

CS381 Assignment 4 Solutions

Due November 10 2023 **Noon** 12:00 PM AoE

1. Consider the following election implementations:

- (a) Plurality: Rank candidates by their number of first place votes
- (b) Borda Count: For each vote of the form $c_1 > c_2 > \dots > c_n$, c_n gets 1 point, c_{n-1} gets 2 points, ..., c_2 gets $n - 1$ points, and c_1 gets n points. When summed over all votes, the candidate with the most points wins.
- (c) The Two-Round System (used by Turkey, etc.): If any candidate gets a majority of first place votes, they immediately win. Otherwise, the two candidates who got the most first place votes go into an instant runoff: for the purpose of implementing this with just the original vote, we will just look at which candidate is preferred by the majority.

Additionally, recall the definition of Independence of Irrelevant Alternatives (IIA): If A is preferred over B by a voting rule for given voter, then if only preferences for X change, the voting rule must not lead to B's being preferred over A.

Show that each of these election schemes do not always satisfy Independence of Irrelevant Alternatives, even with respect to the winning candidate.

Solution:

- (a) Suppose the votes came out as in $<$:

10	12	5
A	B	A
B	C	C
C	A	B

In this voting specification, we have that A has the most first place votes, so $A > B > C$. Now, suppose that the 5 voters in the last column instead put C as their top choice, as in $<'$:

10	12	5
A	B	C
B	C	A
C	A	B

We are only moving C in the voting specification of the last column, so

by IIA we should still have that $A >' B$: however, now B gets the most first place votes, so $B >' A >' C$.

- (b) Suppose the votes came out as in $<$:

15	10	3
A	B	A
B	C	C
C	A	B

In this voting specification, we get:

A	$45 + 10 + 9 = 64$
B	$30 + 30 + 3 = 63$
C	$15 + 20 + 6 = 41$

Now, suppose that the 5 voters in the last column instead put C as their top choice, as in $<'$:

15	10	3
A	B	C
B	C	A
C	A	B

In this voting specification, we get:

A	$45 + 10 + 6 = 61$	As in a, by changing the ranking of C for certain voters, suddenly the victor changed from A to B, violating IIA.
B	$30 + 30 + 3 = 63$	
C	$15 + 20 + 9 = 44$	

(c) Suppose the votes came out as in $<$:

10	14	11
A	B	A
B	C	C
C	A	B

In this voting specification, we have that A and B have the most first place votes, and the majority of people prefer A over B so A wins.

Now, suppose that the 11 voters in the last column instead put C as their top choice, as in $<'$:

10	14	11
A	B	C
B	C	A
C	A	B

Now, B and C have the most first place votes, and a majority of voters prefer B over C, so B wins. That is, by moving C in players' rankings, now B is victor: this breaks IIA with respect to A and B.

- Arrow's Theorem (which we proved in class) shows that for any election system with at least 3 candidates that satisfies both unanimity and independence of irrelevant alternatives must be a dictatorship. Convince me that majority preference satisfies both unanimity and independence of irrelevant alternatives without being a dictatorship when the election is only over two candidates.

Solution: Notice that IIA is automatically satisfied when there's only two candidates. However, if everyone prefers A over B, then the majority prefer A or B, so the mechanism puts A over B: hence, unanimity is satisfied. Further, clearly the mechanism isn't a dictatorship: for every voter i , consider the ranking specification $<^i$ where every voter $j \neq i$ specifies $A <^i_j B$ while $B <^i_i A$. Then, assuming there are at least 3 voters, we have that the majority prefers B to A, making B the victor, contrary to i 's vote. Hence i is not a dictator for every voter i .

Do either 3 or 4, or both for some amount of extra credit

- In class I claimed that a result of Arrow's theorem (stated above) is that any 'reasonable' social choice function is not incentive compatible. In this problem, we'll prove it. Specifically, imagine a social choice function f : it takes as input a specification ($<$) of rankings by the

voters and outputs a single candidate $f(<) = a$. Let's assume that this function isn't *candidate prohibitive*: that is, for every candidate, there is some specification of rankings that results in them winning.

- i) Let's call a social choice function f *monotone* if it satisfies the following: if $<'$ is a separate specification of ranking where exactly one voter i has changed their rankings and $a' = f(<') \neq a$, then $a' <_i a$ and $a <'_i a'$. That is, a' was less preferable to a by voter i in $<$ but is now more preferable in $<'$.

Convince me that if f is incentive compatible, then it must be monotone.

Solution: Suppose f is not monotone. Then there is a separate specifications of rankings $<'$ where exactly one voter i has changed their rankings and $a' = f(<') \neq a$ with $a <_i a'$ or $a' <'_i a$. In the first case, in $<$ voter i is incentivized to vote according to $<'_i$ (as they would get a' elected, whom they prefer in $<_i$). In the second case, in $<'$ voter i would be incentivized to to vote according to $<_i$ (as they would get a elected, whom they prefer in $<'_i$).

Arrow's theorem (above) tells us something about social *welfare* functions (which outputs complete rankings about the candidates), but doesn't directly say anything about social *choice* functions like f . So if we want to say anything about f using Arrow's theorem, we'll need some corresponding social welfare function. Let's try to construct one. To start out, given some collection S of candidates, let's imagine $<^S$ as the ranking produced from $<$ by dragging all the candidates in S and moving them to the top of each voters' rankings *without changing the order between candidates in S* .

- ii) First, imagine we pick some candidates S and think about $f(<^S)$: convince me that the winner is a candidate in S . Hint: since we're assuming that f isn't candidate prohibitive, then there is some ranking specification $<'$ that results in some candidate in S winning. Voter by voter, switch each voters' ranking to the one in $<^S$ and applying monotonicity above.

Solution: Start with the ranking specification $<'$ that results in some candidate in $a \in S$ winning. Now, switch the first voter's vote to the one specified by $<^S$: suppose that now some $b \notin S$ is the victor. Then f wouldn't be monotone, as $b <^S_1 a$. Apply this to every remaining voter sequentially: by the same argument, the victor is a candidate in S every step in the way.

We're now ready to extend f to a social welfare function F with output $<_F$: for any two candidates A, B , we will say $A <_F B$ if $f(<^{AB}) = B$. That is, the resulting net preference between A and B by our social welfare function will be the winner of the election if the votes had been $<^{AB}$.

- iii) Let's start out by verifying this is a valid social welfare function. Basically, the only thing we really need to verify is that the output $<_F$ is *transitive*: that is, we need to make sure that $A <_F B$ and $B <_F C$ means $A <_F C$. More specifically, we need to show that it cannot happen that $A <_F B <_F C <_F A$. Convince me that it cannot happen.

Solution: Suppose it did happen and let $f(<^{ABC}) = A$ (reassign candidates' symbols as necessary). As in ii, sequentially change each voters' preference from $<^{ABC}$ to their vote in $<^{AB}$: by monotonicity, the victor will remain as A every step of the way. As such $f(<^{AB}) = A$, contradicting that $A < B$.

- iv) Next, convince me that F satisfies unanimity. Hint: Observe that if $A <_i B$ for every voter i , then $(<^{AB})^B = <^{AB}$ and then apply ii.

Solution: As in the hint, if $A <_i B$ for every voter i , then $(<^{AB})^B = <^{AB}$. By ii, we know that $f((<^{AB})^B) = B$, so $A < B$.

- v) Next, convince me that F satisfies independence of irrelevant alternatives. Hint: This is easy.

Solution: Let $<$ and $<'$ be two voting specifications where only ranking preferences for a particular candidate c have changed and suppose $A <_F B$. Then we know $f(<^{AB}) = B$. Sequentially, change each vote in $<^{AB}$ to $(<')^{AB}$: by monotonicity, the victor cannot change every step of the way. Hence $f((<')^{AB}) = B$, so $A <'_F B$. This verifies IIA.

- vi) Next, convince me that if f isn't a dictatorship then neither is F .

Solution: If f isn't a dictatorship, then for every voter i there is a voting specification $<'$ where $f(<') = B$ but $B <'_i A$. By monotonicity, we have that $f((<')^{AB}) = B$. This means that $A <'_F B$, showing that F is also not a dictatorship for each voter i .

- vii) Lastly, convince me that if f is an incentive compatible social choice function in an election with at least 3 candidates in which every candidate can win, then f is a dictatorship.

Solution: Putting together all the above parts, we have that we can extend f to a social welfare function F satisfying unanimity and IIA. By Arrow's Theorem, we know that F must be monotone. By vi, this means f must be a dictatorship.

4. One of the interesting questions asked in class was about probabilistic social choice functions. We know from lecture that it's hard to come up with "good" functions of these sort if they have to be deterministic, but perhaps by applying randomization we can achieve better techniques. The difficulty about measuring 'doing well', however, comes from the question of how voters' utilities depend on the different candidates. For example, if I tell you that I prefer candidate A over candidate B, it's not immediately clear to you *how much more* I prefer A over B. Maybe I think B is awful compared to A, maybe I think they're practically the same. One way some folks have used to measure preference of candidates is to refer to the idea of a political compass, where we imagine that candidates are related with some point in the 2D plane. Now we can assume that voters prefer candidates that are closer to their own locations in this 2D plane, and we can literally measure the difference between preference of candidates by how much closer they are to the voter. Of course, the voter isn't exactly revealing these exact distances by their ranked choices, but it does still imply something that we can use to make reasonable estimates. Check out this paper: it supplies a couple interesting techniques that do well when we assume that the underlying voters' preferences are based on a metric space - something that a 2D plane happens to fit! So here's the actual problem: create a program that simulates the following:

- Each candidate and voter is associated with some uniformly random point in the diameter 2 square centered at the origin ($\{\{x, y\} \mid |x| \leq 1, |y| \leq 1\}$)
- For each candidate, determine their ranked choice ballot over the candidates by ordering them according to proximity.
- If we assume that the "Optimal Candidate" is the one that minimizes the sum of distances to every voter, determine the optimal candidate and their objective cost (as perceived by the population).
- Using the collection of ranked choice ballots, determine the winner according to the Randomized dictatorship/plurality method described in the paper and their objective cost (as perceived by the population).

- Using the collection of ranked choice ballots, determine the winner according to the Proportional to squares method described in the paper and their objective cost (as perceived by the population).
 - For the previous two bullets, determine the average ratio of the objective cost of the solution vs the objective cost of the optimal solution.
5. Recall the stable marriage problem from class: in this problem, we imagine that have two sets (canonically described as men and women) who have complete ranking over each other. In class, we showed that the solution produced by the deferred acceptance algorithm was male-optimal: from every man's perspective, they end up with the woman highest on their preferences that they could ever be matched with an a stable assignment. That is, for every man, there is no woman they prefer that they could be matched to in a stable assignment.
- Convince me that the solution produced by the deferred acceptance algorithm is female-pessimal. That is, every woman ends up with the man lowest on their rankings out of all stable assignments. Hint: use the same argument from class.
- Solution:** Suppose there is a woman w who ends up with man m under the matching M produced by DA but for whom there is another stable matching M' in which w and m' are matched, where m' is preferred less than m by w . Let w' be m 's partner in M' . We know that M is male optimal, so if M' is a stable matching then it cannot be the case that m prefers w' over w . Thus m prefers w over w' . But we already know that w prefers m over m' . Hence (m, w) are a rogue couple in M' : that is, they each prefer each other over their assigned partner. Hence M' is unstable, a contradiction, so there is no stable matching in which any woman ends up with a man they prefer less than their partner under DA.
6. For this problem, imagine you own a fleet of freight trains and an equal number of stations. Each one of these trains T_i has a schedule S_i for this month (you can assume months are arbitrarily long in this universe), during which every train visits every station exactly once. You've been informed by your engineers that every one of these trains has a problematic flaw in design and will need to be held in one of your stations for the rest of the month. These problems aren't exactly immediate - they can be held until the end of the month. Unfortunately, your stations aren't very big - each station can only hold one train, and another train cannot pass through the station while another is held there. You've been asked to truncate (end early) each train's schedule at a unique station: whatever this station is, it will remain there for the remainder of the month. Your goal is to do this so that:
- (a) No other train passes through a station that holds another train
 - (b) The number of stops made by all trains is maximized - equivalently, the number of stops lost due to truncation is minimized.

As an example, imagine that we had two trains and two stations, where the schedule was:

Train	Day 1	Day 2	Day 3	Day 4
T_1	S_1	.	S_2	.
T_2	.	S_1	.	S_2

If T_1 were truncated at S_1 , then T_2 wouldn't be able to pass through the station, and thus no valid truncation would follow. Hence T_1 must be truncated at S_2 , meaning T_1 visits both S_1 and S_2 while T_2 must truncate at S_1 , losing its visit to S_2 .

Determine an algorithm that successfully achieves your goals. Hint: Apply deferred acceptance.
Solution: We're going to create an input for the DA algorithm, run the DA algorithm, and

use the assignment produced to choose our truncations. Specifically, we will equate the set of men as the stations and the set of women as the trains. For each man/station, we create rankings over the women/trains according to reverse chronologic order (so each station prefers most the train that arrives last, etc.). For each woman/train, we create rankings over the men/stations according to chonorologic order (so each train most prefers the station it visits first). After running DA and getting the assignment M it produces, we will truncate each train's schedule at the station it gets assigned to. Suppose in the schedule produced that a train t visits a station s after another train has already been truncated at that station. Then s prefers t over its assigned partner, and t prefers s over its assigned partner: this means that (s, t) are a 'rogue couple' in M , so M would be unstable. Note that the reverse also holds: that is, 'rogue couples'/unstability is equivalent with the goal (a) above being violated. Since we know that DA produces stable solutions, we know that the goal (a) is achieved by our schedule. Moreover, we know that DA produces male-optimal solutions: every station ends up holding the train which visits last over every stable assignment, ie over every assignment that satisfies goal (a). Hence there is no way to come up with an assignment which satisfies goal (a) where any station gets assigned a train which visits later. As such, the maximal number of stops are made by this truncated schedule.

7. Recall the details of VCG mechanisms and the Clarke-Pivot rule as discussed in class.

- (a) Imagine a sealed-bid auction where we are auctioning off k identical items and each bidder is only interested in one item. What is the resulting VCG mechanism using Clarke-Pivot rules for this auction?

Solution: The mechanism decides which k bidders get the item. As we know that VCG mechanisms are incentive compatible, I will somewhat sloppily refer to bids and valuations interchangeably. Since the VCG mechanism maximizes social welfare, we know that it will award the items to the k highest bidders. Letting bidder i bid B_i , let's rearrange the bids so that $B_1 > B_2 > \dots B_n$. For each bidder $1 \geq i \leq k$ we have

$$\begin{aligned} p_i &= \max_{\text{choices } C \text{ of who gets the items}} \sum_{j \neq i} v_j(C) - \sum_{j \neq i} v_j(\text{the top } k \text{ bidders get the items}) \\ &= \left(\sum_{j=1}^{k+1} B_j - B_i \right) - \left(\sum_{j=1}^k B_j - B_i \right) \\ &= B_{k+1}, \end{aligned}$$

while for each $i > k$ we get

$$\begin{aligned} p_i &= \max_{\text{choices } C \text{ of who gets the items}} \sum_{j \neq i} v_j(C) - \sum_{j \neq i} v_j(\text{the top } k \text{ bidders get the items}) \\ &= \sum_{j=1}^k B_j - \sum_{j=1}^k B_j \\ &= 0. \end{aligned}$$

Hence the highest k bidders pay B_{k+1} while everyone else pays 0, which aligns with the second price auction for one item already determined in class. Worth noting is also that if $n \leq k$, then the VCG mechanism with Clarke-Pivot payments has everyone pay 0.

- (b) Imagine I am taking bids from contractors for one of the myriad of jobs that I need to get done at my house (anyone reading this know how to service a heating system?). Each contractor values the job negatively: conceptually, we imagine each the contractor summing up expenses of materials and their actual time, so their “bid” is something like what their expectation of costs.

- i. If I applied a VCG mechanism with Clarke-Pivot payments to this problem, what is the result? What is somewhat “unreasonable” about this solution?

Solution: Clearly the ‘least negative bid’ should be awarded the job. Let’s assume that $B_1 > B_2 > \dots > B_n$ (note that these are negative numbers, so $|B_1| < |B_2| < \dots$). Note that h_1 is the option chosen that maximizes the sum of everyone other than the first bidder’s valuation. In this situation, when all these bids are negative, the maximal such option is still awarding the contract to bidder 1: in this case, the sum of valuations of other bidders is 0, while in any other case it would be negative. As such,

$$p_1 = 0 - \sum_{j>1} v_j(\text{bidder 1 getting the contract}) = 0 - 0 = 0.$$

On the other hand, for any bidder $i > 1$ we get

$$p_i = 0 - B_1 = |B_1|,$$

ie every other bidder is paying us how much the winner of the contract loses, ie everyone effectively loses B_1 value. This obviously wouldn’t work and doesn’t satisfy individual rationality (see class notes).

- ii. Determine a reasonable VCG mechanism for this problem that only involves paying one contractor (and all other contractors pay/receive nothing).

Solution: We’ll set $h_i = \max_{j \neq i} B_j$. Now, $p_1 = B_2 - 0 < 0$, and for $i > 2$ we get $p_2 = B_1 - B_1 = 0$. Hence we are paying the highest bid the cost estimate of the second highest. Note that this mechanism is individually rational, but obviously involves a payment made by the mechanism.

- (c) Imagine the town of Waterville is proposing a public project, maybe filling in some pot-holes. The project is known to cost some C dollars and every person i in town i has some private value B_i of the project. Let’s assume that $B_i \geq 0$ for all i (this project won’t hurt anyone) and that $\sum_i B_i \geq C$, as otherwise it wouldn’t make sense to do the project. To fund the project, the town plans to use a VCG mechanism with Clarke-Pivot rules: determine the townspeople who end up paying for this project and how much they pay. Further, show that the amount paid by townspeople is less than C unless it just so happens that $\sum_i B_i = C$: this means the town government must necessarily subsidize the project.

Solution: To model this, we’ll assume the government is a player that values the project at $-C$. As such, with the n townspeople this gives us $n + 1$ total players (we’ll let the government be the $n + 1$ st player). With the creation of this player, we now have that the mechanism will only have the options “do the project” or “don’t do the project”. For each townspeople i , we have $v_i(\text{do the project}) = B_i$ and $v_i(\text{don’t do it}) = 0$. For the government, we have $v_{n+1}(\text{do the project}) = -C$, while $v_{n+1}(\text{don’t do the project}) = 0$.

Hence, the mechanism will do the project if

$$\begin{aligned} \sum_{i=1}^{n+1} v_i(\text{do the project}) &\geq \sum_{i=1}^{n+1} v_i(\text{don't do the project}) \\ &\iff \\ \sum_{i=1}^n B_i &\geq C. \end{aligned}$$

Now, since we're assuming $\sum_i B_i > C$, we can calculate that

$$p_{n+1} = \sum_{j=1}^n B_j - \sum_j j = 1^n B_j = 0.$$

For every townspeople $i \leq n$, let $X_i = \sum_{j=1}^n B_j - B_i$ (ie, everyone else's valuation of the project), giving us

$$\begin{aligned} p_i &= \begin{cases} X_i - C & \text{if } X_i \geq C \\ 0 & \text{otherwise} \end{cases} - (X_i - C) \\ &= \begin{cases} 0 & \text{if } X_i \geq C \\ C - X_i & \text{otherwise.} \end{cases} \end{aligned}$$

Note that this means that the only townspeople who pay are those whose values of the project are so high that without them the project would not be undertaken. Moreover, if $\sum_i B_i = C$, then every townspeople who has $B_i > 0$ is necessary for the project and every person pays their valuation, which is precisely the project cost. On the other hand, if $\sum_i B_i > C$, then we get:

$$\begin{aligned} \sum_i p_i &= \sum_{i:p_i>0} p_i \\ &= \sum_{i:X_i<C} (C - X_i) \\ &= \sum_{i:X_i<C} (C - \sum_{j \neq i} B_j) \\ &\leq \sum_{i=1}^n (C - \sum_{j \neq i} B_j) \\ &= nC - (n-1) \sum_i B_i \\ &= C - (n-1) \left(\sum_i B_i - C \right). \end{aligned}$$

Note that $\sum_i B_i > C$, so we are taking something positive away from C , giving us $\sum_i p_i < C$, meaning the payments collected from everyone are insufficient to pay for the project.