Harvard CS 1360, Spring 2025
Instructor: Ariel Procaccia
Scribe: Sadie Zhao

Fair division 2: Rent division

—

Lecture 11

In this lecture, we introduce the problem of rent division, where the goal is to assign rooms to players and fairly split the rent based on their reported valuations.

Our first approach incorporates the miserly tenant assumption, which allows us to prove the existence of envy-free solutions using Sperner's Lemma. The second approach assumes that players have quasi-linear utilities. Under this model, we explore the existence of envy-free solutions, examine their properties, and discuss how to select optimal envy-free solutions.

1 Approach 1: Model with Miserly Tenant Assumption

1.1 Preliminary: Sperner's Lemma

Lemma 1 (Sperner's Lemma (Two-dimensional Case))**.** If we take a triangulation of a triangle and label its vertices using three labels (e.g., 1, 2, and 3) according to the following **Sperner labeling** rules:

- The three vertices of the large triangle are labeled distinctly (i.e., one with 1, one with 2, and one with 3).
- Any vertex on an edge of the large triangle can only be labeled with one of the two labels of the endpoints of that edge.
- The interior vertices can be labeled arbitrarily with any of the three labels.

Then, there exists at least one elementary triangle whose vertices are labeled with all three numbers (1, 2, and 3).

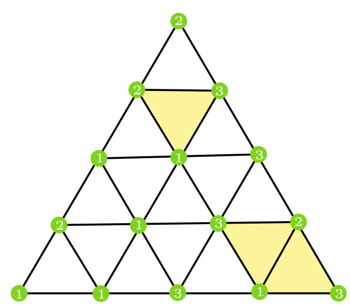

Figure 1: Example of Sperner Labeling

Proof. First, we assume that the vertices A, B, and C are labeled with 1,2, and 3 respectively.

Without loss of generality, we say a "door" is a 1-2 edge, and every elementary triangle is a "room".

First, we claim that there are odd number of "doors" on the boundary of the triangle. This is because the door can only appear on the edge AB whose two ends are labeled by 1 and 2 respectively. Therefore, if you want to start from one vertex labeled 1 and reach the other, you need an odd number of "doors" or you end up with the same label.

Moreover, we can observe that each "room" has at most 2 "doors," and a "room" has 1 "door" if and only if it is "fully labeled" (i.e., its three vertices are labeled 1, 2, and 3).

We can start at a "door" on the boundary and walk into the room. There are two cases:

- The room has only one "door," meaning it is fully labeled and we stop.
- The room has another "door," allowing us to continue moving to the adjacent room through that door.

We repeat this process, continuing until we either:

- Reach a fully labeled "room", or
- Exit the triangle through another "door" on the boundary.

Importantly, no "room" and no "door" is visited twice in this traversal.

If we start at a boundary "door" and exit through another boundary "door," this accounts for two boundary "doors." Since the total number of boundary "doors" is odd, there must be odd number of "doors" on the boundary where the path ends at a fully labeled "room." □

1.2 Model and Existence of EF Solutions

A Rent Division Problem of three players is defined as:

- There are three players A, B, and C,
- Each room i has a price p_i, and the sum of prices $p_1 + p_2 + p_3 = 1$.

The goal is to assign the rooms and divide the rent in a way that is **Envy-Free**: each player prefers their own room at the given price.

Theorem 1. *An envy-free solution always exists under the miserly tenant assumption.*

Proof. Consider the triangle representation of possible division of prices for three rooms (Figure 2). Each point in the triangle corresponds to some price (p_1, p_2, p_3) where

- The bottom left vertex represents $(p_1, p_2, p_3) = (1, 0, 0)$.
- The bottom right vertex represents $(p_1, p_2, p_3) = (0, 1, 0)$.
- The top vertex represents $(p_1, p_2, p_3) = (0, 0, 1)$.

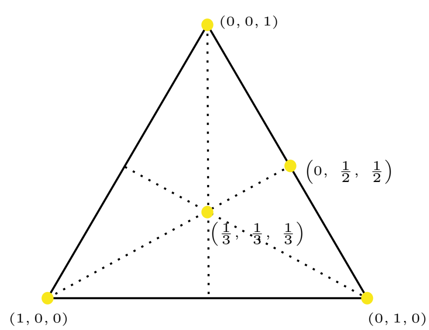

Figure 2: Triangle representation of rent division

Then, we can triangulate this large triangle as Figure 1, and for each elementary triangle, we assign A, B, and C to a vertex so that every elementary triangle is a fully labeled triangle. Next, for every vertex with label A, we ask player A which room in $\{1, 2, 3\}$ they would prefer at the price corresponding to that vertex; and we do that same for vertex with label B and C. This will give us a new labeling by $\{1, 2, 3\}$ of the triangulation.

We further assume the **Miserly Tenant Assumption**: every player would always prefer a free room over a room that costs money. This assumption allows us to determine the labeling along the edges of the triangle.

- **Bottom Edge ($p_3 = 0$)**: Along this edge, room 3 is free ($p_3 = 0$), while rooms 1 and 2 bear the full rent. Thus, any player will always prefer room 3 if given a choice. Consequently, every vertex on this edge is labeled **3**.

- **Left Edge ($p_2 = 0$)**: Along this edge, room 2 is free ($p_2 = 0$), while rooms 1 and 3 bear the full rent. Thus, any player will always prefer room 2 if given a choice. Consequently, every vertex on this edge is labeled **2**.
- **Right Edge ($p_1 = 0$)**: Along this edge, room 1 is free ($p_1 = 0$), while rooms 2 and 3 bear the full rent. Thus, any player will always prefer room 1 if given a choice. Consequently, every vertex on this edge is labeled **1**.

At the three vertices of the triangle, we observe the following:

- **Bottom-Left Vertex** ($(p_1, p_2, p_3) = (1, 0, 0)$): Here, room 1 takes the full rent, and room 2 is free. Since room 2 is free, any rational tenant must prefer it, meaning this vertex is labeled either **2** or **3**.
- **Bottom-Right Vertex** ($(p_1, p_2, p_3) = (0, 1, 0)$): Here, room 2 takes the full rent, and room 3 is free. Since room 3 is free, any rational tenant must prefer it, meaning this vertex is labeled either **1** or **3**.
- **Top Vertex** ($(p_1, p_2, p_3) = (0, 0, 1)$): Here, room 3 takes the full rent, and room 1 is free. Since room 1 is free, any rational tenant must prefer it, meaning this vertex is labeled either **1** or **2**.

These boundary conditions establish a structured labeling that enables the application of a variant of Sperner's Lemma. This variant, which can be derived from the original Sperner's Lemma, guarantees the existence of at least one fully labeled elementary triangle under our labeling scheme.

The presence of such a fully labeled triangle implies the existence of a set of prices at which each player receives their most preferred room, leading to an approximate envy-free rent division. Furthermore, by refining the triangulation, we can achieve increasingly precise approximations of envy-freeness. Finally, if we assume the compactness[1] of the price space, we can extend this argument to establish the existence of a truly envy-free rent division.

□

Remark 1.

- This process naturally leads to an algorithmic approach for computing an envy-free rent division.
- The same technique extends to settings with more than three players by applying a higher-dimensional version of Sperner's Lemma.
- A similar argument, using the original Sperner's Lemma, establishes the existence of an envy-free cake division. In this case, each vertex represents the size of cake pieces rather than room prices, and the induced labeling satisfies the standard Sperner conditions.
- This method requires no assumptions about players' valuations, making it broadly applicable. However, it does not allow control over which envy-free allocation is obtained.

2 Approach 2: Quasi-linear Model

2.1 Model and Existence of EF Solutions

A Rent Division Problem with Quasi-Linear Utilities is defined as:

- Each player $i \in N$ has value v_{ir} for room r.
- For all $i \in N$, $\sum_r v_{ir} = R$, where R is the total rent.
- The utility of player i for getting room R at price p_r is $v_{ir} - p_r$.

A **solution** consists of an **assignment** $\pi : i \mapsto r$ and a price vector $\mathbf{p}$, where p_r is the price of room r. A solution $(\pi, \mathbf{p})$ is **envy-free** if and only if

$$\forall i, j \in N,\ v_{i\pi(i)} - p_{\pi(i)} \geq v_{i\pi(j)} - p_{\pi(j)}$$

[1]A space X is compact if every infinite sequence in X has a convergent subsequence whose limit lies in X.

Remark 2. The miserly tenant assumption is not consistent with this model.

Theorem 2. *An envy-free solution always exists under quasi-linear utilities.*

2.2 Properties of EF Solutions

Assignment π is **welfare-maximizing** if

$$\pi \in \text{argmax}_\sigma \sum_{i \in N} v_{i\sigma(i)}$$

Lemma 2. If $(\pi, \mathbf{p})$ is an EF solution, then π is a welfare-maximizing assignment.

Proof. Let $(\pi, \mathbf{p})$ be an EF solution, and let σ be another assignment.

Then, due to EF, we know that for all $i \in N$, $v_{i\pi(i)} - p_{\pi(i)} \geq v_{i\sigma(i)} - p_{\sigma(i)}$. Then, we can sum up over all $i \in N$ and obtain that

$$\sum_{i \in N} v_{i\pi(i)} - \sum_{i \in N} p_{\pi(i)} \geq \sum_{i \in N} v_{i\sigma(i)} - \sum_{i \in N} p_{\sigma(i)}$$
$$\sum_{i \in N} v_{i\pi(i)} - R \geq \sum_{i \in N} v_{i\sigma(i)} - R$$
$$\sum_{i \in N} v_{i\pi(i)} \geq \sum_{i \in N} v_{i\sigma(i)}$$

Therefore, we can conclude that π is welfare-maximizing. □

Lemma 3. If $(\pi, \mathbf{p})$ is an EF solution and σ is a welfare-maximizing assignment, then $(\sigma, \mathbf{p})$ is an EF solution.

Consider an algorithm that finds a welfare maximizing assignment π, and then finds prices $\mathbf{p}$ that satisfies the EF constraint.

Theorem 3. *The algorithm always returns an EF solution and can be implemented in polynomial time.*

Proof. First, we need to confirm that for any welfare-maximizing assignment π, there exists $\mathbf{p}$ that satisfies the EF constraint for π. We know that an EF solution $(\sigma, \mathbf{p})$ exists, then by Lemma 3, $(\pi, \mathbf{p})$ is also EF, so there exists $\mathbf{p}$ that satisfies the EF constraint for π.

Then, we consider how to implement this algorithm in polynomial time: the first part-find a welfare maximizing assignment-can be done through solving the max weight matching problem in a bipartite graph. In particular, the bipartite graph has players on one side and the rooms on the other side, and weight on any edge (i, r) for player $i \in N$ and room r is v_{ir}. Any max weight matching of this graph corresponds to a welfare maximizing assignment. The second part-find the $\mathbf{p}$ that satisfies the EF constraint-can be done through solving a system of linear inequalities. □

2.3 Optimal EF Solutions

- **Straw Man Solution**: Maximize sum of utilities subject to envy-freeness
  - This is not helpful since all EF solution maximize sum of utilities!
- **Maxmin Solution**: Maximize minimum utility subject to envy-freeness
- **Equitable Solution**: Minimize maximum difference in utilities subject to envy-freeness

Theorem 4. *The maximin and equitable solutions can be computed in polynomial time.*

Theorem 5. *The maximin solution is unique.*

Remark 3. When we say the maximin solution is unique, we mean that while there may be different solutions with varying assignments and prices, the utilities that players receive remain the same across all these solutions.

Theorem 6. *The maximin solution is equitable, but not vice versa.*

3 Comparison Between Two Models

The following table compares the Sperner-based approach (Approach 1) with the quasi-linear utility approach (Approach 2).

	Approach 1	**Approach 2**
Expressiveness	More expressive and general.	Less expressive (e.g. fails to capture budget constraints).
Optimal Envy-Free Solutions	There is no clear way to choose between envy-free solutions.	There are clear ways to choose between envy-free solutions.
Computability	No computational guarantees.	Guarantees polynomial-time algorithm to compute envy-free solutions
Preferences Revealing Difficulties	Requires long sequences of questions to the players (i.e., what do you prefer at this price?)	Only requires the players to reveal their valuations.