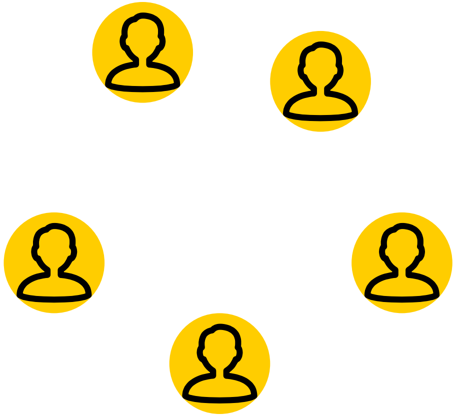
Achieving Fairness in Multi-Round Items Allocation

Gili, Itai, Jack, Shirley

CS 238 Optimized Democracy

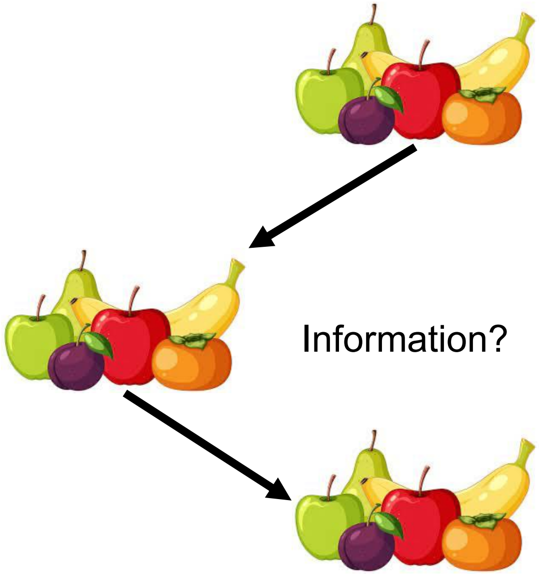
Recall Fair Division

Set of *G* indivisible goods, divided and given to a group of *n* players



Variations on Fair Division

Multi-round



Information?

Adjustments

Restrictions on goods

Real World Examples

The Greater Boston FOOD BANK

https://www.npr.org/2020/11/20/937026003/pfizer-asks-fda-to-approve-its-covid-19-vaccine-for-emergency-use

https://www.lawtechnologytoday.org/2020/09/the-corner-office-a-rusty-artifact-of-the-past/

Previous Research

- Bounding the number of adjustments needed to achieve free at every round (He et al., 2019)
- Bounding the maximum envy between two agents at the end of each round, and decreasing it over time (Benade et al., 2018)
- Analyzing strategy-proofness and envy-freeness in a food bank setting (Alexandrov et al. 2015)

Our Model

Focuses on a **multi-round** and **informed** setting (goods come in batches), and allows agents to **discount the future** with the goal of achieving **low envy** at the end of **every round**.

2 Preliminaries

Definitions

- Let $\{a_1, a_2, \ldots, a_n\}$ be a set of n agents
- Let $\{G^1, \ldots, G^t\}$ be a sequence of T batches of goods such that for every round t, $G^t = \{g_1^{\ t}, \ldots, g_t^{\ mt}\}$
- Let $A = (A_1, \ldots, A_n)$ be an allocation of goods, where A_i is the bundle of goods allocated to agent i.
- EF1: envy-free up to one item

Definitions

- Informed setting: Assume items arrive in order over T rounds
- An algorithm is EF1 if it is EF1 for every round
- Agents have knowledge of the future, but they discount the future and prefer items now
- Let $\delta \in (0,1)$ be the discount factor
- The utility is the sum of $U_i(A_i^t) = \sum_{t'=t}^{T} \delta^{t'-t} v_i(A_i^{t'} \cap G^{t'})$ ns of all items they receive:

Example

- Suppose we have a food bank that receives:
  - 5 apples and 3 oranges at T=1
  - 3 apples and 4 oranges at T=2
  - 7 apples and 2 oranges at T=3
- Suppose two individuals have the following valuations:
  - Apples at 0.2 and oranges at 0.4
  - Apples at 0.6 and oranges at 0.3
- Can we find an algorithm that guarantees EF1?

Example

- Suppose $\delta = 0.5$

	Agent 1	Agent 2
Round 1 3 apples, 2 oranges	3 (0.2) + 2(0.4)	3(0.6) + 2(0.3)
Round 2 4 apples, 1 orange	0.5[4 (0.2) + 1(0.4)]	0.5[4 (0.6) + 1(0.3)]
Round 3 1 apple, 2 oranges	0.5^2[1 (0.2) + 2(0.4)]	0.5^2[1(0.6) + 2(0.3)]
	2.25	4.05

3

No Adjustments

The Impossibility Result & Backwards Induction Envy Balancing

The Impossibility Result

Theorem *No algorithm can guarantee EF1 in multi-round settings with more than two agents*

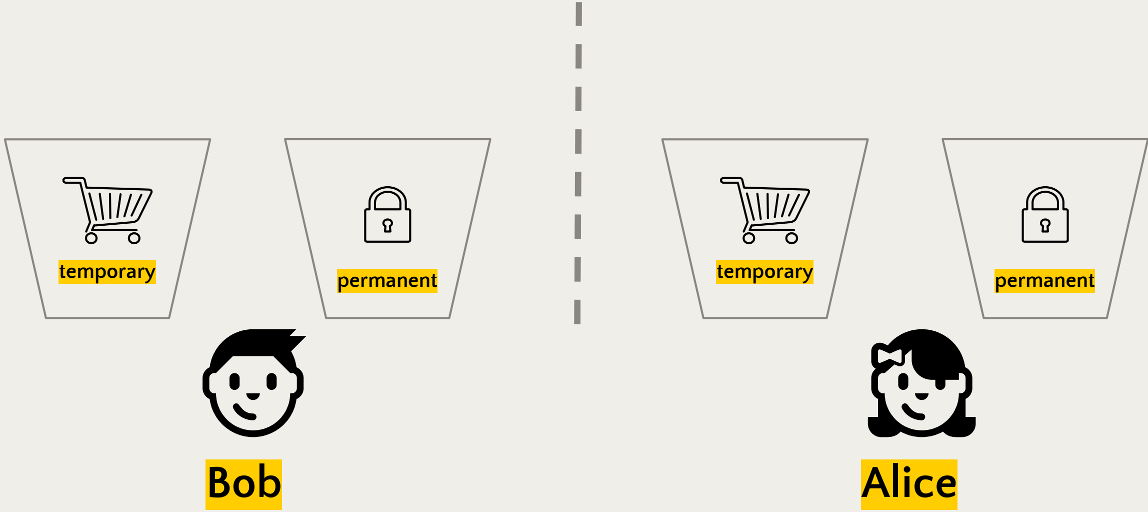
Backwards Induction Envy Balancing Algorithm

- **Two-Player Setting:** Qualitatively Different
- **Backwards Induction Envy Balancing Algorithm**
  - Ensures envy-freeness up to one item (EF1)
- Iteratively in reverse order builds EF1 allocations for two agents (extending He et al. (2019))

Backwards Induction Envy Balancing Algorithm

- Reverse order
- Apply RoundRobin to goods based on envy
- Construct final allocation for each round

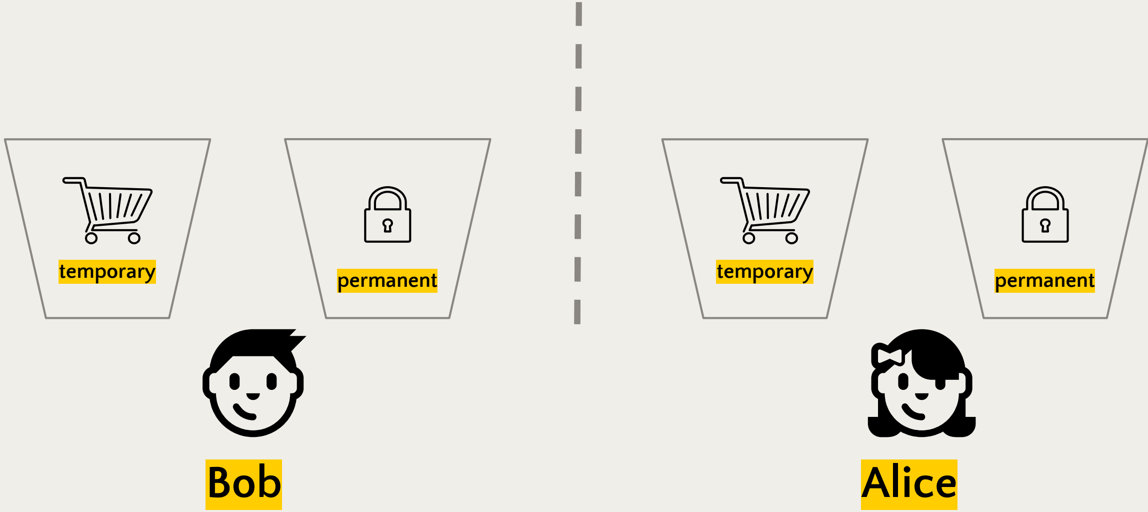
Round 1	 
Round 2	 
Round 3	 
Round 4	 



Backwards Induction Envy Balancing Algorithm

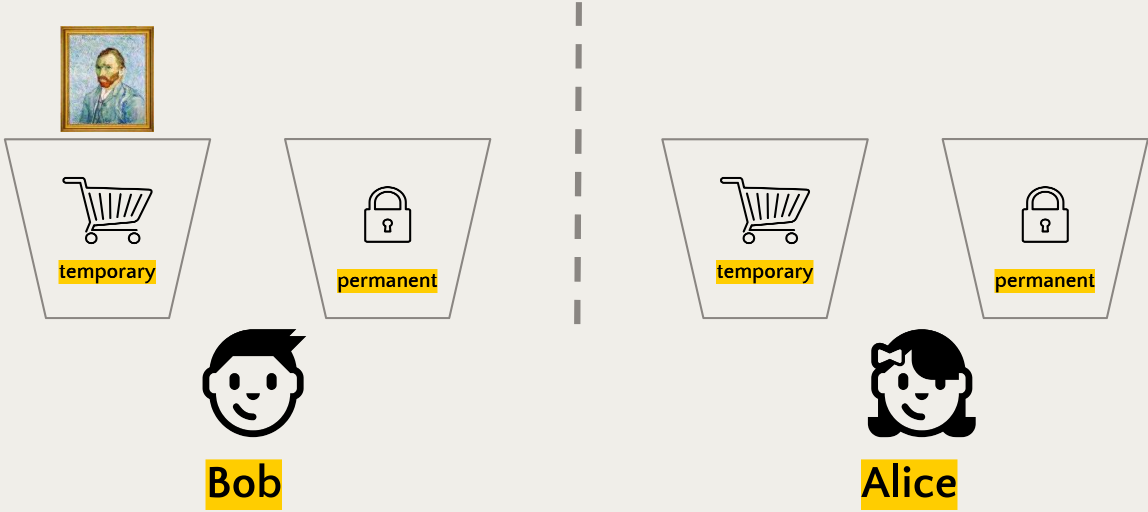
- Apply RoundRobin to goods based on envy
  - EF? Move items to permanent
  - Both envy? substitute the baskets
  - One player envied? continue

Round 1		
Round 2		
Round 3		
Round 4		



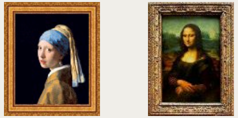
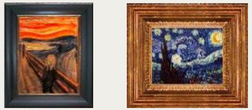
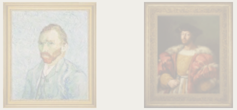
- Apply RoundRobin to goods based on envy
  - EF? Move items to permanent
  - Both envy? substitute the baskets
  - One player envied? continue

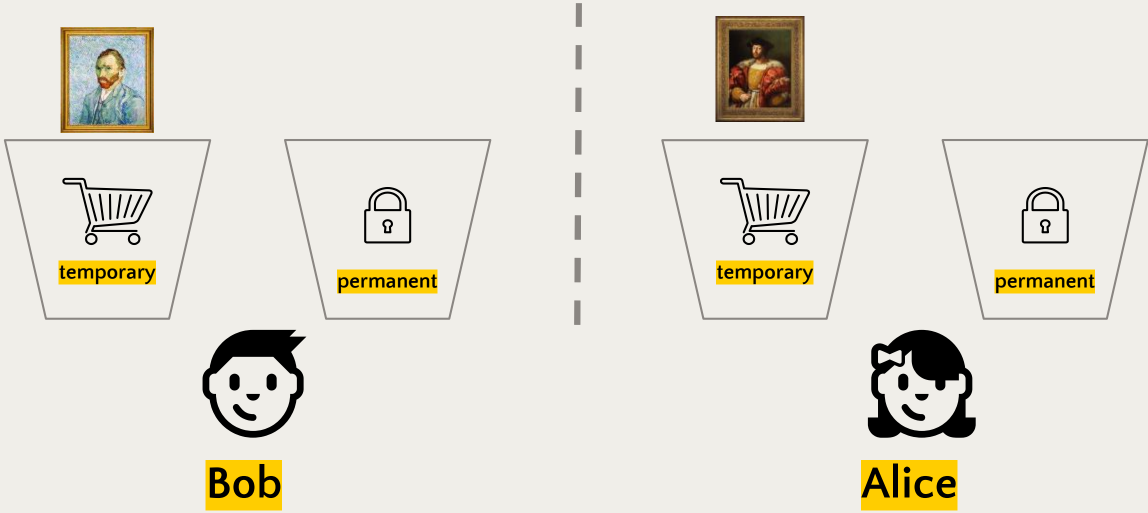
Round 1		
Round 2		
Round 3		
Round 4		



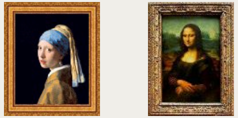
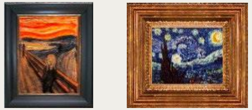
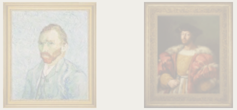
Backwards Induction Envy Balancing Algorithm

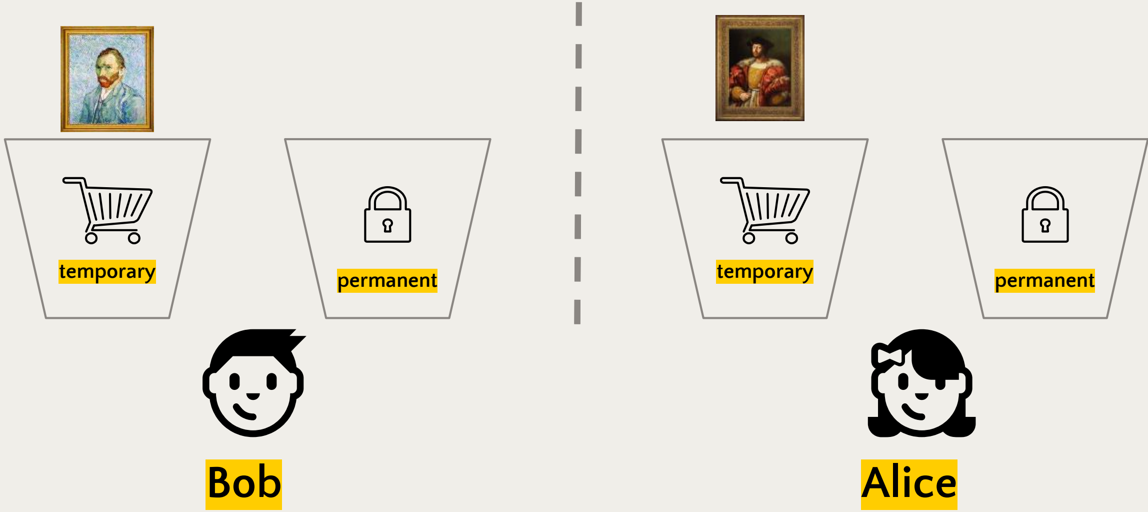
- Apply RoundRobin to goods based on envy
  - EF? Move items to permanent
  - Both envy? substitute the baskets
  - One player envied? continue

Round 1	
Round 2	
Round 3	
Round 4	



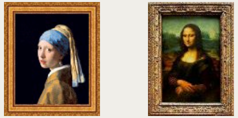
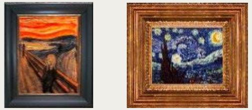
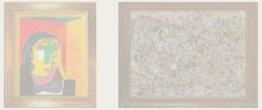
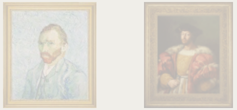
- Apply RoundRobin to goods based on envy
  - EF? Move items to permanent
  - Both envy? substitute the baskets
  - One player envied? continue

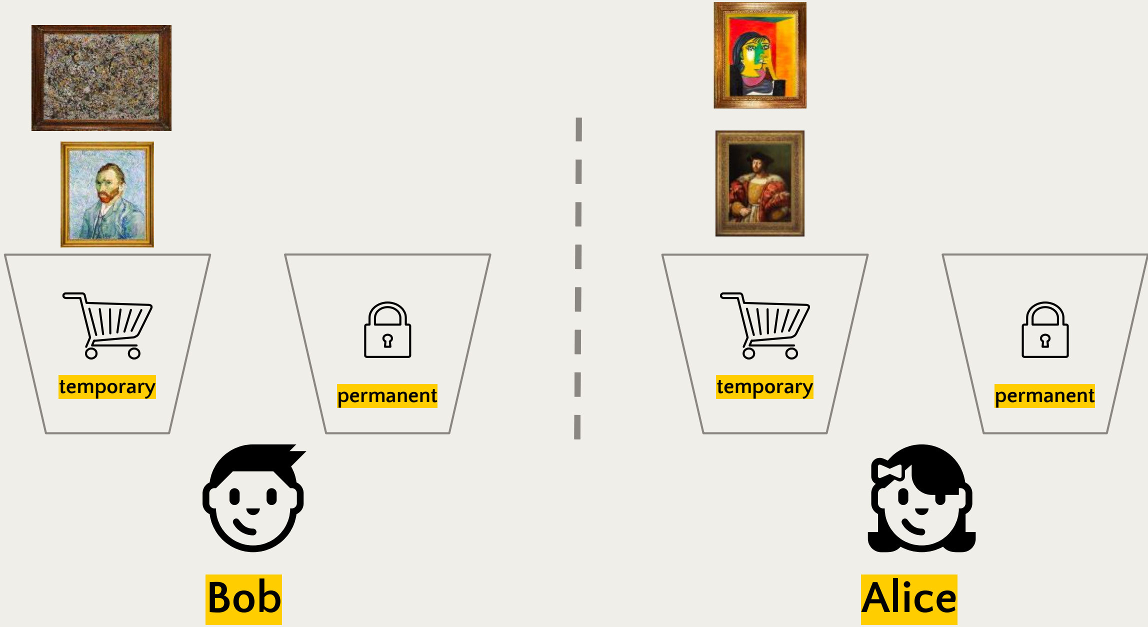
Round 1	
Round 2	
Round 3	
Round 4	



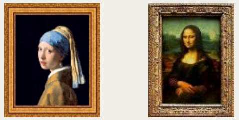
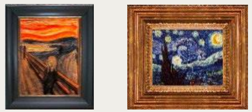
Backwards Induction Envy Balancing Algorithm

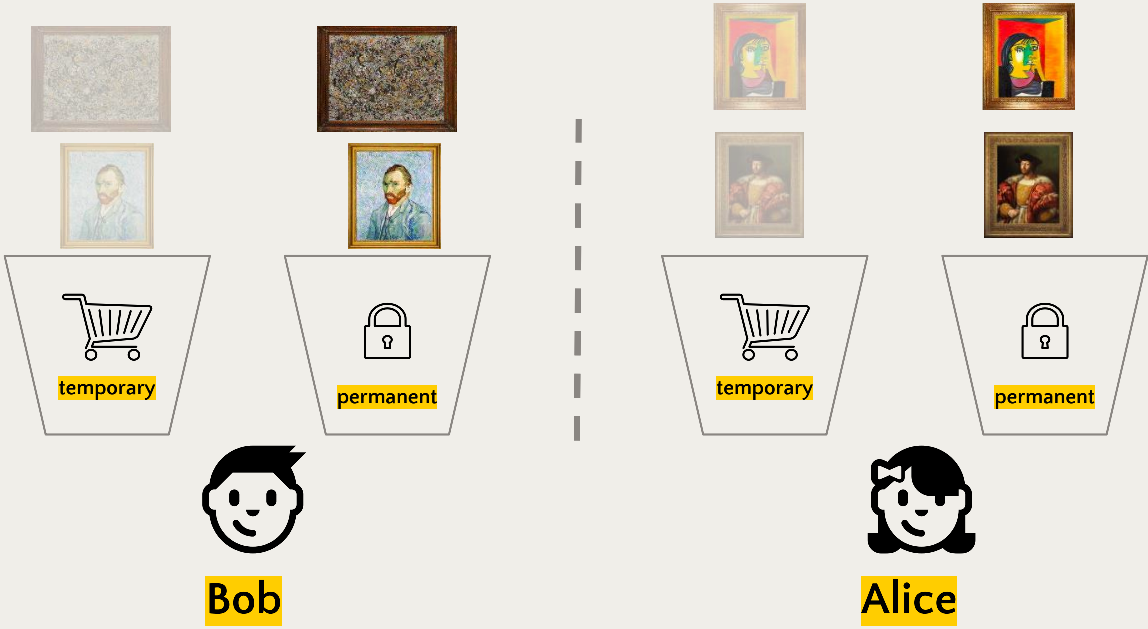
- Apply RoundRobin to goods based on envy
  - EF? Move items to permanent
  - Both envy? substitute the baskets
  - One player envied? continue

Round 1	
Round 2	
Round 3	
Round 4	



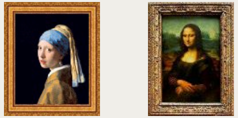
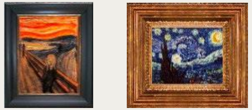
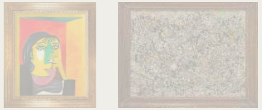
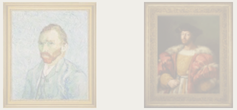
- Apply RoundRobin to goods based on envy
  - EF? Move items to permanent
  - Both envy? substitute the baskets
  - One player envied? continue

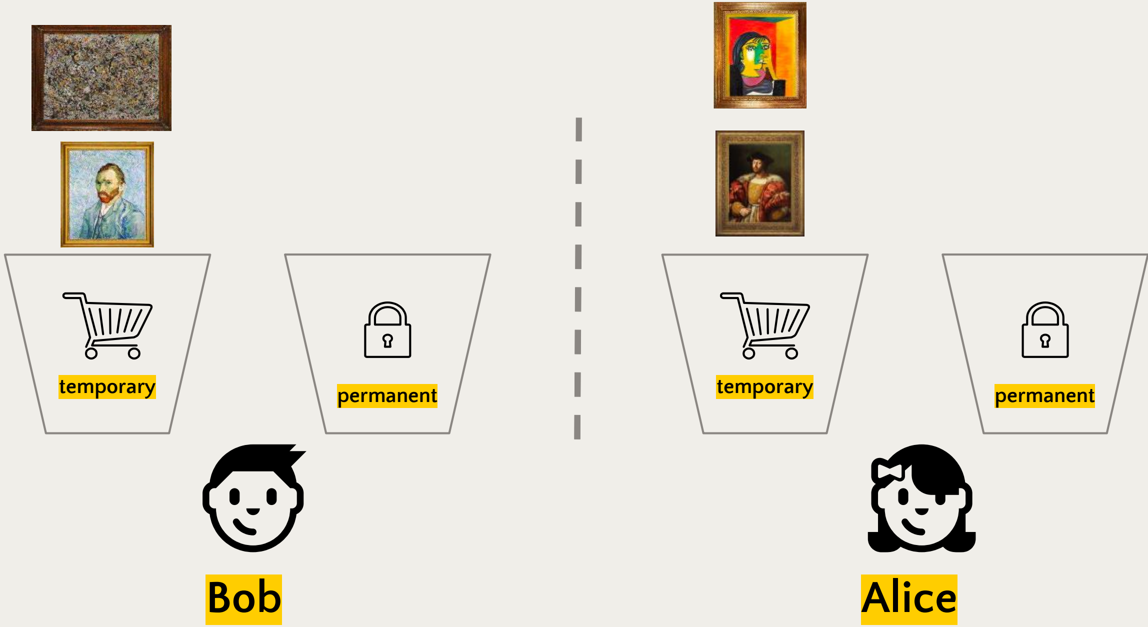
Round 1	
Round 2	
Round 3	
Round 4	



Backwards Induction Envy Balancing Algorithm

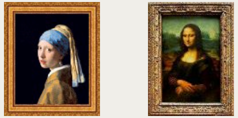
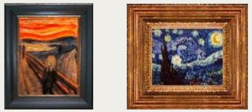
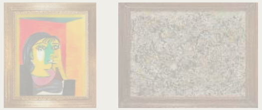
- Apply RoundRobin to goods based on envy
  - EF? Move items to permanent
  - Both envy? substitute the baskets
  - One player envied? continue

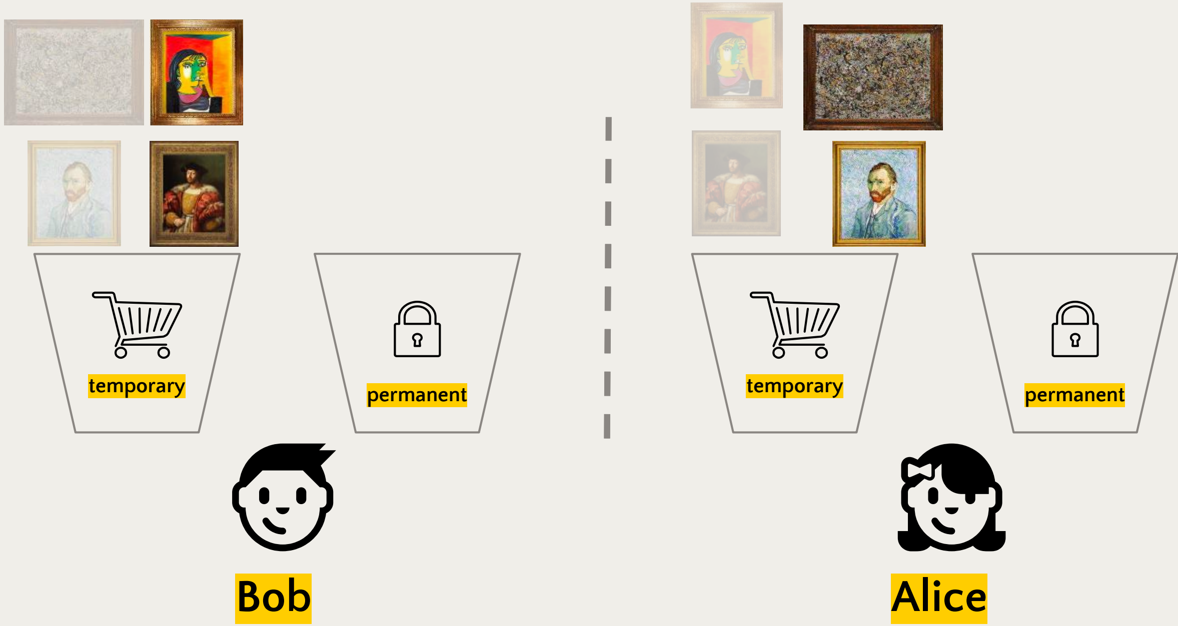
Round 1	
Round 2	
Round 3	
Round 4	



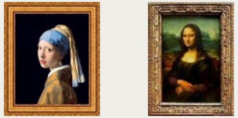
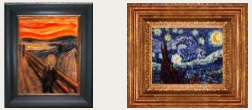
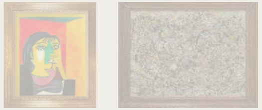
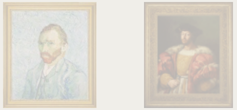
Backwards Induction Envy Balancing Algorithm

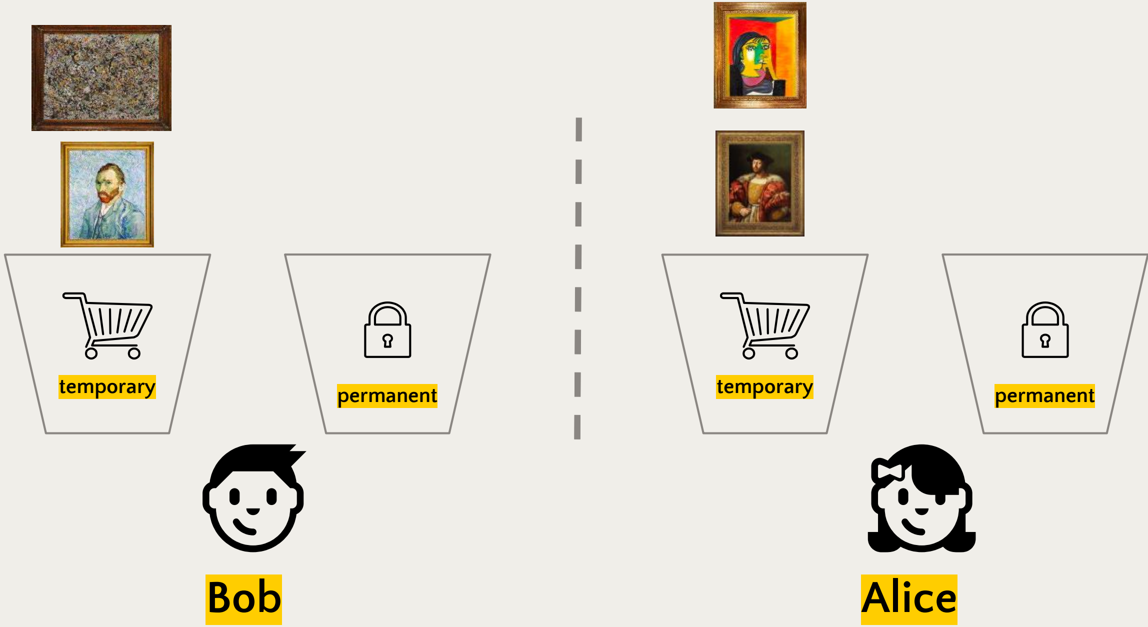
- Apply RoundRobin to goods based on envy
  - EF? Move items to permanent
  - Both envy? substitute the baskets
  - One player envied? continue

Round 1	
Round 2	
Round 3	
Round 4	



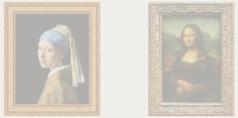
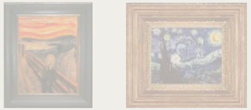
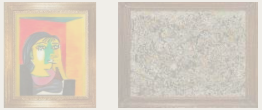
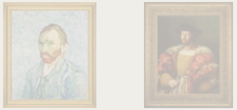
- Apply RoundRobin to goods based on envy
  - EF? Move items to permanent
  - Both envy? substitute the baskets
  - One player envied? continue

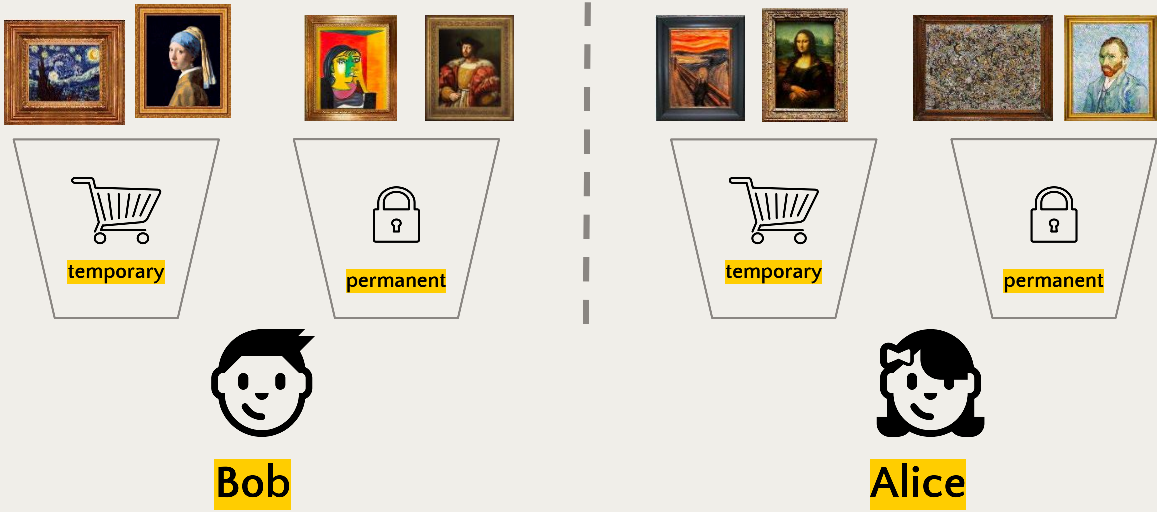
Round 1	
Round 2	
Round 3	
Round 4	



Backwards Induction Envy Balancing Algorithm

- Apply RoundRobin to goods based on envy
  - EF? Move items to permanent
  - Both envy? substitute the baskets
  - One player envied? continue

Round 1	
Round 2	
Round 3	
Round 4	



4

Adjustments

Introducing: Double Round Robin

Setting

- Impossible to achieve EF1 with $n>2$ players
- New tactic: allow **adjustments** to allocations
- Let T = # items, k = # rounds

Theorem: There exists an algorithm that achieves EF1 in every round, using $O(T^{3/2}/\sqrt{k})$ adjustments

Algorithm Profile

- Name: ‘Double Round Robin’

Algorithm Profile

- Name: ‘Double Round Robin’
- Personality traits
  - ⚖️ Balanced: Has a main pile and a side pile

Algorithm Profile

- Name: ‘Double Round Robin’
- <u>Personality traits</u>
  - Balanced: Has a main pile and a side pile
  - Flexible: Allows a complete reallocation of every pile in every round

Algorithm Profile

- Name: ‘Double Round Robin’
- Personality traits

Balanced: Has a main pile and a side pile
Flexible: Allows a complete reallocation of every pile in every round
Growth-mindset: Always adds side pile to main pile eventually

Algorithm Profile

- Name: ‘Double Round Robin’
- <u>Personality traits</u>
  - Balanced: Has a main pile and a side pile
  - Flexible: Allows a complete reallocation of every pile in every round
  - Growth-mindset: Always adds side pile to main pile eventually

Algorithm 2 Double Round Robin

Require: v_i for each agent a_i

$M \leftarrow \emptyset, S \leftarrow \emptyset$
for $t = T/m$ to 1 **do**
  $S \leftarrow S \cup \{G_t\}$
  if $|S| \geq \sqrt{k} \cdot \sqrt{T}$ **then**
    $M \leftarrow M \cup S$
    $S \leftarrow \emptyset$
  end if
  $A_M \leftarrow \text{RoundRobin}(M, a_1 > ... > a_n)$
  $A_S \leftarrow \text{RoundRobin}(S, a_n > ... > a_1)$
  Let A^t be the combination of allocations A_S and A_M
end for
return $[A^1; A^2; \ldots; A^T]$

Extensions

- Can we do better with restricted classes of valuations?
  - E.g. binary valuations
- Are there interesting bounds on other metrics of [approximate] fairness?

Thanks!

Any *questions* ?

- Gili, Itai, Jack, Shirley