Algorithmic Game Theory Homework 2

Zhang Luan 112020

Problem1

My favourite result is **Round Robin Algorithm** .

Round Robin Algorithm :

- Fix an ordering of agents
- While there is an item unallocated
  - i : next agent in the round robin order
  - Allocate i her most valuable item among the unallocated ones

Round Robin Algorithm ensures that the final distribution is EF1.

Round Robin Algorithm enlightens us that we should **relax our mind** in our daily life, **not dwell on the fact that our favorite things in the past are occupied by others**, but **pay attention to the present and grasp our favorite things now** .

Problem2

It is known that the valuation function satisfies ：

$$v(S) + v(T) \geq v(S \cap T) + v(S \cup T)$$

Obviously, the valuation function is a submodular function(concave function). With the increase of indivisible goods, the value of the collection of goods grows more and more slowly.

Proof 1 :

a. $j \in S$

$$\begin{gathered} T + j = T \qquad S + j = S \\ \frac{v(T+j)}{v(T)} = 1 \qquad \frac{v(S+j)}{v(S)} = 1 \\ \frac{v(T+j)}{v(T)} \leq \frac{v(S+j)}{v(S)} \quad establish \end{gathered}$$

b. $j \notin S \ and \ j \in T$

$$\begin{gathered} T + j = T \\ \frac{v(T+j)}{v(T)} = 1 \qquad \frac{v(S+j)}{v(S)} > 1 \\ \frac{v(T+j)}{v(T)} \leq \frac{v(S+j)}{v(S)} \quad establish \end{gathered}$$

c. $j \notin T \ and \ j \in M$

$$v(A)+v(B) \geq v(A\cap B)+v(A\cup B)$$
$$A \leftarrow S+j \ \ and \ \ B \leftarrow T$$
$$v(S+j)+v(T) \geq v(S)+v(T+j)$$
$$v(S+j)-v(S) \geq v(T+j)-v(T)$$
$$\frac{v(S+j)-v(S)}{v(S)} \geq \frac{v(T+j)-v(T)}{v(S)} \geq \frac{v(T+j)-v(T)}{v(T)}$$
$$(since \ S\subseteq T \Rightarrow v(S) \leq v(T))$$
$$\frac{v(S+j)}{v(S)}-1 \geq \frac{v(T+j)}{v(T)}-1$$
$$\frac{v(T+j)}{v(T)} \leq \frac{v(S+j)}{v(S)} \quad establish$$

Proof 2 :

Let $|T|=k$, $T=\{e_1,e_2,\ldots,e_k\}$. We have :

$$v(T)=\sum_{i=1}^{k}(v(\{e_1,e_2,\ldots,e_i\})-v(\{e_1,e_2,\ldots,e_{i-1}\})) \quad (telescopic \ sum)$$

It is known that :

$$v(\{e_1\}) \geq v(T)-v(T-\{e_1\})$$
$$v(\{e_1,e_2\})-v(\{e_1\}) \geq v(T)-v(T-\{e_2\})$$
$$\ldots$$
$$v(\{e_1,e_2,\ldots,e_k\})-v(\{e_1,e_2,\ldots,e_{k-1}\}) \geq v(T)-v(T-\{e_k\})$$
$$(by \ submodularity)$$
$$\Rightarrow \sum_{i=1}^{k}(v(\{e_1,e_2,\ldots,e_i\})-v(\{e_1,e_2,\ldots,e_{i-1}\})) \geq \sum_{i=1}^{k}(v(T)-v(T-\{e_i\}))$$
$$\Rightarrow v(T) \geq \sum_{i=1}^{k}(v(T)-v(T-\{e_i\}))$$

Problem3

Definition 1 :

An allocation $A=(A_1,\ldots,A_n)$ is called $c-EFX$ if for any agents $i,j\in[n]$ and any good $g\in A_j$, we have :

$$v_i(A_i) \geq c\cdot v_i(A_j\setminus g)$$

Definition 2 :

The definition of subadditive valuations is as follows :

$$v(S)+v(T) \geq v(S\cup T)$$

Definition 3 :

The envy graph of an allocation A is defined as follows :

In the envy graph $1.$ each vertex represents an agent $2.$ The directed edge from vertex i to vertex j indicates that agent i envies agent j $(i.e.\ v_i(A_i) < v_i(A_j))$

Lemma 1 :

Let $A = (A_1, A_2 ... A_n)$ be a $c - EFX$ allocation with envy graph $G = (V, E)$, where G contains a cycle. Then there exists another allocation $B = (B_1, B_2 ... B_n)$ with envy graph H where B is also $c - EFX$, and H has no cycles.

Proof :

Let $c = (1, 2.. . |c|)$ be a cycle in G. Thus for all $i \in c$, $v_i(A_i) < v_i(A_{(i\ mod\ |c|)+1})$.

Define a new allocation $A^{'}$ where $A_i' = A_{(i\ mod\ |c|)+1}$ for all $i,$ and let $G' = (V', E')$ be the envy graph for A' . It is clear that A' is a permutation of A .

Suppose A' is not $c - EFX$: there exist $i, j \in [n]$ and $g \in A_j^{'}$ where $v_i(A_i^{'}) < c \cdot v_i(A_j' \setminus g)$. Since A' is a permutation of A , there exists $k \in N$ where $A_k = A_j',$ so $v_i(A_i') < c \cdot v_i(A_k \setminus g)$. Observe that $v_i(A_i') > v_i(A_i)$ if $i \in c,$ and $v_i(A_i') = v_i(A_i)$ if $i \notin c.$ Thus $v_i(A_i) \leq v_i(A_i') < c \cdot v_i(A_k \setminus g),$ so A is also not $c - EFX$. Produce contradictions! Therefore if A is $c - EFX$, then $A^{'}$ is also $c - EFX$.

Since the utility of every agent in c has strictly increased, $1.$ the number of edges from $V^{'} \setminus c$ into c is unchanged $2.$ the number of edges from c into $V^{'} \setminus c$ has decreased or stayed the same $3.$ for each $i \in c$, the number of agents in c whom i envies has decreased by at least one. So $G^{'}$ has strictly fewer edges than G.

Since the number of edges strictly decreases each time, we can execute the above process $|E|$ times at most to get a envy graph without a cycle.

$$
\begin{array}{ll}
\hline
\textit{Algorithm} \quad \textit{Find an } \frac{1}{2} - \textit{EFX allocation for agents with subadditive valuations} & \\
\hline
P \leftarrow [m] & \triangleright \textit{Initially, all goods are in the pool} \\
\textit{for each } i \in [n] \textit{ do} & \\
\qquad A_i \leftarrow \emptyset & \\
\textit{while } P \neq \emptyset \textit{ do} & \\
\qquad g^* \leftarrow \textit{pop}(P) & \triangleright \textit{Remove an arbitrary good from } P \\
\qquad j \leftarrow \textit{FindSourceAgent}(A_1, A_2 ... A_n) & \triangleright \textit{give it to an in} - \textit{degree } 0 \textit{ agent} \\
\qquad A_j \leftarrow A_j \cup \{g^*\} & \\
\qquad \textit{if } \exists i \in [n], g \in A_j \textit{ such that } v_i(A_i) < \frac{1}{2} v_i(A_j \setminus g) \textit{ then} & \\
\qquad\qquad P \leftarrow P \cup A_i & \\
\qquad\qquad A_j \leftarrow A_j \setminus \{g^*\} & \\
\qquad\qquad A_i \leftarrow \{g^*\} & \\
\qquad (A_1, A_2 ... A_n) \leftarrow \textit{EliminateEnvyCycles}(A_1, A_2 ... A_n) & \\
\textit{return } (A_1, A_2 ... A_n) & \\
\hline
\end{array}
$$

Proof :

Let A_k^l be the bundle of agent k at the beginning of round $l,$ and let B_k^l be the bundle of agent k before EliminateEnvyCycles on round l .

We use induction on round l. Initially, all agents have empty bundles, which satisfies $\frac{1}{2}-EFX$. Thus we assume the allocation at the beginning of round l is $\frac{1}{2}-EFX$. Because the allocation at the beginning of round $l+1$ is equal to EliminateEnvyCycles(B^l), and according to **Lemma 1**, if B^l satisfies $\frac{1}{2}-EFX$, then EliminateEnvyCycles(B^l) satisfies $\frac{1}{2}-EFX$. So if we want to prove that the allocation at the beginning of round $l+1$ is $\frac{1}{2}-EFX$, we must prove that B^l is $\frac{1}{2}-EFX$.

$1.$ If the body of the if-statement (lines 9-11) is not executed. The allocation B^l is $\frac{1}{2}-EFX$ by definition.

$2.$ If the body of the if-statement (lines 9-11) is executed. Because g^* was added and then removed, $B_j^l = A_j^l$. For all $k \neq i$, $B_k^l = A_k^l$. Because we assume the allocation at the beginning of round l is $\frac{1}{2}-EFX$, all pairs $(k,k^{'})$ where $k \neq i$ and $k^{'} \neq i$ remain $\frac{1}{2}-EFX$ in B^l. Furthermore, since $B_i^l = \{g^*\}$ and $B_i^l \setminus g = \emptyset \;\; for \; all \; g \in B_i^l$, agent k where $k \neq i$ does not envy agent i under the condition of $\frac{1}{2}-EFX$. Now, we just need to prove that agent i does not envy agent k where $k \neq i$ under the condition of $\frac{1}{2}-EFX$ $(i.e. \;\; v_i(B_i^l) \geq \frac{1}{2} \cdot v_i(B_k^l \setminus g))$.

It is known that $v_i(A_i^l) \geq v_i(A_j^l)$ at the beginning of round l. Because the body of the if-statement (lines 9-11) is executed, there is $g \in A_j^l \cup \{g^*\}$ that satisfies $v_i(A_i^l) < \frac{1}{2} v_i(A_j^l \cup \{g^*\} \setminus g)$. Therefore :

$$v_i(A_i^l) < \frac{1}{2} v_i(A_j^l \cup \{g^*\})$$

$$Because \; v_i \; is \; subadditive$$

$$v_i(A_i^l) < \frac{1}{2}(v_i(A_j^l) + v_i(\{g^*\}))$$

$$Because \; v_i(A_i^l) \geq v_i(A_j^l)$$

$$v_i(A_i^l) < \frac{1}{2}(v_i(A_i^l) + v_i(\{g^*\}))$$

$$v_i(A_i^l) - \frac{1}{2} v_i(A_i^l) < \frac{1}{2} v_i(\{g^*\})$$

$$v_i(A_i^l) < v_i(\{g^*\})$$

$$v_i(A_i^l) < v_i(B_i^l)$$

Since A^l is $\frac{1}{2}-EFX$, $v_i(A_i^l) \geq \frac{1}{2} v_i(A_k^l \setminus g)$ for all $g \in A_k^l$. Since $v_i(A_i^l) < v_i(B_i^l)$ and $A_k^l = B_k^l$ for all $k \neq i$, $v_i(B_i^l) \geq \frac{1}{2} \cdot v_i(B_k^l \setminus g)$ for all $g \in B_k^l$. Therefore, when if-statement is executed, the allocation B^l is also $\frac{1}{2}-EFX$.

To sum up, the allocation returned by the algorithm is $\frac{1}{2}-EFX$. Next, let's discuss the time cost of the algorithm.

For round l, we define a potential function :

$$\phi(l) = \sum_{k=1}^{n} v(A_k^l)$$

If only $i's$ bundle changes in round l (Case 2), we have $v_i(A_i^l) < v_i(B_i^l)$, $\phi(l+1) - \phi(l) > 0$. If only $j's$ bundle changes in round l (Case 1), we have $v_j(A_j^l) < v_j(B_j^l)$, $\phi(l+1) - \phi(l) > 0$. Therefore, in any round which falls under Case 1, $|P|$ decreases by one. If m rounds pass without Case 2 occurring, P becomes empty, and the algorithm terminates.

Because each good can be given to one of the n agents, or left in the unallocated pool, the number of possible allocations is at most $(n+1)^m$. It can be seen from the above content that in any case, ϕ will increase. So ϕ can increase at most $(n+1)^m$ times. Thus the algorithm must have terminated after $m(n+1)^m$ rounds .

To sum up, the algorithm can be computed in pseudo-polynomial time .

Problem4

Problem 4 needs to prove :

$$v_i(A_i) \geq \frac{v_i(M \setminus Y)}{4n} \geq \frac{v_i(M \setminus H_i) - nv_i(y_i^*)}{4n}$$

Proof :

$O(n)\ Algorithm$ naturally guarantees $|A_i| \geq 2$ for any agents $i \in [n]$.

Since A is a $\frac{1}{2} - EFX$ allocation, we have $v_i(A_i) \geq \frac{1}{2}v_i(A_j \setminus g)$ for any good $g \in A_j$. Since $|A_j| \geq 2$, we have $v_i(A_i) \geq \frac{1}{2}v_i(g)$. Combining the above two inequalities, we have :

$$\begin{gathered}
2v_i(A_i) \geq \frac{1}{2}(v_i(A_j \setminus g) + v_i(g)) \\
v_i(A_i) \geq \frac{1}{4}(v_i(A_j \setminus g) + v_i(g)) \\
(by\ subadditivity) \\
v_i(A_i) \geq \frac{1}{4}v_i(A_j) \\
\sum v_i(A_i) \geq \frac{1}{4}\sum_{j \in [n]} v_i(A_j) \\
nv_i(A_i) \geq \frac{1}{4}v_i(M) \geq \frac{1}{4}v_i(M \setminus Y) \\
v_i(A_i) \geq \frac{v_i(M \setminus Y)}{4n}
\end{gathered}$$

In the above, we have proved $v_i(A_i) \geq \frac{v_i(M \setminus Y)}{4n}$, and then we will prove $\frac{v_i(M \setminus Y)}{4n} \geq \frac{v_i(M \setminus H_i) - nv_i(y_i^*)}{4n}$.

$$\begin{gathered}
v_i(M \setminus Y) \\
= v_i((M \setminus (Y \cap H_i)) \setminus (Y \setminus H_i)) \\
\geq v_i(M \setminus (Y \cap H_i)) - v_i(Y \setminus H_i) \\
(by\ subadditivity) \\
\geq v_i(M \setminus H_i) - v_i(Y \setminus H_i) \\
(since\ Y \cap H_i \subseteq H_i)
\end{gathered}$$

It is known that $v_i(y_i^*) \geq v_i(g)$. We can understand y_i^* as the optimal solution and g as the feasible solution . Thus :

$$v_i(Y \setminus H_i) \leq \sum_{g \in Y \setminus H_i} v_i(g)$$
$$(by\ subadditivity)$$
$$v_i(Y \setminus H_i) \leq |Y \setminus H_i| v_i(y_i^*)$$
$$v_i(Y \setminus H_i) \leq n v_i(y_i^*)$$
$$(as\ Y = \cup_i y_i^* \rightarrow |Y| = n \rightarrow |Y \setminus H_i| \leq n)$$

To sum up, we have :

$$v_i(M \setminus Y) \geq v_i(M \setminus H_i) - v_i(Y \setminus H_i) \quad and \quad v_i(Y \setminus H_i) \leq n v_i(y_i^*)$$
$$\Rightarrow v_i(M \setminus Y) \geq v_i(M \setminus H_i) - n v_i(y_i^*)$$
$$\Rightarrow \frac{v_i(M \setminus Y)}{4n} \geq \frac{v_i(M \setminus H_i) - n v_i(y_i^*)}{4n}$$