On the polynomial computation of EFX allocations for 3 agents and 3-valued instances

Francesco Montano

Master of Science in
Engineering in Computer Science
Sapienza, University of Rome

A. Y. 2020 - 2021

Introduction

This thesis is about the Fair Division Problem. Such problem arises in several everyday task:

- Divide Goods
- Distribute Tasks
- Frequency Allocation
- ...

Table of contents

Outline

1. Definitions
2. Envy Free up to Any Item
3. Our Results

Preliminaries

- Set N of n agents
- Set M of m indivisible goods
- $v_i(\cdot)$ is the valuation function of player i
- $\mathcal{A} = (A_1, \ldots, A_n)$ is a partition of the goods, A_i is the set associated to player i.
- The task is producing an allocation that respects a fairness notion for each player.

Envy Free

- An allocation $\mathcal{A} = (A_1, A_2, \ldots, A_n)$ is envy-free (EF) if

$$\forall i, j \in N,\ v_i(A_i) \geq v_i(A_j)$$

- There is **not** always a possible EF allocation for indivisible items. Example: one item and two players that value such item more than zero.
- Because of the impossibility of computing EF allocation in some cases, have been introduced two relaxations of this criterion: *envy free up to one item* and *envy free up to any item*.

Envy Free Up to One Item

- An allocation $\mathcal{A} = (A_1, A_2, \ldots, A_n)$ is envy-free up to one good (EF1) if

$$\forall i, j \in N, A_j \neq \emptyset, \; \exists g \in A_j : v_i(A_i) \geq v_i(A_j \setminus \{g\})$$

- Exists an algorithm that is capable of computing EF1 allocation in polynomial time in the case of monotone valuation functions.

Envy Free Up to Any Item

- An allocation $\mathcal{A} = (A_1, A_2, \dots, A_n)$ is envy-free up to any good (EFX) if

$$\forall i, j \in N, A_j \neq \emptyset, \; \forall g \in A_j : v_i(A_i) \geq v_i(A_j \setminus \{g\})$$

- This is a much stronger version of the EF1 criteria: in the case of additive function
  - EF1: remove from A_j item $argmax_{x \in A_j} v_i(x)$
  - EFX: remove from A_j item $argmin_{x \in A_j} v_i(x)$

Outline

1. Definitions
2. Envy Free up to Any Item
3. Our Results

Envy Free Up to Any Item Recent Studies

In the last years this criteria has been extensively studied:

- in 2016 has been given a formal definition [Car+19],
- in 2018 has been shown that with the divide and choose algorithm we can obtain EFX for two players or n players with identical valuation functions[PR20].
- in 2019 has been shown that we can build allocations that are EFX and have at least half of the maximum possible Nash Welfare by assigning to the agents only a subset of the items and giving the remaining ones to charity[CGH19].
- in 2020 has been shown that for 3 players always exists an EFX allocation[CGM20].

Envy Free Up to Any Item Results

So summarizing till now we have the following results with respect to the number of agents:

- 2 players: the divide and choose algorithm produces an EFX allocation in polynomial time[PR20].
- 3 players: always exists an EFX allocation, but till now we only have a pseudo-polynomial algorithm[CGM20].
- ≥ 4 players: there exists an EFX allocation if we consider only a subset of the entire set of items[CGH19].

Outline

Match&Freeze Algorithm

The work proposed in this thesis is based on the Match&Freeze Algorithm[Ama+21]

- Produces EFX allocation for n players with additive valuation functions that value each item with one out of two possible values.
- Is based on two concepts: assign items at each iteration with the maximum matching algorithm and if one player envies another freeze the envied player.

Our Work

In this thesis we have done the two following things:

- We have build a modified version of the Match&Freeze algorithm that works with additive valuation functions with three values and two players.
- We have exploited the freezing technique to show how to obtain EFX allocation for three players and additive valuation functions with three values with some constraint over the values.

Counterexample of the Match&Freeze Algorithm for Three Values

Example showing that the Match&Freeze algorithm does not work when there are three values valuation functions

	i_1	i_2	i_3	i_4	i_5
p_1	100	50	50	50	50
p_2	50	1	1	1	1

Counterexample of the Match&Freeze Algorithm for Three Values

By following the Match&Freeze algorithm we obtain the allocation shown in bold in table 1. This is not an EFX allocation since

$$v_1(A_1) = 100 < v_1(A_2 \setminus \{i_2\}) = 150$$

	i_1	i_2	i_3	i_4	i_5
p_1	**100**	50	50	50	50
p_2	50	**1**	**1**	**1**	**1**

Table: Counter example for the Match&Freeze algorithm

Match&Freeze++ Algorithm for Three Values

The main idea is:

- We execute the original algorithm till the end
- If we do not obtain an EFX allocation we rollback to the iteration in which we freeze a player and change the assignment ignoring maximum matching.

	i_1	i_2	i_3	i_4	i_5
p_1	100	**50**	**50**	50	**50**
p_2	**50**	1	1	**1**	1

Table: Match&Freeze++ algorithm

Modification for Two Players

- Introduce the concept of **problematic assignment** as an assignment after which the original algorithm could not produce an EFX allocation.
- If we do not obtain an EFX allocation we rollback to the problematic assignment, invert the assignment and resume the algorithm.
- Redefine the number of iterations for which a player is frozen and the items that a non frozen player takes while the other is frozen.

Three Players and Three Values Problems

1. We introduce a constraint over the three values
2. In the case of three players we cannot use maximum matching to assign the items while a player is frozen.

Three Players and Three Values Constraint on the Values

Considering that the three values are $a > b > c$ we have the following constraint

$$c \geq a \mod b$$

Without such constraint we could have that the two non frozen player start to envy each other because of how we assign the items while a player is frozen.

Three Players Three Values Maximum Matching

The allocation shown in the following table is not EFX since p_1 envies p_3: $A_1 = \{i_1, i_{14}\}$, $A_3 = \{i_3, i_4, i_6, i_8, i_{10}, i_{12}\}$ and we have that $v_1(A_1) = 440$ $v_1(A_3 \setminus \{i_{12}\}) = 500$

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8	i_9	i_{10}	i_{11}	i_{12}	i_{13}	i_{14}
p_1	**400**	100	100	100	40	100	40	100	40	100	40	40	40	**40**
p_2	400	**100**	100	100	**100**	100	**100**	40	**40**	40	**40**	40	**40**	40
p_3	100	100	**100**	**100**	100	**100**	100	**100**	40	**100**	40	**40**	40	40

Three Players Three Values Maximum Matching

- If we freeze p_2 rather than p_1 we could obtain the a similar allocation that is not EFX since as we can notice if we invert p_1 and p_2 we have the same number of items for each remaining type.
- A type is a class of items represented by a triple with the value for each player for such class of items in order.
- So I had to define the order of the items to assign to the non frozen players while a player was frozen.

Approach used for the Three Player Case

I have divided the problematic assignments in two types:

- two players envy the frozen player: in such a case I have solved the different problematic assignments one by one, by defining which items give to which players.
- only one player envies the frozen player: in such a case I have defined a unique algorithm.
  - The player that does not envy the others chooses the type of item.
  - The other non frozen player takes an item of the same type.
  - The frozen player will envy or both or none of the two non frozen players.

Bibliography (1)

[Ama+21] Georgios Amanatidis et al. "Maximum Nash welfare and other stories about EFX". In: *Theor. Comput. Sci.* 863 (2021), pp. 69–85. DOI: 10.1016/j.tcs.2021.02.020. URL: https://doi.org/10.1016/j.tcs.2021.02.020.

[Car+19] Ioannis Caragiannis et al. "The Unreasonable Fairness of Maximum Nash Welfare". In: *ACM Trans. Economics and Comput.* 7.3 (2019), 12:1–12:32. DOI: 10.1145/3355902. URL: https://doi.org/10.1145/3355902.

Bibliography (2)

[CGH19] Ioannis Caragiannis, Nick Gravin, and Xin Huang. "Envy-Freeness Up to Any Item with High Nash Welfare: The Virtue of Donating Items". In: *Proceedings of the 2019 ACM Conference on Economics and Computation, EC 2019, Phoenix, AZ, USA, June 24-28, 2019*. Ed. by Anna Karlin, Nicole Immorlica, and Ramesh Johari. ACM, 2019, pp. 527–545. DOI: 10.1145/3328526.3329574. URL: https://doi.org/10.1145/3328526.3329574.

Bibliography (3)

[CGM20] Bhaskar Ray Chaudhury, Jugal Garg, and Kurt Mehlhorn. "EFX Exists for Three Agents". In: *EC '20: The 21st ACM Conference on Economics and Computation, Virtual Event, Hungary, July 13-17, 2020*. Ed. by Péter Biró et al. ACM, 2020, pp. 1–19. DOI: 10.1145/3391403.3399511. URL: https://doi.org/10.1145/3391403.3399511.

[PR20] Benjamin Plaut and Tim Roughgarden. "Almost Envy-Freeness with General Valuations". In: *SIAM J. Discret. Math.* 34.2 (2020), pp. 1039–1068. DOI: 10.1137/19M124397X. URL: https://doi.org/10.1137/19M124397X.