CS 598CCI: Topics in Graph Algorithms — **Lecture date: 7 February, 2020**
Instructor: Chandra Chekuri — **Scribe: Zhoutong Jiang**

1 Introduction

The Gale-Shapley algorithm [1] for the stable marriage problem has been highly influential and even led to a Nobel Prize in Economics. It addresses an important problem that initially arose in matching residents to hospitals. We will discuss the following topics in this lecture.

- Stable marriage (SM) problem
- Simple extensions of SM
- Hospital and residence problem

2 Stable Marriage Problem

We will discuss, first, the most basic setting of the SM problem. We are given n men and n women. Each man m has a total preference order over all the women and similarly, each woman w has a total preference order over the men. We will write $p <_a q$ if according to person $a's$ preference, p is less desirable than q. We will assume for now that there are no ties, i.e., preference orders are strict. In the SM setting, we consider a complete bipartite graph between the men and women.

Before discussing the definition of stable, we define the following,

Definition 1 *(Blocking pair). Given a matching M, a pair (m, w) is a blocking pair for M if $(m, w) \notin M$ and m prefer w to its current partner in M (or m is unmatched in M) and w prefers m over its current partner in M (or w is unmatched in M).*

Consider the following example,

Table 1: Example ($n = 4$)

(a) Men's preference

1	2	4	1	3
2	3	1	4	2
3	2	3	1	4
4	4	1	3	2

(b) Women's preference

1	2	1	4	3
2	4	3	1	2
3	1	4	3	2
4	2	1	4	3

We can observe that $\{(1, 4), (2, 3), (3, 2), (4, 1)\}$ is stable while $\{(1, 1), (2, 3), (3, 2), (4, 4)\}$ is not stable as $(1, 4)$ is a pair of blocking pair.

Now we define stable matching as follows,

Definition 2 *(Stable Matching) A matching M is stable if there is no blocking pair for M.*

It is not clear that even in the SM setting when the graph is bipartite and complete, and the preference lists are strict and complete that there always exists a stable matching. Although it is easy to see that given a matching M, one can check whether it is stable or not. Gale-Shapley showed that every SM instance has a stable matching via an efficient algorithm described below.

Algorithm 1 GS algorithm: men propose, women dispose.

```
all people are free
while ∃ man m that is free do
  w=first woman on m's list that he has not proposed to yet.
  if w is free then
    pair up (m, w)
  else if m is better for w than its current partner m' then
    (m, w) get engaged and m' is free
  else
    w reject m and m is free.
  end if
end while
output engaged pairs
```

Note that the Algorithm 1 as stated is non-deterministic in that one can choose free men in any order. In fact they can all act in parallel in each round. It is not obvious that the algorithm will terminate and that it will output a stable matching. It is not too hard to make it terminate because a man does not propose to a woman more than once (he moves down his list).

Exercise 1 *Rewrite algorithm such that it terminate and argue that it can be implemented in $O(n^2)$ time.*

We will prove this algorithm actually pairs everyone and returns a stable matching.

Claim 3 *At the end of Algorithm 1, all people are paired up*

Proof: Firstly, we have the following **observation**: once a woman gets matched, she will remain matched and her partner ranking in her order can only monotonically improve. To see why every man is paired, suppose all women on m's list reject him. If a woman w rejects m, then w is paired with someone else and by previous observation they will remain paired. But there are n men and n women, which implies that not all women can reject m since they would have to be paired with $n-1$ men. This shows that all men are paired and hence all women are paired as well. □

Lemma 4 *Matching M output by Algorithm 1 is stable.*

Proof: Suppose (m, w) is a blocking pair for M. Since m proposes in decreasing order of preferences in his list, m proposed to w before he proposed to $P_M(m)$ ($P_M(m)$ is the partner of m in the matching M). But w rejected m, which means w had a better partner than m at the time of rejection. As the partner ranking of w can only improve, w has a better partner than m at the end of algorithm; hence $m \prec_w p_M(w)$. Thus (m, w) cannot be a blocking pair. □

Even though the Algorithm 1 seems to suggest that women dispose, the matching in fact prefers men and will result in a men-optimal matching. This is discussed in the Theorem (5) below.

Theorem 5 *In Algorithm 1, each man is matched to the best-ranked woman that he could be matched in any stable matching, and each woman is matched to the least ranked man that she can be matched to in any stable matching.*

Proof: For each m, define $best(m)$ to be the highest ranked woman that m can be matched to in some stable matching. We claim that in any run of Algorithm (1), $best(m)$ will be the partner of m.

We will prove by contradiction. Suppose not. Consider a run of Algorithm (1) and let (m, w) be the first pair that is rejected, in which $w = best(m)$. This means w rejects m because there exists some m' such that $m \prec_w m'$ and w is already matched to m' when m proposed to w. Let S be a stable matching with $(m, w) \in S$; by definition of $best(m)$ such a stable matching exists. Let w' be partner of m' in S. We see that m' prefers w' to w, otherwise (m', w) is a blocking pair for S (why?). But when m was rejected by w in (1) m' was matched to w'; this means that m' had already considered w' before w and must have been rejected. Thus m' was rejected by w' before w rejected m. Note that $best(m')$ is at least as good as w' and hence if w' rejected m' before (m, w) then $best(m')$ (which could be w') must have rejected m' before m proposed to w. This implies $(m, w = best(m))$ is not the first such pair to be rejected in the execution of Algorithm (1), which leads to a contradiction. Hence, $(m, best(m))$ is output of any run of Algorithm (1).

Now for second part, suppose M_0 is a man-optimal stable matching. Let w be a woman and suppose $(m, w) \in M_0$. Suppose there is another stable matching M such that $(m', w) \in M$ and w prefers m over m'. Note that $w = best(m)$ and hence if $(m, w') \in M$ we know that m prefers w to w'. But then (m, w) is a blocking pair in M. □

3 Extensions of SM

3.1 Unequal size

If $|X| \neq |Y|$ but preference lists are complete and strict. We then define (m, w) as a blocking pair for M if (m, w) prefer each other to current partner assuming being married is always preferable to not being married. This can be reduced to $|X| = |Y|$ by adding dummy people to the smaller set. We also have the following theorem.

Theorem 6 *Stable matching always exists. All stable matchings match everyone from the small set and the large set. All nodes can be partitioned into two sets, one of which is matched in all stable matchings and the other unmatched in all stable matchings.*

Exercise 2 *Prove the preceding theorem by reducing to SM.*

3.2 Incomplete lists or unacceptable partners

Consider if the bipartite graph $(A \cup B, E)$ is not complete. Instead, we have the following

- $(u, v) \in E$ means u is willing to match with v and vice versa.
- $(u, v) \notin E$ means u is not willing to match with v or v is not willing to match with u

Also, we consider that each node has a strict preference list over its neighbors. Now some nodes need not be matched in a stable matching. We have the following theorem for this problem.

Theorem 7 *Stable matching exists and nodes are partitioned into two sets, one set is matched in all stable matchings and the other is unmatched in every stable matching.*

Exercise 3 *Prove the preceding theorem by reducing to SM.*

3.3 Ties and indifference

When preference lists have ties or indifference, then matching gets complicated because of different notions of stability. See reference [2, 3] for more details.

4 Hospital Residents Problem (HR)

HR is the original application that motivated the development of stable matchings, and can be considered as the capacitated generalization of SM. We define R as the set of residents and H as the set of hospitals. Each $h \in H$ has a capacity u_h, each resident r has a preference order over hospitals and each hospital has a preference order over residents. Now we are interested in assignment of residents to hospitals that satisfy the capacity constraints at the hospitals.

We can see that it is easy to generalize the notion of stability from SM to HR setting (we leave it as as an exercise. We can reduce HR to SM by duplicating h into u_h nodes and extending the preference over hospitals to these duplicated nodes. We leave this also as an exercise. A nice result with an interesting title is the following.

Theorem 8 *Rural Hospital Theorem [4]: In any stable assignment the number of filled positions in each hospital, are the same in all stable matchings. In particular, if a hospital is unfilled in a stable matching then it remains unfilled in every stable matching.*

References

[1] Gale, David and Shapley, Lloyd S *College admissions and the stability of marriage.* The American Mathematical Monthly, 1962.

[2] Gusfield, Dan and Irving, Robert W. *The stable marriage problem: structure and algorithms.* MIT press, 1989.

[3] David, Manlove. *Algorithmics of matching under preferences.* World Scientific, 2013.

[4] Roth, Alvin E. *On the allocation of residents to rural hospitals: a general property of two-sided matching markets.* Econometrica: Journal of the Econometric Society, 1986.