Fair Division of Indivisible Items (Part II)

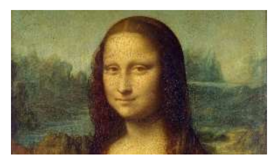
CS 580

Instructor: Ruta Mehta

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Most slides are curtesy Prof. J. Garg

- N: set of n agents, 1 ,…, n,
- M: set of m indivisible items (like cell phone, painting, etc.)



- Agent i has a valuation function $v_i : 2^m \rightarrow \mathbb{R}$ over subsets of items
  - Monotone: the more the happier

Proportionality

- A set N of n agents, a set M of m indivisible goods
- Proportionality: Allocation $A = (A_1, \dots, A_n)$ is proportional if each agent gets at least $1/n$ share of all items:

$$v_i(A_i) \geq \frac{v_i(M)}{n}, \qquad \forall i \in N$$

Cut-and-choose?

Maximin Share (MMS) [B11]

Cut-and-choose.

- Suppose we allow agent i to propose a partition of items into n bundles with the condition that i will choose at the end.
- Clearly, i partitions items in a way that maximizes the value of her least preferred bundle.
- $\mu_i :=$ Maximum value of $i's$ least preferred bundle

Maximin Share (MMS) [B11]

Cut-and-choose.

- Suppose we allow agent i to propose a partition of items into n bundles with the condition that i will choose at the end
- Clearly, i partitions items in a way that maximizes the value of her least preferred bundle
- $\mu_i :=$ Maximum value of $i's$ least preferred bundle

- $\Pi :=$ Set of all partitions of items into n bundles
- $\mu_i := \max_{A \in \Pi} \min_{A_k \in A} v_i(A_k)$

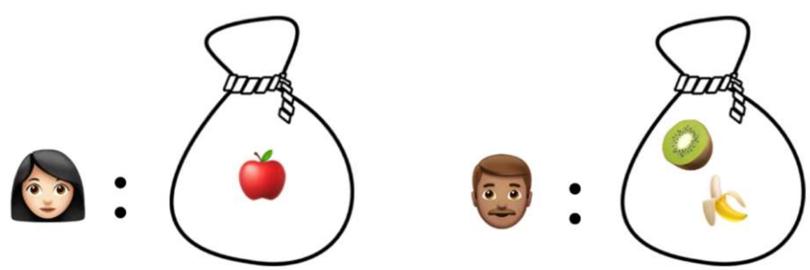
- MMS Allocation: A is called MMS if $v_i(A_i) \geq \mu_i$, $\forall i$
- Additive valuations: $v_i(A_i) = \sum_{j \in A_i} v_{ij}$

MMS value/partition/allocation

Agent\Items	🍎	🍌	🥝
👩	3	1	2
👨	4	4	5

👩	🍎 A	🍌🥝 B
Value	3	3
MMS Value	3	

👨	🍎🍌	🥝
Value	8	5
MMS Value	5	



MMS value/partition/allocation

Agent\Items	🍎	🍌	🥝
👩	3	1	2
👨	4	4	5

👩	🍎	🍌🥝
Value	3	3
MMS Value	3	

👨	🍎🍌	🥝
Value	8	5
MMS Value	5	

👩 : 🍎 👨 : 🥝🍌

Finding MMS value is NP-hard!

What is Known?

- PTAS for finding MMS value [W97]

Existence (MMS allocation)?

- $n = 2$: yes 

  $\Longrightarrow$ A PTAS to find $(1-\epsilon)$-MMS allocation for any $\epsilon > 0$

- $n \geq 3$: NO [PW14]

What is Known?

- PTAS for finding MMS value [W97]

Existence (MMS allocation)?

- $n = 2$: yes EXERCISE

  $\Longrightarrow$ A PTAS to find $(1-\epsilon)$-MMS allocation for any $\epsilon > 0$
- $n \geq 3$: NO [PW14]

- α-MMS allocation for $\alpha \in [0,1]$: $v_i(A_i) \geq \alpha . \mu_i$
  - 2/3-MMS exists [PW14, AMNS17, BK17, KPW18, GMT18]
  - 3/4-MMS exists [GHSSY18]
  - $(3/4 + 1/(12n))$-MMS exists [GT20]

$\frac{39}{40}$ - MMS does not exist

Properties

- Normalized valuations
  - Scale free: $v_{ij} \leftarrow c.v_{ij}, \forall j \in M$
  - $\sum_j v_{ij} = n \quad \Rightarrow \quad \mu_i \leq 1$?

$= v_i(M)$

MMS partition to agent i:

$\underbrace{v_i(A_1) \geq \dots \geq v_i(A_n)}$ $= \mu_i > 1$

$\sum_k v_i(A_k) = v_i(M) > n$!

Properties

- Normalized valuations
  - Scale free: $v_{ij} \leftarrow c.v_{ij}, \forall j \in M$
  - $\sum_j v_{ij} = n \quad \Rightarrow \quad \mu_i \leq 1$
- Ordered Instance: We can assume that agents' order of preferences for items is same: $v_{i1} \geq v_{i2} \geq \cdots v_{im}, \forall i \in N$

Properties

- Normalized valuations
  - Scale free: $v_{ij} \leftarrow c.v_{ij}, \forall j \in M$
  - $\sum_j v_{ij} = n \quad \Rightarrow \quad \mu_i \leq 1$
- Ordered Instance: We can assume that agents' order of preferences for items is same: $v_{i1} \geq v_{i2} \geq \cdots v_{im}, \forall i \in N$

	🍎	🍌	🍓	🍍	🥥
👩	3	1	2	5	4
👨	4	4	5	3	2

	1	**2**	**3**	**4**	**5**
👩	5	4	3	2	1
👨	5	4	4	3	2

Challenge

- Allocation of high-value items!
- If for all $i \in N$
  - $v_i(M) = n \;\Rightarrow \mu_i \leq 1$
  - $\boldsymbol{v_{ij} \leq \epsilon, \forall i, j}$ $\Rightarrow$ $\max_{i,j} v_{ij} \leq \epsilon$

$v_{ij} \leq \epsilon, \forall i, j$

$g_1 > g_2 > \cdots > g_m$

Claim: After round k, if i remains then v_i(remaining goods) $\geq n - k$.

$v_i(M \setminus \{g_1, g_2, g_3, g_4\}) > (n-1)$

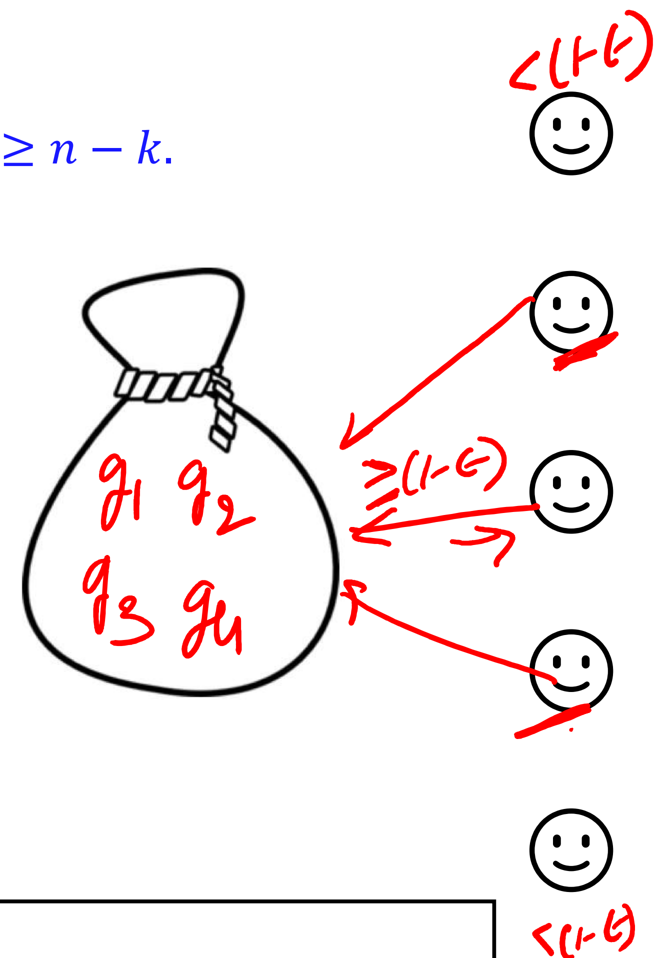
$\Uparrow$

$\forall i: \; v_i(\{g_1, g_2, g_3\}) < (1-\epsilon)$

$v_i(g_4) \leq \epsilon$

$v_i(\{g_1, g_2, g_3, g_4\}) < (1-\epsilon) + \epsilon$

$< 1.$



Bag Filling Algorithm:

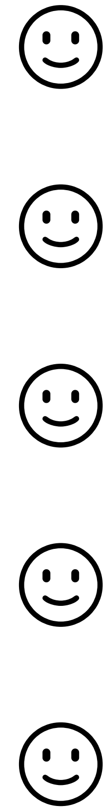
Repeat until every agent is assigned a bag

- Start with an empty bag B
- Keep adding items to B until some agent i values it $\geq (1-\epsilon)$
- Assign B to i and remove both

$v_{ij} \leq \epsilon, \forall i, j$ $\qquad$ $v_i(M) = n$

Claim: After round k, if i remains then v_i(remaining goods) $\geq n - k$.

Pf: In every round, value of the assigned set of items for agent $i < 1$.



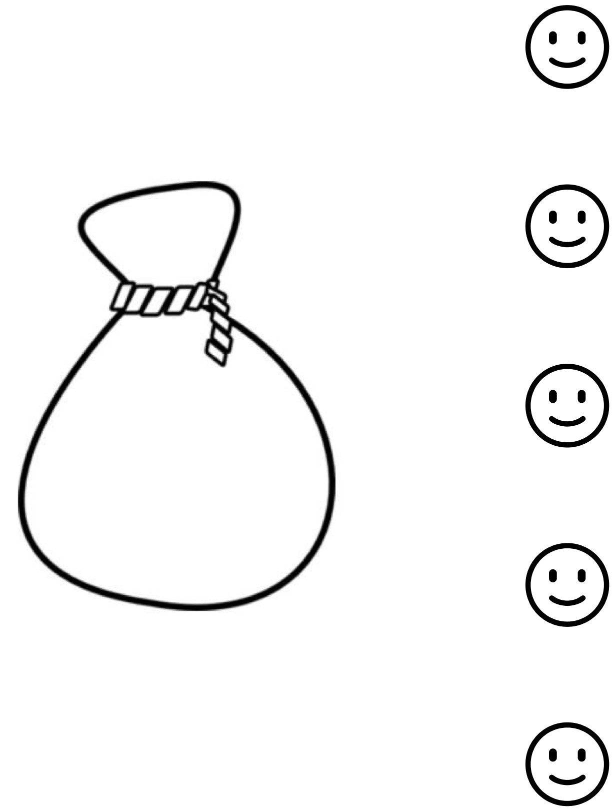
Bag Filling Algorithm:

Repeat until every agent is assigned a bag

- Start with an empty bag B
- Keep adding items to B until some agent i values it $\geq (1 - \epsilon)$
- Assign B to i and remove them both

$v_{ij} \leq \epsilon, \forall i, j$

Thm: Every agent gets at least $(1 - \epsilon)$.



Bag Filling Algorithm:

Repeat until every agent is assigned a bag

- Start with an empty bag B
- Keep adding items to B until some agent i values it $\geq (1 - \epsilon)$
- Assign B to i and remove them both

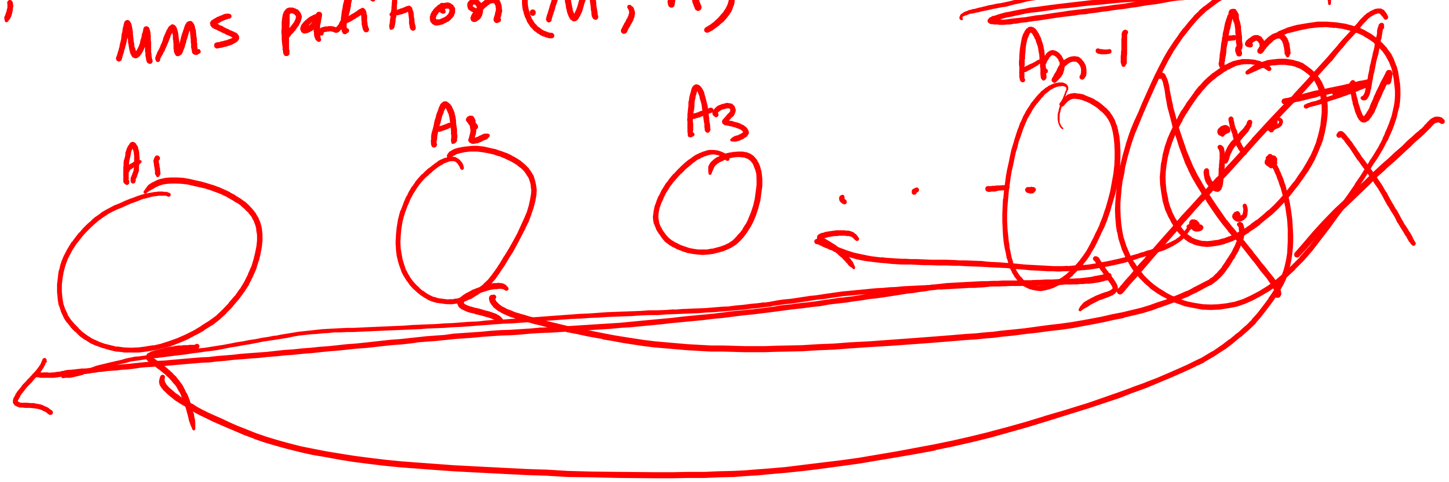
Warm Up: 1/2-MMS Allocation

- If all $v_{ij} \leq 1/2$ then?
  - Done, using bag filling.

$V_{ij} > 1/2$ $\qquad$ $j \Rightarrow i$

- What if some $v_{ij} > \frac{1}{2}$?

i' $\quad$ MMS partition$(M, n) \longrightarrow (M \setminus j, n-1)$



Valid Reductions

- Normalized valuations
  - Scale free: $v_{ij} \leftarrow c.v_{ij}, \forall j \in M$
  - $\sum_j v_{ij} = n \quad \Rightarrow \quad \mu_i \leq 1$
- Ordered Instance: Agents' order of preferences for items is same: $v_{i1} \geq v_{i2} \geq \cdots v_{im}, \forall i \in N$

- Valid Reduction (α-MMS): If there exists $S \subseteq M$ and $i^* \in N$
  - $v_{i^*}(S) \geq \alpha.\mu_{i^*}^n(M)$
  - $\mu_i^{n-1}(M \setminus S) \geq \mu_i^n(M), \forall i \neq i^*$

$\Rightarrow$ reduce the instance size!

Claim. Suppose agent i $\neq i^*$ gets A_i in the be an α-MMS allocation of $M \setminus S$ to agents $N \setminus \{i^*\}$, then $(A_1, \ldots, A_{i^*-1}, S, A_{i^*+1}, \ldots, A_n)$ is an α-MMS allocation in the original instance.

Pf: $\forall i \neq i^*$

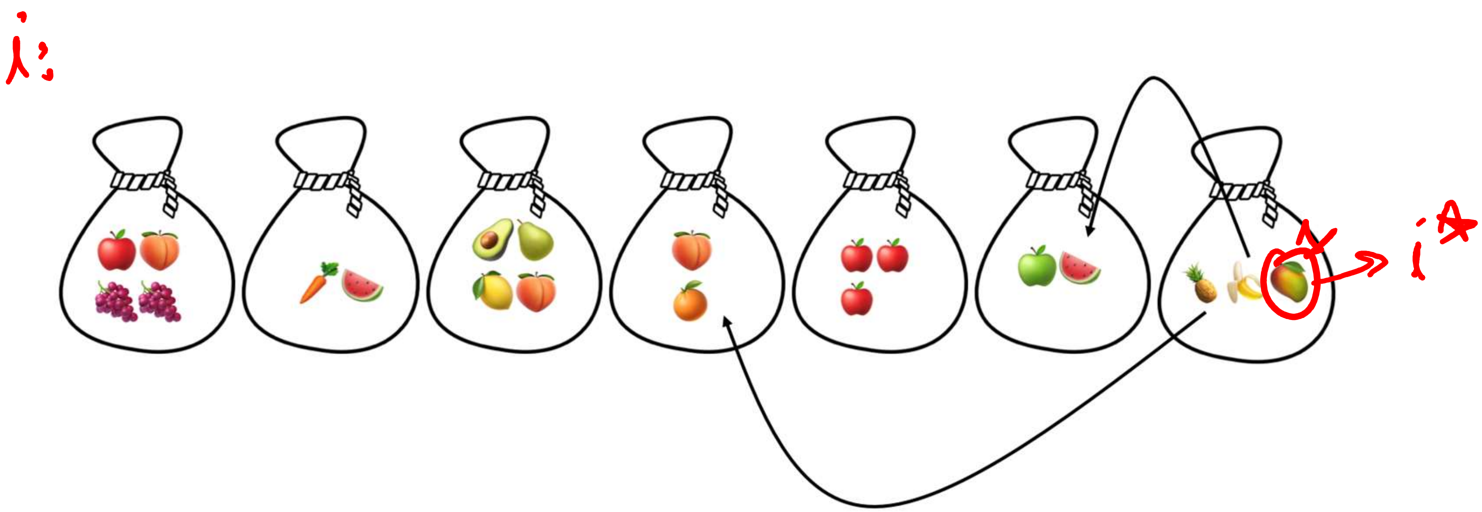
$\underline{V_i(A_i)} \geq \alpha \mu_i^{n-1}(M \setminus S) \geq \underline{\alpha . \mu_i^n(M)}$

$V_{i^*}(S) \geq \alpha \mu_{i^*}^n(M).$

1/2-MMS Allocation

Step 1: Valid Reductions

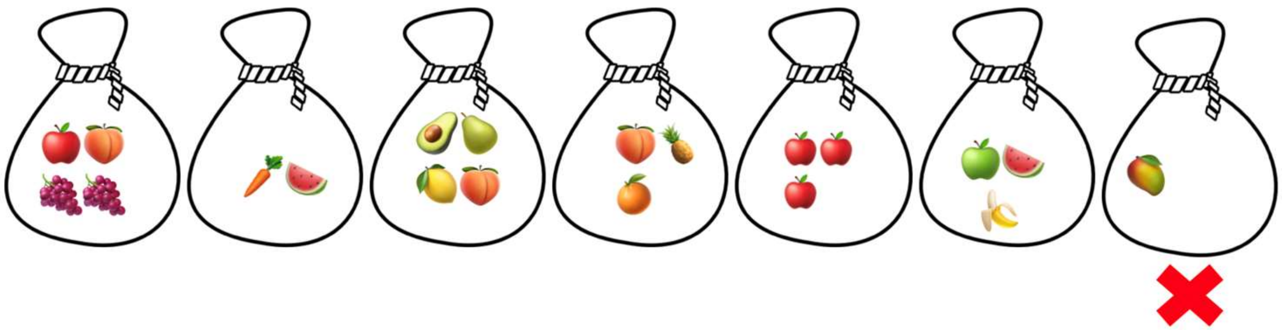
- If $v_{i^*1} \geq 1/2$ then assign item 1 to i^*



1/2-MMS Allocation

Step 1: Valid Reductions

- If $v_{i^*1} \geq 1/2$ then assign item 1 to i^*



1/2-MMS Allocation

- Re-normalization

$\epsilon = 1/2$

Step 0: Normalized Valuations: $\sum_j v_{ij} = n \Rightarrow \mu_i \leq 1$

Step 1: Valid Reductions $(1-\epsilon)$

- If $v_{i^*1} \geq 1/2$ then assign item 1 to i^*. Remove good 1 and agent i^*
- After every valid reduction, normalize valuations

Step 2: Bag Filling $v_{ij} < \frac{1}{2} = \epsilon$

2/3-MMS Allocation [GMT19]

$= (1-\epsilon) \Rightarrow \epsilon = 1/3$

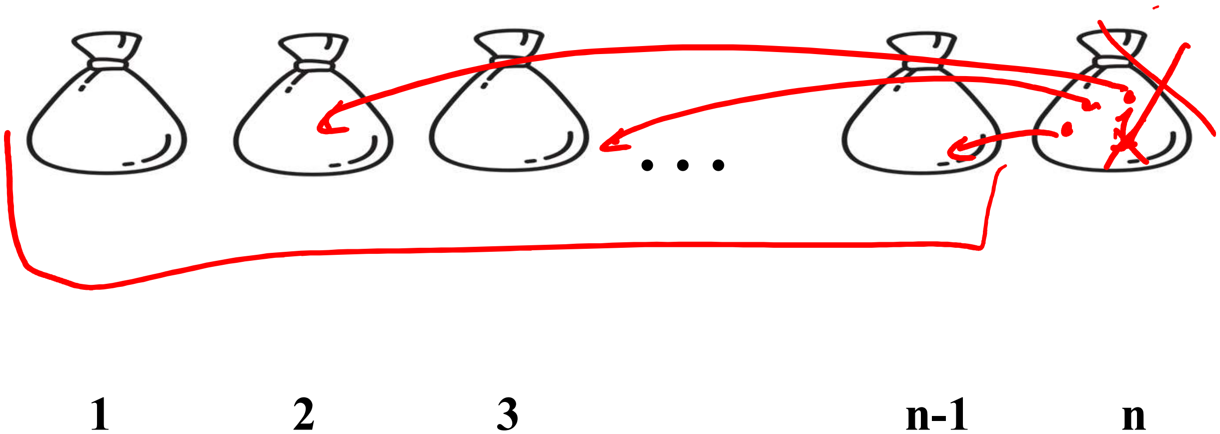
- If all $v_{ij} \leq 1/3$ then?

Step 1: Valid Reductions

- If $v_{i^*1} \geq 2/3$ then assign item 1 to i^*

agent i:

mms part.



2/3-MMS Allocation [GMT19]

Step 1: Valid Reductions

- If $v_{i^*1} \geq 2/3$ then assign item 1 to i^*
- If $v_{i^*n} + v_{i^*(n+1)} \geq 2/3$ then assign $\{n, n+1\}$ to i^*

Q: why valid reduction?

For agent $i \neq i^*$, let the MMS defining partition be

Case I: $\exists k$, $g_n, g_{n+1} \in A_k$

1 2 3 n-1 n

Then move $A_k \setminus \{n, n+1\}$ to other bundles & remove A_k

Partition of $M \setminus \{n, n+1\}$ into $(n-1)$ bundles each with value $\geq \mu_i$

2/3-MMS Allocation [GMT19]

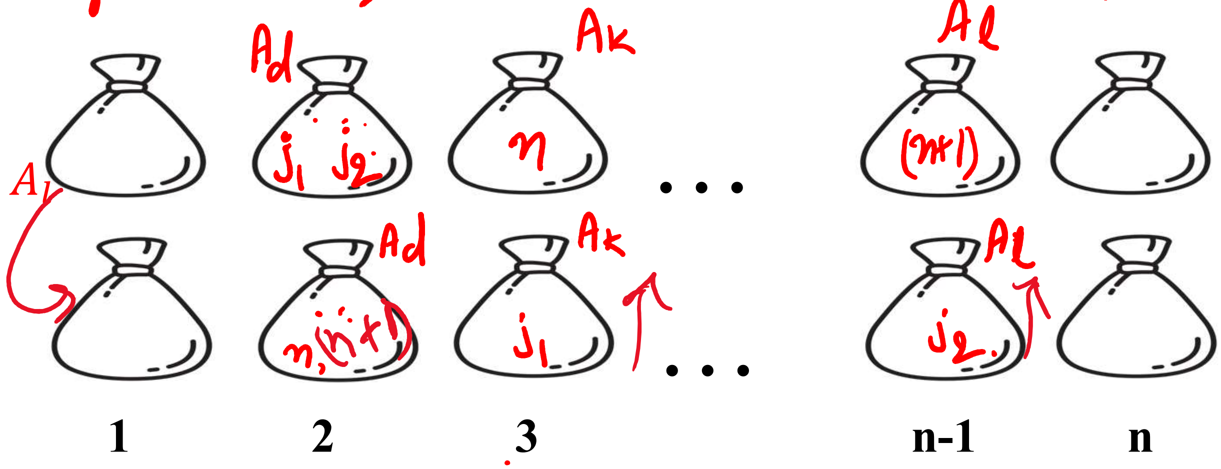
Step 1: Valid Reductions

- If $v_{i^*1} \geq 2/3$ then assign item 1 to i^*
- If $v_{i^*n} + v_{i^*(n+1)} \geq 2/3$ then assign $\{n, n+1\}$ to i^*

Q: Why valid reduction?

For agent $i \neq i^*$, let the MMS defining partition be

Case II:
$n \in A_k$
$(n+1) \in A_l$



$\exists A_d$, with items $j_1 < j_2 \leq (n+1)$. Then, swap items j_1 and n, and items j_2 and (n+1). This may only increase $v_i(A_k)$ & $v_i(A_l)$ because $v_i(j_1) \geq v_i(n)$ & $v_i(j_2) \geq v_i(n+1)$.

2/3-MMS Allocation [GMT19]

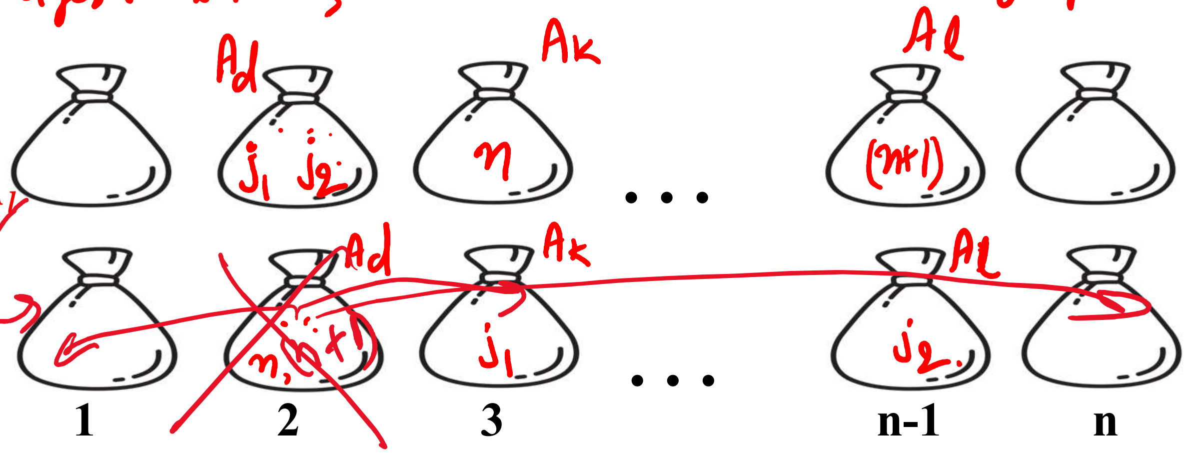
Step 1: Valid Reductions

- If $v_{i^*1} \geq 2/3$ then assign item 1 to i^*
- If $v_{i^*n} + v_{i^*(n+1)} \geq 2/3$ then assign $\{n, n+1\}$ to i^*

Q: Why valid reduction?

For agent $i \neq i^*$, let the MMS defining partition be

Case II:
$n \in A_k$
$(n+1) \in A_l$



$\exists A_d$, with items $j_1 < j_2 \leq (n+1)$.
Then, swap items j_1 and n, and items j_2 and (n+1).
Move remaining items of A_d to other bundles and remove A_d.

2/3-MMS Allocation [GMT19]

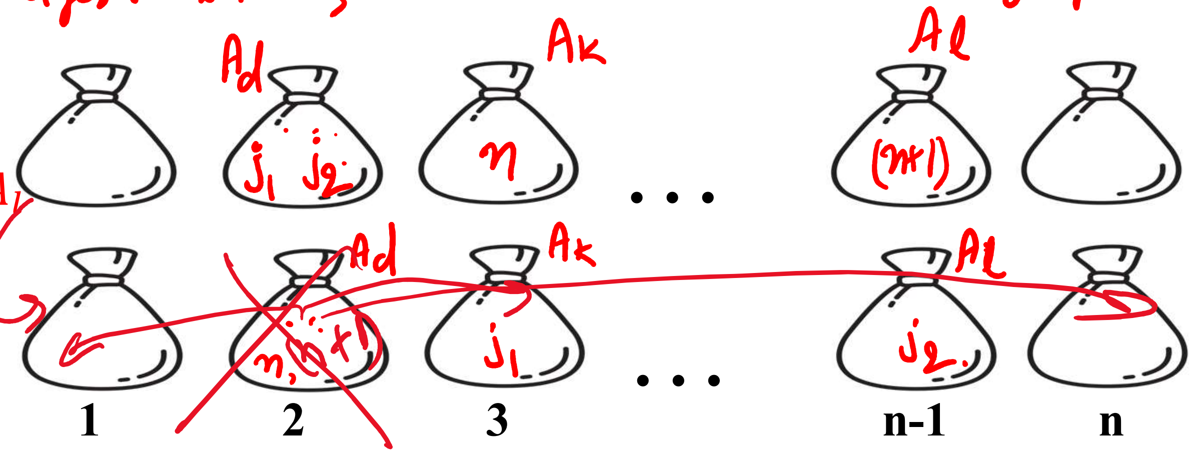
Step 1: Valid Reductions

- If $v_{i^*1} \geq 2/3$ then assign item 1 to i^*
- If $v_{i^*n} + v_{i^*(n+1)} \geq 2/3$ then assign $\{n, n+1\}$ to i^*

Q: Why valid reduction?

For agent $i \neq i^*$, let the MMS defining partition be

Case II:
$n \in A_k$
$(n+1) \in A_l$



$\exists A_d$, with items $j_1 < j_2 \leq (n+1)$. Then, swap items j_1 and n, and items j_2 and (n+1). Move remaining items of A_d to other bundles and remove A_d.

Again, value of none of the remaining bundles has decreased.
$\Rightarrow$ MMS value of agent i has only increased in the reduced instance.

Step 1: Valid Reductions

- If $v_{i^*1} \geq 2/3$ then assign item 1 to i^*
- If $v_{i^*n} + v_{i^*(n+1)} \geq 2/3$ then assign $\{n, n+1\}$ to i^*

Step 2: Generalized Bag Filling with $\epsilon = \frac{1}{3}$

- Initialize n bags $\{B_1, \ldots B_n\}$ with $B_k = \{k\}, \forall k$.
- Assign items starting from $(n+1)$th to the first available bag, and give it to the first agent who shouts (values it at least $2/3 = (1-\epsilon)$).

After Step 1,
For each agent i,
$v_{ij} < \frac{2}{3}, \forall j \leq n$
$v_{ij} < \frac{1}{3}, \forall j > n$

Claim. If agent i^* is the first to shout, then for any agent $i \neq i^*$ the bag is of value at most 1.

1 2 3 n-1 n

2/3-MMS Allocation [GMT19]

- (Re)normalization

Step 0: Normalized Valuations: $\sum_j v_{ij} = n \Rightarrow \mu_i \leq 1$

Step 1: Valid Reductions

- If $v_{i^*1} \geq 2/3$ then assign item 1 to i^*
- If $v_{i^*n} + v_{i^*(n+1)} \geq 2/3$ then assign $\{n, n+1\}$ to i^*
- After every valid reduction, normalize valuations

Step 2: Generalized Bag Filling with $\epsilon = \frac{1}{3}$

- Initialize n bags $\{B_1, \ldots B_n\}$ with $B_k = \{k\}, \forall k$

Chores

- N: set of n agents, 1 ,…, n,
- M: set of m indivisible chores

- Agent i has a disutility function $d_i : 2^m \to \mathbb{R}_+$ over subsets of items
  - Monotone: the more the **un-**happier
- Additive: $d_i(S) = \sum_{j \in S} d_{ij}$, for any subset $S \subseteq M$

- N: set of n agents, 1 ,…, n,
- M: set of m indivisible chores
- Agent i has a disutility function $d_i : 2^m \to \mathbb{R}_-$ over subsets of items
  - Additive: $d_i(S) = \sum_{j \in S} d_{ij}$, for any subset $S \subseteq M$

$$\text{Allocation } A = (A_1, \ldots, A_n)$$

EF1: No agent envies another after removing one of her chores.

$$\forall i, k \in N, \qquad d_i(A_i \backslash c) \leq d_i(A_k), \qquad \exists c \in A_i$$

EF1: Algorithms

Round Robin

1. Order agents arbitrarily.
2. Let them pick their best chore (least painful chore), one-at-a-time, in that order.

Observations:

- If agent k picks the last chore, then agent $(k+1)$ does not envy anyone. Why?

EF1: Algorithms

Envy-cycle-elimination

1. $A = (\emptyset, \ldots, \emptyset)$
2. While there are unassigned chores
   1. Construct envy-graph of A and remove any cycles.
   2. Give an unassigned chore to ….. ??

Observations:

- Cycle elimination does not increase any agent's disutility.
- Giving a chore to sink maintains EF1. Why?

MMS

- N: set of n agents, 1 ,…, n,
- M: set of m indivisible chores
- Agent i has a disutility function $d_i : 2^m \to \mathbb{R}_-$ over subsets of items
  - Additive: $d_i(S) = \sum_{j \in S} d_{ij}$, for any subset $S \subseteq M$

- $\Pi :=$ Set of all partitions of items into n bundles

MMS value: $\mathrm{MMS_i} = \mu_i = \min_{A \in \Pi} \max_{A_k \in A} d_i(A_k)$

α-MMS allocation for $\alpha \geq 1$: $\forall i,\ d_i(A_i) \leq \alpha \mu_i$

1-MMS allocation may not exist!

EF1 to α-MMS

Claim. If $(A_1, \dots, An)$ is EF1 then it is 2-MMS

Observations: $\mu_i \geq \frac{d_i(M)}{n}$ and $\mu_i \geq \max_{j \in M} d_{ij}$

Proof.

Summary

Covered

- Additive Valuations:
  - ½-MMS allocation (poly-time algorithm)
  - $2/3$-MMS allocation (polynomial-time algorithm)

State-of-the-art

- $\left(\frac{3}{4}+\right)$-MMS allocation [GT20]
- More general valuations
  - MMS [GHSSY18]
- Groupwise-MMS [BBKN18]
- Chores: $11/9$-MMS [HL19]

Major Open Questions (additive)

- c-MMS + PO: polynomial-time algorithm for a constant $c > 0$
- Existence of $4/5$-MMS allocation? For 5 agents?

References (Indivisible Case).

[AMNS17] Georgios Amanatidis, Evangelos Markakis, Afshin Nikzad, and Amin Saberi. "Approximation algorithms for computing maximin share allocations". In: *ACM Trans. Algorithms 13.4 (2017)*

[BBKN18] Siddharth Barman, Arpita Biswas, Sanath Kumar Krishnamurthy, and Y. Narahari. "Groupwise maximin fair allocation of indivisible goods". In: *AAAI 2018*

[BK17] Siddharth Barman and Sanath Kumar Krishna Murthy. "Approximation algorithms for maximin fair division". In *EC 2017*

[BK19] Siddharth Barman and Sanath Kumar Krishnamurthy. "On the Proximity of Markets with Integral Equilibria" In *AAAI 2019*

[BKV18] Siddharth Barman, Sanath Kumar Krishnamurthy, and Rohit Vaish. Finding fair and efficient allocations. In: *EC 2018*

[B11] Eric Budish. "The combinatorial assignment problem: Approximate competitive equilibrium from equal incomes". In: *J. Political Economy* 119.6 (2011)

[CKMPSW14] Ioannis Caragiannis, David Kurokawa, Herve Moulin, Ariel Procaccia, Nisarg Shah, and Junxing Wang. "The Unreasonable Fairness of Maximum Nash Welfare". In: *EC* 2016

[GMT19] Jugal Garg, Peter McGlaughlin, and Setareh Taki. "Approximating Maximin Share Allocations". In: *SOSA@SODA 2019*

[GT20] Jugal Garg and Setareh Taki. "An Improved Approximation Algorithm for Maximin Shares". In: *EC 2020*

[GHSSY18] Mohammad Ghodsi, MohammadTaghi HajiAghayi, Masoud Seddighin, Saeed Seddighin, and Hadi Yami. "Fair allocation of indivisible goods: Improvement and generalization". In *EC 2018*

[HL19] Xin Huang and Pinyan Lu. "An algorithmic framework for approximating maximin share allocation of chores". In: *arxiv:1907.04505*

[KBKZ09] Bart de Keijzer, Sylvain Bouveret, Tomas Klos, and Yingqian Zhang. "On the Complexity of Efficiency and Envy-Freeness in Fair Division of Indivisible Goods with Additive Preferences". In: *Algorithmic Decision Theory (ADT)*. 2009

[KPW18] David Kurokawa, Ariel D. Procaccia, and Junxing Wang. "Fair Enough: Guaranteeing Approximate Maximin Shares". In: *J. ACM 65.2 (2018), 8:1–8:27*

[PW14] Ariel D Procaccia and Junxing Wang. "Fair enough: Guaranteeing approximate maximin shares". In *EC 2014*

[W97] Gerhard J Woeginger. "A polynomial-time approximation scheme for maximizing the minimum machine completion time". In: *Operations Research Letters 20.4 (1997)*