Disjoint Stable Matchings

Aadityan Ganesh, Prajakta Nimbhorkar, Geevarghese Philip, Vishwa Prakash HV

January 6, 2021

Chennai Mathematical Institute

Outline

- Background
  - Stable Matchings
  - Gale-Shapley Algorithm
  - Extended Gale-Shapley Algorithm
  - The Lattice Structure
- Disjoint Stable Matchings
  - Disjoint Perfect Matchings
  - Disjoint Stable Matchings
  - Algorithm to find Disjoint Stable Matchings
  - Correctness and Running time of the Algorithm
- Rotations
  - Rotational poset
  - Properties of Rotational poset
  - Analogous Algorithm

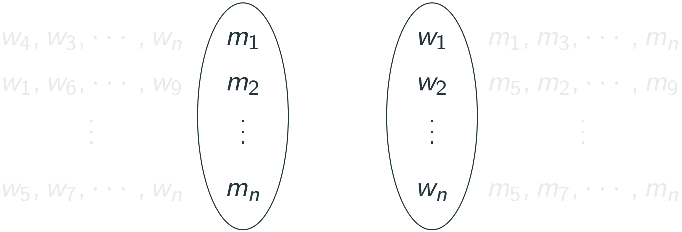
Marriage Matching Instance

A marriage matching instance of *size* n involves two disjoint sets of size n, the men and the women. Associated with each person is a *strictly* ordered *preference list* containing *all* the members of the opposite sex. Person p prefers q to r, where q and r are of the opposite sex of p, if and only if q precedes r on p's preference list.

Stable Matchings

Marriage Matching Instance

A marriage matching instance of *size* n involves two disjoint sets of size n, the men and the women. Associated with each person is a *strictly* ordered *preference list* containing *all* the members of the opposite sex. Person p prefers q to r, where q and r are of the opposite sex of p, if and only if q precedes r on p's preference list.

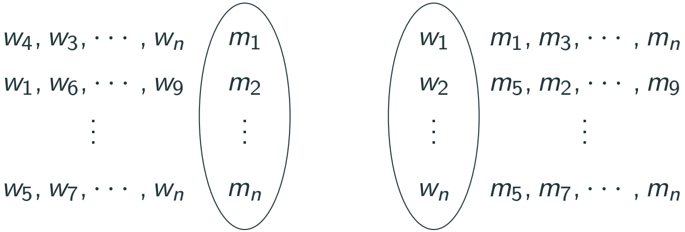


Set of men   Set of women

Stable Matchings

Marriage Matching Instance

A marriage matching instance of *size* n involves two disjoint sets of size n, the men and the women. Associated with each person is a *strictly* ordered *preference list* containing *all* the members of the opposite sex. Person p prefers q to r, where q and r are of the opposite sex of p, if and only if q precedes r on p's preference list.

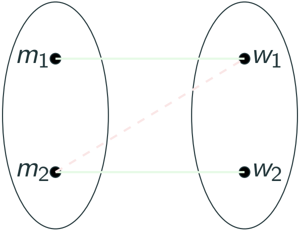


Set of men Set of women

If (m, w) are matched in a matching M, we say $m = p_M(w)$ and $w = p_M(m)$

Blocking Pair

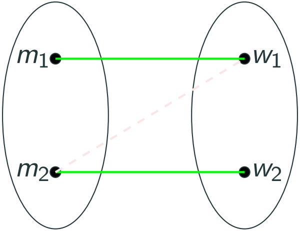
A man m and a woman w are said to *block* a matching M, or the pair (m, w) is said to be a *blocking pair* for M, if m and w are not partners in M, but m prefers w to $p_M(m)$ and w prefers m to $p_M(w)$. A matching with no blocking pair is called a *stable* matching, and is otherwise *unstable*.



If (m, w) are matched in a matching M, we say $m = p_M(w)$ and $w = p_M(m)$

Blocking Pair

A man m and a woman w are said to *block* a matching M, or the pair (m, w) is said to be a *blocking pair* for M, if m and w are not partners in M, but m prefers w to $p_M(m)$ and w prefers m to $p_M(w)$. A matching with no blocking pair is called a *stable* matching, and is otherwise *unstable*.

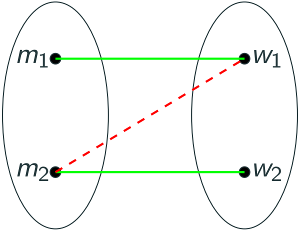


Stable Matchigs

If (m, w) are matched in a matching M, we say $m = p_M(w)$ and $w = p_M(m)$

Blocking Pair

A man m and a woman w are said to *block* a matching M, or the pair (m, w) is said to be a *blocking pair* for M, if m and w are not partners in M, but m prefers w to $p_M(m)$ and w prefers m to $p_M(w)$. A matching with no blocking pair is called a *stable* matching, and is otherwise *unstable*.



Stable Matchings

Stable Matching

A matching with no blocking pair

Checking stability: $O(n^2)$

Stable and Fixed Pairs

Stable Pair

A pair (m, w) is called as a *stable pair* if m and w are partners in at least one stable matching.

Fixed Pair

A pair (m, w) is called as a *fixed pair* if m and w are partners in at all stable matchings.

Stable and Fixed Pairs

Stable Pair

A pair (m, w) is called as a *stable pair* if m and w are partners in at least one stable matching.

Fixed Pair

A pair (m, w) is called as a *fixed pair* if m and w are partners in at all stable matchings.

Gale-Shapley Algorithm

Algorithm 1 Gale-Shapley

```
procedure FIND STABLE MATCHING(M)
    assign each person to be free
    while some man m is free do
        w ← first woman on m's list to whom m hasn't proposed
        if w is free then
            assign m and w to be engaged to each other
        else
            if w prefers m to her current matched partner m′ then
                assign m and w to be engaged and m′ to be free
            else
                w rejects m                                ▷ m remains free
            end if
        end if
    end while
    return Stable matching consisting of n engaged pairs
end procedure
```

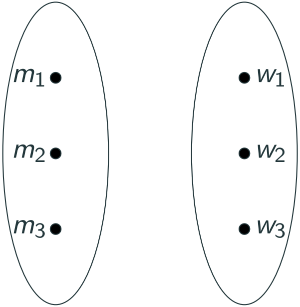
Gale-Shapley Algorithm

m_1 :	w_2	w_3	w_1
m_2 :	w_3	w_1	w_2
m_3 :	w_2	w_1	w_3

Men's Preference

w_1 :	m_1	m_2	m_3
w_2 :	m_2	m_1	m_3
w_3 :	m_3	m_2	m_1

Women's Preference



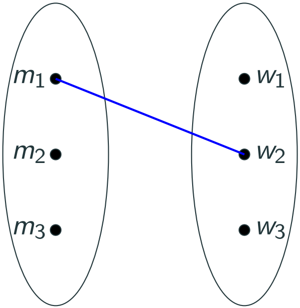
Gale-Shapley Algorithm

m_1 :	w_2	w_3	w_1
m_2 :	w_3	w_1	w_2
m_3 :	w_2	w_1	w_3

Men's Preference

w_1 :	m_1	m_2	m_3
w_2 :	m_2	m_1	m_3
w_3 :	m_3	m_2	m_1

Women's Preference



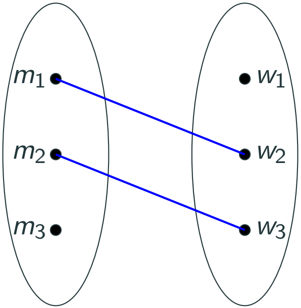
Gale-Shapley Algorithm

m_1 :	w_2	w_3	w_1
m_2 :	w_3	w_1	w_2
m_3 :	w_2	w_1	w_3

Men's Preference

w_1 :	m_1	m_2	m_3
w_2 :	m_2	m_1	m_3
w_3 :	m_3	m_2	m_1

Women's Preference



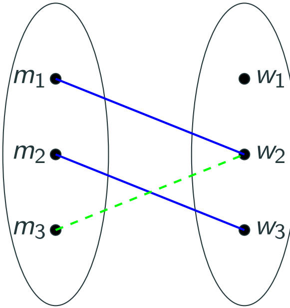
Gale-Shapley Algorithm

m_1 :	w_2	w_3	w_1
m_2 :	w_3	w_1	w_2
m_3 :	w_2	w_1	w_3

Men's Preference

w_1 :	m_1	m_2	m_3
w_2 :	m_2	m_1	m_3
w_3 :	m_3	m_2	m_1

Women's Preference



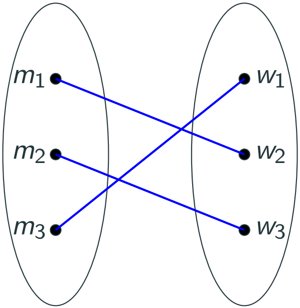
Gale-Shapley Algorithm

m_1 :	w_2	w_3	w_1
m_2 :	w_3	w_1	w_2
m_3 :	w_2	w_1	w_3

Men's Preference

w_1 :	m_1	m_2	m_3
w_2 :	m_2	m_1	m_3
w_3 :	m_3	m_2	m_1

Women's Preference



Gale-Shapley Algorithm - Key Results

1. Every marriage instance has a stable matching.
2. All possible execution of the Gale-Shapley algorithm yields the same result.
3. It results in *"Man-optimal"* stable matching.

   Man-optimal: Every man is matched with his most favored partner among all stable partners.
4. Reversing roles, i.e, women proposing, results in *"Woman-optimal"* stable matching.

   Woman-optimal: Every woman is matched with her most favored partner among all stable partners.
5. The man-optimal stable matching is *woman-pessimal*, and vice-versa.

Gale-Shapley Algorithm - Key Results

1. Every marriage instance has a stable matching.
2. All possible execution of the Gale-Shapley algorithm yields the same result.
3. It results in *"Man-optimal"* stable matching.

   Man-optimal: Every man is matched with his most favored partner among all stable partners.
4. Reversing roles, i.e, women proposing, results in *"Woman-optimal"* stable matching.

   Woman-optimal: Every woman is matched with her most favored partner among all stable partners.
5. The man-optimal stable matching is *woman-pessimal*, and vice-versa.

Gale-Shapley Algorithm - Key Results

1. Every marriage instance has a stable matching.
2. All possible execution of the Gale-Shapley algorithm yields the same result.
3. It results in *"Man-optimal"* stable matching.

   Man-optimal: Every man is matched with his most favored partner among all stable partners.
4. Reversing roles, i.e, women proposing, results in *"Woman-optimal"* stable matching.

   Woman-optimal: Every woman is matched with her most favored partner among all stable partners.
5. The man-optimal stable matching is *woman-pessimal*, and vice-versa.

Gale-Shapley Algorithm - Key Results

1. Every marriage instance has a stable matching.
2. All possible execution of the Gale-Shapley algorithm yields the same result.
3. It results in *"Man-optimal"* stable matching.

   Man-optimal: Every man is matched with his most favored partner among all stable partners.
4. Reversing roles, i.e, women proposing, results in *"Woman-optimal"* stable matching.

   Woman-optimal: Every woman is matched with her most favored partner among all stable partners.
5. The man-optimal stable matching is *woman-pessimal*, and vice-versa.

Gale-Shapley Algorithm - Key Results

1. Every marriage instance has a stable matching.
2. All possible execution of the Gale-Shapley algorithm yields the same result.
3. It results in *“Man-optimal”* stable matching.

   Man-optimal: Every man is matched with his most favored partner among all stable partners.
4. Reversing roles, i.e, women proposing, results in *“Woman-optimal”* stable matching.

   Woman-optimal: Every woman is matched with her most favored partner among all stable partners.
5. The man-optimal stable matching is *woman-pessimal*, and vice-versa.

Extended Gale-Shapley Algorithm

Algorithm 2 Extended Gale-Shapley

```
procedure GS-EXTENDED(M)                              ▷ M is an SM instance
    assign each person to be free
    while some man m is free do
        w ← first woman on m's list
        if some man p is engaged to w then
            assign p to be free
        end if
        assign m and w to be engaged to each other
        for each successor m' of m on w's list do
            delete w on m''s list
            delete m' on w's list                     ▷ deleting the pair (m', w)
        end for
    end while
    return Stable matching consisting of n engaged pairs
end procedure
```

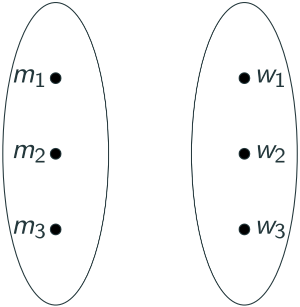
Run of Extended GS Algorithm

m_1 :	w_2	w_3	w_1
m_2 :	w_3	w_1	w_2
m_3 :	w_2	w_1	w_3

Men's Preference

w_1 :	m_1	m_2	m_3
w_2 :	m_2	m_1	m_3
w_3 :	m_3	m_2	m_1

Women's Preference



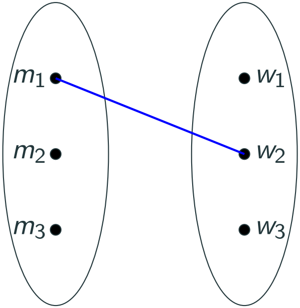
Run of Extended GS Algorithm

m_1 :	w_2	w_3	w_1
m_2 :	w_3	w_1	w_2
m_3 :	w_2	w_1	w_3

Men's Preference

w_1 :	m_1	m_2	m_3
w_2 :	m_2	m_1	m_3
w_3 :	m_3	m_2	m_1

Women's Preference



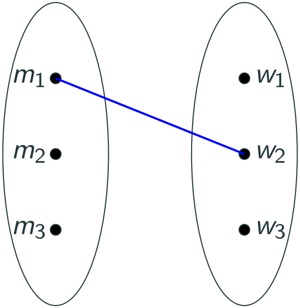
Run of Extended GS Algorithm

m_1 :	w_2	w_3	w_1
m_2 :	w_3	w_1	w_2
m_3 :		w_1	w_3

Men's Preference

w_1 :	m_1	m_2	m_3
w_2 :	m_2	m_1	
w_3 :	m_3	m_2	m_1

Women's Preference



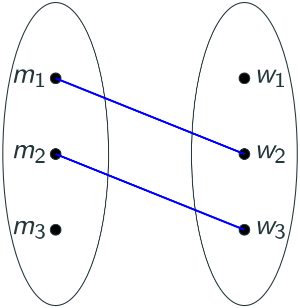
Run of Extended GS Algorithm

m_1 :	w_2	w_3	w_1
m_2 :	w_3	w_1	w_2
m_3 :		w_1	w_3

Men's Preference

w_1 :	m_1	m_2	m_3
w_2 :	m_2	m_1	
w_3 :	m_3	m_2	m_1

Women's Preference



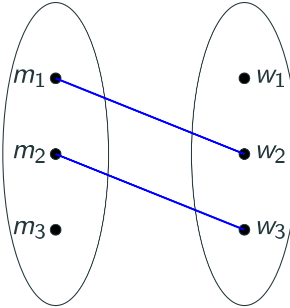
Run of Extended GS Algorithm

m_1 :	w_2		w_1
m_2 :	w_3	w_1	w_2
m_3 :		w_1	w_3

Men's Preference

w_1 :	m_1	m_2	m_3
w_2 :	m_2	m_1	
w_3 :	m_3	m_2	

Women's Preference



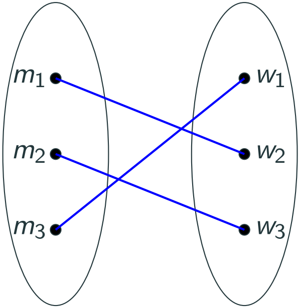
Run of Extended GS Algorithm

m_1 :	w_2		w_1
m_2 :	w_3	w_1	w_2
m_3 :		w_1	w_3

Men's Preference

w_1 :	m_1	m_2	m_3
w_2 :	m_2	m_1	
w_3 :	m_3	m_2	

Women's Preference



GS-lists

MGS-list

Final preference lists generated by the extended Gale-Shapley algorithm with men as proposers are called as *man-oriented Gale-Shapley lists* or *MGS-lists*.

WGS-list

Final preference lists generated by the extended Gale-Shapley algorithm with *women* as proposers are called as *woman-oriented Gale-Shapley lists* or *WGS-lists*.

GS-lists

MGS-list

Final preference lists generated by the extended Gale-Shapley algorithm with men as proposers are called as *man-oriented Gale-Shapley lists* or *MGS-lists*.

WGS-list

Final preference lists generated by the extended Gale-Shapley algorithm with *women* as proposers are called as *woman-oriented Gale-Shapley lists* or *WGS-lists*.

GS-lists

GS-list

Intersection of MGS-list and WGS-list.

Note: GS-lists can be obtained by applying man-oriented extended Gale-Shapley algorithm to get MGS-lists and then, starting with the MGS-lists, applying woman-oriented extended GS algorithm.

Extended GS Algorithm Key Results

1. all stable matchings are contained in the GS-lists.
2. no matching (stable or otherwise) contained in the GS-lists can be blocked by a pair that is not in the GS-lists.
3. In the man-optimal (respectively woman-optimal) stable matching, each man is partnered by the first (respectively last) woman on his GS-list, and each woman by the last (respectively first) man on hers.

Extended GS Algorithm Key Results

1. all stable matchings are contained in the GS-lists.
2. no matching (stable or otherwise) contained in the GS-lists can be blocked by a pair that is not in the GS-lists.
3. In the man-optimal (respectively woman-optimal) stable matching, each man is partnered by the first (respectively last) woman on his GS-list, and each woman by the last (respectively first) man on hers.

Extended GS Algorithm Key Results

1. all stable matchings are contained in the GS-lists.
2. no matching (stable or otherwise) contained in the GS-lists can be blocked by a pair that is not in the GS-lists.
3. In the man-optimal (respectively woman-optimal) stable matching, each man is partnered by the first (respectively last) woman on his GS-list, and each woman by the last (respectively first) man on hers.

Extended GS Algorithm Key Results

1. all stable matchings are contained in the GS-lists.
2. no matching (stable or otherwise) contained in the GS-lists can be blocked by a pair that is not in the GS-lists.
3. In the man-optimal (respectively woman-optimal) stable matching, each man is partnered by the first (respectively last) woman on his GS-list, and each woman by the last (respectively first) man on hers.

The Lattice Structure

A person x is said to *prefer* a matching M to a matching M' if x prefers $p_M(x)$ to $p_{M'}(x)$.

Domination

A stable matching M is said to *dominate* a stable matching M', written $M \preceq M'$, if every man has at least as good a partner in M as he has in M'.i.e., every man either prefers M to M' or is indifferent between them. M *disjointly dominates* $M'(M \prec M')$ if $M \preceq M'$ and $M \cap M' = \varnothing$.

m_1 :				
m_2 :				
m_3 :				
m_4 :				

Men's Preference

The Lattice Structure

A person x is said to *prefer* a matching M to a matching M' if x prefers $p_M(x)$ to $p_{M'}(x)$.

Domination

A stable matching M is said to *dominate* a stable matching M', written $M \preceq M'$, if every man has at least as good a partner in M as he has in M'.i.e., every man either prefers M to M' or is indifferent between them. M *disjointly dominates* $M'(M \prec M')$ if $M \preceq M'$ and $M \cap M' = \varnothing$.

m_1 :				
m_2 :				
m_3 :				
m_4 :				

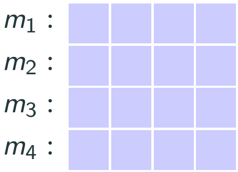
Men's Preference

The Lattice Structure

A person x is said to *prefer* a matching M to a matching M' if x prefers $p_M(x)$ to $p_{M'}(x)$.

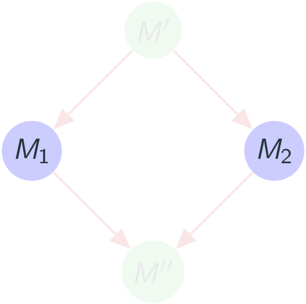
Domination

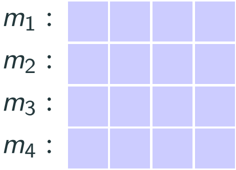
A stable matching M is said to *dominate* a stable matching M', written $M \preceq M'$, if every man has at least as good a partner in M as he has in M'.i.e., every man either prefers M to M' or is indifferent between them. *M disjointly dominates M'* $(M \prec M')$ if $M \preceq M'$ and $M \cap M' = \varnothing$.



Men's Preference

Meet and Join



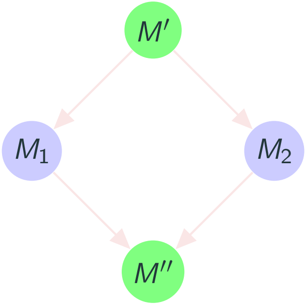


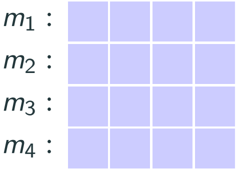
Men's Preference

$$M' = \{(m, w) \mid w = best(p_{M_1}(m), p_{M_2}(m))\}$$

$$M'' = \{(m, w) \mid w = worst(p_{M_1}(m), p_{M_2}(m))\}$$

Meet and Join



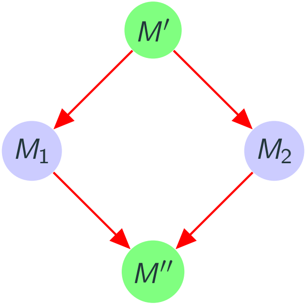


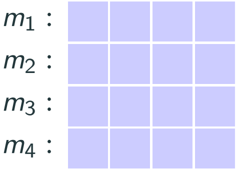
Men's Preference

$$M' = \{(m, w) \mid w = best(p_{M_1}(m), p_{M_2}(m))\}$$

$$M'' = \{(m, w) \mid w = worst(p_{M_1}(m), p_{M_2}(m))\}$$

Meet and Join



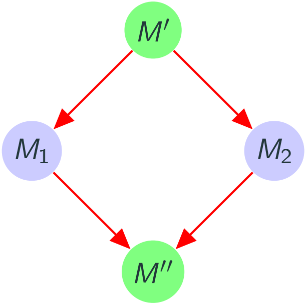


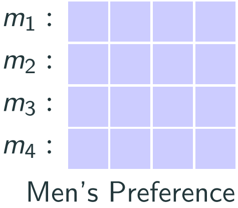
Men's Preference

$$M' = \{(m, w) \mid w = best(p_{M_1}(m), p_{M_2}(m))\}$$

$$M'' = \{(m, w) \mid w = worst(p_{M_1}(m), p_{M_2}(m))\}$$

Meet and Join



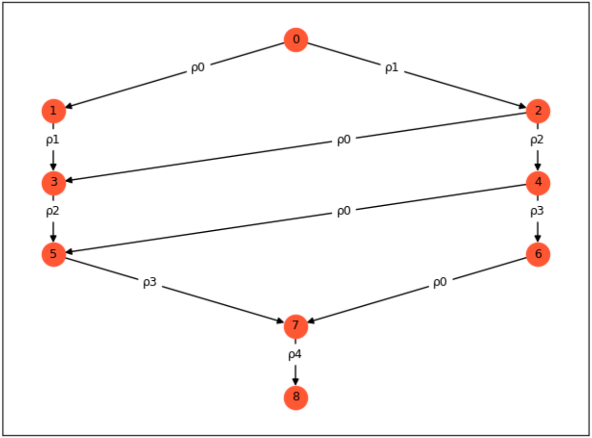


$$M' = \{(m, w) \mid w = best(p_{M_1}(m), p_{M_2}(m))\}$$

$$M'' = \{(m, w) \mid w = worst(p_{M_1}(m), p_{M_2}(m))\}$$

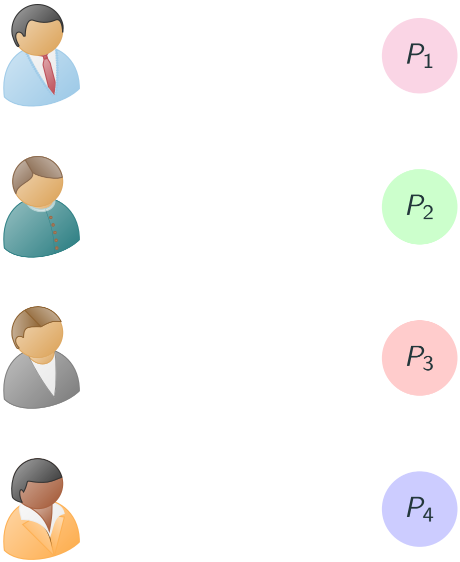
The Lattice Structure

Set of all stable matchings form a distributive lattice under the *Domination* domination.

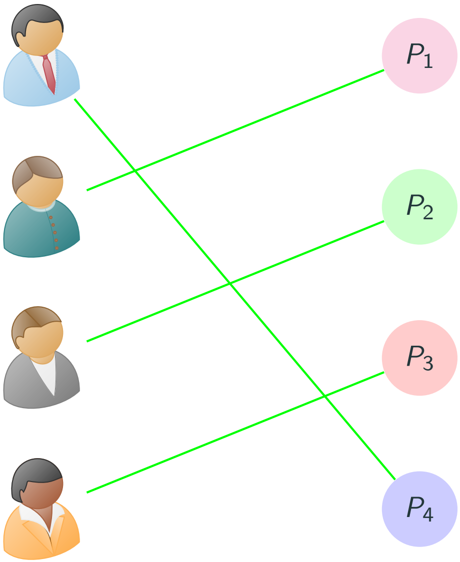


Disjoint Stable Matchings

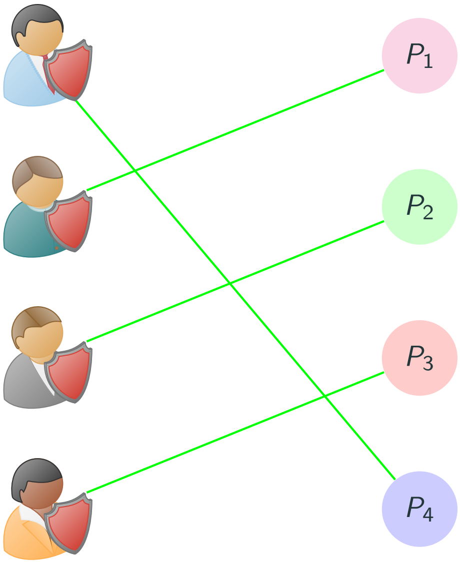
Why do we need them?



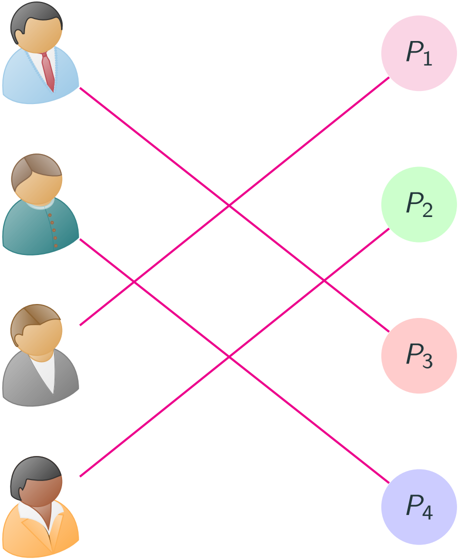
Why do we need them?



Why do we need them?



Why do we need them?



Problem Statement

For a given marriage instance, find a largest set S of disjoint stable matchings.

Existence of Disjoint Stable Matchings

Does there exist a marriage matching instances with disjoint stable matchings?

$m_1 : w_1, w_2, w_3$	$w_1 : m_2, m_3, m_1$
$m_2 : w_2, w_3, w_1$	$w_2 : m_3, m_1, m_2$
$m_3 : w_3, w_1, w_2$	$w_3 : m_1, m_2, m_3$

Existence of Disjoint Stable Matchings

Does there exist a marriage matching instances with disjoint stable matchings?

$$
\begin{array}{ll}
m_1 : w_1, w_2, w_3 & w_1 : m_2, m_3, m_1 \\
m_2 : w_2, w_3, w_1 & w_2 : m_3, m_1, m_2 \\
m_3 : w_3, w_1, w_2 & w_3 : m_1, m_2, m_3
\end{array}
$$

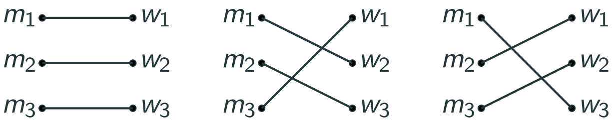
Existence of Disjoint Stable Matchings

Does there exist a marriage matching instances with disjoint stable matchings?

$m_1 : w_1, w_2, w_3$ $\qquad$ $w_1 : m_2, m_3, m_1$

$m_2 : w_2, w_3, w_1$ $\qquad$ $w_2 : m_3, m_1, m_2$

$m_3 : w_3, w_1, w_2$ $\qquad$ $w_3 : m_1, m_2, m_3$



Necessary Condition

If the man-optimal and the woman-optimal stable matchings share a common edge (m, w), then (m, w) is in every stable matching.

This is because w is both the best stable partner and the worst stable partner of m.

So, to have disjoint stable matchings, man-optimal and woman-optimal matchings must be disjoint.

Algorithm: Disjoint Stable Matchings

Algorithm 3 Disjoint Stable Matchings

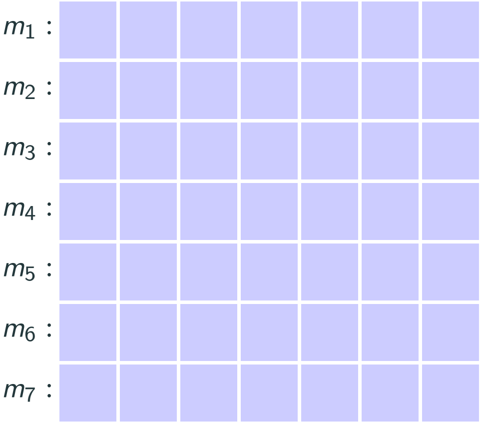
procedure FIND MAXIMUM SET OF DISJOINT STABLE MATCHINGS(M)
    $S \leftarrow \varnothing$ ▷ S: Set of disjoint matchings. Initialize S to be an empty set
    $M' \leftarrow M$.ReverseRoles ▷ Men renamed as women and women as men
    $M_z \leftarrow$FINDSTABLEMATCHING(M') ▷ GS Algorithm: Woman-optimal
    $X \leftarrow$GS-EXTENDED(M) ▷ calling Algorithm 2 modifies M's list
    while $X \cap M_z = \varnothing$ **do**
        $S \leftarrow S \cup \{X\}$
        for every man m **do**
            Delete first woman w on m's list ▷ First woman is $p_X(m)$
            Delete last man on w's list ▷ Last man is $p_X(w)$
            Mark m as *free*
        end for
        $X \leftarrow$GS-EXTENDED(M) ▷ Get a new disjoint matching as X
    end while
    $S \leftarrow S \cup \{M_z\}$
    return S
end procedure

Termination and Time Complexity

In every iteration, we delete at least one entry from the preference list. As the size of preference list is $2n^2$, the algorithm **terminates**.

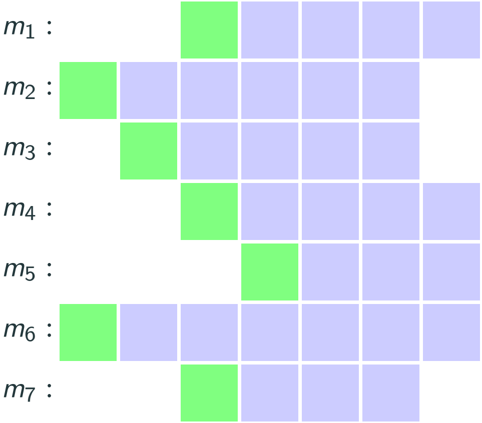
For the same reason, the running time of the algorithm is $\mathrm{O}(\mathrm{n}^2)$.

Run of Disjoint GS Algorithm



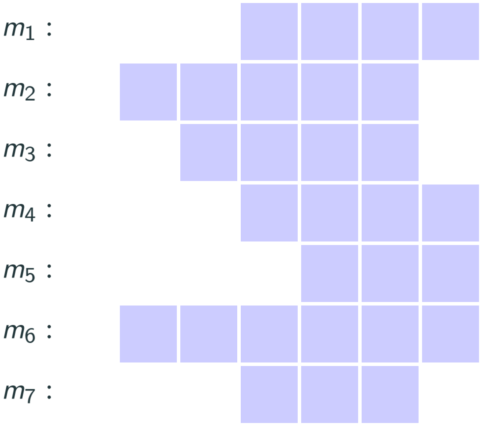
Men's preference list

Run of Disjoint GS Algorithm



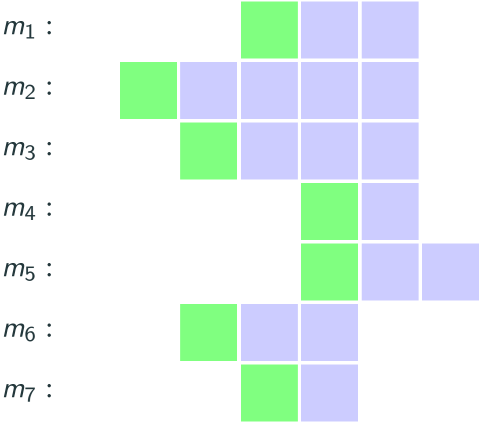
Men's Preference

Run of Disjoint GS Algorithm



Men's Preference

Run of Disjoint GS Algorithm



Men's Preference

Disjoint Stable Matchings

Lemma 1

Each M_i in the set $S = \{M_0, M_1, \cdots, M_n = M_z\}$ is a perfect matching.

Note: It does not come freely from Extended GS!

It only guarantees one-one.

Disjoint Stable Matchings

Lemma 1

Each M_i in the set $S = \{M_0, M_1, \cdots, M_n = M_z\}$ is a perfect matching.

Note: It does not come freely from Extended GS!

It only guarantees one-one.

Disjoint Stable Matchings

Lemma 2

All the matchings in the set S are stable matchings.

Lemma 3

If $M_0, M_1, \cdots, M_n = M_z$ are the matchings discovered by the algorithm 3 in this order, then $M_0 \prec M_1 \prec \cdots \prec M_n = M_z$.

Disjoint Stable Matchings

Lemma 2

All the matchings in the set S are stable matchings.

Lemma 3

If $M_0, M_1, \cdots, M_n = M_z$ are the matchings discovered by the algorithm 3 in this order, then $M_0 \prec M_1 \prec \cdots \prec M_n = M_z$.

Disjoint Stable Matchings

Lemma 4

In any arbitrary execution E of the algorithm 3, for any man m, $p_{M_i}(m)$ is the best stable partner of m when, for every *man, stable partners from $M_0, M_1, \cdots, M_{i-1}$ are disallowed.*

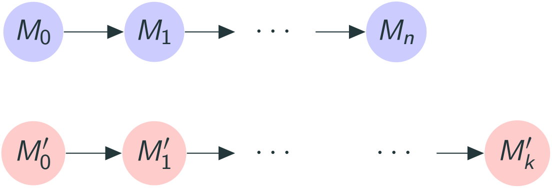
m_1 :							
m_2 :							
m_3 :							
m_4 :							
m_5 :							
m_6 :							

Longest Chain of Disjoint Stable matchings

Lemma 5

The algorithm 3 gives the longest chain of disjoint stable matchings.

Proof:

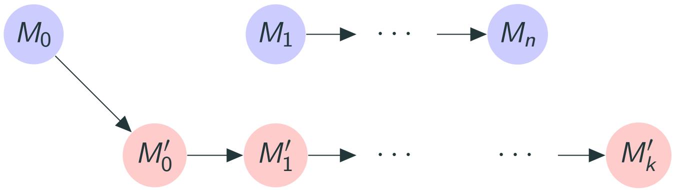


Longest Chain of Disjoint Stable matchings

Lemma 5

The algorithm 3 gives the longest chain of disjoint stable matchings.

Proof:

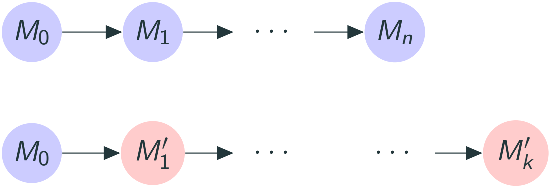


Longest Chain of Disjoint Stable matchings

Lemma 5

The algorithm 3 gives the longest chain of disjoint stable matchings.

Proof:

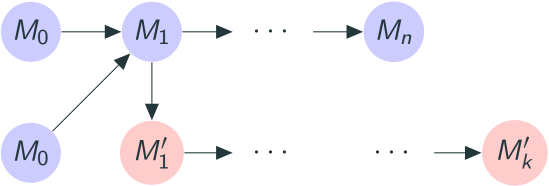


Longest Chain of Disjoint Stable matchings

Lemma 5

The algorithm 3 gives the longest chain of disjoint stable matchings.

Proof:

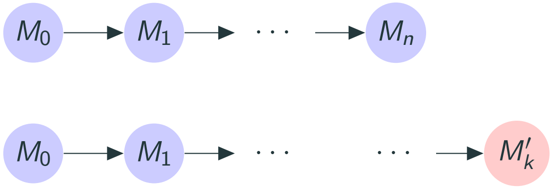


Longest Chain of Disjoint Stable matchings

Lemma 5

The algorithm 3 gives the longest chain of disjoint stable matchings.

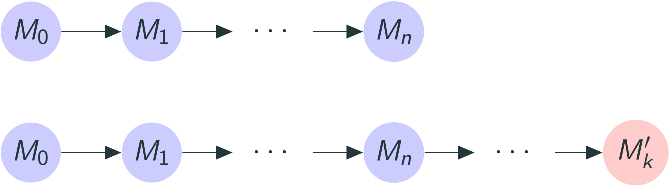
Proof:



Lemma 5

The algorithm 3 gives the longest chain of disjoint stable matchings.

Proof:



Disjoint Stable Matching

Theorem 6 (Teo, C.-P. and Sethuraman, J. (1998))

Let I be an SM instance and let $\mathcal{T}$ be a set of stable matchings in I. Let $\alpha_{j,\mathcal{T}}$ (respectively $\beta_{j,\mathcal{T}}$) denote the set of pairs obtained by assigning each man m_i (woman w_i) to $p_{j,\mathcal{T}}(m_i)$ $(p_{j,\mathcal{T}}(w_i))$, the jth element in the sorted multiset $P_{\mathcal{T}}(m_i) = \{w_i | (m_i, w_i) \in M, M \in \alpha_{j,\mathcal{T}}\}$ (respectively $P_{\mathcal{T}}(w_i)$). Then, each of $\alpha_{j,\mathcal{T}}$ and $\beta_{j,\mathcal{T}}$ is a stable matching.

Given stable matchings $M_1, M_2, \cdots, M_k$,

$$M'_i = \{(m, w) \,|\, w \text{ is the i-th women in the } \textbf{sorted multiset} \; \{p_{M_1}(m), p_{M_2}(m), \cdots, p_{M_k}(m)\}\,\}$$

Disjoint Stable Matching

Theorem 6 (Teo, C.-P. and Sethuraman, J. (1998))

Let I be an SM instance and let $\mathcal{T}$ be a set of stable matchings in I. Let $\alpha_{j,\mathcal{T}}$ (respectively $\beta_{j,\mathcal{T}}$) denote the set of pairs obtained by assigning each man m_i (woman w_i) to $p_{j,\mathcal{T}}(m_i)$ $(p_{j,\mathcal{T}}(w_i))$, the jth element in the sorted multiset $P_{\mathcal{T}}(m_i) = \{w_i | (m_i, w_i) \in M, M \in \alpha_{j,\mathcal{T}}\}$ (respectively $P_{\mathcal{T}}(w_i)$). Then, each of $\alpha_{j,\mathcal{T}}$ and $\beta_{j,\mathcal{T}}$ is a stable matching.

Given stable matchings $M_1, M_2, \cdots, M_k$,

$$M'_i = \{(m, w) \,|\, w \text{ is the i-th women in the } \textbf{sorted multiset} \; \{p_{M_1}(m), p_{M_2}(m), \cdots, p_{M_k}(m)\}\,\}$$

$$M'_1 \to M'_2 \to \cdots \to M'_q$$

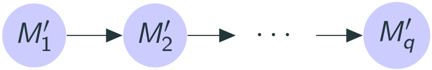
Disjoint Stable Matching

Theorem 6 (Teo, C.-P. and Sethuraman, J. (1998))

Let I be an SM instance and let $\mathcal{T}$ be a set of stable matchings in I. Let $\alpha_{j,\mathcal{T}}$ (respectively $\beta_{j,\mathcal{T}}$) denote the set of pairs obtained by assigning each man m_i (woman w_i) to $p_{j,\mathcal{T}}(m_i)$ $(p_{j,\mathcal{T}}(w_i))$, the jth element in the sorted multiset $P_{\mathcal{T}}(m_i) = \{w_i | (m_i, w_i) \in M, M \in \alpha_{j,\mathcal{T}}\}$ (respectively $P_{\mathcal{T}}(w_i)$). Then, each of $\alpha_{j,\mathcal{T}}$ and $\beta_{j,\mathcal{T}}$ is a stable matching.

Given stable matchings $M_1, M_2, \cdots, M_k$,

$$M'_i = \{(m, w) \mid w \text{ is the i-th women in the } \textbf{sorted multiset} \; \{p_{M_1}(m), p_{M_2}(m), \cdots, p_{M_k}(m)\}\}$$



Disjoint Chain

Corollary 7

Let $S = \{M_1, M_2, \cdots, M_k\}$ *be a set of* disjoint *stable matchings. Let* $\alpha_{j,S}$ *denote the stable matching obtained by matching each man* m_i *to* $p_{i,S}(m_i)$*, the* j*th woman in the sorted set* $P_S(m_i) = \{w_i | (m_i, w_i) \in M, M \in \alpha_{j,S}\}$*. Then, the stable matchings from the set* $C = \{\alpha_{1,S}, \alpha_{2,S}, \cdots, \alpha_{k,S}\}$ *forms a k-length chain* $\alpha_{1,S} \prec \alpha_{2,S} \prec \cdots \prec \alpha_{k,S}$ *of disjoint stable matchings.*

Given stable matchings $M_1, M_2, \cdots, M_k$,

$$M'_i = \{(m, w) \,|\, w \text{ is the i-th women in the } \textbf{sorted set} \; \{p_{M_1}(m), p_{M_2}(m), \cdots, p_{M_k}(m)\}\}$$

Disjoint Chain

Corollary 7

Let $S = \{M_1, M_2, \cdots, M_k\}$ *be a set of* disjoint *stable matchings. Let* $\alpha_{j,S}$ *denote the stable matching obtained by matching each man* m_i *to* $p_{i,S}(m_i)$*, the* j*th woman in the sorted set* $P_S(m_i) = \{w_i | (m_i, w_i) \in M, M \in \alpha_{j,S}\}$*. Then, the stable matchings from the set* $C = \{\alpha_{1,S}, \alpha_{2,S}, \cdots, \alpha_{k,S}\}$ *forms a* k*-length chain* $\alpha_{1,S} \prec \alpha_{2,S} \prec \cdots \prec \alpha_{k,S}$ *of disjoint stable matchings.*

Given stable matchings $M_1, M_2, \cdots, M_k$,

$M'_i = \{(m, w) \mid w$ is the i-th women in the **sorted set** $\{p_{M_1}(m), p_{M_2}(m), \cdots, p_{M_k}(m)\}\}$

Disjoint Chain

Corollary 7

Let $S = \{M_1, M_2, \cdots, M_k\}$ be a set of disjoint *stable matchings. Let $\alpha_{j,S}$ denote the stable matching obtained by matching each man m_i to $p_{i,S}(m_i)$, the jth woman in the sorted set $P_S(m_i) = \{w_i | (m_i, w_i) \in M, M \in \alpha_{j,S}\}$. Then, the stable matchings from the set $C = \{\alpha_{1,S}, \alpha_{2,S}, \cdots, \alpha_{k,S}\}$ forms a k-length chain $\alpha_{1,S} \prec \alpha_{2,S} \prec \cdots \prec \alpha_{k,S}$ of disjoint stable matchings.*

Given stable matchings $M_1, M_2, \cdots, M_k$,

$$M'_i = \{(m, w) \,|\, w \text{ is the i-th women in the } \textbf{sorted set} \; \{p_{M_1}(m), p_{M_2}(m), \cdots, p_{M_k}(m)\}\}$$

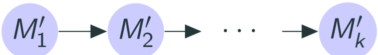
Disjoint Chain

Corollary 7

Let $S = \{M_1, M_2, \cdots, M_k\}$ *be a set of* disjoint *stable matchings. Let* $\alpha_{j,S}$ *denote the stable matching obtained by matching each man* m_i *to* $p_{i,S}(m_i)$*, the* j*th woman in the sorted set* $P_S(m_i) = \{w_i | (m_i, w_i) \in M, M \in \alpha_{j,S}\}$*. Then, the stable matchings from the set* $C = \{\alpha_{1,S}, \alpha_{2,S}, \cdots, \alpha_{k,S}\}$ *forms a* k*-length chain* $\alpha_{1,S} \prec \alpha_{2,S} \prec \cdots \prec \alpha_{k,S}$ *of disjoint stable matchings.*

Given stable matchings $M_1, M_2, \cdots, M_k$,

$$M'_i = \{(m, w) \mid w \text{ is the i-th women in the } \textbf{sorted set} \ \{p_{M_1}(m), p_{M_2}(m), \cdots, p_{M_k}(m)\}\}$$



Maximum Size Set of Disjoint Stable Matchings

Theorem 8

For a given stable marriage instance, algorithm 3 gives the maximum size set of disjoint stable matchings.

Rotations

Rotations

For every stable matching M, we define the following:

$s_M(m)$

For any man m, let $s_M(m)$ denote the first women w on m's list such that w strictly prefers m to $p_M(w)$

$next_M(m)$

For any man m, let $next_M(m)$ denote $p_M(s_M(m))$

Note: $s_M(m)$ might not exist. Example: W_z.
Both $s_M(m)$ and $next_M(m)$ can be easily be found using *Reduced Lists*.

Roations

Definition of Rotations

An ordered list of matched pairs $\rho = (m_0, w_0), (m_1, w_1), \cdots, (m_{r-1}, w_{r-1})$ in a stable matching M is called as a rotation *exposed* in M if for each i $(0 \le i \le r-1)$, m_{i+1} is $next_M(m_i)$ where $i+1$ is taken modulo r.

Elimination of a Rotation

If M is a stable matching and $\rho = (m_0, w_0), (m_1, w_1), \cdots, (m_{r-1}, w_{r-1})$ is a rotation exposed in M, then M/æ is defined to be matching in which each man who is not in ρ stays married to his partner in M, and each man m_i in M is matched to $w_{i+1} = s_M(m_i)$

That is, M/ρ differs from M by one place cyclic shift of each men in ρ

Roations

Definition of Rotations

An ordered list of matched pairs $\rho = (m_0, w_0), (m_1, w_1), \cdots, (m_{r-1}, w_{r-1})$ in a stable matching M is called as a rotation *exposed* in M if for each i $(0 \leq i \leq r-1)$, m_{i+1} is $next_M(m_i)$ where $i+1$ is taken modulo r.

Elimination of a Rotation

If M is a stable matching and $\rho = (m_0, w_0), (m_1, w_1), \cdots, (m_{r-1}, w_{r-1})$ is a rotation exposed in M, then M/æ is defined to be matching in which each man who is not in ρ stays married to his partner in M, and each man m_i in M is matched to $w_{i+1} = s_M(m_i)$

That is, M/ρ differs from M by one place cyclic shift of each men in ρ

Representation

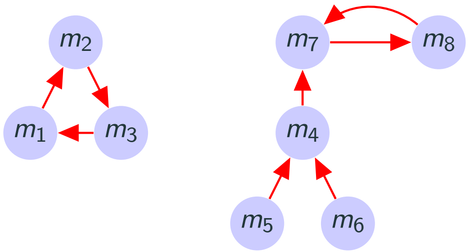


Figure 1: Graph $H(M)$

$(m_i, m_j) \in E(H(m))$ if $m_j = next_M(mi)$.

Properties of Rotations

- M/ρ is a stable matching such that $M \preceq M/\rho$
- Every stable matching except the women optimal matching has at least one rotation exposed in it.
- Every path from M_0 to M_z in $\mathcal{M}$ corresponds to some permutation of set of all rotations.

Properties of Rotations

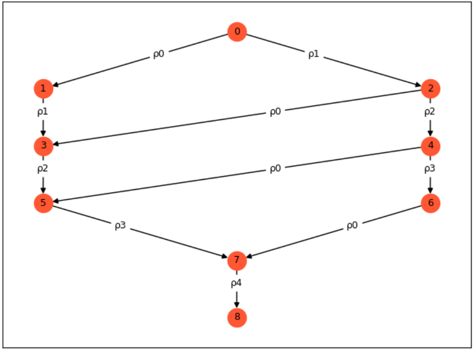
- M/ρ is a stable matching such that $M \preceq M/\rho$
- Every stable matching except the women optimal matching has at least one rotation exposed in it.
- Every path from M_0 to M_z in $\mathcal{M}$ corresponds to some permutation of set of all rotations.

Properties of Rotations

- M/ρ is a stable matching such that $M \preceq M/\rho$
- Every stable matching except the women optimal matching has at least one rotation exposed in it.
- Every path from M_0 to M_z in $\mathcal{M}$ corresponds to some permutation of set of all rotations.

Properties of Rotations

- M/ρ is a stable matching such that $M \preceq M/\rho$
- Every stable matching except the women optimal matching has at least one rotation exposed in it.
- Every path from M_0 to M_z in $\mathcal{M}$ corresponds to some permutation of set of all rotations.



poset of rotations

The set of all rotations forms a partial order under the following relation.

$\rho_1 \prec \rho_2$ iff in *every* path from M_0 to M_z in $\mathcal{M}$ ρ_1 gets eliminated before ρ_2.

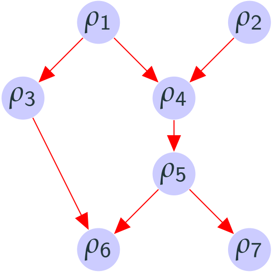


Figure 2: The Rotational Poset $\Pi((M))$

Theorem 9

There is a one-one correspondence between the closed subsets of $\Pi((M))$ *and stable matchings in* (M)

Theorem 10

S is a closed set of rotations of $\Pi((M))$ *corresponding to a stable matching M iff S is the (unique) set of rotations in every* M_0-M *chain in* (M)

Theorem 9

There is a one-one correspondence between the closed subsets of $\Pi((M))$ and stable matchings in (M)

Theorem 10

S is a closed set of rotations of $\Pi((M))$ corresponding to a stable matching M iff S is the (unique) set of rotations in every M_0-M chain in (M)

Algorithm based on $\Pi((M))$

Given a stable matching M_i, we can find the next best disjoint stable matching M_{i+1} by doing the following steps.

1. For every $(m, w) \in M_i$, find $R = \{\rho | (m, w) \in \rho\}$
2. Find $\hat{R}$ = closure of R
3. Return $M_{i+1} = M_i / \hat{R}$.

We use the above steps recursively (until we find a matching that intersects with M_z) to get the maximum chain of disjoint stable matchings.

Algorithm based on $\Pi((M))$

Given a stable matching M_i, we can find the next best disjoint stable matching M_{i+1} by doing the following steps.

1. For every $(m, w) \in M_i$, find $R = \{\rho | (m, w) \in \rho\}$
2. Find $\hat{R}$ = closure of R
3. Return $M_{i+1} = M_i / \hat{R}$.

We use the above steps recursively (until we find a matching that intersects with M_z) to get the maximum chain of disjoint stable matchings.

Algorithm based on $\Pi((M))$

Given a stable matching M_i, we can find the next best disjoint stable matching M_{i+1} by doing the following steps.

1. For every $(m, w) \in M_i$, find $R = \{\rho | (m, w) \in \rho\}$
2. Find $\hat{R}$ = closure of R
3. Return $M_{i+1} = M_i / \hat{R}$.

We use the above steps recursively (until we find a matching that intersects with M_z) to get the maximum chain of disjoint stable matchings.

Algorithm based on $\Pi((M))$

Given a stable matching M_i, we can find the next best disjoint stable matching M_{i+1} by doing the following steps.

1. For every $(m, w) \in M_i$, find $R = \{\rho | (m, w) \in \rho\}$
2. Find $\hat{R}$ = closure of R
3. Return $M_{i+1} = M_i / \hat{R}$.

We use the above steps recursively (until we find a matching that intersects with M_z) to get the maximum chain of disjoint stable matchings.

Algorithm based on $\Pi((M))$

Given a stable matching M_i, we can find the next best disjoint stable matching M_{i+1} by doing the following steps.

1. For every $(m, w) \in M_i$, find $R = \{\rho | (m, w) \in \rho\}$
2. Find $\hat{R}$ = closure of R
3. Return $M_{i+1} = M_i / \hat{R}$.

We use the above steps recursively (until we find a matching that intersects with M_z) to get the maximum chain of disjoint stable matchings.

Future Work

- Disjoint Stable Matchings in the Roommate problem.
- When disjoint stable matchings do not exist, minimize pairwise intersection.

Thank You!