Algorithmic Game Theory Homework 1

Zhang Luan 112020

Problem1

Construct a function:

$$
\begin{gathered}
f:\triangle \to \triangle \\
y = f(x) \\
y_{p,s_p} := \frac{x_{p,s_p} + G_{p,s_p}(x)}{1+\sum_{s_p'} G_{p,s_p'}(x)} \\
G_{p,s_p}(x) := max\{u_p(s_p;x_{-p}) - u_p(x), 0\}
\end{gathered}
$$

Here $\triangle$ indicates that the definition domain and value domain are a set of all mixed strategies.

In other words, $\triangle = \{x_1, x_2, \dots, x_n\}$, where x_i refers to the mixed strategy of the ith individual.

Suppose the ith individual has m pure strategies, then $x_i = (x_{i1}, x_{i2}, \dots, x_{im}), \sum\limits_k x_{ik} = 1, x_{ik} \geq 0$.

Here x_{p,s_p} refers to the probability of the p-th individual executing pure strategy s_p, which is part of the mixed strategy x_p .

Here u_p refers to the income function of individual p, $u_p(s_p;x_{-p})$ refers to the income of individual p when the p-th individual executes the pure strategy s_p and the others' mixed strategy remains unchanged.

Here $G_{p,s_p}(x)$ measures whether pure strategy s_p is a good choice for the p-th individual.

Here y_{p,s_p} refers to the probability of the p-th person executing pure strategy s_p after the update. Obviously, $\sum\limits_{s_p'} y_{p,s_p'} = 1$.

The constructed function f is a continuous function, and $\triangle$ is a good set(convex, compact). Therefore, f satisfies the fixed point theorem.

Fixed point of f is an NE of the game

It can be known from the fixed point theorem.

$$
\begin{gathered}
x = f(x) \\
\Rightarrow G_{p,s_p}(x) = 0, \forall p, s_p \\
\Rightarrow u_p(s_p, x_{-p}) \leq u_p(x) \\
\Rightarrow \sum_{s_p} u_p(s_p, x_{-p}) y_{p,s_p} \leq u_p(x), \sum_{s_p} y_{p,s_p} = 1 \\
\Rightarrow u_p(x_p, x_{-p}) \leq u_p(x) \\
\Rightarrow fixed\ point\ is\ an\ NE
\end{gathered}
$$

Let's prove $x = f(x) \Rightarrow G_{p,s_p} = 0$ by the method of disproof.

$$y_{p,s_p} := \frac{x_{p,s_p} + G_{p,s_p}(x)}{1 + \sum_{s_p'} G_{p,s_p'}(x)}$$

Assume that there exists p , s_p such that $G_{p,s_p}(x) > 0$

$x_{p,s_p} > 0$, otherwise $x_{p,s_p} = 0$ $y_{p,s_p} > 0$, which does not satisfy the fixed point theorem

Since $u_p(x) = \sum_{s_p} u_p(s_p, x_{-p})x_{p,s_p}$, $x_{p,s_p} > 0$, $G_{p,s_p}(x) > 0 \Rightarrow u_p(s_p, x_{-p}) > u_p(x)$.

There exists some other pure strategy $s_p^{'}$ such that $x_{p,s_p'} > 0$ and $u_p(s_p^{'}, x_{-p}) < u_p(x) \Rightarrow G_{p,s_p'=0}$

And because $\sum_{s_p'} G_{p,s_p'}(x) > 0$, so $y_{p,s_p'} < x_{p,s_p'}$, which violates the fixed point law.

Problem2

Definition 1. A pair of strategies(x,y) is NE iff

$$x^T Ry \geq x^{'T} Ry, \forall x^{'} \in \triangle_m$$
$$x^T Cy \geq x^T Cy^{'}, \forall y^{'} \in \triangle_n$$

Definition 1 shows that for both sides of the game, changing their own hybrid strategy cannot increase their own profits.

Definition 2. A pair of strategies(x,y) if NE iff

$$x_i > 0 \Rightarrow e_i^T Ry \geq e_k^T Ry, \forall k \in [m]$$
$$y_j > 0 \Rightarrow x^T Ce_j \geq x^T Ce_l, \forall l \in [n]$$

For the player $X(or\ Y)$, if the profit obtained by executing pure strategy i is less than that obtained by executing another pure strategy j , $(Ry)_i < (Ry)_j$. Then the player $X(or\ Y)$ will tend to execute the pure strategy j .

So we can define support of x (or y) :

$$supp(x) := \{i \in [n] | x_i \neq 0\}.$$

Each pure strategy in the support of x (or y) should be the best responce to the other, and the profit obtained by executing pure strategy in the support should be the same. Therefore, increasing the execution probability of any pure strategy can not increase the profits, reaching the Nash equilibrium.

In summary, two definition of Nash Equilibrium are equivalent , and the interpretation of Definition 2 is more profound !

Problem 3

Definition :

$\epsilon - Approximate\ NE$: given any $\epsilon > 0$

$$x^T Ry \geq x^{'T} Ry - \epsilon, \forall x^{'} \in \triangle_n$$
$$x^T Cy \geq x^T Cy^{'} - \epsilon, \forall y^{'} \in \triangle_n$$

$\epsilon - Well - Supported\, NE$: given any $\epsilon > 0$

$$x_i > 0 \Rightarrow e_i^T Ry \geq e_k^T Ry - \epsilon, \forall k \in [n]$$
$$y_i > 0 \Rightarrow x^T Ce_j \geq x^T Ce_k - \epsilon, \forall k \in [n]$$

We need to prove :

1. each $\epsilon - Well - Supported\, NE$ is also an $\epsilon - Approximate\, NE$

2. from any $\epsilon^2/8 - Approximate\, NE\ (u, v)$, one can find in polynomial time an $\epsilon - Well - Supported\, NE\ (x, y)$

Proof 1:

For player $X,\ e_i^T Ry$ refers to the gain of player X when executing pure strategy i .

Assume that the player X gains the maximum profit when executing pure strategy i_{max} , and the mixed strategy of the player X is x when reaching $\epsilon - Well - Supported\, NE$.

Then $\forall x_i \in x\ and\ x_i > 0,\ e_i^T Ry \geq e_{i_{max}}^T Ry - \epsilon$.

Obviously $(x^{'T} Ry)_{max} = e_{i_{max}}^T Ry,\ x^{'} \in \triangle_n$, so

$$x^T Ry \geq e_j Ry \geq e_{i_{max}}^T Ry - \epsilon = (x^{'T} Ry)_{max} - \epsilon \geq x^{'T} Ry - \epsilon$$
$$j = \underset{i}{argmin}(x_i),\ x_i \in x\ and\ x_i > 0$$

The same is true for player Y, so $1.$ is proved.

Proof 2:

Because (u, v) is an $\epsilon^2/8 - Approximate\, NE$, so

$$u^T Rv \geq u^{'T} Rv - \epsilon^2/8, \forall u^{'} \in \triangle_n$$
$$u^T Rv \geq u^T Rv^{'} - \epsilon^2/8, \forall v^{'} \in \triangle_n$$

We define set J_1 , which satisfies :

$$\exists i,\ (Rv)_i \geq (Rv)_j + \epsilon/2$$
$$j \in J_1 \quad and \quad 1 \leq j \leq n$$

Then we define i^* , which satisfies :

$$i^* = \underset{i}{argmax}(Rv)_i$$

Now by changing $u_j,\ j \in J_1$ to 0 and changing u_{i^*} to $u_{i^*} + \sum_{j \in J_1} u_j$, the profit of the first player increases at least $(\epsilon/2) \sum_{j \in J_1} u_j$. Because the increased profit of the first player must be less than $\epsilon^2/8,\ \sum_{j \in J_1} u_j < \epsilon/4$. Similarly, we can define J_2 , which can deduce $\sum_{j \in J_2} v_j < \epsilon/4$.

Next, we set all $\{u_j | j \in J_1\}$ and $\{v_j | j \in J_2\}$ to 0, and uniformly increase the probabilities of other strategies to obtain a new pair of mixed strategies (x, y).

Because $\sum_{j \in J_2} v_j < \epsilon/4$, $|(Ry)_i - (Rv)_i| \leq \epsilon/4$ (We assume that any element in matrix R is between 0 and 1.) Therefore, $||(Ry)_i - (Ry)_j| - |(Rv)_i - (Rv)_j|| \leq \epsilon/2$. In the above, we set all $\{u_{j^{'}} | j^{'} \in J_1, (Rv)_i \geq (Rv)_{j^{'}} + \epsilon/2\}$ to 0, so:

$$|(Rv)_i - (Rv)_j| \le \epsilon/2,\ j \in \underset{j}{argmax}(u_j > 0)$$
$$|(Ry)_i - (Ry)_j| \le |(Rv)_i - (Rv)_j| + \epsilon/2$$
$$|(Ry)_i - (Ry)_j| \le \epsilon$$
$$x_i > 0 \Rightarrow (Ry)_i \ge (Ry)_k - \epsilon, \forall k \in [n]$$

The same is true for another player, so $2.$ is proved.

Problem 4

4.1

	$\{a\}$	$\{b\}$	$\{c\}$	$\{a,b\}$	$\{a,c\}$	$\{b,c\}$	$\{a,b,c\}$
$agent\ 1$	3	2	2	4	4	3	10
$agent\ 2$	2	2	2	3	3	3	8

Suppose we have a set N of two agents and a set of M of three goods. The valuations of different allocations are as above, which meet the requirement of general monotonicity, i.e., $v_i(\emptyset) = 0$ and $S \subseteq T \to v_i(S) \le v_i(T)$.

We allocate $\{a\}$ to agent 1 and allocate $\{b, c\}$ to agent 2. Since $v_1(\{a\}) \ge v_2(\{b,c\})$ and $v_2(\{b,c\}) \ge v_1(\{a\})$, the allocation is $EF1$. Since $v_1(\{a\} \cup \{b\}) = v_1(\{a,b\}) < \frac{1}{2}v_1(\{a,b,c\})$ and $v_1(\{a\} \cup \{c\}) = v_1(\{a,c\}) < \frac{1}{2}v_1(\{a,b,c\})$, the allocation is not $Prop1$.

4.2

	$\{a\}$	$\{b\}$	$\{c\}$
$agent\ 1$	10	5	1
$agent\ 2$	1	5	10

Suppose we have a set N of two agents and a set of M of three goods. The valuations are as above, which is additive.

round robin algorithm : We arbitrarily fix an ordering of agents, which is $1 \to 2$. In the first round, agent 1 select his favorite good a . In the second round, agent 2 select his favorite good c from the remaining goods. In the final round, agent 1 select good b . The social-welfare is $10 + 10 + 5 = 25$.

envy-cycle-elimination algorithm : We allocate good c to agent 1 and allocate good b to agent 2. Since there is a envy-cycle ,i.e., $v_1(\{c\}) < v_1(\{b\})$ and $v_2(\{b\}) < v_2(\{c\})$, we exchange the bundles of agen 1 and agent 2, i.e., agent 1 $(\{b\})$ and agent 2 $(\{c\})$. Obviously, there is no envy-cycle in the envy-graph at this time, so agent 1 and agent 2 are both source. Then we allocate good a to agent 2. The social-welfare is $5 + 10 + 1 = 16$.

The round robin algorithm achieves better social-welfare than the envy-cycle-elimination algorithm under certain choices !

4.3

	$\{a\}$	$\{b\}$	$\{c\}$
$agent\ 1$	10	5	1
$agent\ 2$	1	0	10

Suppose we have a set N of two agents and a set of M of three goods. The valuations are as above, which is additive.

We allocate $\{b\}$ to agent 1 and allocate $\{a, c\}$ to agent 2. Since $v_1(\{b\}) \geq v_1(\{a, c\} \setminus \{a\})$ and $v_2(\{a, c\}) \geq v_2(\{b\})$, the allocation is $EF1$. In additon, we find that no other allocation is better for all, i.e., there is no allocation Y , s.t., $v_i(Y_i) \geq v_i(X_i)$ for all $i \in [n]$. Thus the allocation is $EF1 + PO$. Since $v_1(\{b\}) < v_1(\{a, c\} \setminus \{c\})$, the allocation is not EFX.

4.4

	$\{a\}$	$\{b\}$	$\{a, b\}$
$agent\ 1$	10	10	10
$agent\ 2$	10	10	10

Suppose we have a set N of two agents and a set of M of two goods. The valuations are as above, which is sub-additive, i.e., $v_1(\{a\}) + v_1(\{b\}) \geq v_1(\{a\} \cup \{b\})$ and $v_2(\{a\}) + v_2(\{b\}) \geq v_2(\{a, b\})$.

Obviously $MMS_1 = v_1(M) = 10$ and $MMS_2 = v_2(M) = 10$.

4.5

Proof :

Suppose we have a set N of n agents , a set of M of m goods and a set K of k $\alpha - MMS$ allocations .

If only one $\alpha - MMS$ allocation exists, i.e., $|K| = 1$, it is obvious that the allocation is PO.

If more than one $\alpha - MMS$ allocation exists, i.e., $|K| \geq 2$, we let agent 1 choose his favorite allocation set K_1 from K . If $|K_1| = 1$, since any allocation not from K_1 is worse for agent 1 , i.e., $v_1(A_1^j) < v_1(A_1^i)$ for any $j \notin K_1$ and $i \in K_1$, the allocation from K_1 is PO . If $|K_1| \geq 2$, we let agent 2 choose his favorite allocation set K_2 from K_1 and judge the size of K_2. We repeat the above operation until the size of allocation set K_i choosed by agent i is 1 . Then the allocation from K_i is PO . Any allocation not from K_i can not guarantee that the valuations from agent 1 to agent i do not decrease at the same time .

Of course, different selection order may lead to different $\alpha - MMS + PO$ allocation. Here we fix the order $(1 \to n)$, just to prove that if an $\alpha - MMS$ allocation exists for an instance, then an $\alpha - MMS + PO$ allocation also exists .