Homework2

Liu Yichen
ID 1120202926

February 3, 2023

1 Problem 1

Problem1: Pick your favourite result in our class and state your reason.

Answer :

In fact, I don't know how to define 'result' clearly.

If teacher means that it is a definition. My favourite definition is Envy-freeness

Envy-freeness, also known as no-envy, is a criterion for fair division. It says that, when resources are allocated among people with equal rights, each person should receive a share that is, in their eyes, at least as good as the share received by any other agent. In other words, no person should feel envy.

In class, teacher introduce a way to use value function V_i, then the definition above is equivalent to : $V_i(X_i) \geq V_i(X_j)$

Or we could say if $\forall i, j\ i\ envy\ j\ V_i(X_i) < V_i(X_j)$.

The definition is really concise. Using math function to depict real-life scenes which give me enlightenment.

If teacher means a result of inference. I choose $pEF1- > EF1 + PO$, though teacher don't tell the reason. I think it use price to evaluate envy freeness problem which makes this problem easier.

We can gain price by the way such as $maxvalue$. And then we can judge whether a strategy meets Pareto Optimal without difficult proof process.

2 Problem 2

The problem background is based on subsets.

Problem 1.

Prove that for any $S \subseteq T \subseteq M \ and \ j \in M, we \ have$

$$\frac{v(T+j)}{v(T)} \leq \frac{v(S+j)}{v(S)} \tag{1}$$

We proof this problem by a lemma which is discovered by me. This lemma is $v(S+i)+v(T) \geq v(S)+v(T+i)$, if $i \subseteq T$ then it is self-evident, if $i \notin T$,then we can use the condition given by the problem to prove it.

Now, we get the lemma which says a profound truth that If two value functions' difference between independent variables is similar , then the larger the independent variable, the smaller the difference. Or in mathematical language, this value function's first derivative is greater than 0, the second derivative is less than 0.

If we want to prove (1) , we can reduce it to $\frac{v(S)}{v(T)} \leq \frac{v(S+j)}{v(T+j)}$

We start to prove now. From $\frac{v(S+j)}{v(T+j)} = \frac{v(S+j)+v(T+j)-V(T+j)}{v(T+j)}$. We have $v(S+i)-v(T+i) \geq v(S)-v(T)$. We can reduce it to $\frac{v(S+j)}{v(T+j)} \geq 1 + \frac{v(S)-v(T)}{v(T+j)}$. Because $S \subseteq T \subseteq M$ and the problem is under general valuation, we can get $v(s)-v(T) \leq 0$, for $v(T+j) \geq v(T)$,than we can reduce the above formula to $\frac{v(S+j)}{v(T+j)} \geq 1 + \frac{v(S)-v(T)}{v(T+j)} \geq 1 + \frac{v(S)-v(T)}{v(T)} = \frac{v(S)}{v(T)}$. Finally ,we have $\frac{v(S)}{v(T)} \leq \frac{v(S+j)}{v(T+j)}$. We prove problem 2.1.

Problem 2.

Prove that for any T $\subseteq$ M we have

$$v(T) \geq \sum_{k \in T} [v(T) - v(T-k)] \tag{2}$$

Because we get subset condition. We can consider the problem this way. We divide T into allocation $(k_1, ..., k_n)$, for $\forall T \subseteq M$.

To prove this problem we need to use the lemma above.We can review what the lemma says. The lemma says that $\forall \ S \subseteq T \subseteq M, \forall j \subseteq M, v(T+j) - V(s+j) \leq v(T) - v(S)$. So in problem 2.2, $v(T) = v(T)-v(T-k_1)+v(T-k_1)-v(T-k_1-k_2)+v(T-k_1-k_2)-...-v(T-k_1-...-k_n)+v(T-k_1-...-k_n)$, because we divide T into allocation $(k_1, ..., k_n)$, for $\forall T \subseteq M$. So $v(T-k_1-...-k_n) = 0$.The last added item can be omitted, so we get n subtraction formulas.

We take a group for example. $v(T)-v(T-k_1) = v(T)-v(T-k_1)$, in this group, the left and right sides of the formula are equal, but when the subscript rises , the situation changes.For k_2, obviously we can find that $v(T)-v(T-k_2) \leq v(T-k_1)-v(T-k_1-k_2)$. Take the sum on both sides of the inequality, the direction of the inequality sign remains unchanged.The result of summing the right-hand side of the inequality is $v(T)$.The result of summing the left-hand side of the inequality is $\sum_{k \in M}[v(T)-v(T-k)]$.

Finally, we get that $v(T) \geq \sum_{k \in M}[v(T) - v(T-k)]$.We prove problem 2.2.

3 Problem 3

Problem :Design a pseudo-polynomial time algorithm to obtain 1/2-EFX allocation when agents have monotone subadditive valuations

Answer:

We first define H_i is a set of n highest-valued items for agent i. Then, we allocate one item per

algorithm: If we want to use this algorithm, we need to make sure that $|A_i| \geq 2$ the reason why is that ,in 1/2-EFX algorithm, $|A_i| \geq 1$. For matching step ,$|A_i| \geq 2$.

First, we need to know what is subadditive valuations. A valuation v is subadditive if$v(S) + v(T) \geq v(S \cup T)$. We can summarize that every additive valuation is submodular, and every submodular valuation is subadditive. It is known that the possible existence of EFX allocations for possibly distinct valuations and $n3$ remains an open question, even for additive valuations. So we talk about the situation about $n = 2$.

We design a algorithm which is reminiscent of our algorithm for polynomial-time computation with identical rankings.

In preparation Phase, initially all goods are in the pool P, and we proceed in rounds until P is empty, maintaining the invariant that the partial allocation at the end of each round is EFX.

We need to introduce a lemma. We define $A = (A_1, A_2, ..., A_n)$ be a a-EFX allocation , envy graph $G = (V, E)$, which contains a cycle. We define $B = (B_1, B_2, ..., B_n)$ be a a-EFX allocation , envy graph $H = (V, E)$, which contains a cycle at the same way.

Since the envy graph is a cyclic, we can always find an unenvied player j and give an arbitrary good g from P to j. So ,there will exist player i to envy j in violation of $\frac{1}{2} - EFX$. Then, we return all of i's current bundle to P and let i's new bundle be justg. In order for i to go from not envying j to envying j in violation of $\frac{1}{2} - EFX$. We add $g^*$to A_j must have caused $v_i(A_j)$ to at least double. With the subadditivity of v_i, to show that $v_i(g)$ must be larger than i's value at the beginning of the round. If i envies any player, it remains consistent with$\frac{1}{2} - EFX$. Under above reason, any envy directed toward i will be EFX. So i will have only one good.

At the time of each round, P will decrease in size or the sum of utilities will increase. Then, we can use potential function argument to show the algorithm terminates in a pseudo-polynomial time.

Or from teacher, I think in this way. We define a g^* as the last good allocated to j. If i is a source allocate the most valuable item which has not been allocated. Then, we have $v_i(A_i) \geq v_i(g^*)$.Because i is a source, we have $v_i(A_i) \geq v_i(A_j \backslash g^*)$. We sum them together ,then we have $2v_i(A_i) \geq v_i(A_j) \geq v_i(A_j \backslash g), \forall g \in A_j$. We prove the algorithm can find $\frac{1}{2} - EFX$ allocation.

Or concisely we can think in this way(Summary in my own language). Obviously we can obtain $\frac{1}{2} - EFX$ in pseudo-polynomial time when agents have subadditive valuations by envy cycle. The process is as follows. By convention we should initial all goods in the pool P and initialize all allocations to the empty set. When the pool is not empty, loop.Remove an arbitrary good from pool and give it to an envied agent. Then we judge whether A_i is an $\frac{1}{2} - EFX$ allocation, if the answer is yes, then return i's old allocation to pool P, and give i to g^*, we ensure the graph is a cyclic by clean envy cycles. Repeat until loop ends.

The algorithm is below.

REFERNECE:Plaut, B.; and Roughgarden, T. 2018. Almost Envy-Freeness with General Valuations. In Proceedings of the 29th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA),2584–2603.

Algorithm 1 Algorithm to find $\frac{1}{2} - EFX$ allocation when n agents have subadditive valuations

function GET EFX ALLOCTION$(n, m, (v_1, ..., v_n))$
  $P \leftarrow [m]$\{put all goods in the pool\}
  for $i \in [n]$ **do**
    $A_i \leftarrow \emptyset$
  end for
  while $P \neq \emptyset$ **do**
    $g^* \leftarrow pop(P)$
    $j \leftarrow GetUnenviedFromAllocation(A_1, ..., A_n)$
    $A_j \leftarrow A_j \cup g^*$
    if $\exists i \in [n], g \in A_j, and\ A_j\ is\ a\ \frac{1}{2} - EFX\ allocation$ **then**
      $P \leftarrow P \cup A_i$
      $A_j \leftarrow (Aj - g^*)$
      $A_i \leftarrow g^*$
    allocation$(A_1, ..., A_n) \leftarrow CleanEnvyCycles(A_1, ..., A_n)$
  end while
  return$(A_1, ..., A_n)$
end function=0

4 Problem 4

Problem: Prove the "Exercise" inequality on Page 40 of our slides lec8-FD-MNW.pdf

$$v_i(A_i) \geq \frac{v_i(M \backslash Y)}{4n} \geq \frac{v_i(M \backslash Y) - nv_i(y_i^*)}{4n} \tag{3}$$

Answer: Here we can partition the instance into n bundles $X_1, X_2, ..., X_n$. Referring to Plaut and Roughgarden 2018, it is known that there exists in the context of indivisible goods ($\frac{1}{2} - EFX$).Then, we have $\forall i, j, v_i(X_i) \geq \frac{1}{2} v_i(X_j \backslash g), for\ all\ g \in X_j$.Actually, we discuss in Problem 3 easily. We know that $|X_i| \geq 2$ for matching step. And we have that $v_i(X_i) \geq \frac{1}{2} V_i(X_j \backslash g)$for all $g \in X$, which haves $v_i(X_i) \geq \frac{1}{2} max(V_i(X_j \backslash g), V_i(g))$. Then, For $nv_i(X_i)$

$$nv_i(X_i) \geq \sum_{|X_j \geq 2|} \frac{1}{2}\frac{1}{2}(V_i(X_j \backslash g) + V_i(g)) \geq \frac{1}{4} \sum_{|X_j \geq 2|} v_i(X_j) \geq \frac{1}{4} v_i(\cup_{|X_j| \geq 2} X_j) \tag{4}$$

The second and third inequality sign above is from subadditivity. We define Y be the set of all the goods in the singleton bundles in X. i.e., $Y = \{y_i | X_i = \{y_i\}\}, y_i$ is allocated to i. From (4) we have the guarantee that every agent $v_i(X_i) \geq \frac{1}{4n} v_i(M \backslash Y)$. Until now, we rename $X$to A, then the first inequlity sign of Problem 4 is proved. Considering the second sign, we can notice that we only need to consider the differences in the numerator. By subadditivity, $v_i(M \backslash H_i) \leq v_i(M \backslash H_i \cap Y) + v_i(M \backslash H_i \backslash Y)$.

Then, we have $v_i(M \backslash H_i) - v_i(M \backslash H_i \cap Y) \leq v_i(M \backslash H_i \backslash Y) \leq v_i(M \backslash Y)$.

Let consider the scenario that we have an envy-free allocation, i.e., a partition of the goods into n bundles $X_1, ..., X_n$, then we have $v_i(X_i) \geq v_i(X_j)$ which implies

$$nv_i(X_i) \geq \sum_{j \in [n]} v_i(X_j) \geq v_i(\cup_{j \in [n]} X_j) = v_i(M) \tag{5}$$

In Problem 4, agent i doesn't envy items in set $(M \backslash H_i \backslash Y)$. Therefore, from (5), we have $nv_i(y_i^*) \geq v_i(M \backslash H_i \backslash Y)$.

So Problem 4 is proved.