Lecture 6
Fair Division w/ Indivisible Items (Contd.)

CS 598RM

15^{th} September 2020

Instructor: Ruta Mehta

ILLINOIS
UNIVERSITY OF ILLINOIS AT URBANA-CHAMPAIGN

Fairness Notions for Indivisible Items

- n agents, m indivisible items (like cell phone, painting, etc.)
- Each agent i has a valuation function over subset of items denoted by $v_i : 2^m \to \mathbb{R}$
- Goal: fair and efficient allocation

Fairness:
  Envy-free (EF)
  Proportionality (Prop)

Efficiency:
  Pareto optimal (PO)

Maximum Nash Welfare (MNW)

EF1	EFX
Prop1	MMS
Guarantees	

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
- μ_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
- $\mu_i := \max_{A \in \Pi} \min_{A_k \in A} v_i(A_k)$

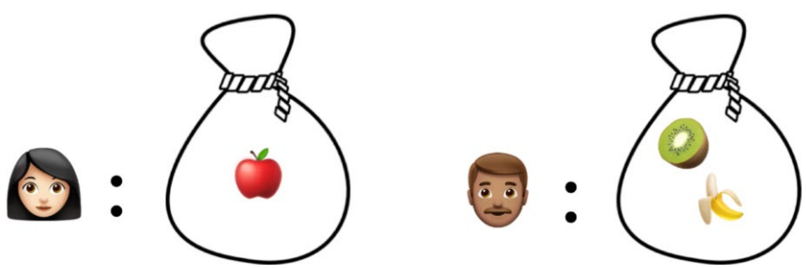
- MMS Allocation: A is called MMS if $v_i(A_i) \geq \mu_i, \forall i$
- Additive valuations: $v_i(A_i) = \sum_{j \in A_i} v_{ij}$

MMS value/partition/allocation

Agent\Items	🍎	🍌	🥝
👩	3	1	2
👨	4	4	5

👩	🍎	🍌 🥝
Value	3	3
MMS Value	3	

👨	🍎 🍌	🥝
Value	8	5
MMS Value	5	

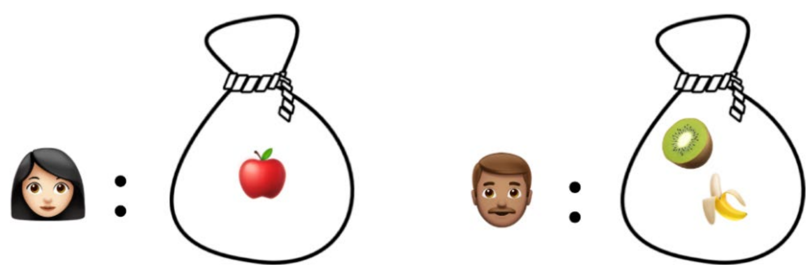


MMS value/partition/allocation

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 

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

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

Properties

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

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 $i \in N$
  - $v_i(M) = n \;\Rightarrow \mu_i \leq 1$
  - $v_{ij} \leq \epsilon, \forall i, j$

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

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

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

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

Warm Up: 1/2-MMS Allocation

- Assume that μ_i is known for all i
  - Scale valuations so that $\mu_i = 1 \Rightarrow v_i(M) \geq n$
- If all $v_{ij} \leq 1/2$ then ?

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: $v_{i1} \geq v_{i2} \geq \cdots v_{im}, \forall i \in N$
- Valid Reduction (α-MMS): If there exists $S \subseteq M$ and $i^* \in N$
  - $v_{i^*}(S) \geq \alpha.\mu_{i^*}^n(M)$
  - $\mu_i^{n-1}(M \setminus S) \geq \mu_i^n(M), \forall i \neq i^*$

  $\Rightarrow$ We can reduce the instance size!

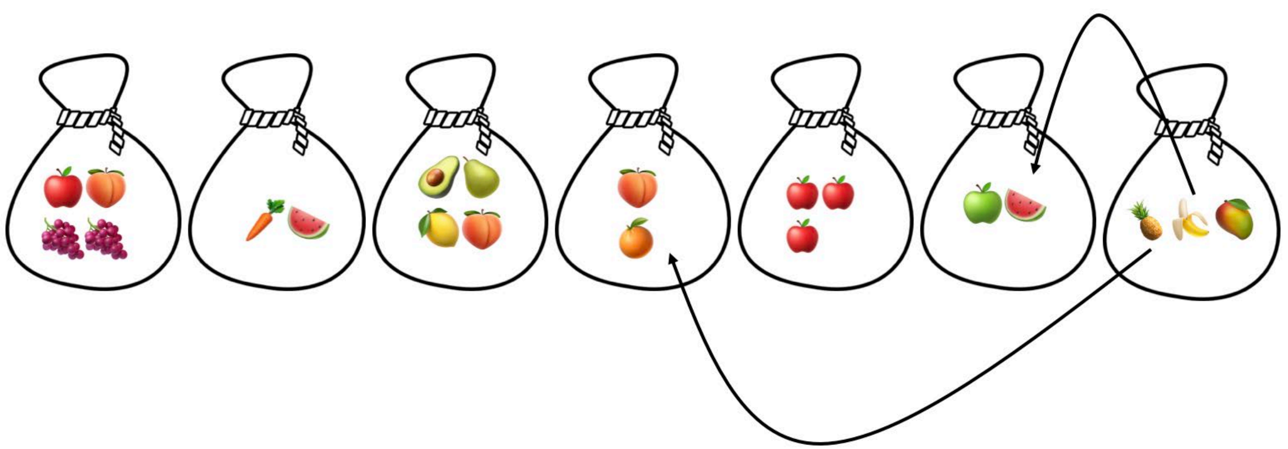
1/2-MMS Allocation

- Assume that μ_i is known for all i
  - Scale valuations so that $\mu_i = 1 \Rightarrow v_i(M) \geq n$

Step 1: Valid Reductions

- If $v_{i1} \geq 1/2$ then assign item 1 to i

Step 2: Bag Filling



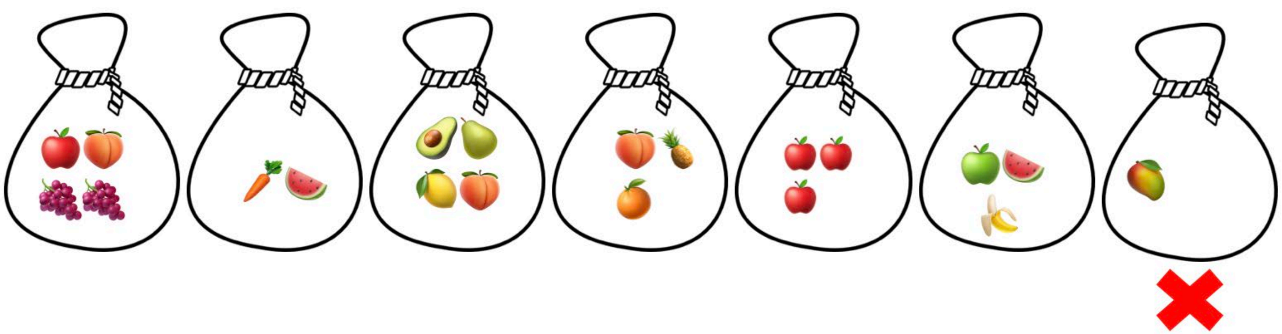
1/2-MMS Allocation

- Assume that μ_i is known for all i
  - Scale valuations so that $\mu_i = 1 \Rightarrow v_i(M) \geq n$

Step 1: Valid Reductions

- If $v_{i1} \geq 1/2$ then assign item 1 to i

Step 2: Bag Filling



1/2-MMS Allocation

- μ_i is not known

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

Step 1: Valid Reductions

- If $v_{i1} \geq 1/2$ then assign item 1 to i
- After every valid reduction, normalize valuations

Step 2: Bag Filling

2/3-MMS Allocation [GMT19]

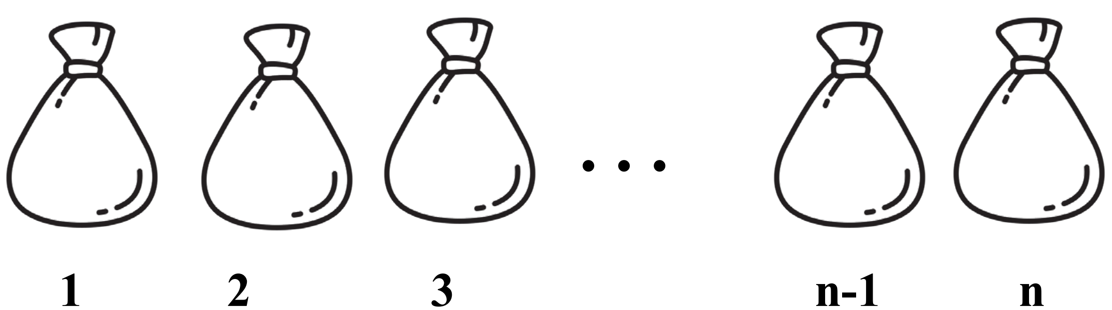
- Assume that μ_i is known for all i
  - Scale valuations so that $\mu_i = 1 \Rightarrow v_i(M) \geq n$
- If all $v_{ij} \leq 1/3$ then ?

Step 1: Valid Reductions

- If $v_{i1} \geq 2/3$ then assign item 1 to i
- If $v_{in} + v_{i(n+1)} \geq 2/3$ then assign $\{n, n+1\}$ to i

Step 2: Generalized Bag Filling

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



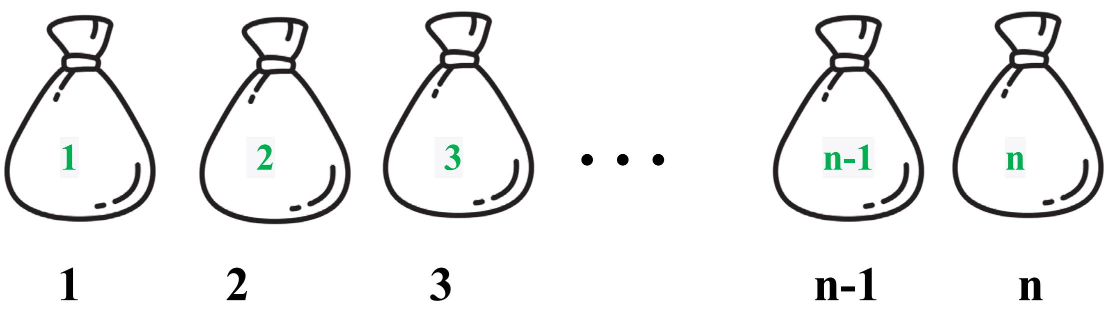
- Assume that μ_i is known for all i
  - Scale valuations so that $\mu_i = 1 \Rightarrow v_i(M) \geq n$

Step 1: Valid Reductions

- If $v_{i1} \geq 2/3$ then assign item 1 to i
- If $v_{in} + v_{i(n+1)} \geq 2/3$ then assign $\{n, n+1\}$ to i

Step 2: Generalized Bag Filling

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



2/3-MMS Allocation [GMT19]

- μ_i is not known

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

Step 1: Valid Reductions

- If $v_{i1} \geq 2/3$ then assign item 1 to i
- If $v_{in} + v_{i(n+1)} \geq 2/3$ then assign $\{n, n+1\}$ to i
- After every valid reduction, normalize valuations

Step 2: Generalized Bag Filling

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

Summary

Covered

- Additive Valuations:
  - Prop1 + PO (polynomial-time algorithm)
  - 2/3-MMS allocation (polynomial-time algorithm)

Not Covered

- $\left(\frac{3}{4}+\right)$-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?

New Fairness Notions

- n agents, m indivisible items (like cell phone, painting, etc.)
- Each agent i has a valuation function over subset of items denoted by $v_i : 2^m \to \mathbb{R}$
- Goal: fair and efficient allocation

Fairness:
Envy-free (EF)
Proportionality (Prop)

Efficiency:
Pareto optimal (PO)

Maximum Nash Welfare (MNW)

EF1	EFX
MMS	Prop1
Guarantees	

Allocation of Indivisible Items to Agents

- Set M of m indivisible items
- Set N of n agents
- Allocation $A = (A_1, \dots, A_n)$ is a partition of items to agents where each item is assigned to at most one agent

Objectives

- Maximize the sum of valuations (**Utilitarian** Welfare):

$$SW(A) = \sum_i v_i(A_i)$$

Objectives

- Maximize the sum of valuations (**Utilitarian** Welfare):

$$SW(A) = \sum_i v_i(A_i)$$

- Maximize the minimum of valuations (Max-Min-Fairness, **Egalitarian** Welfare):

$$SW(A) = \min_i v_i(A_i)$$

Objectives

- Maximize the sum of valuations (**Utilitarian** Welfare):

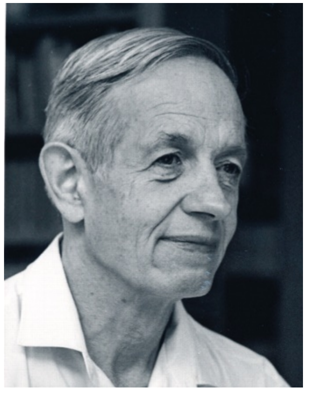
$$SW(A) = \sum_i v_i(A_i)$$

- Maximize the minimum of valuations (Max-Min-Fairness, **Egalitarian** Welfare):

$$SW(A) = \min_i v_i(A_i)$$

- Maximize the geometric mean of valuations (≈ **Efficiency** + **Fairness**, **Maximum Nash Welfare**):

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



Scale invariant

Maximum Nash Welfare (MNW)

- Maximum Nash welfare (MNW): An allocation A that maximizes the Nash welfare among all feasible allocations i.e.,

$$A^* = \arg\max_A (\prod_i v_i(A_i))^{1/n}$$

Additive Valuations ($v_i(A_i) = \sum_{j \in A_i} v_{ij}$):

- Divisible Items: MNW ≡ CEEI ⇒ Envy-free + Prop + PO + …
- Indivisible Items: MNW ⇒ EF1 + PO + $\Omega(\frac{1}{\sqrt{n}})$-MMS [CKMPSW16]
  - Existence of EF1 + PO allocation

MNW (additive)

- APX-hard [Lee17]; 1.069-hardness [G.HM18]

Approximation:

- ρ-approximate MNW allocation A satisfies: $\rho.\,\mathrm{NW}(A) \geq MNW$
  - 2 [CG15, CDGJMVY17], e [AOSS17]
  - 1.45 [BKV18] (pEF1 approach)

Close the gaps!

- Fairness Guarantees
  - Prop1 + PO + $\frac{1}{2n}$-MMS + 2-MNW [GM19]

MNW (additive)

Non-linear integer program:

$$\max \left(\prod_{i \in N} \sum_{j \in M} v_{ij} x_{ij} \right)^{1/n}$$

$$\sum_{i \in N} x_{ij} = 1, \qquad \forall j \in M$$

$$x_{ij} \in \{0, 1\}, \qquad \forall i \in N, j \in M$$

MNW (additive)

Non-linear integer program:

$$\max \frac{1}{n}\left(\sum_{i\in N} \log\left(\sum_{j\in M} v_{ij} x_{ij}\right)\right)$$

$$\sum_{i\in N} x_{ij} = 1, \qquad \forall j \in M$$

$$x_{ij} \in \{0,1\}, \qquad \forall i \in N, j \in M$$

Relaxation: Eisenberg-Gale Convex Program

$$\max \frac{1}{n}\left(\sum_{i\in N} \log(\sum_{j\in M} v_{ij} x_{ij})\right)$$

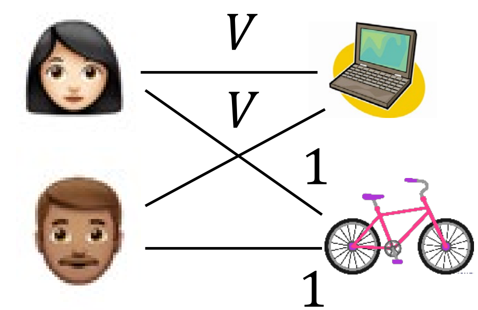
$$\sum_{i\in N} x_{ij} = 1, \qquad \forall j \in M$$

$$x_{ij} \geq 0, \qquad \forall i \in N, j \in M$$

Optimal Solutions ≡ Competitive equilibrium with equal incomes

Natural Approach

MNW problem → E-G convex program
≡ Competitive equilibrium
with linear utilities
↓
Round it to an integral allocation



$$\text{MNW(CEEI)} = \frac{V+1}{2}$$

However, no meaningful approximation guarantee for MNW by rounding [CG15]

$$\text{MNW} = \sqrt{V}$$

$$\text{Integrality Gap} = \Omega(\sqrt{V})$$

Natural Approach

MNW problem → E-G convex program

≡ Competitive equilibrium
with linear utilities

↓

Round it to an integral allocation

However, no meaningful approximation
guarantee for MNW by rounding [CG15]

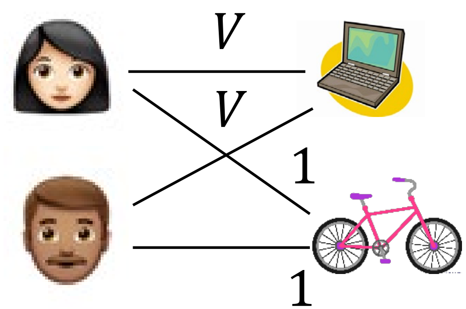
Approach

MNW problem $\rightarrow$ Competitive equilibrium with linear utilities **and spending restriction of \$1 on each item**

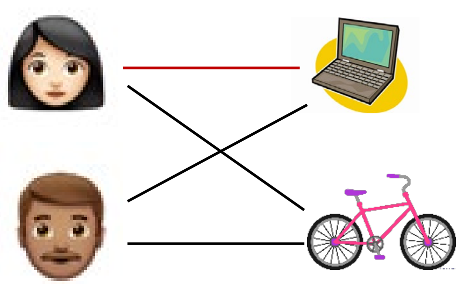
$\downarrow$

Round it to an integral allocation

Competitive Equilibrium vs SR-Equilibrium



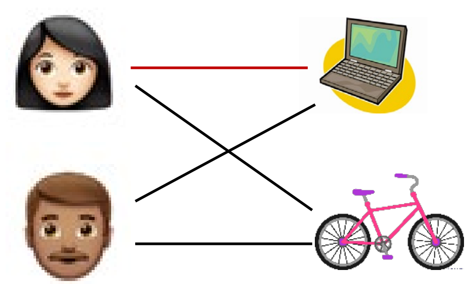
budget	p_{CE}
1	$\frac{2V}{V+1}$
1	$\frac{2}{V+1}$



p_{SR}

V

1



Approach: Spending-Restricted Equilibrium

MNW problem → Competitive equilibrium with linear utilities **and spending restriction of $1 on each item**

↓

Round it to an integral allocation

Extensions:

- Budget-additive (BA) [GHM18]
- Separable Concave (SC) [AMOV18]
- BA + SC [CCGGHM18]

MNW: Generalizations

- Non-symmetric Agents (different entitlements/weights)
  - Weighted envy-free, weighted proportionality
  - MNW (weighted geometric mean)

- Beyond Additive Valuations

Additive ⊂ (SC ⊂ OXS ⊂ GS / Budget additive) ⊂ Submodular ⊂ XOS ⊂ Subadditive

MNW: Generalizations

- Non-symmetric Agents (different entitlements/weights)
  - Weighted envy-free, weighted proportionality
  - MNW (weighted geometric mean)

- Beyond Additive Valuations

Additive $\subset$ SC $\subset$ OXS $\subset$ GS / Budget additive $\subset$ Submodular $\subset$ XOS $\subset$ Subadditive

The non-symmetric MNW Problem

- Non-symmetric MNW was proposed in [HS72, K77] and has been extensively studied and used in many applications
  - Agent i has a weight of w_i
- Allocation $A = (A_1, \dots, A_n)$ is partition of items to agents

$$\mathrm{NW}(A) = \left(\prod_i v_i(A_i)^{w_i}\right)^{1/\sum_i w_i}$$

weighted geometric mean of agents' valuations

- A^*: allocation maximizing the NW
- ρ-approximate MNW allocation A satisfies:

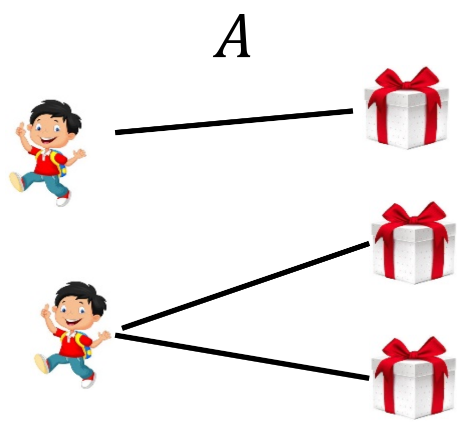
$$\rho.\mathrm{NW}(A) \geq \mathrm{NW}(A^*) = \mathrm{MNW}$$

Example (additive)

w_i

1 [10, 10, 1]

1 [1, 2, 1]



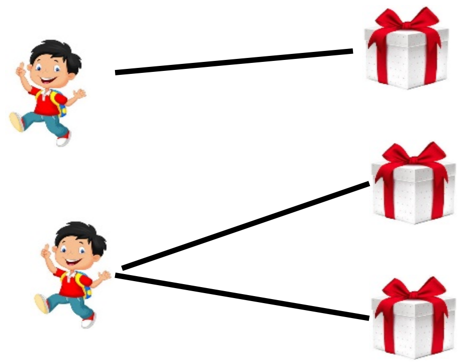
$$\text{MNW=NW}(A) = (10^1 \cdot 3^1)^{1/2}$$

Example (additive)

w_i $\quad$ A

2 $\quad$ [10, 10, 1]

1 $\quad$ [1, 2, 1]



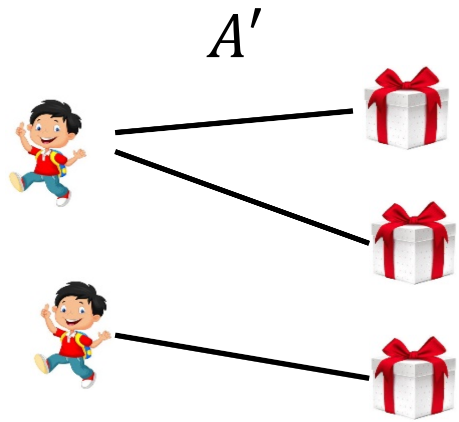
$$\text{NW}(A) = (10^2 \cdot 3^1)^{1/3}$$

Example (additive)

w_i

2 [10, 10, 1]

1 [1, 2, 1]

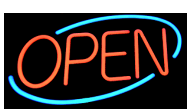


$$\mathrm{NW}(A) = (10^2 \cdot 3^1)^{1/3} < (20^2 \cdot 1^1)^{1/3} = \mathrm{NW}(A') = \mathrm{MNW}$$

MNW Approximations: Additive

	Lower bound	**Upper Bound**
Symmetric	1.069	1.45
Non-symmetric	1.069	$O(n)$

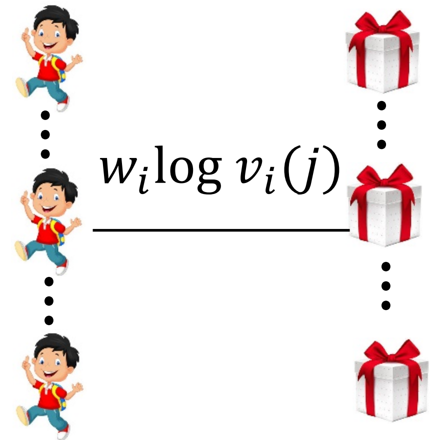
n: # of agents



Constant factor? sublinear?

$m = n$: Matching

$$\mathrm{NW}(A) = \left(\prod_i v_i(A_i)^{w_i}\right)^{1/\sum_i w_i}$$

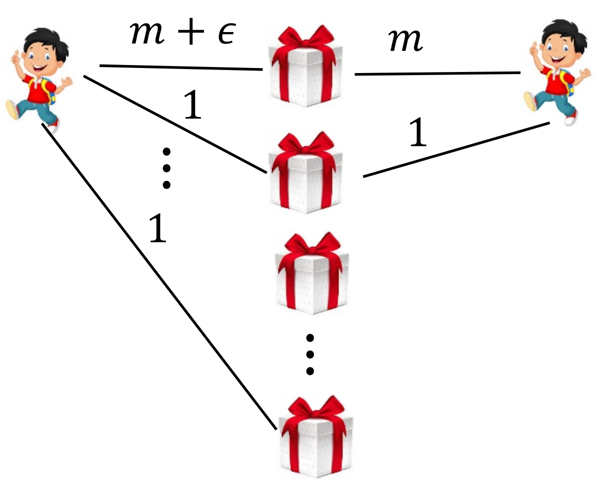


$$\mathrm{MNW} = \max_A \ \mathrm{NW}(A) \ \equiv \ \max_A \sum_i w_i \log v_i(A_i)$$

Claim: If $m = n$, then max-weight matching outputs MNW

$m > n$

- How good is max-weight matching?



$\text{NW}(A^*) \simeq m$

$\text{NW}(A) \simeq \sqrt{2m}$

- Issue: Allocation of high-value items!

Round Robin Procedure

- Guarantee (per agent) ?
- $u_i = v_i(M \setminus H_i)$

- g_i^*: highest-valued item in MNW allocation A_i^*
- $v_i(A_i^*) \leq \; n v_i(g_i^*) + \; u_i \; = n(v_i(g_i^*) + \frac{u_i}{n})$

- If we obtain an allocation A such that $v_i(A_i) \geq v_i(g_i^*) + \frac{u_i}{n}$, then A is $O(n)$-approximation!

$O(n)$-MNW + EF1 [GKK20]

- $H_i = 2n$ highest-valued items for agent i
- $u_i = v_i(M \setminus H_i)$
- Allocate one item to each agent using max-weight matching with weights $w_i \log(v_i(g) + \frac{u_i}{n})$: y_i^* is allocated to i
- $A \leftarrow$ Allocate remaining items using round-robin procedure

- $H_i = 2n$ highest-valued items for agent i
- $u_i = v_i(M \setminus H_i)$
- Allocate one item to each agent using max-weight matching with weights $w_i \log(v_i(g) + \frac{u_i}{n})$: y_i^* is allocated to i
- $A \leftarrow$ Allocate remaining items using round-robin procedure

- g_i^*: highest-valued item in MNW allocation A_i^*
- $v_i(A_i^*) \leq 2n v_i(g_i^*) + u_i \leq 2n(v_i(g_i^*) + \frac{u_i}{n})$
- $v_i(A_i) \geq v_i(y_i^*) + \frac{u_i}{n}$

$$\Rightarrow NW(A) \geq \left(\Pi_i \left(v_i(y_i^*) + \frac{u_i}{n}\right)^{w_i}\right)^{\frac{1}{\sum_i w_i}} \geq \left(\Pi_i \left(v_i(g_i^*) + \frac{u_i}{n}\right)^{w_i}\right)^{\frac{1}{\sum_i w_i}}$$

Generalizations

- Non-symmetric Agents (different entitlements/weights)
  - Weighted envy-free, weighted proportionality
  - MNW (weighted geometric mean)

- Beyond Additive

Additive ⊂ SC ⊂ OXS ⊂ GS / Budget additive ⊂ Submodular ⊂ XOS ⊂ Subadditive

non-negative monotone: $v(S) \leq v(T), \quad S \subseteq T$

Subadditive: $v(A \cup B) \leq v(A) + v(B), \quad \forall A, B$

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)*