Stable Marriages

Let's play matchmaker.

Stable Marriages

Let's play matchmaker.

Given n men and n women, where

- Each man has a complete list of preferences for the women,
- Each woman has a complete list of preferences for the men,

Stable Marriages

Let's play matchmaker.

Given n men and n women, where

- Each man has a complete list of preferences for the women,
- Each woman has a complete list of preferences for the men,

Men's Preferences

	Bob	Doug	Fred
1st	Alice	Alice	Elena
2nd	Clara	Elena	Clara
3rd	Elena	Clara	Alice

Women's Preferences

	Alice	Clara	Elena
1st	Fred	Bob	Doug
2nd	Bob	Fred	Fred
3rd	Doug	Doug	Bob

Stable Marriages

Goal: Create a perfect matching of **stable marriages**.

That is, find a set of n couples where there are no instabilities:

Stable Marriages

Goal: Create a perfect matching of **stable marriages**.

That is, find a set of n couples where there are no instabilities:

Definition: An **instability** is when one man and one woman both prefer each other to their spouses.

Stable Marriages

Goal: Create a perfect matching of **stable marriages**.

That is, find a set of n couples where there are no instabilities:

Definition: An **instability** is when one man and one woman both prefer each other to their spouses.

Example. Suppose:

- Alice is married to Bob
- Clara is married to Doug

Stable Marriages

Goal: Create a perfect matching of **stable marriages**.

That is, find a set of n couples where there are no instabilities:

Definition: An **instability** is when one man and one woman both prefer each other to their spouses.

Example. Suppose:

- Alice is married to Bob
- Clara is married to Doug
- Bob prefers Clara to Alice

Stable Marriages

Goal: Create a perfect matching of **stable marriages**.

That is, find a set of n couples where there are no instabilities:

Definition: An **instability** is when one man and one woman both prefer each other to their spouses.

Example. Suppose:

- Alice is married to Bob
- Clara is married to Doug
- Bob prefers Clara to Alice

If Clara prefers Bob to Doug: ____________________________

Stable Marriages

Goal: Create a perfect matching of **stable marriages**.

That is, find a set of n couples where there are no instabilities:

Definition: An **instability** is when one man and one woman both prefer each other to their spouses.

Example. Suppose:

- Alice is married to Bob
- Clara is married to Doug
- Bob prefers Clara to Alice

If Clara prefers Bob to Doug: ____________________

If Clara prefers Doug to Bob: ____________________

The Gale–Shapley Algorithm

Theorem. (Gale, Shapley, 1962) In the above situation, there always exists a perfect matching of stable marriages.

The Gale–Shapley Algorithm

Theorem. (Gale, Shapley, 1962) In the above situation, there always exists a perfect matching of stable marriages.

Proof. Use the **Gale–Shapley Algorithm** to create the marriages.

The Gale–Shapley Algorithm

Theorem. (Gale, Shapley, 1962) In the above situation, there always exists a perfect matching of stable marriages.

Proof. Use the **Gale–Shapley Algorithm** to create the marriages.

1. Start with no couples engaged.

The Gale–Shapley Algorithm

Theorem. (Gale, Shapley, 1962) In the above situation, there always exists a perfect matching of stable marriages.

Proof. Use the **Gale–Shapley Algorithm** to create the marriages.

1. Start with no couples engaged.
2. As long as at least one man is unengaged, repeat the following: *Each unengaged man* proposes to his next most preferred woman.

The Gale–Shapley Algorithm

Theorem. (Gale, Shapley, 1962) In the above situation, there always exists a perfect matching of stable marriages.

Proof. Use the **Gale–Shapley Algorithm** to create the marriages.

1. Start with no couples engaged.
2. As long as at least one man is unengaged, repeat the following: *Each unengaged man* proposes to his next most preferred woman. *Each woman* then decides whether to accept or reject the proposal(s), as follows:
   - If she is not already engaged, she accepts the proposal.
   - If she is already engaged, she uses her preference list; she accepts the proposal of the man she prefers the most and rejects all others.

The Gale–Shapley Algorithm

Theorem. (Gale, Shapley, 1962) In the above situation, there always exists a perfect matching of stable marriages.

Proof. Use the **Gale–Shapley Algorithm** to create the marriages.

1. Start with no couples engaged.
2. As long as at least one man is unengaged, repeat the following: *Each unengaged man* proposes to his next most preferred woman. *Each woman* then decides whether to accept or reject the proposal(s), as follows:
    - If she is not already engaged, she accepts the proposal.
    - If she is already engaged, she uses her preference list; she accepts the proposal of the man she prefers the most and rejects all others.
3. When all men are engaged, stop. There are n stable marriages.

The Gale–Shapley Algorithm

Theorem. (Gale, Shapley, 1962) In the above situation, there always exists a perfect matching of stable marriages.

Proof. Use the **Gale–Shapley Algorithm** to create the marriages.

1. Start with no couples engaged.
2. As long as at least one man is unengaged, repeat the following: *Each unengaged man* proposes to his next most preferred woman. *Each woman* then decides whether to accept or reject the proposal(s), as follows:
   - If she is not already engaged, she accepts the proposal.
   - If she is already engaged, she uses her preference list; she accepts the proposal of the man she prefers the most and rejects all others.
3. When all men are engaged, stop. There are n stable marriages.

$<<$Time for your moment of zen$>>$

Applying the Gale–Shapley Algorithm

Here is a complete set of preferences for 4 men and 4 women.

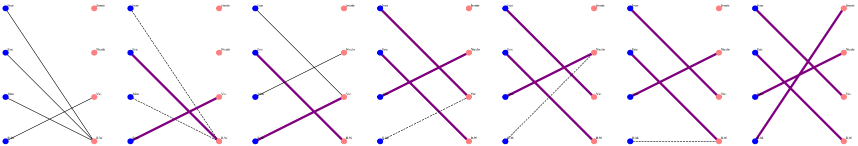
Men's Preferences

	Ivan	Eric	John	R.M.
1^{st}	R.W.	R.W.	R.W.	Vic.
2^{nd}	Vic.	Jennie	Nicole	Nicole
3^{rd}	Jennie	Vic.	Jennie	R.W.
4^{th}	Nicole	Nicole	Vic.	Jennie

Women's Preferences

	Jennie	Nicole	Vic.	R.W.
1^{st}	Eric	John	John	Eric
2^{nd}	John	R.M.	Ivan	R.M.
3^{rd}	R.M.	Eric	R.M.	Ivan
4^{th}	Ivan	Ivan	Eric	John

The Algorithm, Pictorially



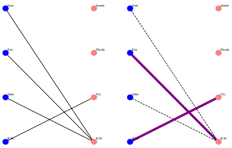
Men's Preferences

Ivan	Eric	John	R.M.
R.W.	R.W.	R.W.	Vic.
Vic.	Jennie	Nicole	Nicole
Jennie	Vic.	Jennie	R.W.
Nicole	Nicole	Vic.	Jennie

Women's Preferences

Jennie	Nicole	Vic.	R.W.
Eric	John	John	Eric
John	R.M.	Ivan	R.M.
R.M.	Eric	R.M.	Ivan
Ivan	Ivan	Eric	John

The Algorithm, Pictorially



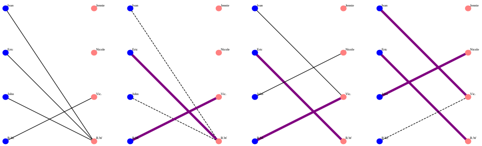
Men's Preferences

Ivan	Eric	John	R.M.
R.W.	**R.W.**	**R.W.**	**Vic.**
Vic.	Jennie	Nicole	Nicole
Jennie	Vic.	Jennie	R.W.
Nicole	Nicole	Vic.	Jennie

Women's Preferences

Jennie	Nicole	Vic.	R.W.
Eric	John	John	**Eric**
John	R.M.	Ivan	R.M.
R.M.	Eric	**R.M.**	**Ivan**
Ivan	Ivan	Eric	**John**

The Algorithm, Pictorially



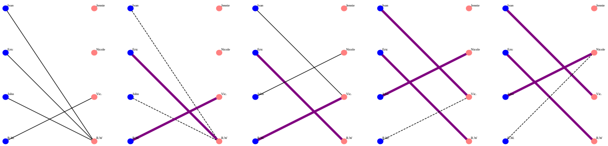
Men's Preferences

Ivan	Eric	John	R.M.
R.W.	**R.W.**	R.W.	**Vic.**
Vic.	Jennie	**Nicole**	Nicole
Jennie	Vic.	Jennie	R.W.
Nicole	Nicole	Vic.	Jennie

Women's Preferences

Jennie	Nicole	Vic.	R.W.
Eric	**John**	John	**Eric**
John	R.M.	**Ivan**	R.M.
R.M.	Eric	**R.M.**	Ivan
Ivan	Ivan	Eric	John

The Algorithm, Pictorially



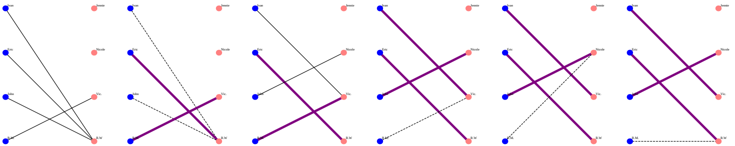
Men's Preferences

Ivan	Eric	John	R.M.
R.W.	**R.W.**	R.W.	Vic.
Vic.	Jennie	**Nicole**	**Nicole**
Jennie	Vic.	Jennie	R.W.
Nicole	Nicole	Vic.	Jennie

Women's Preferences

Jennie	Nicole	Vic.	R.W.
Eric	**John**	John	**Eric**
John	**R.M.**	**Ivan**	R.M.
R.M.	Eric	R.M.	Ivan
Ivan	Ivan	Eric	John

The Algorithm, Pictorially



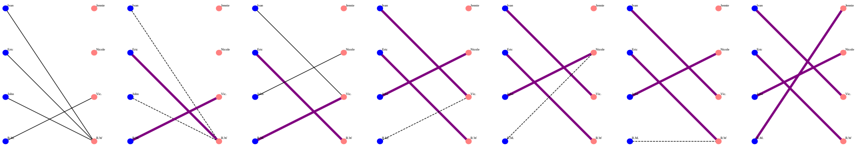
Men's Preferences

Ivan	Eric	John	R.M.
R.W.	**R.W.**	R.W.	Vic.
Vic.	Jennie	**Nicole**	Nicole
Jennie	Vic.	Jennie	**R.W.**
Nicole	Nicole	Vic.	Jennie

Women's Preferences

Jennie	Nicole	Vic.	R.W.
Eric	**John**	John	**Eric**
John	R.M.	**Ivan**	**R.M.**
R.M.	Eric	R.M.	Ivan
Ivan	Ivan	Eric	John

The Algorithm, Pictorially



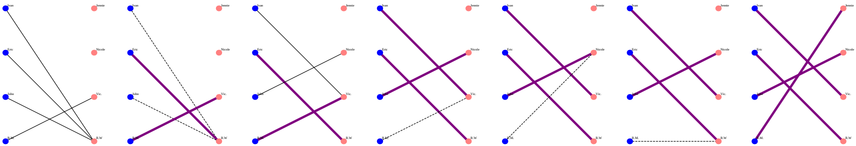
Men's Preferences

Ivan	Eric	John	R.M.
R.W.	**R.W.**	R.W.	Vic.
Vic.	Jennie	**Nicole**	Nicole
Jennie	Vic.	Jennie	R.W.
Nicole	Nicole	Vic.	**Jennie**

Women's Preferences

Jennie	Nicole	Vic.	R.W.
Eric	**John**	John	**Eric**
John	R.M.	**Ivan**	R.M.
R.M.	Eric	R.M.	Ivan
Ivan	Ivan	Eric	John

The Algorithm, Pictorially



Men's Preferences

Ivan	Eric	John	R.M.
R.W.	**R.W.**	R.W.	Vic.
Vic.	Jennie	**Nicole**	Nicole
Jennie	Vic.	Jennie	R.W.
Nicole	Nicole	Vic.	**Jennie**

Women's Preferences

Jennie	Nicole	Vic.	R.W.
Eric	**John**	John	**Eric**
John	R.M.	**Ivan**	R.M.
R.M.	Eric	R.M.	Ivan
Ivan	Ivan	Eric	John

Proof of Correctness

Claim. The Gale–Shapley Algorithm gives a set of n stable marriages.

Proof of Correctness

Claim. The Gale–Shapley Algorithm gives a set of n stable marriages.

Proof. We must show that the algorithm always stops, and that when it stops, the output is indeed a full set of stable marriages.

Proof of Correctness

Claim. The Gale–Shapley Algorithm gives a set of n stable marriages.

Proof. We must show that the algorithm always stops, and that when it stops, the output is indeed a full set of stable marriages.

The algorithm terminates.

Proof of Correctness

Claim. The Gale–Shapley Algorithm gives a set of n stable marriages.

Proof. We must show that the algorithm always stops, and that when it stops, the output is indeed a full set of stable marriages.

The algorithm terminates.

- In each step, at least one proposal occurs
- there are a finite number to be made.

Proof of Correctness

Claim. The Gale–Shapley Algorithm gives a set of n stable marriages.

Proof. We must show that the algorithm always stops, and that when it stops, the output is indeed a full set of stable marriages.

The algorithm terminates.

- In each step, at least one proposal occurs
- there are a finite number to be made.

Claim: Upon termination, everyone is engaged.

Proof of Correctness

Claim. The Gale–Shapley Algorithm gives a set of n stable marriages.

Proof. We must show that the algorithm always stops, and that when it stops, the output is indeed a full set of stable marriages.

The algorithm terminates.

- In each step, at least one proposal occurs
- there are a finite number to be made.

Claim: Upon termination, everyone is engaged.

- Once a woman has been proposed to, she stays engaged.
- If a woman is not engaged at the end, she had no proposal.
- It follows that there is also some man not engaged; however, he must have proposed to the unengaged woman during some round!

Proof of Correctness

The output is a set of stable marriages.

We ask: Is there an instability?

Proof of Correctness

The output is a set of stable marriages.

We ask: Is there an instability?

- Suppose Bob prefers Clara to his current wife.

Proof of Correctness

The output is a set of stable marriages.

We ask: Is there an instability?

- Suppose Bob prefers Clara to his current wife.
- Bob must have proposed to Clara before his current wife.

Proof of Correctness

The output is a set of stable marriages.

We ask: Is there an instability?

- Suppose Bob prefers Clara to his current wife.
- Bob must have proposed to Clara before his current wife.
- Clara must have turned down Bob.

Proof of Correctness

The output is a set of stable marriages.

We ask: Is there an instability?

- Suppose Bob prefers Clara to his current wife.
- Bob must have proposed to Clara before his current wife.
- Clara must have turned down Bob.
  - Clara was proposed to by someone she prefers!

Proof of Correctness

The output is a set of stable marriages.

We ask: Is there an instability?

- Suppose Bob prefers Clara to his current wife.
- Bob must have proposed to Clara before his current wife.
- Clara must have turned down Bob.
  - Clara was proposed to by someone she prefers!
- Hence, Clara must prefer her current husband to Bob.

Proof of Correctness

The output is a set of stable marriages.

We ask: Is there an instability?

- Suppose Bob prefers Clara to his current wife.
- Bob must have proposed to Clara before his current wife.
- Clara must have turned down Bob.
  - Clara was proposed to by someone she prefers!
- Hence, Clara must prefer her current husband to Bob.
- Therefore, there is no instability.

Male-optimality

Claim. The marriages $\mathcal{S}$ generated by the Gale–Shapley Algorithm are **male optimal**.

Male-optimality

Claim. The marriages $\mathcal{S}$ generated by the Gale–Shapley Algorithm are **male optimal**. That is, given any other set of stable marriages, each man will only be paired with a woman lower on his preference list.

Male-optimality

Claim. The marriages $\mathcal{S}$ generated by the Gale–Shapley Algorithm are **male optimal**. That is, given any other set of stable marriages, each man will only be paired with a woman lower on his preference list.

Proof. Suppose that during the Gale–Shapley Algorithm, there is a man who is paired with a "sub-optimal" woman.

Male-optimality

Claim. The marriages $\mathcal{S}$ generated by the Gale–Shapley Algorithm are **male optimal**. That is, given any other set of stable marriages, each man will only be paired with a woman lower on his preference list.

Proof. Suppose that during the Gale–Shapley Algorithm, there is a man who is paired with a "sub-optimal" woman.

- Let M be the first man who is rejected by his optimal woman W during the algorithm.

M-Optimal Pairings ($\mathcal{S}'$)

Relative Pref's

N	W

Male-optimality

Claim. The marriages $\mathcal{S}$ generated by the Gale–Shapley Algorithm are **male optimal**. That is, given any other set of stable marriages, each man will only be paired with a woman lower on his preference list.

Proof. Suppose that during the Gale–Shapley Algorithm, there is a man who is paired with a "sub-optimal" woman.

- Let M be the first man who is rejected by his optimal woman W during the algorithm. [*That is, there is some other set $\mathcal{S}'$ of stable marriages in which M is paired with W.*]

M-Optimal Pairings ($\mathcal{S}'$)

Relative Pref's

N	W

Male-optimality

Claim. The marriages $\mathcal{S}$ generated by the Gale–Shapley Algorithm are **male optimal**. That is, given any other set of stable marriages, each man will only be paired with a woman lower on his preference list.

Proof. Suppose that during the Gale–Shapley Algorithm, there is a man who is paired with a "sub-optimal" woman.

- Let M be the first man who is rejected by his optimal woman W during the algorithm. [*That is, there is some other set $\mathcal{S}'$ of stable marriages in which M is paired with W.*]
- M is rejected because some man N proposes to W whom W prefers to M.

M-Optimal Pairings ($\mathcal{S}'$)

Relative Pref's

N	W

Male-optimality

Claim. The marriages $\mathcal{S}$ generated by the Gale–Shapley Algorithm are **male optimal**. That is, given any other set of stable marriages, each man will only be paired with a woman lower on his preference list.

Proof. Suppose that during the Gale–Shapley Algorithm, there is a man who is paired with a "sub-optimal" woman.

- Let M be the first man who is rejected by his optimal woman W during the algorithm. [*That is, there is some other set $\mathcal{S}'$ of stable marriages in which M is paired with W.*]
- M is rejected because some man N proposes to W whom W prefers to M.
- Since M is the *first* man rejected, we know N likes W at least as much as his optimal woman.

M-Optimal Pairings ($\mathcal{S}'$)

Relative Pref's

N	W

Male-optimality

Claim. The marriages $\mathcal{S}$ generated by the Gale–Shapley Algorithm are **male optimal**. That is, given any other set of stable marriages, each man will only be paired with a woman lower on his preference list.

Proof. Suppose that during the Gale–Shapley Algorithm, there is a man who is paired with a "sub-optimal" woman.

- Let M be the first man who is rejected by his optimal woman W during the algorithm. [*That is, there is some other set $\mathcal{S}'$ of stable marriages in which M is paired with W.*]
- M is rejected because some man N proposes to W whom W prefers to M.
- Since M is the *first* man rejected, we know N likes W at least as much as his optimal woman.
- This, in turn, creates an instability in $\mathcal{S}'$ since W prefers N to M and N prefers W to the woman he is paired with.

M-Optimal Pairings ($\mathcal{S}'$)

Relative Pref's

N	W

Last remarks

- The marriages generated by Gale–Shapley are male optimal.
- The marriages generated by Gale–Shapley are female pessimal.

Last remarks

- The marriages generated by Gale–Shapley are male optimal.
- The marriages generated by Gale–Shapley are female pessimal.
- Run the algorithm with the women proposing to reverse the roles.

  If you do this and get the same marriages, ________________

Last remarks

- The marriages generated by Gale–Shapley are male optimal.
- The marriages generated by Gale–Shapley are female pessimal.
- Run the algorithm with the women proposing to reverse the roles.

  If you do this and get the same marriages, ____________________
- If not all rankings are made, then there may be unmatched people. For example, what if Robot Man did not like Jennie?

Last remarks

- The marriages generated by Gale–Shapley are male optimal.
- The marriages generated by Gale–Shapley are female pessimal.
- Run the algorithm with the women proposing to reverse the roles.

  If you do this and get the same marriages, ________________
- If not all rankings are made, then there may be unmatched people. For example, what if Robot Man did not like Jennie?
- The National Resident Matching Program (`http://www.nrmp.org`) implements this algorithm to match medical students to residency programs.