CS 580RM: Algorithmic Game Theory, Fall 2021
HW 1 Solutions

Instructions:

1. We will grade this assignment out of a total of 40 points.
2. Feel free to discuss with fellow students, but write your own answers. If you do discuss a problem with someone then write their names at the starting of the answer for that problem.
3. Please type your solutions if possible in Latex or doc whichever is suitable, and submit on Gradescope.
4. Even if you are not able to solve a problem completely, do submit whatever you have. Partial proofs, high-level ideas, examples, and so on.
5. Except where otherwise noted, you may refer to lecture slides/notes. You cannot refer to textbooks, handouts, or research papers that have not been listed. If you do use any approved sources, make sure you cite them appropriately, and make sure to write in your own words.
6. No late assignments will be accepted.
7. By AGT book we mean the following book: Algorithmic Game Theory (edited) by Nisan, Roughgarden, Tardos and Vazirani. Its free online version is available at Prof. Vijay V. Vazirani's webpage.

---

1. (*Envy-freeness and Proportionality*)
   (a) (2 points) Give an example with monotone valuations where a Prop1 allocation is not EF1.
   (b) (3 points) Prove that EF1 $\Rightarrow$ Prop1 for instances with additive valuations.
   (c) (5 points) Prove that when agents have (non-identical) additive valuations but the same ordered preferences, an EFX allocation can be found in polynomial time.
   [*Hint: Try to see if any of the algorithms we covered in class works*]

   (a) Consider the following instance with 2 agents a_1, a_2, and 3 goods, g_1, g_2, g_3. The valuations of the agents are identical functions defined as follows. For the empty set, $v(\emptyset) = 0$, and for singleton bundles, $v(g_1) = v(g_2) = 2$, $v(g_3) = 1$. For all two-sized bundles, $v(\{g_1, g_2\}) = v(\{g_2, g_3\}) = v(\{g_1, g_3\}) = 3$ and for the grand bundle, denoted by M, $v(M) = v(\{g_1, g_2, g_3\}) = 3$.

Consider an allocation A for this instance that assigns $A_1 = \{g_3\}$ to a_1, and $A_2 = \{g_1, g_2\}$ to a_2.

For a_2, $v(A_2) \geq v(M)/2$ and for a_1, $v(A_1 \cup \{g_1\}) = 3 \geq v(M)/2$, hence the allocation is Prop1. However, as $v(A_1) = 1$ and $v(A_2 \backslash \{g\}) = 2$ for all g, removing any item g from A_2 does not satisfy $v(A_1) \geq v(A_2 \backslash \{g\})$, hence the allocation is not EF1.

(b) Let M, N and $A = \{A_1 \ldots, A_n\}$ respectively denote the sets of items, agents, and an EF1 allocation of M among N. Consider any agent i. We will show that A is Prop1 for i. The EF1 property implies,

$$\begin{aligned} \forall i, j \in N, \exists g \in M: \ & v_i(A_i) \geq v_i(A_j \backslash \{g\}) \\ & \Rightarrow v_i(A_i) \geq v_i(A_j) - v_i(\{g\}) \\ & \Rightarrow v_i(A_i) \geq v_i(A_j) - v_i(\{g*\}), \text{ where } g* \in \text{argmax}_{g \in M} v_i(\{g\}) \end{aligned} \tag{1}$$

Adding the above inequalities for all j, we get $n \cdot v_i(A_i) \geq \sum_j v_i(A_j) - n \cdot v_i(\{g*\})$, i.e., $v_i(A_i) \geq v_i(M)/n - v_i(\{g*\})$, hence $v_i(A_i) + v_i(\{g*\}) \geq v_i(M)/n$. This shows that A is Prop1 for any agent i, hence is a Prop1 allocation.

(c) Consider the following algorithm. Arrange all goods in a non-decreasing order of preferences according to (all) the agents' valuation functions. Run the envy cycle elimination algorithm that gives an EF1 allocation with general monotone valuations.

We will show that this algorithm gives an EFX allocation when the order of preferences is identical similarly how we show the EF1 guarantee for general valuations, by induction. We will show that the partial allocation obtained after allocating the first k goods is EFX, given that the one obtained after allocating the first k-1 goods is EFX.

The k^{th} good is allocated to some source agent of the (acyclic) envy graph. Let s denote this source agent. Let A^{post} be the allocation obtained after allocating the k^{th} good to s, and A be the allocation obtained after eliminating the cycles in its envy cycle graph. We first show that A^{post} is EFX.

For the partial allocation, say A^{pre} before allocating the k^{th} good, say g^k, we have, for every pair of agents i, j,

$$\forall i, j, g \in A_j^{pre}: \ v_i(A_i^{pre}) \geq v_i(A_j^{pre} \backslash \{g\}) \tag{2}$$

We also know, as s is the source agent of this allocation's envy graph,

$$\forall i: \ v_i(A_i^{pre}) \geq v_i(A_s^{pre}). \tag{3}$$

After assigning g^k to s, for the allocation A^{post} equation (2) is true for all $i, j \neq s$, as none of these bundles have changed. We also have $v_i(A_i^{post}) \geq v_i(A_s^{post} \backslash \{g^k\})$, and as g^k has equal or lower value than all the goods in A^{post} because the goods are assigned in non-increasing order of preference, we have $v_i(A_i^{post}) \geq v_i(A_s^{post} \backslash \{g\})$, for all $i, g \in A_s^{post}$. Hence, equation (2) is true if we replace A^{pre} with A^{post}, proving A^{post} is EFX.

Now note that $v_i(A_i) \geq v_i(A_i^{post})$ for every agent i, as the envy cycle elimination process only gives agents bundles of higher value. Also, the bundles in A^{post} and A are the same, with only their assigned agents possibly changed. Let j' be the agent who obtained j's bundle from A^{post} in A. Hence, we have $v_i(A_i) \geq v_i(A_i^{post}) \geq v_i(A_j^{post} \backslash \{g\}) =$

$v_i(A_{j'} \setminus \{g\})$, for every $i, j, j', g \in A_{j'}$. As the set of inequalities over all i, j give corresponding inequalities over all i, j', we have $v_i(A_i) \geq v_i(A_{j'} \setminus \{g\})$, for all $i, j', g \in A_{j'}$, thus proving A is EFX.

As the initial allocation before assigning any good is trivially EFX, by induction, the final allocation after assigning all goods is also EFX.

---

2. (*MMS*)

   (a) (2 points) For additive valuation functions, we showed $MMS_i \leq \frac{v_i(M)}{n}$ for all agents i. Give an example with sub-additive valuation functions where this is not true.

   (b) (1 point) Prove that if an α-MMS allocation exists for an instance, then an α-MMS+PO allocations also exists.

   (c) (1 point) When agents have identical valuation functions, what is the highest value of $\alpha \in (0, 1]$ for which an α-MMS allocation always exists?

   (d) (6 points) For the case with additive valuation functions when $v_{ij} \leq \epsilon$ for all i, j, prove that an EF1+$(1-\epsilon)$-MMS allocation exists and can be computed in polynomial time.

   (a) Consider 2 agents i, j, and 2 items g, h. Let the valuation function of i be as follows. Their value for the empty set is 0, and the value for each singleton bundle is v. The value of both items together is also v. Then agent i can form 2 bundles of value v, hence their MMS value is v. $v(M)/2 = v/2$, hence $MMS > v(M)/2$.

   (b) Any allocation that Pareto dominates an α-MMS allocation gives an equal or higher valued bundle to every agent, hence is also α-MMS. Thus, the Pareto optimal bundle among all α-MMS allocations is α-MMS+PO.

   (c) 1. (Justification not required: the allocation that defines the MMS value for every agent is the same, as they have identical valuation function. Hence, these MMS defining allocations are also 1-MMS allocations for all agents)

   (d) We will show that any EF1 allocation also ensures a $(1-\epsilon)$-MMS guarantee, thus any EF1 algorithm computes the required allocation.

   As $v(M) = n$, and there are n bundles in the EF1 allocation, by the pigeonhole principle, for every agent there is at least one bundle they value at least 1. Let j denote the agent whose bundle is valued at least 1 by some agent i. Then by EF1, we have, $v_i(A_i) \geq v_i(A_j \setminus \{g\})$ for some $g \in A_j$. As the valuation functions are additive, $v_i(A_i) \geq v_i(A_j) - v_i(\{g\}) \geq 1 - \epsilon$, as $v_i(A_j) \geq 1$ and $v_i(\{g\}) \leq \epsilon$ for all g. As $MMS_i \leq v(M)/n = 1$ when the valuation functions are additive, we have $v_i(A_i) \geq (1-\epsilon)MMS_i$.

---

3. (*Max Nash welfare*) Suppose there are m indivisible goods, and n agents with additive valuations.

   (a) (2 points) Is an MNW allocation always PO? Give a short explanation for your answer.

(b) (8 points) Let $A = (A_1, \ldots, A_n)$ be an MNW allocation. Suppose for every agent i you are given good $g_i \in A_i$ that gives her the maximum value, i.e., $g_i \in \text{argmax}_{g\in A_i} v_{ig}$. Prove that you can now find a (cn)-MNW allocation, where c is a constant, in polynomial time. [Hint: Max weight matchings]

(a) Yes, any MNW allocation is PO. Otherwise, an allocation Pareto dominating it would give every agent an equal or higher valued bundle, and at least one agent a higher valued bundle, and thus have higher Nash welfare value, contradicting the fact that the original allocation is an MNW allocation.

(b) Consider the following algorithm. Give every agent their known highest valued good from some MNW allocation, denoted by A^*. Add all the remaining items in a set S representing all unallocated items. Then order the agents arbitrarily and in a round robin manner, assign every agent their next highest valued good from the remaining unallocated items. Update S after every assignment by removing the assigned item from S. Let the resulting allocation obtained be denoted by A.

We now show that A is a (cn)-MNW allocation. Fix an agent i. Denote their bundle in A^* by $A_i^* = \{g_1, g_2, \ldots, g_l\}$. We know that from the assignment before the round robin phase, agent i has received g_1. Then in the first iteration of round robin, in the worst case their rank is last among all agents, and all goods $g_2, g_3 \ldots, g_n$ are chosen by the other $(n-1)$ agents in this iteration. Even in the worst case, i can still choose g_{n+1} in this round. Similarly, they can choose all goods ranked at most g_{kn+1} or higher for all $k \geq 1$, that is, every n^{th} best good from their MNW bundle. Thus, we have $v_i(A_i) \geq v_i(\cup_{k\geq 0} g_{kn+1})$. Simplifying,

$$\begin{aligned}
v_i(A_i) &\geq v_i(\cup_{k\geq 0}\ g_{kn+1}) \ldots \text{ The good in } k^{th} \text{ round has value at least } g_{kn+1} \\
&\geq \frac{1}{n}\sum_{k\geq 0} n * v_i(g_{kn+1}) \ldots \text{ union taken out as sum as valuation function is additive} \\
&\geq \frac{1}{n}\sum_{k\geq 0} v_i(\cup_{j\in[n]} g_{kn+j}) \ldots \text{ as } v_i(g_{kn+1}) \geq v_i(g_{kn+j}), j \in [n] \\
&= \frac{1}{n}\sum_{k\geq 0}\sum_{j\in[n]} v_i(g_{kn+j}) \ldots \text{ as valuation function is additive} \\
&= \frac{1}{n}\sum_{p\geq 1} v_i(g_p) \\
&= \frac{1}{n} * v_i(A_i^*).
\end{aligned} \tag{4}$$

Hence, every agent i receives a bundle in A of value at least $1/n$ times their value for their bundle in A^*. Hence the Nash welfare of A is at least $1/n$ times the Nash welfare of A^*.

---

4. (*Stable Matchings*)

(a) (2 points) Give an example of a stable matching instance with more than one stable matching.

(b) (2 points) Suppose we consider the variant of finding a stable matching of roommates as follows. There is one set of n people, each person has a preference list for the remaining people, with the highest ranked person in the list most preferred as a roommate. Give an example where a stable matching of roommates does not exist.

(c) (6 points) We discussed in the class that the Differed Acceptance algorithm where men propose gives the worst choice to women among all stable matchings. This indicates the Gale-Shapley algorithm is not truthful. Give an example where a woman can get a better choice by misreporting her preferences.

(a) Consider an instance with 2 men m_1, m_2 and 2 women w_1, w_2. Let the preferences of the men be $m_1 : w_1 \succ w_2$ and $m_2 : w_2 \succ w_1$, and those of the women be $w_1 : m_2 \succ m_1$ and $w_2 : m_1 \succ m_2$. Then the matchings $(m_1, w_1), (m_2, w_2)$ and $(m_1, w_2), (m_2, w_1)$ are both stable.

(b) Consider an instance with 4 people, denoted for ease of exposition by $1, 2, 3, 4$, that must be paired into two pairs of roommates. Let their preferences be,

$$\begin{aligned} 1 &: 2 \succ 3 \succ 4, \\ 2 &: 3 \succ 1 \succ 4, \\ 3 &: 1 \succ 2 \succ 4, \\ 4 &: 3 \succ 1 \succ 2. \end{aligned} \tag{5}$$

It can be verified by enumerating all possible matchings that none is stable.

(c) Consider an instance with 3 men a, b, c and 3 women p, q, r. Their preferences are,

$$\begin{aligned} a &: q \succ p \succ r, \\ b &: p \succ q \succ r, \\ c &: p \succ q \succ r, \\ p &: a \succ c \succ b, \\ q &: c \succ a \succ b, \\ r &: a \succ b \succ c. \end{aligned} \tag{6}$$

If we run the Gale-Shapley algorithm with the men proposing, we get the matching $(a, q), (b, r), (c, p)$, which matches p to her second choice. Now suppose p misreports her preferences as $a \succ b \succ c$, then we get the matching $(a, p), (b, r), (c, q)$, which matches p to her first choice.

---

5. (*Bonus*)

(a) Prove that when agents have binary additive valuation functions (that is, $v_{ij} \in \{0, 1\}$ for all i, j) then an EF1+PO allocation can be found in polynomial time.

(b) An α-EFX allocation is where for all pairs of agents i, j, and every good $g \in A_j$, we have $v_i(A_i) \geq \frac{1}{2} v_i(A_j \setminus \{g\})$. Prove that when the valuation functions of agents are subadditive, $\frac{1}{2}$-EFX $\Rightarrow$ α-MMS for $\alpha = 1/(cn)$, for some constant c where n is the number of agents.

(c) Prove that an MNW allocation is also EF1+PO.

(d) Which of the Arrow's axioms does the instant run-off voting system violate? Give an example to illustrate this. The instant runoff system is as follows: Repeatedly eliminate candidates that have the least number of first-place preferences among all voters, until only one candidate (arbitrarily choose among the last remaining if they all will be eliminated in the next round) remains.

(e) Prove that the stable matching algorithm matches every woman with her worst feasible choice.

(a) We can assume without loss of generality that for every good, there is some agent who values it at 1. Now suppose we run the following modification of the envy cycle elimination algorithm. For every good, say g, if no source agent values it at 1, then consider the source, say s, of the connected component of any agent, say i, who has value 1 for the good. There is a path in the component from the source to this agent, indicating that for each agent along this path, there is some good they value at 1 in their successor's bundle. For each agent from s to i on this path, remove the good valued at 1 from the successor's bundle and re-allocate to the predecessor. Then allocate the new good g to i. Effectively, we increase the value of the source agent by 1 by doing this.

To summarize, we run the envy cycle elimination algorithm with the modification that for each good g, we consider an agent who has value 1 for the good, and swap goods along a path from some source to this agent and allocate g to this agent.

By a similar proof that proves that the envy cycle elimination algorithm gives an EF1 allocation, we can prove that this modified algorithm also gives an EF1 allocation. Additionally, it also assigns every good to an agent who has value 1 for the good, hence has the highest possible social welfare (or sum of valuations of all agents) among all allocations. Hence it is also Pareto optimal, as otherwise, a Pareto dominating allocation would have a higher social welfare.

(b) Consider the set of agents $j \neq i$ that have received singleton bundles in the 1/2-EFX allocation. We have seen that if we remove these agents and their bundles, the MMS value of any agent for allocating the remaining items among the remaining agents is at least that of their original MMS value. Hence, it will suffice to show that every agent gets a bundle of value α-MMS in the reduced instance after removing all singleton bundles and their corresponding agents.

Fix an agent i. For every j, order the (at least two) goods in A_j in non-increasing order of marginal values according to i. Then for the last good, say g_l in this order, we have $v_i(A_j \setminus \{g_l\}) \geq v_i(\{g_l\})$. Then we can say,

$$\begin{aligned} v_i(A_i) &\geq \frac{1}{2} v_i(A_j \backslash \{g_l\}) \\ &\geq \frac{1}{2}(\frac{1}{2}(v_i(A_j \backslash \{g_l\}) + v_i(\{g_l\}))) \\ &\geq \frac{1}{2}(\frac{1}{2} v_i(A_j)) \ldots \text{(by subadditivity)} \\ &= \frac{1}{4} v_i(A_j). \end{aligned} \tag{7}$$

Adding these inequalities over all j, we get $n * v_i(A_i) \geq \frac{1}{4}\sum_j v_i(A_j) \geq v_i(\cup_j A_j) = v_i(M)$, where the last inequality follows from subadditivity.

Next, the MMS value of i cannot be higher than $v_i(M)$, as the valuation function of i is monotone. This implies, $v_i(A_i) \geq \frac{1}{4n} MMS_i$. As this is true for all i, this is a $(1/4n)$-MMS allocation.

(c) PO: Proof by contradiction. Suppose the MNW allocation was not PO. Then there is another Pareto dominating allocation, say P. As all agents receive in P at least the same valued bundles as in the MNW allocation, and at least one agent gets a higher valued bundle, the Nash welfare product of P will be higher than the MNW allocation, a contradiction.

EF1: Proof by contradiction. Suppose the MNW allocation was not EF1. Then there is a pair of agents, say 1 and 2, such that $v_1(A_1) < v_1(A_2 \backslash \{g\})$, for every $g \in A_2$. We will show that there is some good g in A_2 such that removing it from A_2 and giving it to A_1 increases the Nash welfare product, a contradiction.

We want,

$$\frac{(v_1(A_1) + v_1(g))(v_2(A_2) - v_2(g))}{v_1(A_1)v_2(A_2)} > 1 \Leftrightarrow \left(1 + \frac{v_1(g)}{v_1(A_1)}\right)\left(1 - \frac{v_2(g)}{v_2(A_2)}\right) > 1$$

$$\Leftrightarrow 1 + \frac{v_1(g)}{v_1(A_1)} - \frac{v_2(g)}{v_2(A_2)} - \frac{v_1(g)v_2(g)}{v_1(A_1)v_2(A_2)} > 1 \Leftrightarrow \frac{v_1(g)}{v_1(A_1)} > \frac{v_2(g)}{v_2(A_2)}\left(1 + \frac{v_1(g)}{v_1(A_1)}\right)$$

$$\Leftrightarrow \frac{v_1(g)}{v_2(g)} > \frac{v_1(A_1) + v_1(g)}{v_2(A_2)}.$$

If we show some good for which the last inequality is true, then so is the first in this series of implications, completing the proof. We proceed to prove this.

Consider the good from A_2 with the highest value of the ratio $\frac{v_1(g)}{v_2(g)}$, and for which $v_2(g) \neq 0$. As 1 envies 2, there is at least one good positively valued by 1 in A_2; if 2 did not value this good positively, we could re-allocate it to 1 and increase the Nash welfare product. Hence, such a good g is well-defined.

Simple algebra shows for any two fractions $\frac{a}{b} > \frac{c}{d}$, we have $\frac{a}{b} > \frac{a+c}{b+d}$.

Hence,

$$\frac{v_1(g)}{v_2(g)} \geq \frac{\sum_{j \in A_2 : v_2(j) > 0} v_1(j)}{\sum_{j \in A_2 : v_2(j) > 0} v_2(j)} = \frac{\sum_{j \in A_2 : v_2(j) > 0} v_1(j)}{v_2(A_2)} \geq \frac{\sum_{j \in A_2} v_1(j)}{v_2(A_2)} > \frac{v_1(A_1) + v_1(g)}{v_2(A_2)}.$$

Here the second-last inequality follows for the case when there is no good valued positively by 1 and zero by 2. This is without loss of generality, as otherwise we could re-allocate this good to 1 and increase the Nash welfare.

The last inequality follows as 1 envies 2 even upon removing any good, hence even g, from $2's$ allocation.

(d) Ignore.

(e) We first prove that the matching returned by the algorithm matches every man with their best feasible choice, and use this in proving the required statement.

We know that if a man m gets paired with a woman w, then any woman w' more preferred by m must have rejected m during the algorithm. It will suffice to prove that no feasible matching contains the pair (m,w'). We show this by induction. Initially, no woman as rejected any man. Later if some woman w' rejects some man m, then she is paired with a man m' she prefers more. By inductive hypothesis, m' cannot be paired with a woman higher preferred by him in any feasible matching. Hence, if there is a matching where m is paired with w', this matching pairs m' with a woman he prefers lower than w'. But both w' and m' prefer each other over their matches, hence such a matching cannot be stable. Hence, if at any point some w' rejects m, then no feasible matching contains (m,w'), proving that the algorithm matches every m to their best feasible choice.

We now prove that every woman is matched to her worst feasible choice. If there is only one feasible matching, then the statement is true by default. Else, among all stable matchings, consider the one, say M, where each woman is matched with her worst feasible choice. For every woman w matched to some m for whom there is another pair w, m' in another matching, and m' is preferred over m by w, the following is true. As (w, m) is a pair in a *stable* matching M, m' is matched in M with a woman, say w', preferred higher by him. We have seen that when the Gale-Shapley algorithm runs with the men proposing, each man gets matched to their highest preferred and feasible choice. The feasibility of M and (w, m') contradict this, hence the outcome of the algorithm will not contain any such pair w, m'. Hence the stable matching algorithm outputs M.
