Ordered-instance allocation transfer with exact extra-copy preservation

Dev D. Goyal

28 October 2026; revised 1 November 2026

Abstract

An ordered allocation assigns ranks in each agent's decreasing list of good values. Transferring such an allocation to named goods must preserve utilities while respecting the allowed copies of each good. For every finite set of goods, every nonempty finite agent set, and nonnegative additive real valuations, an arbitrary set-valued ordered allocation admits a transfer that weakly increases every agent's utility, preserves the total number of extra copies exactly, respects the same uniform per-good extra-copy cap, and allocates every good. Unused input ranks are permitted. The question is motivated by an ordered-instance reduction attributed to Akrami et al. [1], whose source conventions have not been verified here. The proof completes the input and uses a balanced bipartite matching graph. Incidence ports prevent repeated goods within an agent's bundle, and primary copy vertices force coverage. The explicit graph also gives a polynomial-time transformation. The accompanying Lean formalization covers the existence and utility statements; executable runtime analysis is separate.

1 The transfer problem

Sorting each agent's values turns a named-good instance into an ordered instance. The rank labels are shared, but a particular rank can name a different real good for each agent. An ordered allocation is therefore useful only if it can be transferred back without reducing utilities. When goods can be copied, the transfer must also control both the total number of extra copies and the number of extra copies of each good.

The supplied source context attributes an ordered-instance reduction question to Akrami et al., identified by arXiv:2502.09377v3 [1]. The primary allocation definition, copy convention, restricted theorem, and current source version were not verified for this note. The definitions below are therefore explicit local definitions, and no verified strengthening or resolution of a published question is claimed. If the source conventions agree, this theorem may be compared with that motivating question. No first-priority claim is made. Hall's marriage theorem is the standard matching result used in the proof.

The main difficulty is not just to find enough eligible goods. An agent must receive each good at most once, each good has a multiplicity cap, and all goods must be used to preserve the extra-copy count exactly after completion. The matching graph enforces these conditions together.

Theorem 1 in Section 2 states the result. Lean proves the theorem and its two consequences in Corollary 2 (Section 6); the algorithmic consequence in Proposition 4 has a conventional proof.

2 Definitions and the main theorem

Let N be a nonempty finite set of agents, with $n = |N| \geq 1$. Let M be a finite set of goods, with $m = |M| \geq 0$, and write $[m] = \{1, \ldots, m\}$, with $[0] = \varnothing$. Agent i has a nonnegative real weight $v_i(g)$ for each $g \in M$. Its value for a bundle is additive:

$$u_i(B_i) = \sum_{g \in B_i} v_i(g).$$

For every agent, fix a bijection $r_i : M \to [m]$ such that

$$w_i(a) := v_i(r_i^{-1}(a)) \quad \text{is nonincreasing in } a.$$

Ties can be ordered arbitrarily. The ordered utility of $A_i \subseteq [m]$ is $U_i(A_i) = \sum_{a \in A_i} w_i(a)$.

An allocation on a finite ground set G is a family of sets $X = (X_i)_{i \in N}$ with $X_i \subseteq G$. Different agents may receive the same good. Each individual bundle is a set and contains no repeated good. Define

$$\begin{aligned} \mathrm{mult}_X(g) &= |\{i \in N : g \in X_i\}|, \\ \mathrm{extra}_X(g) &= \max\{\mathrm{mult}_X(g) - 1, 0\}, \\ \mathrm{copies}(X) &= \sum_{g \in G} \mathrm{extra}_X(g). \end{aligned}$$

An unused good has multiplicity zero and contributes zero extra copies. For integers $t, k \geq 0$, a (t, k)-allocation satisfies $\mathrm{copies}(X) \leq t$ and $\mathrm{extra}_X(g) \leq k$ for every $g \in G$. Thus a cap of k extra copies allows multiplicity at most $k + 1$.

Theorem 1 (Exact ordered-instance transfer)**.** *Under the definitions above, let $A = (A_i)_{i \in N}$ be any allocation of $[m]$. Let $k \geq 0$ be an integer and suppose $\mathrm{extra}_A(a) \leq k$ for every rank $a \in [m]$. There is an allocation $B = (B_i)_{i \in N}$ of M such that*

$$u_i(B_i) \geq U_i(A_i) \qquad (i \in N), \tag{1}$$

$$\mathrm{copies}(B) = \mathrm{copies}(A), \tag{2}$$

$$\mathrm{extra}_B(g) \leq k \qquad (g \in M), \tag{3}$$

$$\mathrm{mult}_B(g) \geq 1 \qquad (g \in M). \tag{4}$$

The input may leave ranks unused. There is no restriction on which ranks are copied or on their recipients.

The proof is given in Section 4. The identity of copied goods need not be preserved. Neither must the original bundle sizes $|A_i|$ be preserved. In particular, completion can increase a bundle's size.

Corollary 2 (Budgets and utility thresholds)**.** *If A is a (t, k)-allocation, the output in Theorem 1 is a (t, k)-allocation. If real numbers τ_i satisfy $U_i(A_i) \geq \tau_i$ for all $i \in N$, the output also satisfies $u_i(B_i) \geq \tau_i$ for all agents.*

Proof. Equation (2) gives $\mathrm{copies}(B) = \mathrm{copies}(A) \leq t$, and (3) gives the per-good budget. The threshold assertion follows from (1). □

Why completion matters. Take two goods a, b and two agents. Their weights are

$$(v_1(a), v_1(b)) = (2, 1), \qquad (v_2(a), v_2(b)) = (1, 2).$$

Both ordered weight lists are $(2, 1)$. If $A_1 = A_2 = \{1\}$, both ordered utilities equal 2 and the total extra-copy count is 1. Giving a to agent 1 and b to agent 2 achieves those utilities but uses no extra copy. To preserve the count and cover both goods, the theorem may instead give $B_1 = \{a, b\}$ and $B_2 = \{b\}$. Their utilities are 3 and 2, and exactly one extra copy is used. This example also shows why the theorem does not promise to preserve original bundle sizes.

3 The rank-counting inequality

The counting step converts rank information into a lower bound on eligible agent–good incidences. It is the estimate needed for Hall's condition.

Suppose an ordered allocation C covers every rank. Define its slot set by

$$\mathcal{S} = \{(i, a) : i \in N,\ a \in C_i\}, \qquad D = |\mathcal{S}| = \sum_{i \in N} |C_i|.$$

An incidence $(i, g) \in N \times M$ is eligible for slot (i, a) when $r_i(g) \le a$. For $P \subseteq \mathcal{S}$, let $\mathcal{N}(P)$ be the set of all incidences eligible for at least one slot in P. For $0 \le s \le m$, put

$$c(s) = |\{(i, a) \in \mathcal{S} : a \le s\}|.$$

This counts all slots in the prefix, including slots outside P.

Lemma 3. *For every $P \subseteq \mathcal{S}$ and $I \subseteq N \times M$, let $S = \{g \in M : \exists i,\ (i, g) \in I\}$ and $s = |S|$. Then*

$$|P| \le |\mathcal{N}(P) \setminus I| + c(s). \tag{5}$$

If every rank occurs between one and q times in C, then also

$$c(s) \le qs, \qquad c(s) \le D - m + s. \tag{6}$$

Proof. At most $c(s)$ slots in P have rank at most s. Consider an agent i with at least one selected slot of rank greater than s, and let a_i be its largest selected rank. Since C_i is a set, the selected ranks greater than s are distinct members of $\{s + 1, \dots, a_i\}$. There are at most $a_i - s$ of them.

The agent's top a_i real goods form a set of size a_i, because r_i is a bijection. At most s of those goods belong to S. Hence at least $a_i - s$ of their incidences lie outside I. Each of these incidences is eligible for the selected slot (i, a_i), so it belongs to $\mathcal{N}(P) \setminus I$. Incidences of different agents are distinct. Summing over agents bounds the number of selected slots above s by $|\mathcal{N}(P) \setminus I|$. Adding the selected prefix slots proves (5).

There are s prefix ranks, each occurring at most q times, so $c(s) \le qs$. Each of the last $m - s$ ranks occurs at least once. Thus at least $m - s$ of the D slots lie outside the prefix, which gives $c(s) \le D - (m - s)$. □

4 A matching construction

The proof first covers every rank without changing the copy count. It then uses a matching graph whose ports enforce the per-agent set condition and whose primary vertices enforce coverage.

Proof of Theorem 1. If $m = 0$, all bundles are empty and the conclusions hold with $B_i = \varnothing$. Assume $m \ge 1$.

Complete the input. Fix an agent $i_0 \in N$. Give each unused rank once to i_0, leaving all other assignments unchanged, and denote the result by C. Nonnegativity gives $U_i(C_i) \geq U_i(A_i)$ for every agent. An unused rank changes multiplicity from zero to one, so its extra-copy count remains zero. Consequently,

$$\text{copies}(C) = \text{copies}(A).$$

Set $q = k + 1$. Each rank now occurs between one and q times. With D as in Section 3, double-counting incidences gives

$$m \leq D \leq qm, \qquad \text{copies}(C) = D - m. \tag{7}$$

Build a balanced bipartite graph. The vertex classes are disjoint, even when their labels refer to the same agent or good. The left and right sides have the following vertices.

Left class	Count	Right class	Count
Requested slots $(i, a) \in \mathcal{S}$	D	Right ports (i, g)	nm
Left ports $(i, g) \in N \times M$	nm	Primary copies of goods	m
Dummy vertices	$qm - D$	Secondary copies of goods	$(q-1)m$

Each side has $nm + qm$ vertices. Every good has one primary and $q - 1$ secondary copy vertices. The edges are exactly these:

1. A slot (i, a) is adjacent to right port (i, g) if $r_i(g) \leq a$.
2. Left port (i, g) is adjacent to its identical right port and to all q copy vertices of good g.
3. Every dummy vertex is adjacent to every secondary copy vertex.

In particular, slots and dummies have no primary-copy neighbors. Only the identical left port and eligible slots can reach a given right port.

Verify Hall's condition. Take any subset of the left vertices. Write its slot part as P, its left port part as the incidence set I, and its dummy part as T. Set $S = \{g : \exists i,\ (i, g) \in I\}$ and $s = |S|$. The right-port neighbors include $\mathcal{N}(P) \cup I$, of cardinality

$$|I| + |\mathcal{N}(P) \setminus I|.$$

There are two cases.

If $T = \varnothing$, all q copies of every good in S are also neighbors. They are distinct from right ports. The neighbor count is at least

$$|I| + |\mathcal{N}(P) \setminus I| + qs \geq |I| + |P|,$$

by Lemma 3 and $c(s) \leq qs$.

If $T \neq \varnothing$, all $(q-1)m$ secondary copies are neighbors, as are the s primary copies indexed by S. The neighbor count is at least

$$|I| + |\mathcal{N}(P) \setminus I| + (q-1)m + s.$$

Since $|T| \leq qm - D$, the other bound in Lemma 3 gives

$$\begin{aligned} |P| + |T| &\leq |\mathcal{N}(P) \setminus I| + (D - m + s) + (qm - D) \\ &= |\mathcal{N}(P) \setminus I| + (q-1)m + s. \end{aligned}$$

Adding $|I|$ proves the required bound in this case too.

Thus every left-vertex subset has at least as many neighbors as vertices. Hall's marriage theorem gives a matching saturating the left side. The sides have equal finite cardinality, so the matching also saturates the right side. It is a perfect matching.

Extract sets of goods. Each slot (i, a) is matched to an eligible right port (i, g). Assign that good g to agent i and let B_i be the resulting bundle. Distinct slots use distinct right ports. In particular, two slots of the same agent cannot assign the same good, so every B_i is a set.

If a slot uses right port (i, g), that right port cannot also be matched through its identity edge. The left port (i, g) must therefore be matched to one of the q copy vertices of g. Distinct assigned incidences use distinct copy vertices. This proves $\mathrm{mult}_B(g) \le q$ for every good.

Every primary copy of g has a matched left neighbor because the matching is perfect. Such a neighbor must be a left port (i, g). This left port is occupied by the primary copy. Its right port cannot use the identity edge and must therefore be matched from a slot. That slot assigns g to agent i. Hence $\mathrm{mult}_B(g) \ge 1$ for all g.

All D slots are filled, and no within-agent duplicate is removed in forming the bundles. Thus $\sum_i |B_i| = D$. Coverage and double counting now give

$$\mathrm{copies}(B) = \sum_{g \in M} (\mathrm{mult}_B(g) - 1) = D - m = \mathrm{copies}(C) = \mathrm{copies}(A).$$

Also $\mathrm{extra}_B(g) = \mathrm{mult}_B(g) - 1 \le q - 1 = k$.

Finally, eligibility gives $r_i(g) \le a$ for the good assigned to slot (i, a). Since the ordered values are nonincreasing,

$$v_i(g) = w_i(r_i(g)) \ge w_i(a).$$

Summing over agent i's distinct slots yields $u_i(B_i) \ge U_i(C_i) \ge U_i(A_i)$. This proves all four conclusions. □

5 Algorithmic consequence and scope

Proposition 4. *Given the rankings, a suitable allocation can be obtained by bipartite matching on a graph with at most $2nm$ vertices on each side. For positive n, m, a crude edge bound is $O(n^2m^2)$.*

Proof. Set $\widehat{q} = \min\{k + 1, n\}$. In a set-valued allocation, a rank occurs at most n times. The completed allocation therefore has multiplicities at most $\widehat{q}$. Since $n \ge 1$ and $k \ge 0$, also $\widehat{q} \ge 1$. Repeat the construction with $q = \widehat{q}$. Each side has $nm + \widehat{q}m \le 2nm$ vertices. The slots contribute at most nm^2 edges. Left ports contribute at most $nm(1 + \widehat{q})$ edges. There are at most nm dummies and at most nm secondary copies, giving at most n^2m^2 dummy edges. These counts imply the stated bound. A standard bipartite maximum-matching algorithm finds a perfect matching, and the proof describes how to read off the allocation. □

This is a polynomial number of combinatorial operations, even if k is large and binary encoded. With arbitrary real weights and supplied rankings, no bit-complexity model for real-number comparisons is assumed. No executable matching algorithm or runtime bound is asserted to have been formalized in Lean.

Nonnegativity is used when unused ranks are assigned during completion. The matching and rankwise utility argument for an already covered input only requires real weights ordered by the supplied bijections. The construction preserves $|C_i|$ for each completed bundle, but it need not preserve $|A_i|$. It can also change which named goods are copied. The theorem transfers achieved utility thresholds. It does not by itself prove that one extra copy suffices for exact maximin-share fairness, the separate fair-division question discussed in the source context.

6 Formal statement map and reproducibility

The accompanying source targets Lean 4.19.0 and mathlib revision

`c44e0c8ee63ca166450922a373c7409c5d26b00b`.

The three final declarations, in namespace `OrderedReductionAllocation`, are

```
ordered_allocation_transfer_exact
ordered_allocation_transfer
ordered_threshold_transfer
```

They correspond to Theorem 1, its budget consequence, and the utility-threshold consequence. The definitions of multiplicity, extra copies, total copies, and utility are in namespace `OrderedReductionCompletion`. Natural-number subtraction there is truncated at zero, so its extra-copy definition agrees with the maximum used above. Lean uses zero-based `Fin m` ranks; this paper uses one-based ranks. The eligibility order and all counts agree under that change of labels.

The formal source includes the prefix estimates, the rank-counting inequality, the Hall gadget and assignment extraction, completion, and the allocation-level utility and exact-copy statements. A declaration map and reproduction instructions accompany this manuscript. A fresh Lean 4.19.0 build succeeded for the import-only source variant, which replaces the umbrella import by two explicit imports and leaves the proof body unchanged. A separate audit with `--trust=0` checked the three final declarations and an expanded statement of the exact transfer. All four reports contain only the standard foundational axioms `propext`, `Classical.choice`, and `Quot.sound`. A formal proof checks the stated proposition, not historical novelty or external expert validation.

7 Research process and AI disclosure

This work is fully autonomous agentic research. Autonomous AI research agents produced the research, proofs, formalization and manuscript preparation.

Opus 5.5 by Anthropic used with Claude Code revised this manuscript editorially on 1 November 2026: it moved this disclosure from the title page into its own section, revised the abstract and the first section, and completed the reference entry. It did not change the mathematics.

Dev D. Goyal is responsible for publication. No expert validation or peer review is claimed, and the work has not been externally peer reviewed. Section 6 states the formal scope, and Section 5 states the limits of the result.

References

[1] Akrami et al. arXiv:2502.09377v3. https://arxiv.org/abs/2502.09377v3, 2025. Supplied attribution to Akrami et al.; full bibliographic metadata, source conventions, and current-version comparison not verified here.