin square.

Since $\lfloor \frac{n}{2} \rfloor$ labels have not appeared at row 1 in the first $\lceil \frac{n}{2} \rceil$ columns, these labels must appear in the first row in the last $\lfloor \frac{n}{2} \rfloor$ columns to obtain a latin square. This implies that the resulting latin square remains top-balanced. $\square$

5 Stronger Fairness After Every Day: Full-Balance

The top-balance condition considers only the top items for each player. The allocations of other items can be arbitrary, and thus unfair. The condition can be strengthened as follows:

- In the first two days, each player should receive at least one item from the top $\lceil n/2 \rceil$ items — this condition is the same as in top-balance.
- In the first three days, each player should receive at least one item from the top $\lceil n/3 \rceil$ items, *and at least two items from the top $\lceil 2n/3 \rceil$ items* — the latter condition is new.
- In general, in the first t days, for every $j \in \{1, \dots, t\}$, each player should receive at least j items from the highest-valued $\lceil jn/t \rceil$ items.

All these conditions are formalized succinctly below.

Definition 2. A permutation sequence is called fully-balanced if after every day $t \in [n]$, the bundle Z_i^t of every player i satisfies the following for every $j \in [t]$:

$$Z_i^t[j] \leq \lceil jn/t \rceil. \quad (1)$$

We denote by $\text{BAL}(t, j)$ the condition (1) for some particular $t \in [n]$ and $j \in [t]$. So a repeated assignment is fully-balanced if it satisfies $\text{BAL}(t, j)$ for every day $t \in [n]$ and item $j \in [t]$.

Remark 2. (a) $\text{BAL}(t = n, j)$ implies that, after n days, the j -best highest item of each player is item j ; this implies that each player has exactly one copy of each item, which is the Latin-square condition of Section 3.

(b) $\text{BAL}(t, j = 1)$ is exactly the top-balance condition of Section 4.

(c) $\text{BAL}(t = 1, \cdot)$ and $\text{BAL}(t, j = t)$ hold trivially, so in fact we only have to check $t \in \{2, \dots, n\}$ and $j \in \{1, \dots, t-1\}$.

Do fully-balanced sequences exist for all n ?

It is easy to find fully-balanced sequences for $n \in \{3, 4, 5, 6\}$; the readers may want to try it themselves before looking at Table 2. To verify that a permutation sequence is fully-balanced, one should check, for each $t \in [n]$ and $j \in [t]$, that the top-right rectangle with t columns and $\lceil jn/t \rceil$ rows contains each integer in $1, \dots, n$ at least j times. For example, for $n = 6$ and $t = 4$, we check that the 2×4 top-left rectangle contains $1, \dots, 6$ at least once; the 3×4 top-left rectangle contains $1, \dots, 6$ at least twice; and the 5×4 top-left rectangle contains $1, \dots, 6$

1 2 3	1 4 3 2	1 4 2 3 5	1 4 3 5 2 6
2 3 1	2 3 4 1	2 5 3 4 1	2 5 6 4 1 3
3 1 2	3 2 1 4	3 1 4 5 2	3 6 2 1 4 5
	4 1 2 3	4 2 5 1 3	4 1 5 6 3 2
		5 3 1 2 4	5 2 1 3 6 4
			6 3 4 2 5 1

Table 2. Fully-balanced permutation sequences for $n \in \{3, 4, 5, 6\}$.

at least three times. We were able to handcraft sequences for $n \leq 8$. For larger values of n , we implemented a backtracking algorithm, and found such sequences for all $n \leq 11$. Appendix A shows the fully-balanced sequences that we found for $7 \leq n \leq 11$.

Theorem 2. *For all $n \leq 11$, a fully-balanced permutation sequence exists.*

Twelve (12) is the first natural number that does not admit a fully-balanced sequence. Below we show there are infinitely many natural numbers with no fully-balanced sequence, including $n = 12$.

Proposition 1. *There is no fully-balanced sequence of length $n = 6k$ for $k \geq 2$.*

Proof. Let A_i represent the player who goes i^{th} in the first day. Let B_i represent the player who goes i^{th} in the second day; C_i the player who goes i^{th} in the third day and D_i the player who goes i^{th} in the fourth day. We will show that balance must break down within the fourth day. Notice that for $k \geq 2$, $\lceil \frac{6k}{4} \rceil < 2k$. Here is the list of implications that prevent balance:

1. $\{A_i : 3k + 1 \leq i \leq 6k\} = \{B_i : 1 \leq i \leq 3k\}$ by BAL(2, 1). Each player that appears in the second half in the first day must appear in the first half in the second day, since $\lceil \frac{6k}{2} \rceil = 3k$.
2. $\{A_i : 1 \leq i \leq 3k\} = \{B_i : 3k + 1 \leq i \leq 6k\}$. This follows by (1).
3. $\{C_i : 1 \leq i \leq 2k\} = \{A_i : 2k + 1 \leq i \leq 3k\} \cup \{B_i : 2k + 1 \leq i \leq 3k\}$ by BAL(3, 1). Since $\lceil \frac{6k}{3} \rceil = 2k$, each player must go at least once in the top $2k$ positions after the first 3 days. But, $\{A_i : 2k + 1 \leq i \leq 3k\} \cup \{B_i : 2k + 1 \leq i \leq 3k\}$ are $2k$ distinct players that have not gone in the top $2k$ positions after the first 2 days.
4. $\{C_{2k}\} \subset \{D_i : 1 \leq i \leq 3k\}$ by BAL(4, 1). Let $z := \lceil \frac{6k}{4} \rceil$. Since $z < 2k$, then player C_{2k} did not get any of the top z items in any of the first three days. Thus, $C_{2k} \in \{D_i : 1 \leq i \leq z\}$ (Moreover, $\{C_i : z + 1 \leq i \leq 2k\} \subseteq \{D_i : 1 \leq i \leq z\}$).
5. $\{C_i : 3k + 1 \leq i \leq 6k\} \subseteq \{D_i : 1 \leq i \leq 3k\}$ by BAL(4, 2). Each player is in the first $3k$ positions in the first or second day, but not both days. This follows from (1) and (2). Since $\frac{2(6k)}{4} = 3k$, then each player goes in the top $3k$ positions at least twice in the first 4 days. But, each of $C_{3k+1}, \dots, C_{6k}$ have gone once in the first 3 days.

But there is no room for all of $C_{2k}, C_{3k+1}, \dots, C_{6k}$ to go in the top $3k$ positions in the fourth day. That is,

$$|\{C_{2k}, C_{3k+1}, C_{3k+2}, \dots, C_{6k}\}| = |\{D_1, D_2, \dots, D_{3k}\}| + 1$$

which leads to a contradiction.

Proposition 1 can be extended to all n . The proof varies with the remainder of n divided by 6, so there are six different proofs (including Proposition 1). The table below shows the lower bounds for each remainder j in $\{0, 1, 2, 3, 4, 5\}$. The proofs are given in Appendix B. Table 3 shows there are at most finitely

n	$6k$	$6k+1$	$6k+2$	$6k+3$	$6k+4$	$6k+5$
bounds on k	$k \geq 2$	$k \geq 11$	$k \geq 3$	$k \geq 8$	$k \geq 4$	$k \geq 9$
bounds on n	$n \geq 12$	$n \geq 67$	$n \geq 20$	$n \geq 51$	$n \geq 28$	$n \geq 59$

Table 3. Values of $n = 6k + j$ with no fully-balanced sequence

many values n which admit a fully-balanced sequence. In particular, the following theorem follows from Table 3.

Theorem 3. *There exist finitely many n such that there is a fully-balanced sequence of length n . Moreover, there is no fully-balanced sequence for $n > 61$.*

Proof. Since every natural number can be expressed uniquely in the form $n = 6k + j$ for $j = 0, 1, 2, 3, 4, 5$, then we can list the natural numbers that fall under the bounds in Table 3. It is straightforward to check that $n = 61$ is the largest natural number which possibly has a fully-balanced sequence. See the Appendix B for a proof of Table 3 and this theorem.

6 Valuations and Proportionality: Weak Full-Balance

As the concept of a fully-balanced sequence is too strong, we would like to relax it. To do so (and also to link our balance concepts to the literature on fair allocation), we assume that each item $j \in [n]$ has a *value* $v_j \in \mathbb{R}$, such that $v_1 > v_2 > \dots > v_n$.⁵ The total value of any bundle (multiset of items) Z is denoted $v(Z) := \sum_{k \in Z} v_k$.

A natural fairness concept is that each player should receive a bundle of items that is worth at least $1/n$ of the total value, that is, $v(Z_i^t) \geq v(Z_N^t)/n$.⁶ This

⁵ The literature on fair division studies a more general setting, in which different players may have different rankings and different valuations over the items. In Appendix E we prove that our main theorem holds in this more general setting too.

⁶ The phrase “at least” comes from the more general setting (footnote 5). In the case of a common valuation, proportionality implies that each player receives exactly $1/n$ of the total value.

concept is prominent both in law and in the literature on fair division ever since its inception [47]. Nowadays, it is called *proportionality* [16, 42]. We require a strong variant of proportionality, which holds not only for specific valuations but for *any* valuations that satisfy the condition $v_1 > \dots > v_n$. We call this variant *ordinal proportionality*, as it is based only on the order among items rather than on their exact values.⁷

Given the latin square constraint (Section 3), the allocation after day n is ordinally proportional. This is because the bundle of every player i satisfies $v(Z_i^n) = \sum_{j \in M} v_j$ and the total value of all items is $v(Z_N^n) = n \cdot \sum_{j \in M} v_j$. However, before day n it is impossible to guarantee ordinal proportionality. For example, if v_1 is very high and the other item values are very low, then at any day before the n -th day, the value of some players (those who did not get item 1 so far) will be below the proportionality threshold.

A natural relaxation of proportionality for indivisible items is *proportionality up to c items*, abbreviated PROP c [22, 21, 27, 17, 33]. It means that, for each player i , if we add to i 's bundle some c items not allocated to i yet, then i 's value will be at least $1/n$ of the total value. This definition implicitly assumes that all items are goods (have positive values); we use a more symmetric definition, that applies equally to items with positive or negative values: instead of just adding c items, we both add and remove the same number of items.

Definition 3. Let $c \geq 1$ be an integer.

- (a) A repeated assignment is called perpetually PROP c for valuation v if for every time $t \in [n]$ and player i , there is a set $C' \subseteq N \setminus Z_i^t$ and a set $C'' \subseteq Z_i^t$, with $|C''| = |C'| \leq c$, such that $v(Z_i^t \setminus C'' \cup C') \geq v(Z_N^t)/n$.
- (b) A repeated assignment is called perpetually ordinally PROP c if it is perpetually PROP c for every valuation v satisfying $v_1 > \dots > v_n$.

Note that, when all values are positive, Definition 3(a) implies the standard definition of PROP c , as removing items only makes the value smaller.

The sequence for $n = 6$ at Table 1 fails ordinal PROP1 already after the second day. This is because the total value allocated up to this time is $2 \cdot (v_1 + v_2 + v_3 + v_4 + v_5 + v_6)$, whereas the bundle of player 5 is $\{5, 6\}$. Even if we add the most valuable item 1, the resulting bundle value is $v_1 + v_5 + v_6$, which might be smaller than $1/6$ of the total value (e.g. if $v_5 = v_6 = 0$ and $v_1 = \dots = v_4 = 1$). Hence, the repeated assignment at Table 1 is not perpetually ordinally PROP1. Below we relate full balance to perpetual proportionality.

Theorem 4. A fully-balanced permutation sequence generates a perpetually ordinally PROP1 repeated assignment.

Theorem 4 follows directly from Theorem 5 below, which shows that ordinal PROP1 follows from an a slightly weaker balance condition.

⁷ Ordinal fairness is also known as *necessary fairness* or *stochastic-dominance fairness* (*sd-fairness*). Ordinal fairness in item allocation has been studied e.g. by Bouveret et al. [10], Pruhs and Woeginger [41], Aziz et al. [3], Segal-Halevi et al. [45], Brams et al. [14, 15].

Definition 4. A permutation sequence is called weakly fully-balanced if for all $t \in [n]$, the bundle Z_i^t of every player i satisfies the following for all $j \in [t]$:

$$Z_i^t[j] \leq \lfloor jn/t + 1 \rfloor. \quad (2)$$

We call this condition WEAKBAL(t, j).

Note that WEAKBAL(t, j) is weaker than BAL(t, j) only when jn/t is an integer; otherwise they are equivalent.

Theorem 5. A weakly fully-balanced permutation sequence generates a perpetually ordinal PROP1 repeated assignment.

Proof. We fix a day t , and prove that the cumulative allocation up to day t satisfies ordinal PROP1.

So far, nt items have been allocated, so $|Z_N^t| = nt$. Note that the multiset Z_N^t contains exactly t clones of every item in $1, \dots, n$.

Fix a player i . Let us add to i 's bundle Z_i^t a copy of the highest-valued item (1), and remove the least-valuable item from i 's bundle, so that i 's new bundle still has t items overall. Denote this new bundle C_i . To show PROP1, it is sufficient to show that the $v(C_i) \geq v(Z_N^t)/n$. Construct a new multiset D_i , containing n clones of each item in C_i ; now, it is sufficient to show that $v(D_i) \geq v(Z_N^t)$. We do this by showing a bijection $f : D_i \rightarrow Z_N^t$, such that $x \leq f(x)$ for all $x \in D_i$; hence $v(x) \geq v(f(x))$ for all $x \in D_i$.

We first show the bijection for the special case that n/t is an integer.

- We first map the n clones of item 1 in D_i (the added item). We map them to the t clones of items $1, 2, \dots, n/t$ in Z_N^t .
- Next, we map the n clones of the best item held by i , denoted $Z_i^t[1]$. By WEAKBAL($t, j = 1$), $Z_i^t[1] \leq \lfloor n/t + 1 \rfloor = n/t + 1$. Hence, we can map it to the t clones of items $n/t + 1, 2, \dots, 2n/t$ in Z_N^t .
- Similarly, by WEAKBAL(t, j) for all $j \in [t - 1]$, $Z_i^t[j] \leq jn/t + 1$; hence, we can map the n clones of $Z_i^t[j]$ to the t clones of items $jn/t + 1, 2, \dots, (j+1)n/t$ in Z_N^t .

When n/t is not an integer, the mapping works as follows:

- The n clones of item 1 in D_i are mapped to the t clones of items $1, 2, \dots, \lfloor n/t \rfloor$ in Z_N^t , as well as to some $n\%t$ clones of item $\lfloor n/t + 1 \rfloor$ (here $\%$ denotes the integer remainder operation).
- The n clones of $Z_i^t[1]$ in D_i , which is at most $\lfloor n/t + 1 \rfloor$ by WEAKBAL($t, j = 1$), are mapped to the remaining $t - n\%t$ clones of item $\lfloor n/t + 1 \rfloor$ in Z_N^t , as well as to the t clones of $\lfloor n/t + 2 \rfloor$ and some lower-valued items. Overall, the $2n$ clones of 1 and $Z_i^t[1]$ in D_i are mapped to the t clones of items $1, \dots, \lfloor 2n/t \rfloor$ as well as to some clones of $\lfloor 2n/t + 1 \rfloor$ in Z_N^t .
- Similarly, for all $j \in [t - 1]$, the $(j - 1)n$ clones of items $1, Z_i^t[1], \dots, Z_i^t[j - 1]$ in D_i are mapped to the t clones of items $1, \dots, \lfloor jn/t \rfloor$, as well as to some clones of $\lfloor jn/t + 1 \rfloor$ in Z_N^t . By WEAKBAL(t, j) $Z_i^t[j] \leq \lfloor jn/t + 1 \rfloor$, so the clones of $Z_i^t[j]$ can be mapped to the remaining clones of $\lfloor jn/t + 1 \rfloor$ and to lower-valued items.

1	1	1	1	1	1
2	2	2	2	2	2
3	3	3	3	3	3
4	4	4	4	4	4
5	5	5	5	5	5
6	6	6	6	6	6
7	7	7	7	7	7
8	8	8	8	8	8
9	9	9	9	9	9
10	10	10	10	10	10

Z_N^t

1	1	1	1	1	1
1	1	1	1	2	2
2	2	2	2	2	2
2	2	4	4	4	4
4	4	4	4	4	4
6	6	6	6	6	6
6	6	6	6	7	7
7	7	7	7	7	7
7	7	9	9	9	9
9	9	9	9	9	9

D_i

Table 4. Example for the proof of Theorem 5. Here $n = 10, t = 6$.

For illustration, Table 4 shows the items in Z_N^t (left), and the items 1 and $\lfloor jn/t + 1 \rfloor$ for $j \in [t - 1]$ in D_i (right) for $n = 10, t = 6$. One can see that each item in the D_i table is at least as good as the item at the same location in the Z_N^t table, as required.

We proved in Section 4 that there is no fully-balanced sequence for $n = 12$. In contrast, we could find a weakly-fully-balanced sequence for $n = 12$ does exist; see Table 8 in Appendix A.

Corollary 1. *A perpetually ordinal PROP1 repeated assignment exists for all $n \leq 12$.*

Unfortunately, the impossibility result for a fully-balanced sequence extends to weakly-fully-balanced sequences, though for larger values of n , as shown in Table 5. The proof is in Proposition 3 in Appendix C.

n	$6k$	$6k + 1$	$6k + 2$	$6k + 3$	$6k + 4$	$6k + 5$
bounds on k	$k \geq 19$	$k \geq 11$	$k \geq 16$	$k \geq 12$	$k \geq 17$	$k \geq 9$
bounds on n	$n \geq 114$	$n \geq 67$	$n \geq 98$	$n \geq 75$	$n \geq 106$	$n \geq 59$

Table 5. Values of $n = 6k + j$ with no weakly fully-balanced sequence

7 Ordinal PROPc: Open Question

Can we satisfy perpetual ordinal PROPc for some $c \geq 2$? For this, it is sufficient to satisfy the following condition for all players $i \in N$, after every day $t \in [n]$, for every $j \in [t]$:

$$Z_i^t[j] \leq \lfloor (j + c - 1)n/t + 1 \rfloor. \quad (3)$$

We call this condition $\text{WEAKBAL}(c, t, j)$; the condition previously called $\text{WEAKBAL}(t, j)$ is actually $\text{WEAKBAL}(1, t, j)$. For $c = 2$, Condition (3) is non-trivial only for $t \geq 3$ and $j \leq n - 2$, for example:

- In the first three days, each player gets at least one item from the top $\lfloor 2n/3 + 1 \rfloor$;
- In the first four days, each player gets at least one item from the top $\lfloor 2n/4 + 1 \rfloor$ and at least two items from the top $\lfloor 3n/4 + 1 \rfloor$;

We call a permutation sequence *c-balanced* if it satisfies $\text{WEAKBAL}(c, t, j)$ for all $t \in [n]$ and all $j \in [t]$. A *c-balanced* permutation sequence generates a perpetually ordinal PROP_c assignment; the proof is similar to Theorem 5.

Open Question 1 *Is there, for every $n \geq 1$, a 2-balanced permutation sequence? A c-balanced permutation sequence for some constant $c \geq 2$?*

We have preliminary results showing that a *c-balanced* permutation exists for $c \in O(\log n)$, but strengthening these results to $c \in O(1)$ remains challenging.

Another future work direction involves a different possible strengthening of the top-balance condition: every player should get at least one item from the top $\lceil n/t \rceil$, not only at the first t days, but also on every sequence of t consecutive days. This condition cannot be satisfied already for $n = 12$. This is because, in the first three days, there must be three pairwise-disjoint subsets of four players each, who receive the top four items. In particular, the player who gets item 4 on day 3 has not received any top-3 item so far, and thus must receive one of the top 3 items in day 4. This player gets one of the top 6 items both in day 3 and in day 4. Hence, some other player does not get any of the top 6 items in these two consecutive days. Is there a weaker variant of the top-balance condition, that can be guaranteed to hold on every sequence of t consecutive days?

8 Related Work

8.1 Fair division

The literature on fair division is very large, with new papers appearing every several weeks. See Amanatidis et al. [2], Aziz et al. [4], Nguyen and Rothe [40], Lang and Rothe [35] for some recent surveys. Most works on fair allocation consider a one-round division, but in recent years there is a growing interest in multiple-round division. Below we present several settings that are related to ours. They are ordered by increasing difficulty — from the “easiest” ones (where strongest fairness guarantees are possible) to the “hardest” ones (where only weak fairness guarantees are possible).

Repeated fair assignment. This setting is most closely related to ours, as it involves a single assignment problem (of exactly one item to each player) that is repeated every day. It was recently studied by Micheel and Wilczynski [38]. They studied the computational complexity of three separate fairness notions: (a)

Mirrored envy: if some player envies another player a certain number of times, then the latter envied them the same number of times. (b) *Equal treatment of equals*: players with the same preferences should get exactly the same final bundle. (c) *Minimizing the number of times a player is envious*.

Caragiannis and Narang [20] study a generalization of repeated fair assignment, in which the value of a player for an item may depend on the number of rounds the item was previously used by the same player. In this setting, even obtaining fairness at the last day becomes challenging, as the round-robin technique might not work. They study the run-time of computing an allocation that maximizes the sum of values (“utilitarian”), and present some special cases in which it is possible to ensure EF1 (envy-free up-to one item) at the last day.

Lim et al. [36] study the computational problem of finding a repeated allocation that maximizes the smallest utility overall, or every individual day. This is NP-hard in general, but can be solved efficiently in some special cases.

Repeated fair allocation. This setting is more general than ours, as it allows the number of items allocated each day to be different than the number of players, and do not require that each player receives exactly one item per day [30]. Naturally, the attainable fairness guarantees are more restricted. In particular: (a) When $n \geq 3$ and the number of days is fixed in advance (as in our setting), only last-day fairness is guaranteed, and only when the number of days is a multiple of n ; there are no guarantees on fairness after each day. (b) When $n = 2$ and the number of days is fixed in advance and is even, there are repeated allocations that are envy-free after the last day, and in addition, the allocation in each individual day satisfies a weak variant of EF1. (c) When the number of rounds can be chosen by the algorithm, there are sequences that are simultaneously last-day envy-free and Pareto-efficient, and individual-day PROP1.

Cookson et al. [23] strengthen their results to *ordinal fairness*, as in our setting. They show polynomial-time algorithms that guarantee the following: (a) For $n = 2$ players: ordinal EF1 after each day, as well as in each individual day; (b) For $n \geq 3$ players: ordinal EF1 at each individual day, as well as ordinal PROP1 after the last day. They also prove that, for $n \geq 3$, ordinal EF1 after each day cannot be guaranteed even when all players have identical preferences.

In contrast to these works, we focus on the important special case of one item per player per day, and consider ordinal fairness guarantees that should hold after each day, for any n .

Temporal fair allocation. This setting is even more general, as it allows the number and values of items to be different in each day. The guarantees are naturally weaker; see He et al. [29], Elkind et al. [25], Cookson et al. [23]. Still, the fact that the same players participate in each daily allocation is advantageous, as it allows us to compensate players who got a low value in early days, by giving them a higher value in later days.

One-shot allocation. The most well-studied fair division setting involves a one-shot allocation (a single day). For this setting, there are several algorithms

that attain an allocation that is *envy-free up to one item (EF1)* — a condition stronger than PROP1 [37, 18, 19]. These cannot be applied in our setting, as they might allocate more than one item to the same player in the same day.

One-shot assignment. Here, each player must receive a single item. In this setting every allocation is trivially PROP1 (and EF1), which makes these conditions uninteresting. Fairness is usually attained either by randomization or by monetary transfers; see Basteck [5] and Segal-Halevi [44] respectively for some recent surveys. In our *repeated* assignment setting, we expect to attain fair outcomes even without randomization or money.

One-shot allocation with cardinality constraints. Here, items are partitioned into pre-specified categories, and each player can receive at most a fixed number of items (called the “capacity”) of each category [8, 24, 46]. The repeated assignment setting can be reduced to this setting: define, for each day, a category containing all items allocated at that day, with a capacity of 1. There are algorithms that guarantee EF1 (hence also PROP1), but only for specific numeric valuations. For example, the algorithm of Biswas and Barman [8], translated to our setting, works as follows: (1) Assign the items in day 1 arbitrarily. (2) In each day $t \geq 2$, order the players by ascending value, from the player with the smallest to the one with the largest total value; give item 1 to the first player in the order, item 2 to the second player, etc. Suppose that, after some day t , some player i has a smaller value than some other player j , i.e., $v(Z_i^t) < v(Z_j^t)$. Let t' be the maximum $t' < t$ in which $v(Z_i^{t'}) \geq v(Z_j^{t'})$ held. Then at day $t' + 1$, player j received an item with a higher value than i , and at days $t' + 2, \dots, t$, player i received an item with an equal-or-higher value than j . Hence, if the item of day $t' + 1$ is removed from j ’s bundle, i does not envy. Hence, the allocation is EF1.

In contrast, our aim is to guarantee *ordinal approximate fairness* — a condition that holds simultaneously for all numeric valuations consistent with the ranking. Another advantage of this approach is that our algorithms do not need to know the valuations, as they can allocate positions in a picking sequence.

Picking sequences The output of our algorithms can also be understood as a *picking sequence* — an order by which each player picks a favorite item from the remaining items. Picking sequences have a very low informational requirements, as players do not have to reveal their valuations — they only have to pick the best remaining item when their turn arrives [11, 13, 43].

Online fair allocation. This is the hardest setting studied, as it assumes that the items and their valuations are *not known in advance* (in contrast to temporal fair division). Each day, a new item arrives, and must be allocated immediately and irrevocably to one of the players. As the future is not known, the algorithm might make allocations that are bad in hindsight. This translates to much weaker fairness guarantees. It is not possible to guarantee PROP_c for any constant c , even in the last round; such a guarantee is possible only if c grows at least like $\sqrt{t}$. See e.g. Benadè et al. [6], Benadè et al. [7], Kahana and Hazon [32], He et al. [29], Neoh et al. [39], Zeng and Psomas [48], Jiang et al. [31].

Latin squares. Kawase et al. [34] studied an allocation problem with “latin square” constraints. Their goal is not to attain proportionality, but rather to maximize the social welfare.

Acknowledgments. This work has started at the following post on Mathematics Stack Exchange: <https://math.stackexchange.com/q/5014281>.

Erel is funded by Israel Science Foundation grant no. 1092/24.

The proof of Proposition 3 in Appendix C was written by Aristotle [1] and verified manually by the authors.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

Bibliography

- [1] Achim, T., Best, A., Bietti, A., Der, K., Federico, M., Gukov, S., Halpern-Leistner, D., Henningsgard, K., Kudryashov, Y., Meiburg, A., Michelsen, M., Patterson, R., Rodriguez, E., Scharff, L., Shanker, V., Sicca, V., Sowrirajan, H., Swope, A., Tamas, M., Tenev, V., Thomm, J., Williams, H., Wu, L.: Aristotle: Imo-level automated theorem proving (2025), URL <https://arxiv.org/abs/2510.01346>
- [2] Amanatidis, G., Birmpas, G., Filos-Ratsikas, A., Voudouris, A.A.: Fair division of indivisible goods: A survey. arXiv preprint arXiv:2202.07551 (2022)
- [3] Aziz, H., Gaspers, S., MacKenzie, S., Walsh, T.: Fair assignment of indivisible objects under ordinal preferences. *Artificial Intelligence* **227**, 71–92 (2015), arXiv:1312.6546
- [4] Aziz, H., Li, B., Moulin, H., Wu, X.: Algorithmic fair allocation of indivisible items: A survey and new questions. *ACM SIGecom Exchanges* **20**(1), 24–40 (2022)
- [5] Basteck, C.: Fair solutions to the random assignment problem. *Journal of Mathematical Economics* **79**, 163–172 (2018)
- [6] Benade, G., Kazachkov, A.M., Procaccia, A.D., Psomas, C.A.: How to make envy vanish over time. In: *Proceedings of the 2018 ACM Conference on Economics and Computation*, pp. 593–610, EC ’18, Association for Computing Machinery, Ithaca, NY, USA (2018), ISBN 978-1-4503-5829-3
- [7] Benadè, G., Kazachkov, A.M., Procaccia, A.D., Psomas, A., Zeng, D.: Fair and efficient online allocations. *Operations Research* **72**(4), 1438–1452 (2024), ISSN 0030-364X
- [8] Biswas, A., Barman, S.: Fair division under cardinality constraints. In: *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pp. 91–97 (2018)
- [9] Bolzano, B.: Rein analytischer Beweis des Lehrsatzes, das zwischen je zwey Werthen, die ein entgegengesetztes Resultat gewähren, wenigstens eine reelle Wurzel der Gleichung liege. gedruckt bei Gottlieb Haase (1817)
- [10] Bouveret, S., Endriss, U., Lang, J.: Fair division under ordinal preferences: Computing envy-free allocations of indivisible goods. In: *Proceedings of the 2010 Conference on ECAI 2010: 19th European Conference on Artificial Intelligence*, pp. 387–392, IOS Press, NLD (2010), ISBN 978-1-60750-605-8

- [11] Bouveret, S., Lang, J.: A general elicitation-free protocol for allocating indivisible goods. In: IJCAI 2011 (2011)
- [12] Bouveret, S., Lemaître, M.: Characterizing conflicts in fair division of indivisible goods using a scale of criteria. *Autonomous Agents and Multi-Agent Systems* **30**(2), 259–290 (2016)
- [13] Brams, S.J., Kaplan, T.R.: Dividing the indivisible. *Journal of Theoretical Politics* **16**(2), 143 (2004)
- [14] Brams, S.J., Kilgour, D.M., Klamler, C.: Two-person fair division of indivisible items when envy-freeness is impossible. In: *Operations Research Forum*, vol. 3, p. 24, Springer (2022)
- [15] Brams, S.J., Kilgour, D.M., Klamler, C.: Fair division of indivisible items: Envy-freeness vs. efficiency revisited. *Games* **17**(1), 4 (2026)
- [16] Brams, S.J., Taylor, A.D.: *Fair Division: From cake-cutting to dispute resolution*. Cambridge University Press (1996)
- [17] Bu, X., Li, Z., Liu, S., Song, J., Tao, B.: Fair division with allocator’s preference. In: *International Conference on Web and Internet Economics (WINE 2023)*, pp. 77–94 (2023)
- [18] Budish, E.: The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes. *Journal of Political Economy* **119**(6), 1061–1103 (2011)
- [19] Caragiannis, I., Kurokawa, D., Moulin, H., Procaccia, A.D., Shah, N., Wang, J.: The unreasonable fairness of maximum nash welfare. *ACM Transactions on Economics and Computation (TEAC)* **7**(3), 1–32 (2019)
- [20] Caragiannis, I., Narang, S.: Repeatedly matching items to agents fairly and efficiently. *Theoretical Computer Science* **981** (2024), ISSN 0304-3975, arXiv:2207.01589
- [21] Chakraborty, M., Igarashi, A., Suksompong, W., Zick, Y.: Weighted envy-freeness in indivisible item allocation. *ACM Transactions on Economics and Computation* **9**(3), 1–39 (2021)
- [22] Conitzer, V., Freeman, R., Shah, N.: Fair public decision making. In: *Proceedings of the 2017 ACM Conference on Economics and Computation*, pp. 629–646 (2017)
- [23] Cookson, B., Ebadian, S., Shah, N.: Temporal fair division. *Proceedings of the AAAI Conference on Artificial Intelligence* **39**(13), 13727–13734 (2025), ISSN 2374-3468
- [24] Dror, A., Feldman, M., Segal-Halevi, E.: On fair division under heterogeneous matroid constraints. *Journal of Artificial Intelligence Research* **76**, 567–611 (2023)
- [25] Elkind, E., Lam, A., Latifian, M., Neoh, T.Y., Teh, N.: Temporal fair division of indivisible items (2024), to appear in the *Proceedings of AAMAS 2025*
- [26] Euler, R.: On the completability of incomplete latin squares. *European Journal of Combinatorics* **31**(2), 535–552 (2010), ISSN 0195-6698, combinatorics and Geometry
- [27] Gourvès, L., Lesca, J., Wilczyński, A.: On fairness via picking sequences in allocation of indivisible goods. In: *International Conference on Algorithmic Decision Theory (ADT 2021)*, pp. 258–272 (2021)
- [28] Hall, M.: An existence theorem for latin squares. *Bulletin of the American Mathematical Society* **51**(6), 387–388 (1945)
- [29] He, J., Procaccia, A.D., Psomas, A., Zeng, D.: Achieving a fairer future by changing the past. In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019*, pp. 343–349, Macao, China (2019), ISBN 978-0-9992411-4-1

- [30] Igarashi, A., Lackner, M., Nardi, O., Novaro, A.: Repeated fair allocation of indivisible items. *Proceedings of the AAAI Conference on Artificial Intelligence* **38**(9), 9781–9789 (2024), ISSN 2374-3468, arXiv:2304.01644
- [31] Jiang, H., Kulkarni, J., Singla, S.: Online geometric discrepancy for stochastic arrivals with applications to envy minimization (2019)
- [32] Kahana, I., Hazon, N.: The leximin approach for a sequence of collective decisions. In: *ECAI 2023*, pp. 1198–1206, *Frontiers in Artificial Intelligence and Applications*, IOS Press (2023), ISBN 978-1-64368-436-9, arXiv:2305.18024
- [33] Kahana, I., Segal-Halevi, E., Hazon, N.: Perpetual fully-online approximate fairness. arXiv preprint arXiv:2605.19844 (2026)
- [34] Kawase, Y., Roy, B., Sanpui, M.A.: Resource allocation under the latin square constraint (2025)
- [35] Lang, J., Rothe, J.: Fair division of indivisible goods. In: *Economics and Computation: An Introduction to Algorithmic Game Theory, Computational Social Choice, and Fair Division*, pp. 605–680, Springer (2024)
- [36] Lim, E., Neoh, T.Y., Teh, N.: Fairness in repeated matching: A maximin perspective. In: *Proceedings of the AAAI Conference on Artificial Intelligence*, pp. 17111–17119 (2026)
- [37] Lipton, R.J., Markakis, E., Mossel, E., Saberi, A.: On approximately fair allocations of indivisible goods. In: *Proceedings of the 5th ACM conference on Electronic commerce - EC '04*, p. 125 (2004), ISBN 1-58113-771-0
- [38] Micheel, K.J., Wilczynski, A.: Fairness in repeated house allocation. In: *27th European Conference on Artificial Intelligence (ECAI 2024)*, *Frontiers in Artificial Intelligence and Applications*, IOS Press (2024), ISBN 978-1-64368-548-9
- [39] Neoh, T.Y., Peters, J., Teh, N.: Online fair division with additional information (2025)
- [40] Nguyen, T.T., Rothe, J.: Complexity results and exact algorithms for fair division of indivisible items: a survey. In: *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*, pp. 6732–6740 (2023)
- [41] Pruhs, K., Woeginger, G.J.: Divorcing made easy. In: *Fun with Algorithms*, *Lecture Notes in Computer Science*, vol. 7288, pp. 305–314, Springer, Berlin, Heidelberg (2012), ISBN 978-3-642-30347-0
- [42] Robertson, J., Webb, W.: *Cake-cutting algorithms: Be fair if you can*. AK Peters/CRC Press (1998)
- [43] Segal-Halevi, E.: Competitive equilibrium for almost all incomes: existence and fairness. *Autonomous Agents and Multi-Agent Systems* **34**(1), 26 (2020), ISSN 1573-7454, arXiv:1705.04212
- [44] Segal-Halevi, E.: Generalized rental harmony. *The American Mathematical Monthly* **129**(5), 403–414 (2022)
- [45] Segal-Halevi, E., Hassidim, A., Aziz, H.: Fair allocation with diminishing differences. *Journal of Artificial Intelligence Research* **67**, 471–507 (2020)
- [46] Shoshan, H., Hazon, N., Segal-Halevi, E.: Efficient nearly-fair division with capacity constraints. In: *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, pp. 206–214 (2023)
- [47] Steinhaus, H.: The problem of fair division. *Econometrica* **16**, 101–104 (1948)
- [48] Zeng, D., Psomas, A.: Fairness-efficiency tradeoffs in dynamic fair division. In: *Proceedings of the 21st ACM Conference on Economics and Computation*, pp. 911–912, EC '20, Association for Computing Machinery, Virtual Event, Hungary (2020), ISBN 978-1-4503-7975-5, arXiv:1907.11672

A Balanced Sequences

1	5	7	4	6	3	2
2	7	6	3	1	4	5
3	4	5	7	2	6	1
4	6	2	1	5	7	3
5	3	1	6	7	2	4
6	2	4	5	3	1	7
7	1	3	2	4	5	6

1	8	5	6	7	4	3	2
2	7	4	3	6	8	5	1
3	6	8	2	5	1	4	7
4	5	7	1	3	2	8	6
5	4	2	8	1	7	6	3
6	3	1	7	8	5	2	4
7	2	6	5	4	3	1	8
8	1	3	4	2	6	7	5

1	9	8	5	7	6	4	3	2
2	6	7	3	4	8	5	9	1
3	5	4	9	1	2	7	8	6
4	8	9	6	2	3	1	5	7
5	7	2	1	9	4	8	6	3
6	3	1	8	5	7	9	2	4
7	4	6	2	8	9	3	1	5
8	2	5	7	3	1	6	4	9
9	1	3	4	6	5	2	7	8

Table 6. Fully-balanced permutation sequences for $n \in \{7, 8, 9\}$.

1	10	8	6	7	4	9	5	2	3
2	9	3	4	5	6	10	1	8	7
3	7	5	2	9	1	8	4	10	6
4	6	10	1	8	3	7	2	5	9
5	8	9	7	10	2	6	3	4	1
6	4	2	3	1	9	5	8	7	10
7	5	1	10	4	8	3	9	6	2
8	3	6	9	2	10	4	7	1	5
9	2	7	8	6	5	1	10	3	4
10	1	4	5	3	7	2	6	9	8

1	9	11	7	8	6	10	2	3	5	4
2	11	10	5	3	4	8	7	9	6	1
3	8	6	4	10	9	5	1	7	11	2
4	7	5	8	2	1	11	10	6	3	9
5	6	9	2	1	7	3	11	4	8	10
6	10	3	1	11	8	9	4	2	7	5
7	5	2	9	4	10	6	3	8	1	11
8	4	1	11	7	3	2	9	5	10	6
9	3	8	10	6	11	4	5	1	2	7
10	2	7	6	9	5	1	8	11	4	3
11	1	4	3	5	2	7	6	10	9	8

Table 7. Fully-balanced permutation sequences for $n \in \{10, 11\}$.

1	12	11	8	9	10	4	7	2	6	5	3
2	11	10	6	7	5	3	9	12	8	4	1
3	10	12	4	5	11	6	8	1	7	9	2
4	9	7	5	2	8	1	10	11	3	6	12
5	8	6	3	1	9	12	2	10	4	11	7
6	7	9	2	11	12	8	3	4	5	1	10
7	6	3	1	10	4	2	11	5	12	8	9
8	5	2	12	4	7	10	1	9	11	3	6
9	4	1	10	8	3	11	12	6	2	7	5
10	3	8	11	12	6	9	5	7	1	2	4
11	2	5	9	6	1	7	4	3	10	12	8
12	1	4	7	3	2	5	6	8	9	10	11

Table 8. weakly-fully-balanced sequence for $n = 12$

B Integers for which No Fully-Balanced Sequence Exists

We prove the following extension of Proposition 1.

Proposition 2. *There is no fully-balanced permutation sequence of length n in any of the following ranges:*

$n \bmod 6$	n	first value covered
0	$n = 6k, k \geq 2$	12
1	$n = 6k + 1, k \geq 11$	67
2	$n = 6k + 2, k \geq 3$	20
3	$n = 6k + 3, k \geq 8$	51
4	$n = 6k + 4, k \geq 4$	28
5	$n = 6k + 5, k \geq 9$	59.

In particular, no fully-balanced sequence exists for any $n > 61$.

The technique used here is similar to that in the proof of Proposition 1. We also use the same notation A_i, B_i, C_i, D_i as in that proof.

B.1 Case $n = 6k + 1$

The following formulas hold:

$$\left\lceil \frac{6k+1}{2} \right\rceil = 3k+1, \quad \left\lceil \frac{6k+1}{3} \right\rceil = 2k+1$$

and

$$\left\lceil \frac{6k+1}{4} \right\rceil = \begin{cases} (6k+4)/4 & \text{if } k \text{ is even} \\ (6k+2)/4 & \text{if } k \text{ is odd} \end{cases}$$

We have the following: $\{A_{3k+2}, \dots, A_{6k+1}\} \subset \{B_1, \dots, B_{3k+1}\}$ and $\{B_{3k+2}, \dots, B_{6k+1}\} \subset \{A_1, \dots, A_{3k+1}\}$. Let

$$\mathcal{A}_k = \{A_{3k+2}, \dots, A_{6k+1}\} \cap \{B_{2k+2}, \dots, B_{3k+1}\}$$

and

$$\mathcal{B}_k = \{B_{3k+2}, \dots, B_{6k+1}\} \cap \{A_{2k+2}, \dots, A_{3k+1}\}.$$

Thus, $\mathcal{A}_k$ and $\mathcal{B}_k$ are disjoint and $|\mathcal{A}_k| \geq k-1$, $|\mathcal{B}_k| \geq k-1$.

Even k Let $a_k = \frac{6k+4}{4}$ and $\mathcal{C}_k = \{C_{a_k+1}, \dots, C_{2k+1}\}$. Since $\mathcal{A}_k \cup \mathcal{B}_k \subset \{C_1, \dots, C_{2k+1}\}$, then

$$\left| (\mathcal{A}_k \cup \mathcal{B}_k) \cap \mathcal{C}_k \right| \geq 2k + 1 - a_k - 3 = \frac{1}{2}k - 3.$$

We know that $(\mathcal{A}_k \cup \mathcal{B}_k) \cap \mathcal{C}_k \subset \{D_1, \dots, D_{a_k}\}$. Also,

$$\left| \{C_{3k+2}, \dots, C_{6k+1}\} \cap \{D_1, \dots, D_{3k+1}\} \right| \geq 3k - 1.$$

There is a contradiction, if $3k - 1 + \frac{1}{2}k - 3 > 3k + 1$. Hence, there is no solution, if $k > 10$, or equivalently, $k \geq 12$ and $n \geq 73$.

Odd k Using an argument similar to even k , we get a contradiction, if

$$3k - 1 + \frac{1}{2}k - \frac{5}{2} > 3k + 1.$$

Hence, there are no solutions if $k \geq 11$ or $n \geq 67$.

B.2 Case $n = 6k + 2$

The following formulas hold:

$$\left\lceil \frac{6k+2}{2} \right\rceil = 3k + 1, \quad \left\lceil \frac{6k+2}{3} \right\rceil = 2k + 1$$

and

$$\left\lceil \frac{6k+2}{4} \right\rceil = \begin{cases} (6k+4)/4 & \text{if } k \text{ is even} \\ (6k+2)/4 & \text{if } k \text{ is odd} \end{cases}$$

We have the following: $\{A_{3k+2}, \dots, A_{6k+2}\} = \{B_1, \dots, B_{3k+1}\}$ and $\{B_{3k+2}, \dots, B_{6k+2}\} = \{A_1, \dots, A_{3k+1}\}$. Let

$$\mathcal{A}_k = \{A_{3k+2}, \dots, A_{6k+2}\} \cap \{B_{2k+2}, \dots, B_{3k+1}\}$$

and

$$\mathcal{B}_k = \{B_{3k+2}, \dots, B_{6k+2}\} \cap \{A_{2k+2}, \dots, A_{3k+1}\}.$$

Thus, $\mathcal{A}_k$ and $\mathcal{B}_k$ are disjoint and $|\mathcal{A}_k| = k$, $|\mathcal{B}_k| = k$.

Even k Let $a_k = \frac{6k+4}{4}$ and $\mathcal{C}_k = \{C_{a_k+1}, \dots, C_{2k+1}\}$. Since $\mathcal{A}_k \cup \mathcal{B}_k \subset \{C_1, \dots, C_{2k+1}\}$, then

$$\left| (\mathcal{A}_k \cup \mathcal{B}_k) \cap \mathcal{C}_k \right| \geq 2k + 1 - a_k - 1 = \frac{1}{2}k - 1.$$

We know that $(\mathcal{A}_k \cup \mathcal{B}_k) \cap \mathcal{C}_k \subset \{D_1, \dots, D_{a_k}\}$. However,

$$\{C_{3k+2}, \dots, C_{6k+2}\} = \{D_1, \dots, D_{3k+1}\}.$$

There is a contradiction, if $\frac{1}{2}k - 1 > 0$. Hence, there is no solution, if $k > 2$, or equivalently, $n \geq 26$.

Odd k Using an argument similar to even k , we get a contradiction, if

$$\frac{1}{2}k - \frac{1}{2} > 0.$$

Hence, there are no solutions if $k > 1$ or $n \geq 20$.

B.3 Case $n = 6k + 3$

The following formulas hold:

$$\left\lceil \frac{6k+3}{2} \right\rceil = 3k+2, \quad \left\lceil \frac{6k+3}{3} \right\rceil = 2k+1$$

and

$$\left\lceil \frac{6k+3}{4} \right\rceil = \begin{cases} (6k+4)/4 & \text{if } k \text{ is even} \\ (6k+6)/4 & \text{if } k \text{ is odd} \end{cases}$$

We have the following: $\{A_{3k+3}, \dots, A_{6k+3}\} \subset \{B_1, \dots, B_{3k+2}\}$ and $\{B_{3k+3}, \dots, B_{6k+3}\} \subset \{A_1, \dots, A_{3k+2}\}$. Let

$$\mathcal{A}_k = \{A_{3k+3}, \dots, A_{6k+3}\} \cap \{B_{2k+2}, \dots, B_{3k+2}\}$$

and

$$\mathcal{B}_k = \{B_{3k+3}, \dots, B_{6k+3}\} \cap \{A_{2k+2}, \dots, A_{3k+2}\}.$$

Thus, $\mathcal{A}_k$ and $\mathcal{B}_k$ are disjoint and $|\mathcal{A}_k| = k$, $|\mathcal{B}_k| = k$.

Even k Let $a_k = \frac{6k+4}{4}$ and $\mathcal{C}_k = \{C_{a_k+1}, \dots, C_{2k+1}\}$. Since $\mathcal{A}_k \cup \mathcal{B}_k \subset \{C_1, \dots, C_{2k+1}\}$, then

$$\left| (\mathcal{A}_k \cup \mathcal{B}_k) \cap \mathcal{C}_k \right| \geq 2k+1 - a_k - 1 = \frac{1}{2}k - 1.$$

We know that $(\mathcal{A}_k \cup \mathcal{B}_k) \cap \mathcal{C}_k \subset \{D_1, \dots, D_{a_k}\}$. However,

$$\left| \{C_{3k+3}, \dots, C_{6k+3}\} \cap \{D_1, \dots, D_{3k+2}\} \right| \geq 3k.$$

There is a contradiction, if $\frac{1}{2}k - 1 > 2$. Hence, there is no solution, if $k > 6$, or equivalently, $n \geq 51$.

Odd k Using an argument similar to even k , we get a contradiction, if

$$\frac{1}{2}k - \frac{3}{2} > 2.$$

Hence, there are no solutions if $k > 7$ or $n \geq 57$.

B.4 Case $n = 6k + 4$

The following formulas hold:

$$\left\lceil \frac{6k+4}{2} \right\rceil = 3k+2, \quad \left\lceil \frac{6k+4}{3} \right\rceil = 2k+2$$

and

$$\left\lceil \frac{6k+4}{4} \right\rceil = \begin{cases} (6k+4)/4 & \text{if } k \text{ is even} \\ (6k+6)/4 & \text{if } k \text{ is odd} \end{cases}$$

We have the following: $\{A_{3k+3}, \dots, A_{6k+4}\} = \{B_1, \dots, B_{3k+2}\}$ and $\{B_{3k+3}, \dots, B_{6k+4}\} = \{A_1, \dots, A_{3k+2}\}$. Let

$$\mathcal{A}_k = \{A_{3k+3}, \dots, A_{6k+4}\} \cap \{B_{2k+3}, \dots, B_{3k+2}\}$$

and

$$\mathcal{B}_k = \{B_{3k+3}, \dots, B_{6k+4}\} \cap \{A_{2k+3}, \dots, A_{3k+2}\}.$$

Thus, $\mathcal{A}_k$ and $\mathcal{B}_k$ are disjoint and $|\mathcal{A}_k| = k$, $|\mathcal{B}_k| = k$.

Even k Let $a_k = \frac{6k+4}{4}$ and $\mathcal{C}_k = \{C_{a_k+1}, \dots, C_{2k+2}\}$. Since $\mathcal{A}_k \cup \mathcal{B}_k \subset \{C_1, \dots, C_{2k+2}\}$, then

$$|(\mathcal{A}_k \cup \mathcal{B}_k) \cap \mathcal{C}_k| \geq 2k+2 - a_k - 2 = \frac{1}{2}k - 1.$$

We know that $(\mathcal{A}_k \cup \mathcal{B}_k) \cap \mathcal{C}_k \subset \{D_1, \dots, D_{a_k}\}$. However,

$$\{C_{3k+3}, \dots, C_{6k+4}\} = \{D_1, \dots, D_{3k+2}\}.$$

There is a contradiction, if $\frac{1}{2}k - 1 > 0$. Hence, there is no solution, if $k > 2$, or equivalently, $n \geq 28$.

Odd k Using an argument similar to even k , we get a contradiction, if

$$\frac{1}{2}k - \frac{3}{2} > 0.$$

Hence, there are no solutions if $k > 3$ or $n \geq 34$.

B.5 Case $n = 6k + 5$

The following formulas hold:

$$\left\lceil \frac{6k+5}{2} \right\rceil = 3k+3, \quad \left\lceil \frac{6k+5}{3} \right\rceil = 2k+2$$

and

$$\left\lceil \frac{6k+5}{4} \right\rceil = \begin{cases} (6k+8)/4 & \text{if } k \text{ is even} \\ (6k+6)/4 & \text{if } k \text{ is odd} \end{cases}$$

We have the following: $\{A_{3k+4}, \dots, A_{6k+5}\} \subset \{B_1, \dots, B_{3k+3}\}$ and $\{B_{3k+4}, \dots, B_{6k+5}\} \subset \{A_1, \dots, A_{3k+3}\}$. Let

$$\mathcal{A}_k = \{A_{3k+4}, \dots, A_{6k+5}\} \cap \{B_{2k+3}, \dots, B_{3k+3}\}$$

and

$$\mathcal{B}_k = \{B_{3k+4}, \dots, B_{6k+5}\} \cap \{A_{2k+3}, \dots, A_{3k+3}\}.$$

Thus, $\mathcal{A}_k$ and $\mathcal{B}_k$ are disjoint and $|\mathcal{A}_k| = k$, $|\mathcal{B}_k| = k$.

Even k Let $a_k = \frac{6k+8}{4}$ and $\mathcal{C}_k = \{C_{a_k+1}, \dots, C_{2k+2}\}$. Since $\mathcal{A}_k \cup \mathcal{B}_k \subset \{C_1, \dots, C_{2k+2}\}$, then

$$|(\mathcal{A}_k \cup \mathcal{B}_k) \cap \mathcal{C}_k| \geq 2k + 2 - a_k - 2 = \frac{1}{2}k - 2.$$

We know that $(\mathcal{A}_k \cup \mathcal{B}_k) \cap \mathcal{C}_k \subset \{D_1, \dots, D_{a_k}\}$. However,

$$| \{C_{3k+4}, \dots, C_{6k+5}\} \cap \{D_1, \dots, D_{3k+3}\} | \geq 3k + 1.$$

There is a contradiction, if $\frac{1}{2}k - 2 > 2$. Hence, there is no solution, if $k > 8$, or equivalently, $n \geq 65$.

Odd k Using an argument similar to even k , we get a contradiction, if

$$\frac{1}{2}k - \frac{3}{2} > 2.$$

Hence, there are no solutions if $k > 7$ or $n \geq 59$.

C Integers for which No Weakly Fully-Balanced Sequence Exists

Proposition 3. *There is no weakly-fully-balanced permutation sequence of length n in any of the following ranges:*

$n \bmod 6$	n	first value covered
0	$n = 6k, k \geq 19$	114
1	$n = 6k + 1, k \geq 11$	67
2	$n = 6k + 2, k \geq 16$	98
3	$n = 6k + 3, k \geq 12$	75
4	$n = 6k + 4, k \geq 17$	106
5	$n = 6k + 5, k \geq 9$	59.

In particular, no weakly-fully-balanced sequence exists for any $n > 108$.

Proof. Put

$$a = \left\lfloor \frac{n}{2} \right\rfloor + 1, \quad b = \left\lfloor \frac{n}{3} \right\rfloor + 1, \quad d = \left\lfloor \frac{n}{4} \right\rfloor + 1, \quad e = 2a - n.$$

The number e is 2 for even n and 1 for odd n . By WEAKBAL(2,1), every player receiving an item ranked above a in day 1 receives an item of rank at most a in day 2. Consequently, at most e players receive top- a items in both days 1 and 2: the $2a$ top-item incidences are distributed among n players.

Using the notation A_i, B_i, C_i, D_i as in the proof of Proposition 1, define:

$$\begin{aligned} Z_1 &= \{A_i : b + 1 \leq i \leq a\} \cap \{B_i : a + 1 \leq i \leq n\}, \\ Z_2 &= \{B_i : b + 1 \leq i \leq a\} \cap \{A_i : a + 1 \leq i \leq n\}. \end{aligned}$$

At most e players in $\{A_i : b + 1 \leq i \leq a\}$ can have received a top- a item in day 2. So at most e players in $\{A_i : b + 1 \leq i \leq a\}$ are not in $\{B_i : a + 1 \leq i \leq n\}$. Hence, $|Z_1| \geq a - b - e$. Analogous arguments imply $|Z_2| \geq a - b - e$.

The sets Z_1, Z_2 are disjoint by construction. By $\text{WEAKBAL}(3, 1)$, all their players must receive a top- b item in day 3, so

$$Z_1 \cup Z_2 \subseteq \{C_i : 1 \leq i \leq b\}.$$

After discarding the d players receiving a top- d item in day 3, at least

$$2(a - b - e) - d$$

of these players must receive a top- d item in day 4, by $\text{WEAKBAL}(4, 1)$. Independently, consider the $n - a$ players in $\{C_i : a + 1 \leq i \leq n\}$. At most e of them received top- a items in both A and B. Every other such player has received a top- a item at most once in the first three days, and hence must receive a top- a item in D by $\text{WEAKBAL}(4, 2)$. This contributes at least $n - a - e$ players to $\{D_i : 1 \leq i \leq a\}$. They are disjoint from the former contribution, since their C-ranks are respectively above a and between $d + 1$ and b . Therefore a necessary condition is

$$a \geq (2(a - b - e) - d) + (n - a - e),$$

or equivalently

$$n \leq 2b + d + 3e. \tag{4}$$

It remains to check what values of n satisfy (4). Here we split by $n \bmod 6$:

- For $n = 6k$ and k even, we have $b = 2k + 1, d = 1.5k + 1, e = 2$, so (4) becomes $6k \leq 4k + 2 + 1.5k + 1 + 6 = 5.5k + 9$, equivalently $k \leq 18$.
If k is odd, then d becomes $1.5k + 0.5$, so (4) becomes $6k \leq 5.5k + 8.5$, equivalently $k \leq 17$.
- For $n = 6k + 1$ and k even, we have $b = 2k + 1, d = 1.5k + 1, e = 1$, so (4) becomes $6k + 1 \leq 4k + 2 + 1.5k + 1 + 3 = 5.5k + 6$, equivalently $k \leq 10$.
If k is odd, then (4) becomes $6k + 1 \leq 5.5k + 5.5$, equivalently $k \leq 9$.
- For $n = 6k + 2$ and k even, we have $b = 2k + 1, d = 1.5k + 1, e = 2$, so (4) becomes $6k + 2 \leq 4k + 2 + 1.5k + 1 + 6 = 5.5k + 9$, equivalently $k \leq 14$.
If k is odd, then d becomes $1.5k + 1.5$, so (4) becomes $6k + 2 \leq 5.5k + 9.5$, equivalently $k \leq 15$.
- For $n = 6k + 3$ and k even, we have $b = 2k + 2, d = 1.5k + 1, e = 1$, so (4) becomes $6k + 3 \leq 4k + 4 + 1.5k + 1 + 3 = 5.5k + 8$, equivalently $k \leq 10$.
If k is odd, then d becomes $1.5k + 1.5$, so (4) becomes $6k + 3 \leq 5.5k + 8.5$, equivalently $k \leq 11$.
- For $n = 6k + 4$ and k even, we have $b = 2k + 2, d = 1.5k + 2, e = 2$, so (4) becomes $6k + 4 \leq 4k + 4 + 1.5k + 2 + 6 = 5.5k + 12$, equivalently $k \leq 16$.
If k is odd, then d becomes $1.5k + 1.5$, so (4) becomes $6k + 4 \leq 5.5k + 11.5$, equivalently $k \leq 15$.
- For $n = 6k + 5$ and k even, we have $b = 2k + 2, d = 1.5k + 2, e = 1$, so (4) becomes $6k + 5 \leq 4k + 4 + 1.5k + 2 + 3 = 5.5k + 9$, equivalently $k \leq 8$.
If k is odd, then d becomes $1.5k + 1.5$, so (4) becomes $6k + 5 \leq 5.5k + 8.5$, equivalently $k \leq 7$.

These thresholds correspond to the thresholds in the proposition statement.

D Example of Top-Balanced Sequence Algorithm

Here, we demonstrate a top-balanced sequence for $n = 15$ players. This example is sufficiently complex to show how this procedure works for any positive integer n .

For day 1, assign items 1 through 15 to players 1 through 15 in increasing order. For day 2, assign items 1 through 7 to players 9 through 15. The remaining items can be assigned arbitrarily to the remaining players for day 2. For day $t = 3$, we have $\lceil \frac{15}{3} \rceil = 5$. Notice that for the first 2 days, there are 10 unique players who were assigned one of the items 1 through 5. There are 5 players not in this group. In particular, these are players 6, 7, 8, 14, 15. Assign these players items 1 through 5. Assign the remaining items arbitrarily to the remaining players for day 3.

For day $t = 4$, we have $\lceil \frac{15}{4} \rceil = 4$. For the first 3 days, the players assigned one of the items 1 through 4 are 12 unique players. For day 4, choose the 3 players who are not in this group of 12 players, and assign items 1 through 3 to these players. These will be players 5, 13, 15. Assign the remaining items arbitrarily for day 4.

For day $t = 5$, we have $\lceil \frac{15}{5} \rceil = 3$. A key point is that the group of players assigned one of items 1 through 3 on the first 4 days are unique. There are 3 players outside this group. Given the previous assignments, these will be players 4, 12, 14. Assign items 1 through 3 to these 3 players on day 5. Assign the remaining items to the remaining players arbitrarily.

For day $t = 6$, we have $\lceil \frac{15}{6} \rceil = 3$. Every player has already been assigned one of items 1 through 3. However, we still need some care in assigning items on day 6. Choose 2 players that were not assigned items 1 or 2, and assign them items 1 and 2 on day 6. These could be players 3 and 11. Assign the remaining items arbitrarily.

For day $t = 7$, we still have $\lceil \frac{15}{7} \rceil = 3$. There are at least 2 players who were not assigned items 1 or 2 on the first 6 days. Assign items 1 and 2 to these players on day 7. These could be players 8 and 15.

On day $t = 8$, there is one player not previously assigned items 1 or 2. Based on previous assignments, this would be player 14. Assign item 1 to this player. The remaining items may be assigned arbitrarily on day 8.

Every player has been assigned either item 1 or item 2 after 8 days. In particular, 8 different players were assigned item 1. Arbitrarily order the remaining 7 players and assign item 1 to each of the 7 players on each of the remaining days. The remaining 7 players could be ordered as

$$2, 10, 7, 13, 12, 8, 14.$$

This procedure can be applied to an arbitrary positive integer n . The key property is that at each stage all players are assigned one of the top $\lceil \frac{n}{t} \rceil$ items in the first t days. If $\frac{n}{t}$ is an integer, then these players will be unique. Moving to the next day, we pick the players that have not been assigned any items from 1 through $\lceil \frac{n}{t+1} \rceil$ and assign those items to these players on day $t+1$. As soon as those players have been exhausted, we focus on the players that have not been assigned any items from 1 through $\lceil \frac{n}{t+2} \rceil$ and so on. This procedure will produce a top-balanced sequence.

E Heterogeneous valuations

Throughout the paper we assumed for simplicity that all players have the same ranking and the same value function. Here we prove that our main positive result, namely Theorem 5, holds even when players have different valuations and different rankings.

Theorem 6. *A weakly fully-balanced permutation sequence generates a perpetually ordinaly PROP1 repeated assignment, even when players have different rankings and different valuations.*

Proof. As different players may rank the items differently, we can no longer identify a number j with “the j -th highest ranked item”. Instead, we interpret the entries of the permutation sequence as *picking rights*: the number j received by a player on a given day is *the right to be the j -th one to pick an item* on that day. That is, on each day the players pick items one at a time, in increasing order of their number, so that the player holding right 1 picks first, the player holding right 2 picks second, and so on. A player holding right j always gets an item that is *at least as good, from that player’s own viewpoint*, as their j -th best item. Hence, if the player’s valuation is v_i , the value they obtain from a single pick with right j is at least $v_i(j)$, where we rename the items so that $v_i(1) \geq \dots \geq v_i(n)$.

Crucially, the weak-balance condition constrains the picking rights only, and is independent of any player’s valuation. For a player i and a day t , let P_i^t denote the multiset of picking rights that i has accumulated over days $1, \dots, t$, and let $P_i^t[j]$ denote the j -th smallest (that is, j -th best) right in P_i^t . In this notation, WEAKBAL(t, j) states that

$$P_i^t[j] \leq \lfloor jn/t + 1 \rfloor \quad \text{for all } j \in [t].$$

We now fix a day t and a player i with valuation v_i , rename the items as above so that $v_i(1) \geq \dots \geq v_i(n)$, and prove that the cumulative allocation up to day t satisfies PROP1 for i .

So far, nt picking rights have been allocated in total, so $|Z_N^t| = nt$, where Z_N^t now denotes the multiset of all allocated rights. Since each day is a permutation of the rights $1, \dots, n$, the multiset Z_N^t contains exactly t clones of every right in $1, \dots, n$. Because the actual item allocated for right j is worth at least $v_i(j)$ to player i , it suffices to bound i ’s value from below by the value of the rights: writing $v_i(P_i^t) := \sum_{p \in P_i^t} v_i(p)$, the true value of i ’s bundle is at least $v_i(P_i^t)$, while the true value of all allocated items equals $v_i(Z_N^t) = t \sum_{k=1}^n v_i(k)$, which is exactly the value obtained by reading Z_N^t as a multiset of rights. Hence it is enough to establish PROP1 at the level of rights.

Let us add to P_i^t a copy of the best right (1) and remove its worst (highest-numbered) right, so that the modified multiset still has t rights overall. Denote this new multiset C_i . Adding right 1 corresponds to adding a single most-preferred item to i ’s bundle, and removing the worst right corresponds to removing a single item; this is a valid PROP1 add-one/remove-one modification. To show PROP1, it is therefore sufficient to show that $v_i(C_i) \geq v_i(Z_N^t)/n$.

Construct a new multiset D_i , containing n clones of each right in C_i ; now it is sufficient to show that $v_i(D_i) \geq v_i(Z_N^t)$. We do this by exhibiting a bijection $f : D_i \rightarrow Z_N^t$ such that $x \leq f(x)$ for all $x \in D_i$; since v_i is non-increasing in the right index, this gives $v_i(x) \geq v_i(f(x))$ for all $x \in D_i$, and summing yields the claim.

We first show the bijection for the special case that n/t is an integer.

- Map the n clones of right 1 in D_i (the added right) to the t clones of rights $1, 2, \dots, n/t$ in Z_N^t .

- Next, map the n clones of the best right held by i , namely $P_i^t[1]$. By $\text{WEAKBAL}(t, j = 1)$, $P_i^t[1] \leq \lfloor n/t + 1 \rfloor = n/t + 1$. Hence we can map these clones to the t clones of rights $n/t + 1, n/t + 2, \dots, 2n/t$ in Z_N^t .
- Similarly, by $\text{WEAKBAL}(t, j)$ for all $j \in [t - 1]$, $P_i^t[j] \leq jn/t + 1$; hence we can map the n clones of $P_i^t[j]$ to the t clones of rights $jn/t + 1, jn/t + 2, \dots, (j + 1)n/t$ in Z_N^t .

When n/t is not an integer, the mapping works as follows:

- The n clones of right 1 in D_i are mapped to the t clones of rights $1, 2, \dots, \lfloor n/t \rfloor$ in Z_N^t , as well as to some $n\%t$ clones of right $\lfloor n/t + 1 \rfloor$ (here $\%$ denotes the integer remainder operation).
- The n clones of $P_i^t[1]$ in D_i , which is at most $\lfloor n/t + 1 \rfloor$ by $\text{WEAKBAL}(t, j = 1)$, are mapped to the remaining $t - n\%t$ clones of right $\lfloor n/t + 1 \rfloor$ in Z_N^t , as well as to the t clones of $\lfloor n/t + 2 \rfloor$ and some lower-ranked rights. Overall, the $2n$ clones of 1 and $P_i^t[1]$ in D_i are mapped to the t clones of rights $1, \dots, \lfloor 2n/t \rfloor$ as well as to some clones of $\lfloor 2n/t + 1 \rfloor$ in Z_N^t .
- Similarly, for all $j \in [t - 1]$, the $(j - 1)n$ clones of $1, P_i^t[1], \dots, P_i^t[j - 1]$ in D_i are mapped to the t clones of rights $1, \dots, \lfloor jn/t \rfloor$, as well as to some clones of $\lfloor jn/t + 1 \rfloor$ in Z_N^t . By $\text{WEAKBAL}(t, j)$, $P_i^t[j] \leq \lfloor jn/t + 1 \rfloor$, so the clones of $P_i^t[j]$ can be mapped to the remaining clones of $\lfloor jn/t + 1 \rfloor$ and to lower-ranked rights.

1	1	1	1	1	1
2	2	2	2	2	2
3	3	3	3	3	3
4	4	4	4	4	4
5	5	5	5	5	5
6	6	6	6	6	6
7	7	7	7	7	7
8	8	8	8	8	8
9	9	9	9	9	9
10	10	10	10	10	10

Z_N^t

1	1	1	1	1	1
1	1	1	1	2	2
2	2	2	2	2	2
2	2	4	4	4	4
4	4	4	4	4	4
6	6	6	6	6	6
6	6	6	6	7	7
7	7	7	7	7	7
7	7	9	9	9	9
9	9	9	9	9	9

D_i

Table 9. Example for the proof of Theorem 6. Here $n = 10, t = 6$, and the entries are picking rights.

For illustration, Table 9 shows the rights in Z_N^t (left), and the rights 1 and $\lfloor jn/t + 1 \rfloor$ for $j \in [t - 1]$ in D_i (right) for $n = 10, t = 6$. One can see that each right in the D_i table is at least as good as the right at the same location in the Z_N^t table, as required.