Compsci 590T Homework 7

Student: Brian Sun, Collaborator: Thai On

1 Problem 1.a

$(\star)$ Base on the given information we know $A^* = max_{A \in A(N,G)} NW(A)$
If A^* is not Pareto efficient, then base on the Pareto efficient statement we know for any other allocation A', it is either the cases that there exists no player $i \in N$ for which $v_i(A'_i) < v_i(A^*_i)$, and the two allocations cannot offer the same utility to all players: $v_i(A^*_i) = v_i(A'_i)$ for all $i \in N$. Therefore we know there should not have any player that $v_i(A'_i) < v_i(A^*_i)$ and at least one player such that $v_i(A'_i) > v_i(A^*_i)$.

There will have two cases:
1. There are some players utility are the same and the rest of the players have greater utility in A' than A^*.
2. There are no player utility are the same, which all the players have greater utility in A' than A^*.

Case 1 :
Set N_{same} be all the players who have the same utility between A' and A^*
$a.$ $i \in N_{same},\ v_i(A'_i) = v_i(A^*_i)$ therefore $i \in N_{same},\ \prod v_i(A'_i) = \prod v_i(A^*_i)$
$b.$ $i \in N \backslash N_{same},\ v_i(A'_i) > v_i(A^*_i)$ therefore $i \in N \backslash N_{same},\ \prod v_i(A'_i) > \prod v_i(A^*_i)$
Base on a, b we have $i \in N,\ \prod v_i(A'_i) > \prod v_i(A^*_i)$
Since $\prod v_i(A'_i) > \prod v_i(A^*_i)$ which contradict $(\star)$ therefore case 1 cannot happen.

Case 2 :
Since $i \in N,\ v_i(A'_i) > v_i(A^*_i)$ so $\prod v_i(A'_i) > \prod v_i(A^*_i)$ since this contradict $(\star)$ therefore case 2 cannot happen as well.

Since both cases cannot happen, therefore A^* is Pareto efficient.

2 Problem 1.b

Assume the allocation A^* that maximizes the Nash welfare is not envy-free up to one good then that means for any two players $i, j \in N$, there exists no item $g \in A_j^*$ such that $v_i(A_i^*) < v_i(A_j^* \backslash \{g\})$.

If this is true, consider the new allocation A' by move the item $g^* \in A_j^*$ to A_i^*. Defined g^* as an item that maximizes the value $v_i(g) - v_j(g), \forall g \in A_j^*$.

Then we know:

$$\begin{aligned} NW(A') &= v_i(A_i^* \cup g^*) \times v_j(A_j^* \backslash g^*) \\ &= \left(\sum_{g \in A_i} v_i(g) + v_i(g^*) \right) \times v_j(A_j^* \backslash g^*) \\ &= (v_i(A_i^*) + v_i(g^*)) \times v_j(A_j^* \backslash g^*) \\ &= v_i(A_i^*) v_j(A_j^* \backslash g^*) + v_i(g^*) v_j(A_j^* \backslash g^*) \end{aligned}$$

$$\begin{aligned} NW(A^*) &= v_i(A_i^*) \times (v_j(A_j^* \backslash g^*) + v_j(g^*)) \\ &= v_i(A_i^*) v_j(A_j^* \backslash g^*) + v_i(A_i^*) v_j(g^*) \end{aligned}$$

compare $NW(A^*)$ and $NW(A')$:

$$\begin{gathered} NW(A^*) : NW(A') \\ = v_i(A_i^*) v_j(A_j^* \backslash g^*) + v_i(A_i^*) v_j(g^*) : v_i(A_i^*) v_j(A_j^* \backslash g^*) + v_i(g^*) v_j(A_j^* \backslash g^*) \\ = v_i(A_i^*) v_j(g^*) : v_i(g^*) v_j(A_j^* \backslash g^*) \longrightarrow (\star) \end{gathered}$$

Then base on the inequality we know:

$$\frac{v_j(g^*)}{v_i(g^*)} \leqslant \frac{v_j(A_j^*)}{v_i(A_j^*)}$$

Since $v_i(A_i^*) < v_i(A_j^* \backslash g^*)$ so $v_i(A_i^* \cup g^*) < v_i(A_j^*)$ therefore:

$$\frac{v_j(g^*)}{v_i(g^*)} \leqslant \frac{v_j(A_j^*)}{v_i(A_j^*)} < \frac{v_j(A_j^*)}{v_i(A_i^* \cup g^*)} \longrightarrow (1)$$

Base on (1) we know:

$$\begin{aligned} \frac{v_j(g^*)}{v_i(g^*)} &< \frac{v_j(A_j^*)}{v_i(A_i^* \cup g^*)} \\ &< \frac{\sum_{g \in A_j^* \backslash g^*} v_j(g) + v_j(g^*)}{\sum_{g \in A_i^*} v_i(g) + v_i(g^*)} \\ &< \frac{v_j(A_j^* \backslash g^*) + v_j(g^*)}{v_i(A_i^*) + v_i(g^*)} \\ v_j(g^*) \times (v_i(A_i^*) + v_i(g^*)) &< v_i(g^*) \times (v_j(A_j^* \backslash g^*) + v_j(g^*)) \\ v_i(A_i^*) v_j(g^*) + v_i(g^*) v_j(g^*) &< v_i(g^*) v_j(A_j^* \backslash g^*) + v_i(g^*) v_j(g^*) \\ v_i(A_i^*) v_j(g^*) &< v_i(g^*) v_j(A_j^* \backslash g^*) \longrightarrow (\blacklozenge) \end{aligned}$$

Base on $(\star)$ and $(\blacklozenge)$ we know $NW(A^*) < NW(A')$
Since $NW(A')$ is greater than $NW(A^*)$ which contradict A^* is the maximizes the Nash welfare. Therefore A^* that maximizes the Nash welfare is envy-free up to one good

3 Problem 2

Assume g^* is not exists, then that means no matter which item $g \in A_i$ we remove from A_i there will at least exist one player $j \in E_i(A)$ still envy i. Which another way to say this is in the envy graph if there exist an node n_i that has incoming edge, then no matter which item you remove from A_i, n_i will still has at least one incoming edges.

Since no matter which item we remove from A_i, n_i will still has at least one incoming edges. Which means i got assign item at least one time when i already cost someone envy him.

Then there will have 2 possible situations for when we assign item to player i when i already cost someone envy him.
1. when there are still players exist that don't have incoming edges.
2. when every players have at least one incoming edges.

For situation 1: this situation cannot happen, since the algorithm will only assign item to player i with no incoming edges. So the algorithm will not assign item to player i since he has incoming edges.

For situation 2: this situation cannot happen. Since if all players has incoming edges, that means in the envy graph every nodes has at least one path to go back to itself, which there will have at least one cycle on the graph. Since if the graph have cycles, the algorithm will decycle the cycle, to make the envy graph to a acyclic envy graph. Since there are no cycle in the graph, which means there are no path exist that can make any nodes go back to itself, which means there exist at least one node don't have incoming edges. So the situation of every players have at least one incoming edges cannot happen.

Since both possible situations for when we assign item to i when i already has incoming edges cannot happen in the algorithm. Therefore the contrition is false, so g^* exist.

4 Problem 3

Given there are at least one MMS allocation, then to show there will exist at least one Pareto-optimal MMS allocation. We can prove it by pick a random MMS allocation point and try to make Pareto improvement to reach to the Pareto-optimal point. Since Pareto improvement is a action that increase player's utility value without decrease any other player's utility value. So keep repeating this type of action until we get to a stage that no matter what we do, there will always exist at least one player decrease his utility value, which base on the Pareto-optimal definition, this stage will be the Pareto-optimal allocation point.

There will have two situation:
1. when there are only one Pareto-optimal allocation.
2. when there are more than one Pareto-optimal allocation.

Situation (1): If there are only one Pareto-optimal allocation, then we need to check if that Pareto-optimal point satisfy MMS or not.

a. Assume the Pareto-optimal allocation point does not satisfy MMS.
Given the fact that there exist at least one MMS allocation, but Pareto-optimal allocation does not satisfy MMS which means we cannot make make any Pareto improvement action from this MMS allocation point to the Pareto-optimal point.
Because the Pareto-optimal point does not satisfy MMS, which means there will have at least one value in this point that does not reach the MMS threshold. Therefore we cannot make any Pareto improvement from the MMS allocation point to the Pareto-optimal point without decrease any value in the MMS allocation point. Since the space is bounded therefore the Pareto-optimal point must satisfy MMS so we can reach there starting from the MMS allocation point we pick.

b. Assume the Pareto-optimal allocation point satisfy MMS.
Randomly pick a MMS allocation point and try to make Pareto improvement actions. Since Pareto improvement is increase player's utility value without decrease any other player's utility value, therefore any Pareto improvement starting from any MMS allocation point will still satisfy MMS. Therefore starting from a MMS allocation point and try to make Pareto improvement until we cannot, then that stage will be the Pareto-optimal allocation point and is satisfy MMS.

Situation (2): base on situation 1.a, we know the space is bounded, therefore there must exist at least one Pareto-optimal allocation point that satisfy MMS.
Since there are at least one Pareto-optimal allocation point that satisfy MMS, then do the same thing in situation 1.b, then we can reach to the Pareto-optimal allocation point and is satisfy MMS.

Therefore if there exists at least one MMS allocation, then there exists a Pareto-optimal MMS allocation.

5 Problem 4.a

Assume the result can output a not EF-1 allocation, then we can try to make a case that given any two player $a, b \in N$ that a envy b more than one good.

Therefore two different order:
1. The order is a starts ahead of b
2. The order is b starts ahead of a

For order (1): Every round a will pick his most preferred available course $j*$ if $v_{aj*} > 0$. Since b pick the class after a, therefore the best b can do is to pick the class that is at least as good as what a pick. Therefore we have:

$$\begin{aligned}
\text{first round: } & v_a(c_{a1}) \geqslant v_a(c_{b1}) \qquad \text{define } c_{ik} \text{ is the class } i \text{ pick in the } k \text{ round} \\
\text{second round: } & v_a(c_{a2}) \geqslant v_a(c_{b2}) \\
\text{third round: } & v_a(c_{a3}) \geqslant v_a(c_{b3}) \\
& \quad . \qquad\quad . \\
& \quad . \qquad\quad . \\
& \quad . \qquad\quad . \\
\text{last round: } & v_a(c_{an}) \geqslant v_a(c_{bn})
\end{aligned}$$

Sum all the value from both side we will get:

$$\sum_{k=1}^{n} v_a(c_{ak}) \geqslant \sum_{k=1}^{n} v_a(c_{bk})$$

Since student a value his classes weakly greater than what student b has, therefore if a pick before b, we cannot construct an allocation that make a envy b more than one good.

For order (2): Every round a will pick his most preferred available course $j*$ if $v_{aj*} > 0$. Therefore the best b can do after a pick his class is to pick an class that is at least as good as what a pick. Therefore we have:

$$\begin{aligned}
v_a(c_{a1}) &\geqslant v_a(c_{b2}) \\
v_a(c_{a2}) &\geqslant v_a(c_{b3}) \\
v_a(c_{a3}) &\geqslant v_a(c_{b4}) \\
. \qquad & \qquad . \\
. \qquad & \qquad . \\
. \qquad & \qquad . \\
v_a(c_{an-1}) &\geqslant v_a(c_{bn})
\end{aligned}$$

Sum all the value from both side we will get:

$$\sum_{k=1}^{n} v_a(c_{ak}) \geqslant \sum_{k=2}^{n} v_a(c_{bk})$$

Since student a value his classes always weakly greater than student b's class without the first class (first class must be $v_a(c_{b1}) \geqslant v_a(c_{a1})$, since we are trying to construct an allocation that make a envy b), therefore if b pick before a, we still cannot construct an allocation that make a envy b more than one good.

Therefore the algorithm output a EF-1 allocation, since both order cannot construct an EF-n allocation.

6 Problem 4.b

For example:

	590T (1)	589 (1)
Alice	1	1
Bob	2	0

We set the order to be A → B. The algorithm proceeds as follows
(i) Alice gets 590T
(ii) Bob gets 589
(iii) Since all courses are at capacity, we terminate.

The result is not PO allocation. Since if Bob gets 590T and Alice gets 589, the utility for Bob increase and Alice stay the same, which is a parental improvement. At That stage we cannot make parental improvement any more, therefore Bob gets 590T and Alice gets 589 is the PO allocation which is different than the result that RR algorithm gives us.

7 Problem 5