Homework2

Problem 1. (5pt) Problem 5 in Ex. 1

Consider the following convex programming in f_{ij} variables and p_j variables.

$$\begin{aligned} \max \quad & \sum_{i\in A, j\in G} f_{ij}\log(v_{ij}) - \sum_{j\in G} p_j \log p_j \\ \text{s.t.} \quad & \sum_{i\in A} f_{ij} = p_j, \forall j \in G \\ & \sum_{j\in G} f_{ij} = B_i, \forall i \in A \\ & f_{ij} \geq 0, \forall (i,j) \in A \times G \end{aligned}$$

Show that the solutions of the above formulation gives prices and ($ spent) allocation at a CE.

proof:

$$\begin{aligned} & max \sum_{i\in A, j\in G} f_{ij}log(v_{ij}) - \sum_{j\in G} p_j log\ p_j \\ \Leftrightarrow & \ min \sum_{j\in G} p_j log\ p_j - \sum_{i\in A, j\in G} f_{ij}log(v_{ij}) \end{aligned}$$

Its Lagrange function is:

$$L(p_j, f_{ij}, \alpha_i, \beta_j, \gamma_{ij}) = \sum_{j\in G} p_j log\ p_j - \sum_{i\in A, j\in G} f_{ij}log(v_{ij}) + \alpha_i(\sum_{i\in A} f_{ij} - p_j) + \beta_j(\sum_{j\in G} f_{ij} - B_i) + \gamma_{ij}(-f_{ij})$$

According to KKT conditions:

$$\begin{cases} \dfrac{\partial L(p_j, f_{ij}, \alpha_i, \beta_j, \gamma_{ij})}{\partial p_j} = & 0 \quad \dots\dots\dots\dots (1) \\ \dfrac{\partial L(p_j, f_{ij}, \alpha_i, \beta_j, \gamma_{ij})}{\partial f_{ij}} = & 0 \quad \dots\dots\dots\dots (2) \\ \gamma_{ij} * f_{ij} = & 0 \quad \dots\dots\dots\dots (3) \\ \sum_{i\in A} f_{ij} - p_j = & 0 \quad \dots\dots\dots\dots (4) \\ \sum_{j\in G} f_{ij} - B_i = & 0 \quad \dots\dots\dots\dots (5) \\ f_{ij} \geq & 0 \quad \dots\dots\dots\dots (6) \\ \alpha_i, \beta_j \neq & 0 \quad \dots\dots\dots\dots (7) \\ \gamma_{ij} \geq & 0 \quad \dots\dots\dots\dots (8) \end{cases}$$

For CE, it is necessary to prove OPTIAML Bundle (OB) and market clearing.

Among them, condition (4) can prove to be market clear, and condition (5) proves that each agent has spent all its property.

From (1) and (2):

$$\begin{aligned} \frac{\partial L(\dots)}{\partial p_j} &= log\ p_j + p_j * \frac{1}{p_j} - \alpha_i \\ &= log\ p_j + 1 - \alpha_i \\ &= 0 \end{aligned}$$

$$\frac{\partial L(\ldots)}{\partial f_{ij}} = -log\, v_{ij} + \alpha_i + \beta_j - \gamma_{ij}$$
$$= 0$$

Add the above two formulas:

$$-log\, v_{ij} + \alpha_i + \beta_j - \gamma_{ij} + log\, p_j + 1 - \alpha_i = 0$$
$$\Rightarrow log(\frac{v_{ij}}{p_j}) - 1 - \beta_j + \gamma_{ij} = 0$$
$$\Rightarrow log(\frac{v_{ij}}{p_j}) = 1 + \beta_j - \gamma_{ij}$$

Thus,

$$\frac{v_{ij}}{p_j} = e^{1+\beta_j-\gamma_{ij}},\ \gamma_{ij} \geq 0$$

therefore, in the solution when $f_{ij} > 0$ (*it means individual i spents some money on good j to buy it*),

$$\because f_{ij} * \gamma_{ij} = 0$$
$$\therefore \begin{cases} \gamma_{ij} = & 0 \\ \alpha_i = & log\, p_j + 1 \\ \beta_j = & log\, \frac{v_{ij}}{p_j} - 1 \end{cases}$$

Thus,

$$\frac{v_{ij}}{p_j} = e^{1+\beta_j}$$
$$= max_{K \in M} \frac{v_{ik}}{p_k},\ for\ all\ good\ j$$

Thus, the solution of the formulation gives the price and allocation at CE.

Problem 2. (5pt) Problem 5 in Ex. 2

$$\begin{aligned} \max \quad & \sum_{i \in A, j \in G} f_{ij} \log(v_{ij}) - \sum_{j \in G} (q_j \log q_j - q_j) \\ \text{s.t.} \quad & \sum_{i \in A} f_{ij} = q_j, \forall j \in G \\ & \sum_{j \in G} f_{ij} = B_i, \forall i \in A \\ & q_j \leq c_j, \forall j \in G \\ & f_{ij} \geq 0, \forall (i,j) \in A \times G \end{aligned}$$

Show that an optimal solution $\left((f_{ij})_{(i,j) \in A \times G}, (q_j)_{j \in G}\right)$ of the above convex programming captures a CE in terms of money allocations of agent i on good j in f_{ij} and total spending on good j in q_j at the equilibrium. The actual price of good j will come from an expression in involving both q_j and the dual variable for constraint $q_j \leq c_j$.

proof:

$$max \sum_{i \in A, j \in G} f_{ij} log(v_{ij}) - \sum_{j \in G} (q_j log\, q_j - q_j)$$
$$\Leftrightarrow min \sum_{j \in G} (q_j log\, q_j - q_j) - \sum_{i \in A, j \in G} f_{ij} log(v_{ij})$$

Its Lagrange function is:

$$
\begin{aligned}
&L(q_j, f_{ij}, \alpha_i, \beta_j, \gamma_j, \sigma_{ij}) \\
&= \sum_{j\in G}(q_j log\, q_j - q_j) - \sum_{i\in A, j\in G} f_{ij} log(v_{ij}) + \alpha_i(\sum_{i\in A} f_{ij} - q_j) + \beta_j(\sum_{j\in G} f_{ij} - B_i) + \gamma_j(q_j - c_j) + \sigma_{ij}(-f_{ij})
\end{aligned}
$$

According to KKT conditions:

$$
\begin{cases}
\frac{\partial L(q_j, f_{ij}, \alpha_i, \beta_j, \gamma_j, \sigma_{ij})}{\partial q_j} = & 0 \quad \ldots\ldots\ldots\ldots (1) \\
\frac{\partial L(q_j, f_{ij}, \alpha_i, \beta_j, \gamma_j, \sigma_{ij})}{\partial f_{ij}} = & 0 \quad \ldots\ldots\ldots\ldots (2) \\
\sigma_{ij} * f_{ij} = & 0 \quad \ldots\ldots\ldots\ldots (3) \\
\gamma_j(q_j - c_j) = & 0 \quad \ldots\ldots\ldots\ldots (4) \\
\sum_{i\in A} f_{ij} - q_j = & 0 \quad \ldots\ldots\ldots\ldots (5) \\
\sum_{j\in G} f_{ij} - B_i = & 0 \quad \ldots\ldots\ldots\ldots (6) \\
f_{ij} \geq & 0 \quad \ldots\ldots\ldots\ldots (7) \\
q_j - c_j \leq & 0 \quad \ldots\ldots\ldots\ldots (8) \\
\alpha_i, \beta_j \neq & 0 \quad \ldots\ldots\ldots\ldots (9) \\
\gamma_j, \sigma_{ij} \geq & 0 \quad \ldots\ldots\ldots\ldots (10)
\end{cases}
$$

Similarly, condition (5) can prove to be market clear, and condition (6) proves that each agent has spent all its property.

From (1) and (2):

$$
\begin{aligned}
\frac{\partial L(\ldots)}{\partial q_j} &= log\, q_j + q_j * \frac{1}{q_j} - 1 - \alpha_i + \gamma_j \\
&= log\, q_j + 1 - \alpha_i + \gamma_j - 1 \\
&= 0
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial L(\ldots)}{\partial f_{ij}} &= -log\, v_{ij} + \alpha_i + \beta_j - \sigma_{ij} \\
&= 0
\end{aligned}
$$

Add the above two formulas:

$$
\begin{gathered}
log\, q_j - \alpha_i + \gamma_j - log\, v_{ij} + \alpha_i + \beta_j - \sigma_{ij} = 0 \\
\Rightarrow log(\frac{v_{ij}}{p_j}) - \beta_j - \gamma_j + \sigma_{ij} = 0 \\
\Rightarrow log(\frac{v_{ij}}{p_j}) = \beta_j + \gamma_j - \sigma_{ij}
\end{gathered}
$$

Thus,

$$
\frac{v_{ij}}{p_j} = e^{\beta_j - \sigma_{ij} + \gamma_j},\ \sigma_{ij} \geq 0
$$

therefore, in the solution when $f_{ij} > 0$, we can get

$$
\begin{gathered}
\because f_{ij} * \sigma_{ij} = 0 \\
\therefore \sigma_{ij} = 0
\end{gathered}
$$

Thus,

$$\frac{v_{ij}}{p_j} = e^{\beta_j + \gamma_j}$$
$$= max_{K \in M} \frac{v_{ik}}{p_k}, \ for\ all\ good\ j$$

Thus, the optimal solution of the formulation captures the price and allocation at CE.

Problem 3. (5pt) Problem 5 in Ex. 3
An allocation $A = (A_1, \ldots, A_n)$ is called α-EFX if

$$v_i(A_i) \geq \alpha \cdot v_i(A_j \setminus g), \forall g \in A_j, \forall i, j.$$

Design a polynomial-time algorithm to obtain $\frac{1}{2}$-EFX allocation when agents have monotone subadditive valuations.

This is a polynomial-time algorithm given in [HJ19]. And the allocation this algorithm outputs is $\frac{1}{2}$-EFX when all agents have subadditive valuations.

Algorithm 2 Algorithm for (sub-)additive valuations

1: Initiate $L = N$ and $R = M$.
2: Initiate $A_i = \emptyset$ for all $i \in N$.
3: **while** $R \neq \emptyset$ **do**
4:     Compute a maximum weight matching $\mathcal{M}$ between L and R, where the weight of edge between $i \in L$ and $j \in R$ is given by $v_i(A_i \cup \{j\}) - v_i(A_i)$. If all edges have weight 0, then we compute a maximum cardinality matching $\mathcal{M}$ instead.
5:     For every edge $(i, j) \in \mathcal{M}$, allocate j to i: $A_i = A_i \cup \{j\}$ and exclude j from R: $R = R \setminus \{j\}$.
6:     As long as there is an envy-cycle w.r.t. $A = (A_i)_{i \in N}$, invoke procedure $\mathcal{P}$ (to be described later).
7:     Update $A = (A_i)_{i \in N}$ to be the allocations after $\mathcal{P}$.
8:     Update the set of agents not envied by any other agents: $L = \{i \in N : \forall j \in N, v_j(A_j) \geq v_j(A_i)\}$.
9: **end while**

Here lists same conditions we have talked in PPT before:

- L means N agents.
- R means M unallocated goods.
- A_i means an allocation to agent i.
- There is an $envy - graph\ G$, any agent is represented by a node. Also, there is a direct edge from node i to j iff i is envies j. A directed cycle in G is called an $envy - cycle$.
- procedure P: Let $C = i_1 \to i_2 \to \cdots \to i_t \to i_1$ be such a cycle. If we reallocate $A_{i_{k+1}}$ to agent i_k for all $k \in [t-1]$, and reallocate A_{i_1} to agent i_t, the number of edges of G will be strictly decreased. Thus, by repeatedly using this procedure, we eventually get another allocation whose envy-graph is acyclic. It is shown in [Lipton et al., 2004] that P can be done in time $O(nm^3)$.

Algorithm 2 computes an allocation in polynomial time that is $\frac{1}{2}$-EFX when all agents have subadditive valuations.

Proof:

In each round, there must be at least one agent who is not envied by anybody, since after Step 6, the envy-graph is acyclic. Thus there will be at least one agent whose in-degree is $0, i.e., L \neq \emptyset$. Moreover, since at least one good will be allocated in each round (each while loop), Algorithm 2 terminates in at most m rounds. Given that each round (including procedure P, the computation of maximum weight matching, and the updates of edge weights and unenvied agents L) can be done in polynomial time, Algorithm 2 runs in polynomial time.

Fix any agent i. We classify other agents into three sets:

- agents not envied by i: $N1 = \{j \in N/\{i\} : v_i(A_i) \geq v_i(A_j)\}$;
- agents envied by i that receive only one good:
  $N2 = \{j \in N/\{i\} : v_i(A_i) < v_i(A_j) \; and \; |A_j| = 1\}$;
- agents envied by i that receive more than one good:
  $N3 = \{j \in N/\{i\} : v_i(A_i) < v_i(A_j) and |A_j| \geq 2\}$.

By definition we have $N1 \cup N2 \cup N3 = N \; \{i\}$.

We first show that the final allocation A is $\frac{1}{2}$-EFX when the valuations are subadditive. Note that it suffices to consider $N3$, as the other agents are either not envied by i, or allocated only one good.

Fix any agent $j \in N3$. Let e_j be the last good allocated to j. As we only allocate goods to unenvied agents, we have $v_i(A_i) \geq v_i(A_j/\{e_j\})$. On the other hand, we also have $v_i(A_i) \geq v_i(e_j)$, since otherwise in the first round, matching e_j (which is not allocated yet) with i (which is not envied) gives a matching with strictly larger weight. Combing the two inequalities and by subadditivity of valuation v_i , we have

$$2 \cdot v_i(A_i) \geq v_i(A_j/\{e_j\}) + v_i(e_j)$$

which implies that agent i is satisfied $\frac{1}{2}$-EFX .

Problem 4. (5pt) Problem 5 in Ex. 4

Assume that agents have additive valuations. Show that an allocation that maximizes the Nash welfare (MNW) (1) is EF1+PO. (2) may not be EFX. (3) is EFX when agents have identical valuations.

(1)

It is the same problem that we have discussed in the PPT.

What we need to do is achieving EF1 while maintaining PO. Also, it should satisfy Competitive Equilibrium(CE).

Here is the allocation:

- Start with $A = (A_1, A_2, \ldots, A_n)$
- each item j is assigned to an agent with the highest valuation
- set price of item j as $p_j = max_i \; v_{ij}$

This allocation satisify:

1. Optimal bundle:

$$f_{ij} > 0 \Rightarrow \frac{v_{ij}}{p_j} = max_{k \in G} \frac{v_{ik}}{p_k}$$

2. Market clearing:

$$\sum_j f_{ij} = p(A_i), \; \sum_i f_{ij} = p_j$$

(2)

Here is the EFX example:

agent A's budget: **[15,10,20]**, agent B's budget: **[14,30,18]**. The number in each person's budgut means the individual value of each good. Thus, if the allocation maximizes the Nash welfare, then with the solution in (1), the allocation should be: items 1 and 3 to the agent A and item 2 to the agent B. And of course, B envies A.

However, for agent B:

$$\begin{gathered} v_B(A_B) \leq v_B(A_A), \\ v_B(A_B) \geq v_B(A_A \setminus \{item\ 1\}), \\ v_B(A_B) \geq v_B(A_A \setminus \{item\ 3\}), \\ \Rightarrow it\ is\ EFX. \end{gathered}$$

Here is the counter-example:

agent A's budget: **[15,10,20]**, agent B's budget: **[8,13,18]**. The number in each person's budgut means the individual value of each good. Thus, if the allocation maximizes the Nash welfare, then with the solution in (1), the allocation should be: items 1 and 3 to the agent A and item 2 to the agent B. And of course, B envies A.

However, for agent B:

$$\begin{gathered} v_B(A_B) \leq v_B(A_A), \\ v_B(A_B) \leq v_B(A_A \setminus \{item\ 1\}), \\ v_B(A_B) \geq v_B(A_A \setminus \{item\ 3\}), \\ \Rightarrow it\ is\ not\ EFX. \end{gathered}$$

To a conclude, it may not be EFX.

(3)

Here is the algorithm:

- *MNW algorithm*: with additive valuations, (1) it computes a *largest* subset of agents with positive valuations for goods, and (2) it then computes an MNW allocation over this subset [CI16].

And with additive identical valuations, an MNW allocation is EFX.

Proof:

Let $A = (A_1, \ldots, A_n)$ be an MNW allocation. Hence, it is EF1 and PE. MUW follows with identical valuations. We next show that A is even EFX. We consider two cases. In the first case, there are $m \leq n$ goods. The allocation A is EFX because each agent receives in it at most one good. In the second case, there are m>n goods. The Nash welfare of A is positive in this case. For the sake of contradiction, assume that A is not EFX. WLOG, we can suppose that agent 1 is not EFX of agent 2. Therefore, $u_1(A_1) < u_1(A_2/\{o\})$ for some $o \in A_2$. We next consider the allocation $B = (B_1, \ldots, B_n)$ with $B_1 = A1 \cup \{o\}$, $B_2 = A_2/\{o\}$ and $B_i = A_i$ for each $i \in 3, \ldots, n$. We next show that the Nash welfare of B is strictly greater than the Nash welfare of A and thus reach a contradiction with the MNW of A. To do this, consider the ratio between the welfare of B and the welfare of A. We derive that the value of this ratio is greater than one iff agent 1 is not EFX of agent 2. The result follows. We can compute an MNW allocation by using the *MNW algorithm*.

Reference

[PR18] Benjamin Plaut and Tim Roughgarden. Almost envy-freeness with general valuations. In: SODA 2018

[BD11] Plaut, B., Roughgarden, T.: Almost envy-freeness with general valuations. In: Proceedings of the Twenty-Ninth Annual ACM-SIAM Symposium on Discrete Algorithms, SODA 2018, New Orleans, LA, USA, January 7-10, 2018. pp. 2584–2603 (2018)

[HJ19] Hau Chan , Jing Chen , Bo Li and Xiaowei Wu: Maximin-Aware Allocations of Indivisible Goods. In: IJCAI 2019

[CI16] Caragiannis, I., Kurokawa, D., Moulin, H., Procaccia, A.D., Shah, N., Wang, J.: The unreasonable fairness of maximum nash welfare. In: Proceedings of the 2016 ACM Conference on Economics and Computation, EC '16, Maastricht, The Netherlands, July 24-28, 2016. pp. 305–322 (2016)