1 Universal Preference

Note 4

Suppose that preferences in a stable matching instance are universal: all n jobs share the preferences $C_1 > C_2 > \cdots > C_n$ and all candidates share the preferences $J_1 > J_2 > \cdots > J_n$.

(a) What pairing do we get from running the algorithm with jobs proposing? Prove that this happens for all n.

(b) What pairing do we get from running the algorithm with candidates proposing? Justify your answer.

(c) What does this tell us about the number of stable pairings? Justify your answer.

Solution:

(a) The pairing results in (C_i, J_i) for each $i \in \{1, 2, ..., n\}$. This result can be proved by induction:

Our base case is when $n = 1$, so the only pairing is (C_1, J_1), and thus the base case is trivially true.

Now assume this is true for some $n \in \mathbb{N}$. On the first day with $n+1$ jobs and $n+1$ candidates, all $n+1$ jobs will propose to C_1. C_1 prefers J_1 the most, and the rest of the jobs will be rejected. This leaves a set of n unpaired jobs and n unpaired candidates who all have the same preferences (after the pairing of (C_1, J_1)). By the process of induction, this means that every i^{th} preferred candidate will be paired with the i^{th} preferred job.

(b) The pairings will again result in (J_i, C_i) for each $i \in \{1, 2, ..., n\}$. This can be proved by induction in the same as above, but replacing "job" with "candidate" and vice-versa.

(c) We know that job-proposing produces a candidate-pessimal stable pairing. We also know that candidate-proposing produces a candidate-optimal stable pairing. We found that candidate-optimal and candidate-pessimal pairings are the same. This means that there is only one stable pairing, since both the best and worst pairings (for candidates) are the same pairings.

2 Pairing Up

Note 4

Prove that for every even $n \geq 2$, there exists an instance of the stable matching problem with n jobs and n candidates such that the instance has at least $2^{n/2}$ distinct stable matchings.

(*Hint*: It can help to start with some small examples; find an instance for $n = 2$, and think about how you can use these preference lists to construct an instance for $n = 4$. After this, you should be in a good position to generalize the construction for all even n.)

Solution:

To prove that there exists such a stable matching instance for any even $n \geq 2$, it suffices to construct such an instance. But first, we look at the $n = 2$ case to generate some intuition. We can recognize that for the following preferences:

J_1	$C_1 > C_2$
J_2	$C_2 > C_1$

C_1	$J_2 > J_1$
C_2	$J_1 > J_2$

both $S = \{(J_1, C_1), (J_2, C_2)\}$ and $T = \{(C_1, J_2), (C_2, J_1)\}$ are stable pairings.

The $n/2$ in the exponent motivates us to consider pairing the n jobs into $n/2$ groups of 2 and likewise for the candidates. We pair up job $2k-1$ and $2k$ into a pair and candidate $2k-1$ and $2k$ into a pair, for $1 \leq k \leq n/2$.

From here, we recognize that for each pair (J_{2k-1}, J_{2k}) and (C_{2k-1}, C_{2k}), mirroring the preferences above would yield 2 stable matchings from the perspective of just these pairs. If we can extend this perspective to all $n/2$ pairs, this would be a total of $2^{n/2}$ stable matchings.

Our construction thus results in preference lists like follows:

J_1	$C_1 > C_2 > \dots$
J_2	$C_2 > C_1 > \dots$
$\vdots$	$\vdots$
J_{2k-1}	$C_{2k-1} > C_{2k} > \dots$
J_{2k}	$C_{2k} > C_{2k-1} > \dots$
$\vdots$	$\vdots$
J_{n-1}	$C_{n-1} > C_n > \dots$
J_n	$C_n > C_{n-1} > \dots$

C_1	$J_2 > J_1 > \dots$
C_2	$J_1 > J_2 > \dots$
$\vdots$	$\vdots$
C_{2k-1}	$J_{2k} > J_{2k-1} > \dots$
C_{2k}	$J_{2k-1} > J_{2k} > \dots$
$\vdots$	$\vdots$
C_{n-1}	$J_n > J_{n-1} > \dots$
C_n	$J_{n-1} > J_n > \dots$

Each match will have jobs in the kth pair paired to candidates in the kth pair for $1 \leq k \leq n/2$.

A job j in pair k will never form a rogue couple with any candidate c in pair $m \neq k$ since it always prefers the candidates in this pair over all candidates across other pairs. Since each job in pair k can be stably matched to either candidate in pair k, and there are $n/2$ total pairs, the number of stable matchings is $2^{n/2}$.

3 Upper Bound

Note 4

In the notes, we show that the stable matching algorithm terminates in at most n^2 days. Prove the following stronger result: the stable matching algorithm will always terminate in at most $(n-1)^2 + 1 = n^2 - 2n + 2$ days.

Solution: Recall that there is always a candidate who receives only one proposal (on the last day). Other than that candidate, every other candidate can reject up to $n-1$ jobs. Thus, there's a total of $(n-1)^2 = n^2 - 2n + 1$ rejections. Conceptually, in the worst case scenario, there would be exactly one rejection per day; if we were to hand out none, then the algorithm would terminate. On the final day, the candidate who is proposed to only once receives their offer. Thus, the process takes at most $(n-1)^2+1 = n^2-2n+2$ days.

4 Short Tree Proofs

Note 5

Let $G = (V,E)$ be an undirected graph with $|V| \geq 1$.

(a) Prove that every connected component in an acyclic graph is a tree.

(b) Suppose G has k connected components. Prove that if G is acyclic, then $|E| = |V| - k$.

(c) Prove that a graph with $|V|$ edges contains a cycle.

Solution:

(a) Every connected component is connected, and acyclic because the graph is acyclic; by definition, this is a tree.

(b) Because each connected component is a tree, each connected component has $|V_i|-1$ edges. The total number of edges is thus $\sum_i(|V_i|-1) = |V|-k$.

(c) An acyclic graph has $|V|-k$ edges which cannot equal $|V|$, thus if a graph has $|V|$ edges it has a cycle.

5 Proofs in Graphs

Note 5

(a) Suppose California has n cities ($n \geq 2$) such that for every pair of cities X and Y, either X has a road to Y or Y has a road to X. Further, suppose that all roads are one-way streets.

Prove that regardless of the configuration of roads, there always exists a city which is reachable from every other city by traveling through at most 2 roads.

[*Hint:* Induction]

(b) Consider a connected graph G with n vertices which has exactly $2m$ vertices of odd degree, where $m > 0$. Prove that there are m walks that *together* cover all the edges of G (i.e., each edge of G occurs in exactly one of the m walks, and each of the walks should not contain any particular edge more than once).

[*Hint:* In lecture, we have shown that a connected undirected graph has an Eulerian tour if and only if every vertex has even degree. This fact may be useful in the proof.]

(c) Prove that any graph G is bipartite if and only if it has no tours of odd length.

[*Hint:* In one of the directions, consider the lengths of paths starting from a given vertex.]

Solution:

(a) We prove this by induction on the number of cities n.

Base case: For $n = 2$, there's always a road from one city to the other.

Inductive Hypothesis: When there are k cities, there exists a city c that is reachable from every other city by traveling through at most 2 roads.

Inductive Step: Consider the case where there are $k+1$ cities. Remove one of the cities d and all of the roads to and from d. Now there are k cities, and by our inductive hypothesis, there exists some city c which is reachable from every other city by traveling through at most 2 roads. Let A be the set of cities with a road to c, and B be the set of cities two roads away from c. The inductive hypothesis states that the set S of the k cities consists of $S = \{c\} \cup A \cup B$.

Now add back d and all roads to and from d.

Between d and every city in S, there must be a road from one to the other. If there is at least one road from d to $\{c\} \cup A$, c would still be reachable from d with at most 2 road traversals. Otherwise, if all roads from $\{c\} \cup A$ point to d, d will be reachable from every city in B with at most 2 road traversals, because every city in B can take one road to go to a city in A, then take one more road to go to d. In either case there exists a city in the new set of $k+1$ cities that is reachable from every other city by traveling at most 2 roads.

Alternate Solution :Alternatively, we can prove this using properties of directed graphs. Let c be the city with the largest in-degree. Note that this graph is essentially a complete graph, where each edge is a directed edge instead of an undirected edge. Therefore, the total in degree sums to $n(n-1)/2$, and so does the total out degree. In addition, the in degree + out degree of any vertex must add up to $n-1$.

Because the total in-degree of all vertices is $n(n-1)/2$, The largest in-degree is $d \geq (n-1)/2$. Let S be the these d cities that can reach c by one edge.

For any other city x, it has to have at least $(n-1)-d$ out-degree (because in-degree $<= d$). Notice that there are n total vertices, two of which are x or c, and d vertices that connect to c through one edge. Thus, there are $n-2-d$ other vertices. Since x has out degree at least $n-1-d > n-2-d$, it must therefore connect to at least one vertex in S by the pigeonhole principle.

Thus, all vertices are either connected to c through 1 or 2 edges.

(b) We split the $2m$ odd-degree vertices into m pairs, and join each pair with an edge, adding m more edges in total. (Here, we allow for the possibility of multi-edges, that is, pairs of vertices with more than one edge between them.) Notice that now all vertices in this graph are of even degree. Now by Euler's theorem the resulting graph has an Eulerian tour. Removing the m added edges breaks the tour into m walks covering all the edges in the original graph, with each edge belonging to exactly one walk.

(c) To prove the claim, we need to prove two directions: if G is bipartite, it contains no tours of odd length, and if G contains no tours of odd length, it must be bipartite.

Suppose G is bipartite, and let L and R be the two disjoint sets of vertices such that there does not exist any edge between two vertices in L or two vertices in R. Further, suppose there is some tour in G, and we start traversing this tour at $v_0 \in L$.

Since each edge in G connects a vertex in L to a vertex in R, the first edge in the tour connects the start vertex v_0 to a vertex $v_1 \in R$. Similarly, the second edge connects $v_1 \in R$ to $v_2 \in L$. In general, it must be the case that the $2k$th edge connects vertex $v_{2k-1} \in R$ to $v_{2k} \in L$, and the $2k+1$th edge connects vertex $v_{2k} \in L$ to $v_{2k+1} \in R$.

Since only even numbered edges connect to vertices in L, and we started our tour in L, the tour must end with an even number of edges.

For the opposite direction, suppose G contains no tours of odd length. Without loss of generality, let us consider one connected component of G; the following reasoning can be applied to all of the connected components of G.

Let v be an arbitrary vertex in G; we can divide all of the vertices in G into two disjoint sets:

$$R = \{u \mid \text{the shortest path from } u \text{ to } v \text{ is even}\}$$
$$L = \{u \mid \text{the shortest path from } u \text{ to } v \text{ is odd}\}$$

We claim that no two vertices in L are adjacent. For contradiction, suppose there do exist adjacent vertices $u_1, u_2 \in L$. Consider the tour consisting of:

- the shortest path from v to u_1 (odd length)
- the edge (u_1, u_2) (length 1)
- the shortest path from u_2 to v (odd length)

This tour has odd length, and contradicts our assumption that G has no tours of odd length. This means that no two vertices in L are adjacent.

Similarly, we claim that no two vertices in R are adjacent. For contradiction, suppose there do exist adjacent vertices $u_1, u_2 \in R$. Consider the tour consisting of:

- the shortest path from v to u_1 (even length)
- the edge (u_1, u_2) (length 1)
- the shortest path from u_2 to v (even length)

This tour has odd length, and contradicts our assumption that G has no tours of odd length. This means that no two vertices in R are adjacent.

We've just shown that there are no edges between two vertices in L, and no edges between two vertices in R. If there are multiple connected components in G, the same partition can be applied to all of the components. Together, this means that G is bipartite.