Honor Among Bandits: No-Regret Learning for Online Fair Division

Ariel D. Procaccia, Benjamin Schiffer, Shirley Zhang

NeurIPS 2024

Speaker: Joseph Chuang-Chieh Lin

Economics and Computation Lab,
Department of Computer Science & Engineering,
National Taiwan Ocean University

20 June 2025

Outline

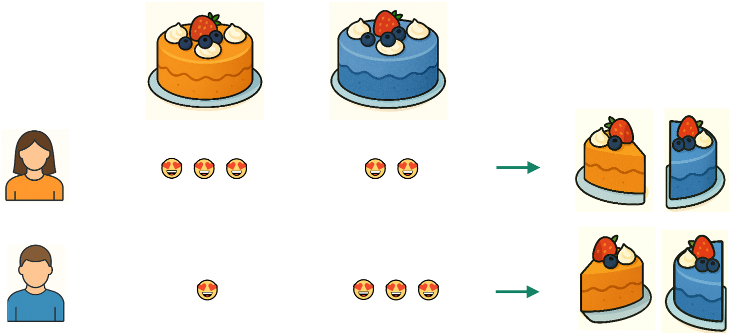
Online Fair Division Problem

- We have n players and m item types. Items arrive over time (rounds $t = 1, 2, \ldots, T$) and one at a time.
- Each arriving item j_t has a type $k_t \in [m]$, where $k_t \sim \mathcal{D}$ not depending on T.
- Allocate each item immediately and irrevocably to a single player.
- Player i's value for an item of type k is an unknown random variable $V_i(j)$ (sub-Gaussian) with mean μ^*_{ik}.
- **Goal:** Maximize social welfare under fairness constraints.
  - social welfare: Utilitarian Social Welfare
  - fairness: envy-free and proportionality in expectation.

Some fairness concepts

Some fairness concepts

Some fairness concepts



Envy-freeness for allocating indivisible goods

NP-complete

Two-Partition Problem

Given a multiset S of positive integers, determine if it is possible to partition S into two disjoint subsets, say S_1 and S_2, such that the sum of the integers in S_1 is equal to the sum of the integers in S_2.

- $S = \{1, 5, 11, 5\}$
- $S_1 = \{11\}$,
  $S_2 = \{1, 5, 5\}$.

- $S = \{3, 5, 8, 10, 11, 14, 17, 19, 21, 22, 25, 33\}$.
- $S_1 = \{33, 25, 22, 14\}$,
  $S_2 = \{3, 5, 8, 10, 11, 17, 19, 21\}$.

Motivating Example: Food Bank

- A food bank receives **perishable** food donations **sequentially**.
- Must allocate each donation **immediately** to one of several food pantries.
- Each pantry has **unknown** true utility for different food types.
- Need to allocate **fairly** (no pantry envies another) while maximizing total utility distributed.

Key Goals and Challenges

- **Fairness:** Envy-freeness (EFE) or proportionality (PE) in expectation, enforced *every round*.
- **Learning:** Player values μ^*_{ik} unknown, must be learned via observed rewards.
- **Online Allocation:** Must balance exploration (learning values) and exploitation (maximizing welfare).
- **Metric:** Regret against optimal fair allocation (if μ^* were known).

Fractional Allocations and Welfare

- A **fractional allocation** is a matrix $X \in \mathbb{R}^{n \times m}$ with

$$X_{ik} \geq 0, \quad \sum_{i=1}^{n} X_{ik} = 1 \quad (\forall k \in [m]).$$

- Interpret X_{ik} as the probability that a type-k item is given to player i.
- If $\mu^* \in \mathbb{R}^{n \times m}$ is the matrix of true means, the expected welfare of X is:

$$\langle X, \mu^* \rangle_F = \sum_{i=1}^{n} \sum_{k=1}^{m} X_{ik}\, \mu^*_{ik}.$$

- $Y^{\mu^*} = \arg\max_{X \in \mathcal{F}(\mu^*)} \langle X, \mu^* \rangle_F$ is the optimal fair allocation if μ^* is known.
  - F: Frobenius inner product of two matrices.
  - $\mathcal{F}(\mu^*)$: the set of all fair, feasible fractional allocations under the true means μ^*.

Solving the LP when μ^* is known

$$
\begin{aligned}
Y^{\mu^*} := \; & \arg\max \langle X, \hat{\mu^*} \rangle_F \\
\text{s.t. } & \langle B_\ell(\mu^*), X \rangle_F \geq c_\ell, \quad \forall \ell = 1 \ldots L, \\
& \sum_{i=1}^{n} X_{ik} = 1, \quad \forall k = 1 \ldots m, \quad X_{ik} \geq 0.
\end{aligned}
$$

Fairness notions as linear constraints

Fairness in expectation relative to the mean values.

- Represent $\langle B, X \rangle_F \geq c$ as (B, c).
- a set if L linear constraints: $\{B_\ell, c_\ell\}_{\ell=1}^{L}$
  $\Leftrightarrow \quad \langle B_\ell, X \rangle_F \geq c_\ell$ for all $\ell \in [L]$.
- $B_\ell(\mu^*)$: a function of the mean value matrix μ^*.

Nash Social Welfare (NSW)

- For a discrete allocation $A = (A_1, A_2, \ldots, A_n)$ of indivisible goods, each player i has utility $v_i(A_i)$.
- The **Nash Social Welfare** of allocation A is defined as:

$$\mathrm{NSW}(A) = \left(\prod_{i=1}^{n} v_i(A_i)\right)^{1/n}.$$

- In the fractional setting with mean values μ^*, player i's utility is $v_i(X) = \sum_{k=1}^{m} X_{ik}\mu^*_{ik}$. [additive] Therefore,

$$\mathrm{NSW}(X) = \left(\prod_{i=1}^{n}\sum_{k=1}^{m} X_{ik}\mu^*_{ik}\right)^{1/n}.$$

- NSW allocations are known to achieve *Pareto optimality* and *EF1* (envy-freeness up to one good) [e.g., Caragiannis et al., 2016].

Nash Social Welfare (NSW) vs. Sum-of-Utilities (SW)

- **Sum-of-Utilities (SW):** The utilitarian social welfare (USW) used in this paper is $\mathrm{SW}(X) = \langle X, \mu^* \rangle_F = \sum_{i=1}^{n} \sum_{k=1}^{m} X_{ik} \mu^*_{ik}$.
- Connection: NSW balances fairness (geometric mean) and efficiency; SW focuses purely on total welfare (arithmetic sum).
  - NSW $\Rightarrow$ fairness: EF1; efficiency: PO.
- This work maximizes SW under fairness constraints (EFE or PE), rather than optimizing NSW.
- Computational hardness:
  - Maximizing USW with EF1 is strongly NP-hard [Aziz et al. 2023].
  - Maximizing NSW is NP-hard [Lipton et al. EC'04] and APX-hard [Lee 2017]. Best known approx. ratio: 2.889 [Cole & Gkatzelis STOC'15]

Criterion	Utilitarian Social Welfare (USW) / "Welfare" in (Individual Utility)	Nash Social Welfare (NSW)
Definition (Individual/Social)	Sum of values in an agent's bundle (Individual Utility); Sum of all individual utilities (Social Welfare)	Geometric mean of agents' individual utilities (Social Welfare)
Mathematical Objective	$\sum_{j \in A_i} v_i(j)$ (for individual i) / $\sum_{i=1}^{n} v_i(A_i)$ (for social)	$(\prod_{i=1}^{n} v_i(A_i))^{1/n}$ or $\sum_{i=1}^{n} log\ v_i(A_i)$
Primary Focus	Maximizing total aggregate utility/efficiency	Balancing efficiency with fairness/equity
Treatment of Agent Utilities	Summation; zero utility for one agent does not zero out total social welfare	Product/Geometric Mean; zero utility for one agent zeros out total NSW
Impact on Minorities/Least Satisfied Agents	Can lead to highly unequal distributions; potentially unfair to those with low values [2]	Encourages more balanced distributions; implicitly protects agents from receiving very low utility [4]
Key Properties (for maximization)	Pareto Optimal (PO) [6]	Pareto Optimal (PO), Envy-Freeness up to One Good (EF1), Scale-Free [1]
General Computational Complexity (for maximization of indivisible goods)	NP-hard [1]; often requires additional constraints for fairness	NP-hard [1]; challenging to approximate; FPT for small 'n' in some cases

Online Allocation Process

- Time steps $t = 1, 2, \ldots, T$. At round t:
  1. An item j_t of type $k_t \sim D$ (e.g. Uniform($[m]$)) arrives.
  2. The algorithm chooses a fractional allocation $X_t = \mathrm{ALG}(H_t)$ based on history H_t.
  3. The item of type k_t is given to player i_t drawn from distribution $X_{:,k_t}$.
  4. The algorithm observes reward $V_{i_t}(j_t)$ (value of that item to i_t).
- History $H_t = \{(k_1, i_1, V_{i_1}(j_1)), \ldots, (k_{t-1}, i_{t-1}, V_{i_{t-1}}(j_{t-1}))\}$.

Online Item Allocation (Pseudo-code summary)

Algorithm 2 [Online Item Allocation]

Require: ALG

1: $\forall i,\ A_i^0 \leftarrow \{\}, H_0 \leftarrow \{\}$
2: **for** $t \leftarrow 1$ to T **do**
3: $\quad X_t \leftarrow \text{ALG}(H_t)$
4: $\quad k_t \sim \mathcal{D}$
5: $\quad$ Generate item j_t of type k_t (i.e. $V_i(j_t) \sim N(\mu^*_{ik_t}, 1),\ \forall i \in N$)
6: $\quad i_t \leftarrow$ Sample from $(X_t)^\top_{k_t}$
7: $\quad A^t_{i_t} = A^{t-1}_{i_t} + \{j_t\}$
8: $\quad H_t \leftarrow H_{t-1} + (k_t, i_t, V_{i_t}(j_t))$
9: **end for**
10: **return** $A = (A_1^T, A_2^T, ..., A_n^T)$

Multi-Armed Bandit Perspective

- There exists an arm for each player's value for each type of good.
- Pulling an arm represents allocating a specific item type to a specific player.

Fairness Definitions (In Expectation)

Envy-Freeness in Expectation (EFE)

For each time t and history H_t, the chosen X_t must satisfy, for every pair $i, i' \in [n]$:

$$\langle X_{i,\cdot}^{(t)}, \mu_i^* \rangle \geq \langle X_{i',\cdot}^{(t)}, \mu_i^* \rangle.$$

No player i expects to prefer another player's allocation over their own.

Proportionality in Expectation (PE)

For each time t and history H_t, X_t must also satisfy, for all $i \in [n]$:

$$\langle X_{i,\cdot}^{(t)}, \mu_i^* \rangle \geq \frac{1}{n} \sum_{i'=1}^{n} \langle X_{i',\cdot}^{(t)}, \mu_i^* \rangle.$$

Each player's expected share is $\geq 1/n \times \{$they would get from all items$\}$.

Equivalence of EFE and PE for Two Players

When $n = 2$, the two fairness notions coincide:

EFE

$$X_1 \cdot \mu_1 \;\geq\; X_2 \cdot \mu_1,$$
$$X_2 \cdot \mu_2 \;\geq\; X_1 \cdot \mu_2.$$

PE

$$X_i \cdot \mu_i \geq \frac{(X_1 + X_2) \cdot \mu_i}{2} = \tfrac{1}{2} \sum_k \mu_{ik}, \; \forall i.$$

$$X_2 \cdot \mu_1 = \sum_k (1 - X_{1k})\, \mu_{1k} = \sum_k \mu_{1k} - X_1 \cdot \mu_1$$

Thus, $X_1 \cdot \mu_1 \geq X_2 \cdot \mu_1 \iff X_1 \cdot \mu_1 \geq \tfrac{1}{2} \sum_k \mu_{1k}.$

Fairness Definitions (In Terms of Linear Constraints)

envy-freeness in expectation; $\mathrm{efe}(\mu^*) := \{(B_\ell^{\mathrm{efe}}(\mu^*), 0)\}_{\ell=1}^{n^2}$

For every $\ell \in [n^2]$, construct $B_\ell^{\mathrm{efe}}(\mu^*)$:

- Define $i = \lceil \frac{\ell}{n} \rceil$ and $i' = (\ell \mod n) + 1$.
- For every $k \in [m]$, let $(B_\ell^{\mathrm{efe}}(\mu^*))_{ik} = \mu_{ik}^*$ and $(B_\ell^{\mathrm{efe}}(\mu^*))_{i'k} = -\mu_{ik}^*$.
- Let $(B_\ell^{\mathrm{efe}}(\mu^*))_{i''k} = 0$ for all $i'' \notin \{i, i'\}$, $k \in [m]$.

proportionality in expectation; $\mathrm{pe}(\mu^*) := \{(B_\ell^{\mathrm{pe}}(\mu^*), 0)\}_{\ell=1}^{n}$

For every $\ell \in [n]$, construct $B_\ell^{\mathrm{pe}}(\mu^*)$:

- For every $k \in [m]$, let $(B_\ell^{\mathrm{pe}}(\mu^*))_{\ell k} = \frac{n-1}{n}\,\mu_{\ell k}^*$ and $(B_\ell^{\mathrm{pe}}(\mu^*))_{\ell k} = -\frac{1}{n}\,\mu_{\ell k}^*$ for every $i \neq \ell$.

Regret

Regret

Let Y^{μ^*} be the optimal fair allocation (fraction) if μ^* is known. If the algorithm uses allocations $X_1, \ldots, X_T$, then

$$R(T) = T \langle Y^{\mu^*}, \mu^* \rangle_F - \sum_{t=1}^{T} \mathbb{E}\big[\langle X_t, \mu^* \rangle_F\big]$$

is the regret compared to the optimal fair policy.

An Illustrating Example

Say there are $n = 2$ players, $m = 2$ item types, Bernoulli rewards, and WLOG $\mu^* \in [0, 1]^{n \times m}$. Define

$$\mu^{(1)} = \begin{pmatrix} 1/T^2 & 0 \\ 1 & 0.5 \end{pmatrix} \quad , \quad \mu^{(2)} = \begin{pmatrix} 0 & 1/T^2 \\ 1 & 0.5 \end{pmatrix}.$$

Any EFE-satisfying algorithm must behave (nearly) uniformly to cover both cases.

Indistinguishability Argument

- Under either $\mu^{(1)}$ or $\mu^{(2)}$, Player 1's chance of "seeing an item" in any round is $\leq 1/T^2$.
- Over T rounds, with probability $\geq 1/2$, Player 1 sees no successes in both worlds (using Markov's inequality).
- Thus no strategy can, with probability $> 1/2$, reliably tell which of $\mu^{(1)}, \mu^{(2)}$ holds.

Regret of the Only Safe Allocation

- The *only* fractional allocation that remains envy-free for both instances is *Uniform-At-Random*: $X_{ik} = 1/2$.
- But under $\mu^{(2)}$, the optimal EFE allocation is

$$Y^{\mu^{(2)}} = \begin{pmatrix} 0 & 0.5 \\ 1 & 0.5 \end{pmatrix},$$

  which gives Player 2 all items of type 1.
- Uniform-at-Random incurs $\Omega(T)$ regret in this case.
  - Regret 1 in each iteration.

$$\langle \mu^{(2)}, Y^* \rangle_F = \left\langle \begin{pmatrix} 0 & 1/T^2 \\ 1 & 0.5 \end{pmatrix}, \begin{pmatrix} x & y \\ 1-x & 1-y \end{pmatrix} \right\rangle_F$$

$$= (1-x) + \frac{1}{2} + \left(\frac{1}{T^2} - \frac{1}{2}\right) y$$

$\Rightarrow$ max : $x = 0$
$y = 0.5$

envyness for player 1 :

$$\frac{1}{T^2} y - \left(0 \cdot (1-x) + \frac{1}{T^2} \cdot (1-y)\right) = \frac{1}{T^2}(y - (1-y))$$

$$= \frac{1}{T^2}(2y - 1) \Rightarrow y \geq 0.5$$

envyness for player 2 :

$$(1-x) + \frac{1}{2}(1-y) - \left(1 \cdot x + \frac{1}{2} \cdot y\right) = \frac{3}{2} - 2x - y \Rightarrow x \leq 0.5$$

Lower bound on means

- No algorithm can enforce envy-freeness in expectation at each round *and* achieve $o(T)$ regret if means can be arbitrarily close to zero.
- This justifies the lower bound on means ($\mu_{ik}^* \geq a > 0$) in our upper-bound results.

Problem Statement

Problem

- Given n, m, a, b such that $0 < a \leq \mu^*_{ik} \leq b$ for all $i \in [n], k \in [m]$.
- Given a family of fairness constraints $\left\{ \{B_\ell(\mu), c_\ell\}_{\ell=1}^{L} \right\}$.

Goal: Design an online algorithm ALG such that, with prob. $\geq 1 - 1/T$,

1. X_t satisfies EFE (or PE) at every round t (fairness).
2. $R(T) = o(T)$ sublinear; specifically, achieve $\tilde{O}(T^{2/3})$ regret.

Property 1: Equal Treatment Guarantees Fairness

- If players involved in a constraint share identical $X_{i,\cdot}$, the fairness constraint holds.

Property 1

For any $\ell \in [L]$, suppose that a fractional allocation $X \in \mathbb{R}^{n \times m}$ satisfies $X_{i_1} = X_{i_2}$ for any $i_1, i_2 \in \{i : B_\ell(\mu)_i \neq \mathbf{0}\}$. Then, $\langle B_\ell(\mu), X \rangle_F \geq c_\ell$.

- **Uniform-at-Random (UAR)** ($X_{ik} = 1/n$) satisfies all EFE and PE constraints.
- Ensure safe exploration: allocate uniformly to remain fair without any knowledge.

Observation 1

The EFE and PE constraints satisfy Property 1.

Explicit Constraint Formulation: Cake Example

Explicit Constraint Formulation: Cake Example

Define fractional allocations and valuations:

$$X = \begin{pmatrix} X_{\text{Alice},\text{Orange}} & X_{\text{Alice},\text{Blue}} \\ X_{\text{Bob},\text{Orange}} & X_{\text{Bob},\text{Blue}} \end{pmatrix}, \quad \mu = \begin{pmatrix} 3 & 2 \\ 1 & 3 \end{pmatrix}$$

Envy-Freeness Constraints (EFE) expressed as $\langle B_\ell(\mu), X \rangle_F \geq c_\ell$:

$$B_1(\mu) = \begin{pmatrix} 3 & 2 \\ -3 & -2 \end{pmatrix}, \; c_1 = 0, \quad B_2(\mu) = \begin{pmatrix} -1 & -3 \\ 1 & 3 \end{pmatrix}, \; c_2 = 0$$

These matrices illustrate Property 1:

- (Property 1) Equal allocations ($X_{A,O} = X_{B,O}$, $X_{A,B} = X_{B,B}$) imply constraints hold trivially.

Property 2: Near-Optimal Fair Allocation with Slack

Property 2

- For the optimal fair allocation Y^{μ^*}, there exists an X' such that:
  1. $\langle X', \mu^* \rangle_F \geq \langle Y^{\mu^*}, \mu^* \rangle_F - O(\gamma)$ (near-optimal),
  2. For each fairness constraint ℓ, either:
     - $\langle B_\ell(\mu^*), X' \rangle_F \geq c_\ell + \gamma$ (slack γ),
     - or all players involved in constraint ℓ have equal allocation in X' (Property 1 holds).

- Key for handling unknown μ^*: we can tolerate small estimation errors and still find a feasible fair X'.
- The loss $O(\gamma)$ has a (hidden) factor of $O(n^3)$ and $\gamma = O(T^{-1/3})$.

Property 3: Lipschitz Continuity of Constraints

- The fairness constraints (EFE/PE) depend linearly on μ.
- Thus, for any X, if $\|\mu - \mu'\|_1 \leq \epsilon$, then:

$$|\langle B_\ell(\mu), X\rangle_F - \langle B_\ell(\mu'), X\rangle_F| \leq K\epsilon$$

- Implies that if X satisfies a constraint for μ, then for any μ' close by, X still nearly satisfies it.

Property 3

There exists $K > 0$ such that $\forall \mu, \mu' \in [a, b]^{n\times m}$, $\forall X$ and $\forall \epsilon > 0$, if $\|\mu - \mu'\|_1 \leq \epsilon$, then $\|\langle B_\ell(\mu), X\rangle_F - \langle B_\ell(\mu'), X\rangle_F\|_1 \leq K\epsilon$.

Property 4: Invariance of Constraint Structure

- For a given constraint ℓ (e.g., envy between i and i'), the *set of players* it compares does not depend on the actual μ.
- The indices appearing in $B_\ell(\mu)$ (the non-zero rows) are fixed.
- Ensures we know exactly which players each constraint refers to, regardless of unknown means.

Property 4

For any $\mu, \mu' \in [a, b]^{n \times m}$, $\{i : B_\ell(\mu)_i \neq \mathbf{0}\} = \{i : B_\ell(\mu')_i \neq \mathbf{0}\}$.

Explicit Constraint Formulation: Cake Example

Define fractional allocations and valuations:

$$X = \begin{pmatrix} X_{\text{Alice},\text{Orange}} & X_{\text{Alice},\text{Blue}} \\ X_{\text{Bob},\text{Orange}} & X_{\text{Bob},\text{Blue}} \end{pmatrix}, \quad \mu = \begin{pmatrix} 3 & 2 \\ 1 & 3 \end{pmatrix}, \; \mu' = \begin{pmatrix} 4 & 1 \\ 2 & 5 \end{pmatrix},$$

Envy-Freeness Constraints (EFE) expressed as $\langle B_\ell(\mu), X \rangle_F \geq c_\ell$:

$$B_1(\mu') = \begin{pmatrix} 4 & 1 \\ -4 & -1 \end{pmatrix}, \; c_1 = 0, \quad B_2(\mu') = \begin{pmatrix} -2 & -5 \\ 2 & 5 \end{pmatrix}, \; c_2 = 0$$

These matrices illustrate Property 4:

- (Property 4) The locations of nonzero entries are independent of actual valuations.

Lemmas for Property 2

Lemma 1 (EFE satisfies Property 2)

There is a constructive algorithm (Algorithms 3 & 4) that transforms the optimal envy-free allocation Y^{μ^*} into an allocation X' satisfying Property 2.

- It uses "envy-with-slack-α" graphs, equivalence classes, and iterative merging/removal steps to ensure either slack or equal treatment, while losing only $O(\gamma)$ welfare.

Lemma 2 (PE satisfies Property 2)

The family of PE constraints satisfies Property 2.

- Check total slack in the proportionality constraints. One can either directly use $X' = \text{UAR}$ if slack is small, or transfer allocations from high-slack players to a communal pot and redistribute evenly if slack is large.

Proof Sketch of Lemma 1

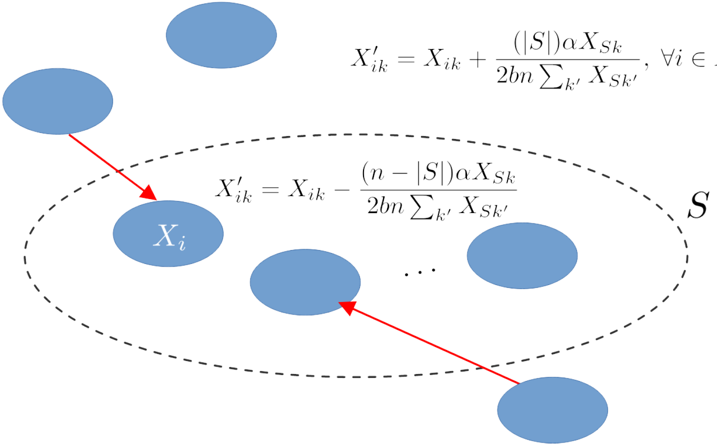
- envy-with-slack-α graphs: track whether a player prefers their allocation by at least α over another players' allocation.
- Given μ, X, α, construct a graph with a set N of vertices, a set E of edges such that a directed edge from i to $i' \Leftrightarrow X_i \cdot \mu_i - X_{i'} \cdot \mu_i < \alpha$.
  - The weight of such edge: $X_i \cdot \mu_i - X_{i'} \cdot \mu_i$.
- Construct such graphs with progressively smaller α, for $\alpha \geq \gamma$.
- The algorithm operates on sets of nodes: equivalence classes.
  - Every pair of nodes in an equivalence class has the same allocation.
- The algorithm makes progress in every iteration by either
  1. merging two equivalence classes, or
  2. removing an edge from the graph.

Algorithm 3: Envy-with-Slack Refinement (Overview)

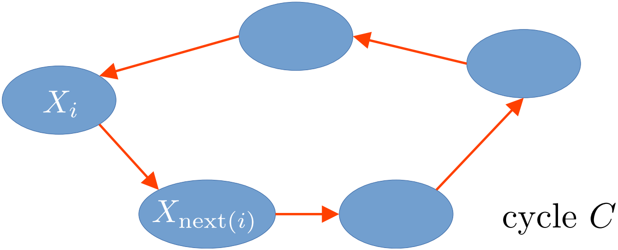
- Maintain an "envy-with-slack-α" directed graph whose nodes are players and edges $i \to i'$ mean player i's slack over i' is less than α.
- Track equivalence classes of players with identical allocations.
  - Each node in the graph is actually an equivalence class.
- Repeatedly do one of three operations to remove edges or merge classes:
  1. **remove-incoming-edge**: If a class S has in-edges but no out-edges, transfer its allocation to all other players to eliminate all in-edges.
  2. **cycle-shift**: Find a directed cycle (each points to minimal-slack neighbor). If some i^* has edges only to some but NOT all members of the cycle, split each cycle member's allocation half–half with its successor to remove one out-edge.
  3. **average-clique**: Otherwise, merge all classes in the cycle into one class, averaging their allocations.

$$X'_{ik} = X_{ik} + \frac{(|S|)\alpha X_{Sk}}{2bn \sum_{k'} X_{Sk'}}, \ \forall i \in N \setminus S$$

$$X'_{ik} = X_{ik} - \frac{(n - |S|)\alpha X_{Sk}}{2bn \sum_{k'} X_{Sk'}}$$



$$X'_{ik} = \frac{1}{2}\left(X_{ik} + X_{\text{next}(i)k}\right),\ \forall i \in V(C)$$



Merging two equivalence classes

- Merge two equivalence classes S and T: for each item type k,

$$X_k = \frac{1}{|S| + |T|} \left(\sum_{i \in S} X_{ik} + \sum_{j \in T} X_{jk} \right).$$

⋆ This operation might incur envy with respect to some equivalence class $U \notin S \cup T$.

Algorithm 4: Envy Removal Subroutine

- After merging (average-clique), envy may appear along some edges.
- Repeatedly find a directed cycle in the envy graph where each edge has non-negative envy.
- Rotate allocations along that cycle: each node takes its successor's allocation.
- This strictly reduces the number of envious edges and preserves the number of slack-edges.
- Welfare loss per call is bounded by $O(\alpha)$.

Termination and Complexity of Algorithm 3 + 4

- Start with an envy-free allocation. Each iteration removes either:
  - At least one edge from the slack graph (every n steps), or
  - At least one envious edge via Algorithm 4.
- There are at most n^2 edges total, so after $O(n^3)$ iterations all edges gone.
- Final allocation has slack $\geq \gamma$ on all constraints or equal treatment, satisfying Property 2.
- Total welfare loss is $O(\gamma)$, as each iteration costs at most $O(\gamma)$.

Proof Sketch of Lemma 2 (for PE)

- Define the slack $S_i := Y_i^{\mu} \cdot \mu_i - \frac{1}{n}\|\mu_i\|_1$ of player i.
- **Case 1:** $\sum_{i=1}^{n} S_i \leq \frac{b}{a} n\gamma$.
  - Take $X' = \text{UAR}$.
- **Case 2:** $\sum_{i=1}^{n} S_i > \frac{b}{a} n\gamma$.
  - Define $\Delta_{ik} = \frac{Y_{ik}^{\mu}}{\sum_{k'=1}^{m} Y_{ik'}^{\mu}} \cdot \frac{S_i}{\sum_{i'}^{n} S_{i'}} \cdot \frac{n\gamma}{a}$.
  - Construct X' as $X'_{ik} := Y_{ik}^{\mu} - \Delta_{ik} + \frac{1}{n}\sum_{i'=1}^{n} \Delta_{i'k}$ (redistribution).
  - By carefully deductions, we can prove that
    - $X'_i \cdot \mu_i - \frac{1}{n}\|\mu_i\|_1 \geq \gamma$.
    - $\langle Y^{\mu}, \mu \rangle_F - \langle X', \mu \rangle_F \leq \frac{b}{a} n\gamma$.

The main algorithm

Algorithm 1: Fair Explore-Then-Commit (Fair-ETC)

Input: n, m, T. Bounds $a \le \mu^*_{ik} \le b$. Fairness constraints $\{(B_\ell(\mu), c_\ell)\}_{\ell=1}^L$.

1. **Explore Phase (Rounds $t = 1$ to $T^{2/3} - 1$):**
   - Use Uniform-at-Random: $X_t(i,k) = 1/n$ for all i, k.
   - Collect observations: Let $N_{ik} = \#$ times player i got type-k item.
   - Compute empirical means $\hat{\mu}_{ik} = (1/N_{ik}) \sum V_i(j)$ over those samples.
   - Set confidence radius $\epsilon_{ik} = \sqrt{\frac{\log^2(4Tnm)}{2N_{ik}}}$.
2. **Commit Phase (Rounds $t = T^{2/3}$ to T):**
   - Define confidence set $\hat{\mu} \pm \epsilon$ (i.e., $\mu^* \in [\hat{\mu}_{ik} \pm \epsilon_{ik}]\ \forall i, k$ with prob. $1 - 1/T$).
   - Solve the semi-infinite LP:

$$
\begin{aligned}
X^{\hat{\mu}} = \ & \arg\max_X \ \langle X, \hat{\mu} \rangle_F \\
& \text{s.t. } \langle B_\ell(\mu), X \rangle_F \ge c_\ell, \quad \forall \ell = 1, \dots, L, \ \forall \mu \in [\hat{\mu} \pm \epsilon], \\
& \sum_{i=1}^n X_{ik} = 1, \quad \forall k = 1 \dots m, \quad X_{ik} \ge 0.
\end{aligned}
$$

   - For each subsequent round, use fixed fractional allocation $X_t = X^{\hat{\mu}}$.

Implementation Details

- The exploration phase yields $N_{ik} = \Omega(T^{2/3})$ samples for each (i, k) w.h.p.
  - Thus $\epsilon_{ik} = O(T^{-1/3}\sqrt{\log T})$, $\|\epsilon\|_1 = \tilde{O}(T^{-1/3})$.
- The LP has infinitely many constraints.
- However, since each constraint is linear in μ, it suffices to enforce it at extreme points of $[\hat{\mu} \pm \epsilon]$ — a finite (exponential) set.
- Alternatively, use a separation oracle + ellipsoid method to solve in polynomial time.
- Key property: any X' from Lemma 1 & 2 is feasible for the LP, so the LP is not empty.
- The solution $X^{\hat{\mu}}$ ensures fairness for all μ in $\hat{\mu} \pm \epsilon$, so in particular for μ^* w.h.p.

Linear Dependence on μ & Finite Constraint Reduction

- Suppose each fairness constraint has the form

$$\langle B(\mu), X\rangle_F = \sum_{i,k} (\beta_{ik}\,\mu_{ik})\,X_{ik} = \sum_{i,k} \alpha_{ik}\,\mu_{ik}\,.$$

- As a function of μ, this is just the linear map $\mu \mapsto \sum_{i,k} \alpha_{ik}\,\mu_{ik}$.
- We require this to hold for *all* μ in the confidence region $\big[\hat{\mu}-\epsilon,\ \hat{\mu}+\epsilon\big]$:

$$\sum_{i,k} \alpha_{ik}\,\mu_{ik} \geq c \quad \forall\,\mu \in [\hat{\mu}-\epsilon,\ \hat{\mu}+\epsilon].$$

- A linear functional achieves its minimum over a convex polytope at one of the polytope's *vertices* $\Rightarrow$ enforce $\sum_{i,k} \alpha_{ik}\,\mu_{ik} \geq c$ only at the finitely many (i.e., 2^{nm}) extreme points of the hyperrectangle $[\hat{\mu} \pm \epsilon]$.

Theorem 1: Regret Upper Bound (Main Theorem)

Theorem 1

With probability $1 - 1/T$, Fair-ETC achieves:

- X_t satisfies fairness constraints (EFE or PE) for all rounds t
- $R(T) = O\left(T^{2/3} \log T\right)$

Proof Sketch of Theorem 1 (1/2)

1. **Exploration Phase Regret:** Each of the first $T^{2/3}$ rounds uses UAR instead of Y^{μ^*}. Regret per round at most b, so total $O(T^{2/3})$.
2. **High-Probability Event:** UAR sampling yields $N_{ik} = \Omega(T^{2/3})$ for each (i, k). Then $|\hat{\mu}_{ik} - \mu^*_{ik}| \leq \epsilon_{ik} = \tilde{O}(T^{-1/3})$ w.p. $\geq 1 - \frac{1}{T}$ (Hoeffding's inequality).
3. **Existence of Near-Optimal X':** By Property 2 (Lemma 1 & 2), there is X' with $\langle X', \mu^* \rangle \geq \langle Y^{\mu^*}, \mu^* \rangle - O(T^{-1/3})$ that satisfies constraints for μ^*.

Proof Sketch of Theorem 1 (2/2)

5. **Robustness to Estimation:** By Property 3, X' satisfies constraints for all $\mu \in [\hat{\mu} \pm \epsilon]$ because slack γ can dominate $K\|\epsilon\|_1 = O(T^{-1/3} \log T)$; or by equality in Property 2 and Property 4, X' remains feasible.

6. **Commit Phase Regret:** The LP solution $\hat{X}$ has welfare at least $\langle X', \hat{\mu} \rangle$. Relate $\langle X', \hat{\mu} \rangle$ to $\langle Y^{\mu^*}, \mu^* \rangle$ via Lipschitz bounds:

$$\begin{aligned}
\langle Y^{\mu^*}, \mu^* \rangle_F - \langle \hat{X}, \mu^* \rangle_F &= \langle Y^{\mu^*}, \mu^* \rangle_F - \langle X', \mu^* \rangle_F + \langle X', \mu^* \rangle_F - \langle \hat{X}, \mu^* \rangle_F \\
&\leq \langle Y^{\mu^*}, \mu^* \rangle_F - \langle X', \mu^* \rangle_F + (\langle X', \hat{\mu} \rangle_F - \langle \hat{X}, \hat{\mu} \rangle_F) K\|\epsilon\|_1 \\
&= O(T^{-1/3} \log T).
\end{aligned}$$

Thus per-round loss in commit phase is $O(T^{-1/3} \log T)$. Over T rounds, gives $O(T^{2/3} \log T)$.

Theorem 2: Regret Lower Bound

Theorem 2

There exists a, b, n, m such that NO algorithm can, for all $\mu^* \in [a, b]^{n \times m}$, both satisfy EFE constraints (PE, resp.) and achieve regret $< \frac{T^{2/3}}{\log T}$ w.p. $\geq 1 - 1/T$.

Proof Idea of Theorem 2

Construct two instances ($\mu^{(1)}$ & $\mu^{(2)}$) on $n = 2$ players, $m = 2$ types:

$$\mu^{(1)} = \begin{pmatrix} 2 & 3 \\ 1 & 1 \end{pmatrix} \quad , \quad \mu^{(2)} = \begin{pmatrix} 2 & 3 \\ 1 & 1 + T^{-1/3} \end{pmatrix}.$$

- For $\mu^{(1)}$:
  - Optimal EFE gives all type-1 items to Player 2 and all type-2 items to Player 1.
- For $\mu^{(2)}$:

In $\mu^{(2)}$, to be envy-free, we must give some type-2 items to Player 2. In $\mu^{(1)}$, giving type-2 to Player 2 is suboptimal. Distinguishing these requires $\Omega(T^{2/3})$ samples of type-2 by Player 2. Hence any fair algorithm suffers $\Omega(T^{2/3})$ regret in at least one instance.

Open Questions

- **Poly(n, m) Regret:** Can we avoid exponential dependence on n and m in regret for EFE?
- **$\sqrt{T}$-Regret?** Is $\tilde{O}(\sqrt{T})$ possible if optimal fair solution has slack?
- **Other Fairness Notions:** Extend to equitability, EFX, MMS, etc.
- **Wider Applications:** Online cake cutting, resource scheduling with fairness, etc.
- **Dealing with changing μ_t?**
- **Gradient-based approaches?**

Thank you!

Questions & Discussions