Stable Marriage Problem

Debarka Sengupta

Slides are developed with help from

- Shreepriya Dogra
- Nupur Ahluwalia

Background

1.1 Hospitals/Residents Problem

- Each college can admit multiple students, One student can study in only one college (One to Many)
- Cropped up in the mid-40's
- Medical students, when they received offers from hospitals, they used to wait in case a better offer would present itself from a hospital more to their liking.
- The situation would end up in:
  - unhappy students who accepted their first offers
  - unhappy hospitals when students did not keep their earlier commitments.

1.2 Stable Allocation Problem

- Each server can be matched to multiple clients, Each client’s request may be processed at multiple server (Many to Many)
- Key criteria is performance enhancement which may be based on assignment of servers at a closer geographical location.
- The problem is similar to TA assignment to courses in universities.

1.3 Kidney Allocation Problem

- Many donors may be a match for a kidney transplant recipient (Many to One)
- The donors may be deceased or living
- Many willing donors for a specific recipient may not be a match
- There are many criteria in an allocation:
  - matching (compatible immune system, blood type & tissue type)
  - time of transplant (in case of deceased donor)
- Main goal/challenge: thicken the kidney exchange market to enable as many matches as possible

Stable Marriage Problem

2.1 Problem Statement

- The Stable Marriage Problem (SMP) states that:
  - given *n* men and *n* women (where each person has ranked all members of the opposite sex in order of preference)
  - marries the men and women together such that there are no two people of opposite sex who would both rather have each other than their current partners.

- If there are no such people, all the marriages are “stable”

2.2 Analysis of Stable Marriage Problem - 1

- Problem of finding a stable matching between two equally sized sets of elements given an ordering of preferences for each element.
- A matching is a mapping from the elements of one set to the elements of the other set.
- A matching is not stable if:
    - There is an element A of the first matched set which prefers some given element B of the second matched set over the element to which A is already matched, and
    - B also prefers A over the element to which B is already matched.

2.2 Analysis of Stable Marriage Problem - 2

- One man will matched to one woman (One to One)
- Complete Bipartite Graph with *2n* vertices:
  - *n* vertices representing men
  - *n* vertices representing women
- In 1962, David Gale and Lloyd Shapley proved that, for any equal number of men and women, it is always possible to solve the SMP and make all marriages stable.

2.3 Gale Shapley Algorithm (GS)

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m*'
     - *w* and *m* are engaged, *m*' is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Example matching 1

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Is this a stable matching?

Example matching 1

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

No!

Albert and Emily form a **blocking pair**

1. Everyone is unmatched

2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*

3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man m is unmatched:
   - w := m's most-preferred woman to whom he has not proposed yet
   - If w is also unmatched:
     - w and m are engaged
   - Else if w prefers m to her current match m'
     - w and m are engaged, m' is unmatched
   - Else: w rejects m
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man m is unmatched:
   - w := m's most-preferred woman to whom he has not proposed yet
   - If w is also unmatched:
     - w and m are engaged
   - Else if w prefers m to her current match m'
     - w and m are engaged, m' is unmatched
   - Else: w rejects m
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m*'
     - *w* and *m* are engaged, *m*' is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m*'
     - *w* and *m* are engaged, *m*' is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

1. Everyone is unmatched
2. While some man *m* is unmatched:
   - *w* := *m*'s most-preferred woman to whom he has not proposed yet
   - If *w* is also unmatched:
     - *w* and *m* are engaged
   - Else if *w* prefers *m* to her current match *m'*
     - *w* and *m* are engaged, *m'* is unmatched
   - Else: *w* rejects *m*
3. Return matched pairs

Albert	Diane	Emily	Fergie
Bradley	Emily	Diane	Fergie
Charles	Diane	Emily	Fergie

Men's Preference

Diane	Bradley	Albert	Charles
Emily	Albert	Bradley	Charles
Fergie	Albert	Bradley	Charles

Women's Preference

Analysis

Claim 1 - GS terminates in polynomial time (at most n^2 iterations of the outer loop)

Proof:

- Each iteration, one man proposes to someone to whom he has never proposed before
- n men, n women -> $n \times n$ possible events

(Can tighten a bit to $n(n - 1) + 1$ iterations.)

Claim 2 - GS results in a perfect matching

Proof by contradiction:

- Suppose by way of contradiction that *m* is unmatched at termination
- *n* men, *n* women -> *w* is unmatched, too
- Once a woman is matched, she is never unmatched; she only swaps partners. Thus, nobody proposed to *w*
- *m* proposed to everyone (by def. of GS)

Claim 3 - GS results in a stable matching, i.e., there are no blocking pairs

Proof by contradiction (1):

- Assume *m* and *w* form a blocking pair

Case #1: *m* never proposed to *w*

  - GS: men propose in order of preferences
  - *m* prefers current partner *w'* > *w*
  - -> *m* and *w* are not blocking

Claim 3 contd. - GS results in a stable matching, i.e., there are no blocking pairs

Proof by contradiction (2):

Case #2: *m* proposed to *w*

- *w* rejected *m* at some point
- GS: women only reject for better partners
- *w* prefers current partner *m'* > *m*
- -> *m* and *w* are not blocking

Case #1 and #2 exhaust space.

Man Optimality/Pessimality

- Let *S* be the set of stable matchings
- *m* is a **valid partner** of *w* if there exists some stable matching *S* in *S* where they are paired
- A matching is **man optimal** if each man receives his *best* valid partner
  - Is this a perfect matching? Stable?
- A matching is **man pessimal** if each man receives his *worst* valid partner

Claim 4 - GS with the man proposing – results in a man-optimal matching

Proof by contradiction (1):

- Men propose in order -> at least one man was rejected by a valid partner
- Let *m* and *w* be the first such reject in *S*
- This happens because *w* chose some *m'* > *m*
- Let *S'* be a stable matching with *m, w* paired (*S'* exists by def. of valid)

Claim 4 contd. - GS with the man proposing – results in a man- optimal matching

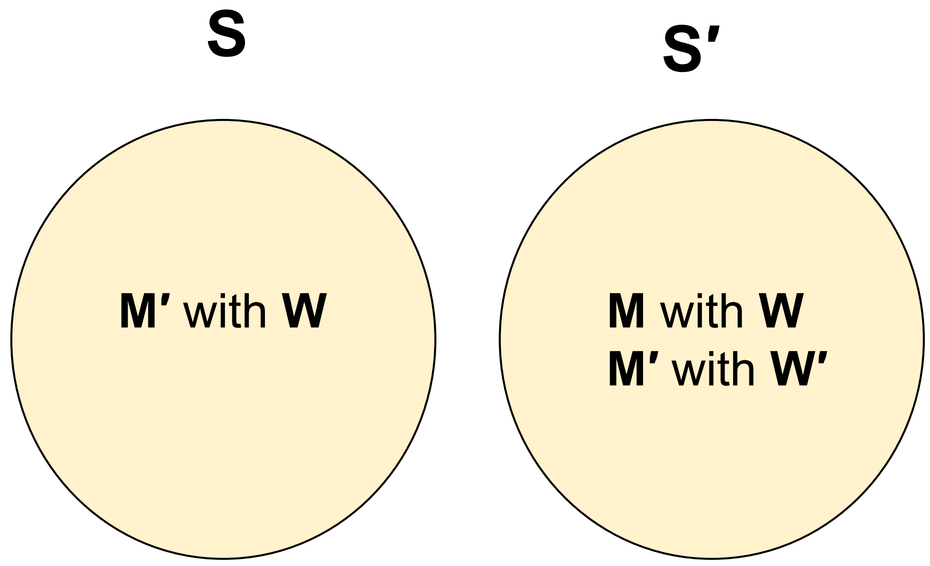
Proof by contradiction (2):

- Let w' be partner of m' in S'
- m' was not rejected by valid woman in S before m was rejected by w (by assump.)

  -> m' prefers w to w'
- Now w prefers m' over m, her partner in S'

  -> m' and w form a blocking pair in S'

So what’s going on here?



1. For $W \rightarrow M' > M$ (W rejects M in S)
2. For $M' \rightarrow W > W'$ (As M was the first to be rejected in S)
3. If the above are true, in S′, M′ and W creates blocking pair

Claim 5 - Order in which men propose does not impact the solution

- *Is the above claim just?*

Claim 6 - GS with the man proposing – results in a woman- pessimal matching

Claim 6 - GS with the man proposing – results in a woman- pessimal matching

- *m* and *w* matched in *S, m* is not worst valid
- -> exists stable *S'* (Non GS) with *w* paired to *m'* < *m*
- Let *w'* be partner of *m* in *S'*
- *m* prefers *w* over *w'* (by man-optimality)
- -> *m* and *w* form blocking pair in *S'*

Last but not least

Gale and Shapley proved that it is always possible to find a matching that makes all marriages stable, and provided a quadratic time algorithm which can be used to find one of two extreme stable marriages, the so-called male optimal or female optimal solutions.

For n men, n women Exponentially large number of stable matchings are possible. Giving a tight upper bound is an open challenge.

4. Miscellaneous

Nobel Prize

- Lloyd Shapley and Alvin E. Roth won the Nobel Prize in 2012 for their work on matching theory, including the kidney donor matching problem.

5. References

1. The Stable Marriage Problem – Optimizing Different Criteria using Genetic Algorithms
https://pdfs.semanticscholar.org/4d33/ecab3803df26695bc1d285ad17ca55192b76.pdf

2. Matching Theory: Kidney Allocation by Kyle Luong (Meds 2016)
http://www.uwomj.com/wp-content/uploads/2013/10/v82no1_6.pdf

3. Matching Kidney Donors with Those Who Need Them & Other Explorations in Economics
https://www.nap.edu/read/23508/pdf/frtr_annotated_kidney.pdf

4. CS364A: Algorithmic Game Theory Lecture #10: Kidney Exchange and Stable Matching∗
Tim Roughgarden
https://theory.stanford.edu/~tim/f13/l/l10.pdf

5. Wikipedia page on Stable Marriage Problem
https://en.wikipedia.org/wiki/Stable_marriage_problem

6. Stable Matching John P. Dickerson, CMU
http://www.cs.cmu.edu/~arielpro/15896s16/slides/896s16-16.pdf

7. Algorithm Design (Tardos and Kleinberg)