The Stable Marriage Problem and the Complexity of Counting Stable Matchings

Master Thesis

Despina Bourou

Supervisor:
Mihalis Kolountzakis

Department of Mathematics & Applied Mathematics
University of Crete
Greece

September 2022

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέπων κ. Μιχάλη Κολουντζάκη για την καθοδήγηση και τη βοήθεια του κατά τη διάρκεια αυτής της εργασίας.

Επίσης θα ήθελα να ευχαριστήσω την κα. Μαρία Λουκάκη και τον κ. Θεόδουλο Γαρεφαλάκη που αποτέλεσαν μέλη της επιτροπής αξιολόγησης καθώς και για την καθοδήγηση τους κατά τη διάρκεια των σπουδών μου.

Τέλος, ευχαριστώ την οικογένεια μου για τη στήριξή τους καθ'όλη τη διάρκεια των σπουδών μου.

Στη γιαγιά μου, Βαγγελιώ

Contents

Preface

The Stable Marriage Problem was first introduced and solved by David Gale and Lloyd Shapley in their paper entitled "College Admissions and the Stability of Marriage" in 1962 [1]. The problem is formulated as follows: Suppose we have 2 finite equally sized sets M, the set of men, and W, the set of women, such that each element of M has a preference list of all the elements in W and, similarly, each element of W has a preference list of all the elements in M. Then, a stable matching or stable marriage is a bijection of M onto W such that there are no $m \in M$ and $w \in W$ that prefer each other to their assigned partners in the matching.

In [1] Gale and Shapley also introduced a polygamous version of the problem and presented an algorithm for finding stable matchings in both the polygamous and monogamous versions. On the practical side, the polygamous version of the Gale-Shapley algorithm has been used for many years, in fact preceding the Gale-Shapley paper, to assign "job applicants" to "job positions". The best known amongst them is the placement of resident doctors in training programs. Moreover, since the publication of their paper several additional papers have been published studying different variations of the problem.

In 2012, Lloyd S. Shapley and Alvin E. Roth were awarded a Nobel Memorial Prize in Economic Sciences for "the theory of stable allocations and practice of market design".

In this thesis, we are doing a review of literature on the stable marriage problem, starting by presenting the Gale-Shapley algorithm and its correctness and proceeding to also present a number of different results and variations of the problem.

More specifically, we start by formally presenting the stable marriage problem along with some basic notations, terminologies and definitions about the stable marriage problem and complexity classes that will be used throughout the thesis. In section 2, we firstly present the Gale-Shapley algorithm along with a proof of the algorithm's correctness and a number of consequences of the algorithm, such as the fact that the matching obtained by the algorithm is male-optimal and woman-pessimal. We also study the expected value of the number of proposals during the course of the algorithm for arbitrary but fixed lists of preferences of the women when the lists of men are constructed independently and uniformly at random. In section 3 we present 2 variations of the problem: The problem of stable matchings with incomplete lists and the problem of admitting n students to m universities. Next, we study the set of stable matchings, proving that is has in fact a distributive lattice structure. Furthermore, we prove that the number of stable matchings can grow exponentially with the size of the instance, meaning that any algorithm generating all of the stable matchings could be of exponential time complexity in the worst case. In section 5, continuing our study of the number of stable matchings we prove that the problem of determining the number of stable matchings is $\#P-$complete and therefore cannot be solved in polynomial time if $P \neq NP$. In the process of doing that, we introduce the concept of rotations presenting also a number of results concerning the structure of the set of stable matchings.

Εισαγωγικό σημείωμα

Το πρόβλημα του Σταθερού Γάμου εισήχθηκε για πρώτη φορά και λύθηκε από τους David Gale και Lloyd Shapley στο paper τους με τίτλο "College Admissions and the Stability of Marriage" το 1962 [1]. Το πρόβλημα διατυπώνεται ως εξής: Υποθέτουμε πως έχουμε 2 πεπερασμένα σύνολα ίσου μεγέθους, το σύνολο M των ανδρών και το σύνολο W των γυναικών τέτοια ώστε κάθε στοιχείο του M έχει μια λίστα προτίμησης με όλα τα στοιχεία του W και, όμοια, κάθε στοιχείο του W έχει μια λίστα προτίμησης με όλα τα στοιχεία του M. Τότε, ένα σταθερό ταίριασμα ή αλλιώς σταθερός γάμος είναι μία 1-1 απεικόνιση από το M επί του W τέτοια ώστε να μην υπάρχει άντρας $m \in M$ και γυναίκα $w \in W$ που προτιμούν ο ένας τον άλλον από ότι τους "συζύγους" που τους έχουν ανατεθεί στο ταίριασμα.

Στο [1] οι Gale και Shapley εισήγαγαν επίσης μία πολυγαμική εκδοχή του προβλήματος και παρουσίασαν έναν αλγόριθμο για την εύρεση σταθερού ταιριάσματος τόσο στην μονογαμική όσο και στην πολυγαμική εκδοχή. Μάλιστα, η πολυγαμική εκδοχή του αλγορίθμου των Gale και Shapley χρησιμοποίουνταν για πολλά χρόνια, στην πραγματικότητα πριν απο το paper των Gale-Shapley, για την αντιστοίχηση "αιτούντων για δουλειά" σε "θέσεις εργασίας". Η πιο γνωστή εφαρμογή τέτοιων αντιστοιχίσεων είναι η τοποθέτηση φοιτητών ιατρικής σε νοσοκομεία. Επιπλέον, από την δημοσίευση του paper τους, έχουν δημοσιευθεί αρκετά επιπλέον paper που μελετούν διάφορες παραλλαγές του αρχικού προβλήματος.

Το 2012, απονεμήθηκε στους Lloyd S. Shapley και Alvin E. Roth το βραβείο Νόμπελ Οικονομικών Επιστημών για την "θεωρία των σταθερών κατανομών και την εφαρμογή του σχεδιασμού αγοράς".

Σε αυτή την εργασία, μέσω μιας ανασκόπησης της βιβλιογραφίας, ξεκινάμε παρουσιάζοντας τον αλγόριθμο των Gale και Shapley και την απόδειξη της ορθότητάς του και συνεχίζουμε με την παρουσίαση διαφόρων αποτελεσμάτων και παραλλαγών του προβλήματος.

Πιο συγκεκριμένα, ξεκινάμε ορίζοντας αυστηρά το πρόβλημα του γάμου και παρουσιάζουμε κάποιους βασικούς συμβολισμούς, ορολογίες και ορισμούς σχετικά με το πρόβλημα του γάμου αλλά και τις κλάσεις πολυπλοκότητας που θα μας φανούν χρήσιμοι κατά τη διάρκεια της εργασίας. Στη δεύτερη ενότητα, ξεκινάμε με τον αλγόριθμο των Gale και Shapley, συνεχίζοντας με την απόδειξη της ορθότητάς του καθώς και με διάφορες ιδιότητες του ταιριάσματος του αλγορίθμου, όπως για παράδειγμα το γεγονός πως το ταίριασμα που προκύπτει από τον αλγόριθμο είναι το βέλτιστο για τους άντρες και το χειρότερο για τις γυναίκες. Μελετάμε ακόμη την μέση τιμή των προτάσεων κατά τη διάρκεια του αλγορίθμου για αυθαίρετες αλλά δοσμένες λίστες προτίμησης των γυναικών όταν οι λίστες προτίμησης των ανδρών κατασκευάζονται ανεξάρτητα και ομοιόμορφα στην τύχη. Στην ενότητα 3 παρουσιάζουμε 2 παραλλαγές του προβλήματος: το πρόβλημα του σταθερού γάμου όταν οι λίστες προτίμησης είναι ανολοκλήρωτες καθώς και το πρόβλημα εισόδου n φοιτητών σε m πανεπιστήμια. Στη συνέχεια, μελετάμε το σύνολο των σταθερών ταιριασμάτων αποδεικνύοντας πως έχει τη δομή distributive lattice. Έπειτα αποδεικνύουμε πως ο αριθμός των σταθερών ταιριασμάτων είναι δυνατό να μεγαλώνει εκθετικά με το μέγεθος του προβλήματος, που σημαίνει πως οποιοσδήποτε αλγόριθμος παράγει όλα τα σταθερά ταιριάσματα σε ένα δοσμένο στιγμιότυπο του προβλήματος θα μπορούσε να είναι εκθετικού χρόνου στη χειρότερη περίπτωση. Στο κεφάλαιο 5, συνεχίζοντας τη μελέτη του πλήθους των σταθερών ταιριασμάτων, αποδεικνύουμε ότι το πρόβλημα της εύρεσης του αριθμού των σταθερών ταιριασμάτων σε ένα πρόβλημα γάμου είναι $\#P-$πλήρες και, ως αποτέλεσμα, δε μπορεί να λυθεί σε πολυωνυμικό χρόνο αν $P \neq NP$. Για να το αποδείξουμε αυτό, εισάγουμε την έννοια των περιστροφών παρουσιάζοντας παράλληλα ένα πλήθος διαφορετικών αποτελεσμάτων σχετικά με τη δομή του συνόλου σταθερών ταιριασμάτων.

1 Preliminaries

1.1 Stable matchings: an introduction

Definition 1.1. *Let M and W be two finite sets of n elements. A matching is a bijection of M onto W.*

Definition 1.2. *An instance of size n of the stable marriage problem involves two disjoint sets, the set M of men $\{m_1, ..., m_n\}$ and the set W of women $\{w_1, ..., w_n\}$. Each person has a strictly ordered preference list of the members of the opposite sex.*

Definition 1.3. *A matching in a stable marriage problem instance is called unstable if a man m and a woman w mutually prefer each other to their spouses.*

Definition 1.4. *A matching is called stable if there is no man m and no woman w that both prefer each other to their assigned partners.*

The stable marriage problem asks for a stable matching in a given instance of the problem. Naturally occurs the question of whether there is always a stable matching in a given stable marriage instance. In the next chapter we will prove, through Gale-Shapley's algorithm, that indeed there is always at least one stable matching. In fact, through the following example we can see that it is possible that more than one stable matchings exist.

Example 1.5. *Suppose we have the sets $M = \{m_1, m_2\}$ and $W = \{w_1, w_2\}$ of men and women respectively, along with the following preference lists:*

men's preference lists			*women's preference lists*		
m_1 :	w_1	w_2	w_1 :	m_2	m_1
m_2 :	w_2	w_1	w_2 :	m_1	m_2

Here we have 2 possible matchings:
M_1=\{ $(m_1,w_1),(m_2,w_2)$\} *and* M_2=\{$(m_1,w_2),(m_2,w_1)$\}
and we can easily check that, in fact, both of them are stable.

We now introduce some basic notations and terminology that will be used throughout the next chapters.

Notation 1.6.

- *Let m be a man and w_1, w_2 be 2 distinct women. Then* $w_1 m w_2 \iff$ *man m prefers woman w_1 to woman w_2.*

- *Similarly, if w is a woman and m_1, m_2 are 2 distinct men then* $m_1 w m_2 \iff$ *woman w prefers man m_1 to man m_2*

Notation 1.7.

- $mr(i,k) = j \iff$ *woman k is in position j in the preference list of man i*

- $wr(i,k) = j \iff$ *man k is in position j in the preference list of woman i*

Notation 1.8.

- $mp(i,j) = k \iff$ *woman k is in position j in the preference list of man i*

- $wp(i,j) = k \iff$ *man k is in position j in the preference list of woman i*

Notation 1.9. $p_M(x) = y \iff$ *person's x partner in matching M is person y.*

Terminology 1.10. *We say that a person x prefers a matching M to a matching M' and we write MxM' if he prefers his partner in M to his partner in M', meaning that his partner in M is higher in his preference list than his partner in M'. Note that this is strict preference. Generally, a person might prefer a matching M to M' or M' to M or he/she might be indifferent between them if $p_M(x) = p_{M'}(x)$.*

Terminology 1.11. *We say that a man m and a woman w block a matching M or that m-w are a blocking pair for M if m, w are not partners in M but they prefer each other to their assigned partners in the matching, that is $wmp_M(m)$ and $mwp_M(w)$. Obviously, a matching for which there is at least one blocking pair is not stable, whereas a matching for which there are no blocking pairs is stable.*

1.2 Computational complexity

In this section we present a few basic definitions and results about complexity classes that will be used in chapter 5.

Generally, an instance of a problem is obtained by specifying all problem parameters and any problem Π is associated with a set D_Π of instances.

Definition 1.12. *A decision problem is a problem that can be answered by "yes" or "no".*

Definition 1.13. *P is the class of decision problems that can be solved by an algorithm running in time polynomial in the size of the input.*

For example, the problem of deciding whether 2 numbers a, b are relatively prime is in class P: Using the euclidean algorithm, which runs in polynomial time, we can find the greatest common divisor of a and b denoted by $gcd(a, b)$. If $gcd(a, b) = 1$ then a, b are relatively prime. Otherwise, a, b are not relatively prime.

Definition 1.14. *NP is the class of decision problems such that, given a "yes" instance of the problem and a polynomial-size "certificate" of a solution, there is a polynomial-time algorithm verifying that the certificate is correct.*

For example let's consider the problem of examining whether a number is composite. Given a composite number n and one of its divisors as a certificate, we can check in polynomial time through the euclidean algorithm that n is indeed composite.
Obviously $P \subseteq NP$ since if a problem is in P we can solve it in polynomial time without even being supplied a certificate.

Definition 1.15. *A decision problem D_1 is said to be polynomial-time reducible to a decision problem D_2 if there exists a function $f : D_{D_1} \to D_{D_2}$ that transforms an instance d_1 of D_1 to an instance d_2 of D_2 such that:*

- *The transformation takes polynomial time*
- *d_1 is a "yes" instance of D_1 $\iff$ d_2 is a "yes" instance of D_2*

Definition 1.16. *A problem Π is said to be $NP-$complete if:*

- *Π is in NP*
- *every other problem in NP is polynomial-time reducible to Π*

Informally, that means that if a problem Π is $NP-$complete then it is at least as "hard" as any other problem in NP.
An immediate result of definitions 1.15 and 1.16 is the following:

Theorem 1.17. *If a problem B is $NP-$complete and B is polynomial-time reducible to A for a problem A in NP then A is also $NP-$complete.*

A more general class of problems is the class of "search problems". A search problem Π consists of a set D_Π of instances and, for each instance $I \in D_\Pi$, a set $S_\Pi(I)$ of finite objects called solutions for I. An algorithm is said to solve a search problem Π if, given as an input any instance $I \in D_\Pi$, it returns the answer "no" if $S_\Pi(I) = \emptyset$ and otherwise returns some solution $s \in S_\Pi(I)$ (the corresponding decision problem asks whether or not $S_\Pi(I)$ is empty).

The counting problem based on a search problem Π is, "Given I what is the cardinality of $S_\Pi(I)$ that is, how many solutions are there?".

Definition 1.18. *$\#P$ is the class of all counting problems associated with the decision problems in the class NP.*

The concept of "completeness" for $\#P$ is once again used to capture the notion of a "hardest" problem in a class.

Definition 1.19. *If A and B are problems in $\#P$, a polynomial time counting reduction from A to B is a pair of polynomial-time computable functions f, g such that*

- *function f maps an instance x of A to an instance $f(x)$ of B and*
- *function g recovers from the number of solutions n of $f(x)$ the number of solutions $g(n)$ of x.*

A special case of polynomial-time counting reductions are parsimonious reductions, where f preserves the number of solutions during the transformation, that is g is the identity function. Thus, a parsimonious reduction from a problem A to a problem B is a polynomial-time computable function that maps an instance x of A to an instance y of B such that x and y have the same number of solutions.

We say that a problem A is $\#P-$complete if it is in $\#P$ and for every other problem B in $\#P$, there is a polynomial-time counting reduction from B to A.

2 Existence of a stable matching: The Gale-Shapley Algorithm

A fundamental algorithm, leading to the construction of a stable matching, was developed by Gale and Shapley in 1962, constituting a proof by construction of the existence of at least one stable matching in any given instance of the problem.

2.1 Description of the algorithm

The idea of the algorithm is as follows: Suppose that initially no one is married. Lets suppose that a man m proposes to a woman w. It would be a mistake to say that the pair (m, w) would surely belong to the final stable matching since the woman w might receive a proposal from another man who she prefers more. It would also be a mistake to rule out the possibility that the pair (m, w) is in the final matching, as the woman w might not receive another proposal by a man she ranks as highly as m. As a result we introduce an intermediate state, the engagement.

Algorithm 1 Gale-Shapley algorithm for stable matchings

```
initially all women and all men are free
while there is a man who is free do
    choose a man m that is free
    w ← first woman on m's list to whom m has not yet proposed
    if w is free then
        engage m and w
    else if w is already married to a man m′ then
        if w prefers m to m′ then
            engage m and w
            m′ becomes free
        else
            m remains free
        end if
    end if
end while
```

Observation 2.1. *The algorithm above involves an element of non determinism since the order in which the free men propose is not specified. However, as we will see, this is of no consequence: the order in which the men* *propose is of no importance to the final outcome, which is always the same for a specific stable marriage instance.*

Note 2.2. *The algorithm presented in Gale and Shapley's paper was running in stages: In the first stage, all men propose to their favorite woman. Each woman that receives more than one proposals rejects all but her favorite man. However, she does not accept him yet but keeps him on a string to allow for the possibility that someone better might propose to her later. In the second stage, the rejected men propose to their second choices, the women reject all but their favorite amongst the group of the new proposers and the boy at the string-if any- and again keeps her favorite in a string and we continue in a similar manner. In [7], McVitie and Wilson presented a recursive version of the Gale-Shapley algorithm, which was essentially the* ***Algorithm 1*** *presented above with the difference that every time, if there is a man that is rejected he is the next one to propose. All algorithms give the same result and the same proposals and rejections are made (although in general in different order).*

2.2 Proof of the algorithm's correctness

Theorem 2.3. *The algorithm terminates.*

Proof. The algorithm would not terminate if the while loop did not terminate, that is if there was a man m that always remained free. m would always remain free if and only if he got rejected by all the women in his list.

Each woman can reject only when she is engaged and once she is engaged she never again becomes free. So when m got rejected by the last woman on his preference list that would mean that all n women were already engaged. But since the number of men is equal to the number of women and there are no men married to 2 women then all n men would also have to be engaged, which is a contradiction. In fact, for an instance of size n, the total number of iterations cannot exceed n^2. □

Theorem 2.4. *The matching obtained from the algorithm is stable.*

Proof. Suppose that the matching M obtained from the algorithm is not stable, that is there is a man m and a woman w such that they both prefer each other to their partners assigned by the algorithm. Suppose that m's partner in M is w_1.

Since man m prefers woman w to w_1 that means that woman w is higher in his preference list than w_1. However, m did not end up marrying w but w_1 who is lower in his preference list, meaning that, during the course of the algorithm, m proposed to w but w rejected him for someone else that she preferred more. As a result, w prefers her current spouse to m, which is a contradiction. $\square$

We have thus proved that the algorithm yields a stable matching. However, we have seen that it is possible that more than 1 stable matchings exist. A natural question is whether all executions of the algorithm yield the same results. The answer is yes and, in fact, every execution of the algorithm gives the optimal result for each man, meaning that there is no other stable matching in which he is paired with a woman that is ranked higher in his preference list than the one he is assigned to by the algorithm. We prove this in Theorem 2.7 below.

At the same time, this result constitutes the worst solution for the women, meaning that in every other possible stable matching, each woman is assigned to a man that is equal or superior in her preference list to the one she is assigned to by the algorithm. We prove this result in Theorem 2.8 below.

Definition 2.5. *A woman w is called a valid partner for a man m if there is a stable matching containing the pair (m, w). Similarly, a man m is called a valid partner for a woman w if there is a stable matching containing the pair (m, w).*

Definition 2.6. *We say that a woman w is the best valid partner for a man m if every other valid partner of m is after w in the preference list of m, whereas a man m is the worst valid partner for a woman w if m is after any other valid partner of w in the preference list of w.*

Theorem 2.7. *The matching obtained by the algorithm assigns each man to his best valid partner.*

Proof. Suppose that there is an execution of the algorithm such that, in the matching M obtained by the algorithm, at least one man is not married to his best valid partner. That means that at some point during the course of the algorithm he was rejected by a valid partner.

Consider the first moment t_1 in the algorithm in which a man m is rejected by a valid partner w. The woman w rejects m at the moment t_1 either because she was already married to another man she prefers more or because she was married to m but, at the moment t_1, another man who she prefers more proposes to her so she breaks up with m. Either way, at the moment t_1, w forms or continues an engagement with a man m' she prefers more than m, that is $m'wm$. That means that w got engaged to m' at the moment t_1 or earlier, that is, if t_0 is the moment when w got engaged to m', then $t_0 \leq t_1$. As a result, at t_0, m' had not been rejected by any valid partner (because we defined t_1 to be the first moment in the algorithm in which a man m is rejected by a valid partner).

Moreover, since w is a valid partner for m there exists a stable matching N containing the pair (m, w). Suppose that in N, m' is paired with a woman w'. Thus, at t_0, m' had not been rejected by w' since w' is obviously a valid partner for m'. Since the proposals in the algorithm happen in decreasing order of preference, that means that $wm'w'$. Thus we have $wm'w'$ and $m'wm$ which is a contradiction since we supposed that the matching N is stable. □

Theorem 2.8. *The matching obtained by the algorithm assigns each woman to her worst valid partner.*

Proof. Suppose that there is a woman w that is not assigned to her worst valid partner by the algorithm. That is, if (m, w) is a pair in the matching M obtained by the algorithm, there is a stable matching N containing the pairs (m', w) and (m, w') such that mwm'. Thus $w'mw$ (or else the matching N would not be stable). That contradicts the fact that the matching M of the algorithm yields the best possible solution for m. □

Observation 2.9. *If we swap the roles of women and men, that is if women make the proposals instead of men, then a woman-optimal and man-pessimal stable matching would occur.*

In fact, the man-optimal and woman-pessimal stable solutions possess a further optimality property, often referred to as Weak Pareto Optimality:

Theorem 2.10 (Weak Pareto Optimality)**.** *For a given instance of the stable marriage problem, there is no matching, stable or unstable, in which every man has a partner whom he strictly prefers to his partner in the matching M obtained by the algorithm.*

Proof. Obviously, from theorem 2.7, that is true for all stable matchings. Suppose now that there is an unstable matching M_0 with this property, meaning that every man prefers his partner in M_0 to his partner in M. Observe that the Gale-Shapley algorithm terminates when the last woman w receives her first proposal (since she was the last to receive a proposal, that means all the other $n-1$ women are already married to $n-1$ different men and since that was the first proposal she received that means she was free before so she now becomes engaged to the $n-th$ man and the algorithm terminates). As a result, no man was rejected by w during the course of the algorithm. Let m and m_0 be w's partners in the matchings M and M_0 respectively. Then m_0 prefers w to his partner in M so w must have rejected him during the course of the algorithm, which is a contradiction. $\square$

2.3 Conflict of Interest

The fact that the best stable matching for the men is the worst for the women is a special case of the following more general result.

Theorem 2.11. *Let M and M' be two stable matchings and suppose that m and w are partners in M but not in M'. Then one of m and w prefers M to M' and the other prefers M' to M. In other words, if w' is the partner of m in M' and m' is the partner of w in M' then either*

$$wmw' \text{ and } m'wm$$

$$or$$

$$w'mw \text{ and } mwm'.$$

Proof. By the definition of a stable matching, m and w cannot both prefer M to M' (or else in M' where we have the pairs (m, w') and (m', w) we would have mwm' and wmw' thus m and w would mutually prefer each other to their spouses in M' which is a contradiction). It remains to show that they also cannot both prefer M' to M.

Suppose $w'mw$, that is m prefers M' to M. It remains to show that mwm'. Let $m = X_0$, $w = x_0$ and $w' = x_1$ so we have $x_1X_0x_0$. Since m prefers M' to M, w' must prefer M to M' (or else M would not be stable). Let X_1 be the partner of w' in M. Then $X_1x_1X_0$. Similarly, since w' prefers M to M', X_1 must prefer M' to M (or else M' would not be stable). Let x_2 be X_1's spouse in M', then $x_2X_1x_1$.
Continuing in a similar manner, we obtain the sequence
(X_0, x_0), (X_1, x_1), (X_2, x_2) etc in M and
(X_0, x_1), (X_1, x_2), (X_2, x_3) etc in M'
where $X_{k+1}x_{k+1}X_k$ and $x_{k+1}X_kx_k$ $\forall k \geq 0$

Since the number of people is finite, there exist X_i and X_j with $i \neq j$, let $i > j$, such that $X_i = X_j$. Let j be the smallest integer with that property and, for that j, let i be the smallest integer with that property such that $i > j$. Since $X_i = X_j$, we have $x_i = x_j$. Furthermore, we have $j = 0$ because, if $j \neq 0$, then (X_{j-1}, x_j) would belong in M' along with $(X_{i-1}, x_i) = (X_{i-1}, x_j)$. Hence we would have $X_{i-1} = X_{j-1}$ which is a contradiction since we assumed that j is the smallest integer with that property.

Thus $j = 0$ and so $x_i = x_0 = w$ and (X_{i-1}, x_0) appears in the matching M'. As a result, $X_{i-1} = m'$ and $X_i = m$. Thus, since $X_i x_i X_{i-1}$, we have that mwm', which is what we wanted to prove. $\square$

We now also present a second proof.

Proof. Let $\mathcal{X}$ and $\mathcal{Y}$ (respectively $\mathcal{X}'$ and $\mathcal{Y}'$) denote the sets of men and women that prefer M to M' (respectively M' to M).

In M there can be no pair (m, w) with m in $\mathcal{X}$ and w in $\mathcal{Y}$ because then, both m and w would prefer matching M to M' meaning that, both m and w would prefer each other to their assigned partners in M', leading to a contradiction since we supposed that M' is stable. As a result, every man in $\mathcal{X}$ has an $M-$partner in $\mathcal{Y}'$ and, as a result, $|\mathcal{X}| \leq |\mathcal{Y}'|$. In other words, if wmw' then $m'wm$. Now all we have to prove is that if $w'mw$ then mwm' or, in other words, that every man in $\mathcal{X}'$ has an $M-$partner in $\mathcal{Y}$.

Similarly as before, in M' there can be no pair (m_1, w_1) with m_1 in $\mathcal{X}'$ and w_1 in $\mathcal{Y}'$ because then, both m_1 and w_1 would prefer M' to M meaning that, both m_1 and w_1 would prefer each other to their assigned partner in M which is a contradiction. As a result, every man in $\mathcal{X}'$ has an $M'-$partner in $\mathcal{Y}$ and, as a result, $|\mathcal{X}'| \leq |\mathcal{Y}|$.

We also have that $|\mathcal{X}| + |\mathcal{X}'| = |\mathcal{Y}| + |\mathcal{Y}'|$ since the left hand side is the number of men and the right hand side is the number of women that have a different partner in the two matchings. As a result, $|\mathcal{X}| = |\mathcal{Y}'|$ and $|\mathcal{X}'| = |\mathcal{Y}|$ or else we would have $|\mathcal{X}| + |\mathcal{X}'| < |\mathcal{Y}| + |\mathcal{Y}'|$ which is a contradiction. Thus, every man in $\mathcal{X}'$ has an $M-$partner in $\mathcal{Y}$ (because if a man in $\mathcal{X}'$ had an $M-$partner in $\mathcal{Y}'$ then $|\mathcal{Y}'| > |\mathcal{X}|$ which is a contradiction) and the statement of the theorem follows. $\square$

2.4 Analysis of the algorithm: Expected value of the number of proposals

We are interested in performing an average-case analysis of the Gale-Shapley algorithm. We suppose that the lists of men are chosen independently and uniformly at random, whereas the lists of women can be arbitrary but are fixed in advance. Let N the random variable expressing the total number of proposals made by the men during the course of the algorithm. It is clear that the running time of the algorithm is proportional to N. We want to study the expected value $\mathbb{E}(N)$ of N.

In order to study the expected value of N we use a technique that will simplify the problem, the **principle of deferred decisions**: The idea behind it is that the entire set of random choices are not fixed in advance but they are made during the course of the algorithm exactly when they are needed.

To better understand this technique, we present the game of the clock solitaire: We divide a deck of 52 cards in 13 stacks consisting of 4 cards each. We place the stacks face down, arbitrarily, as the face of the clock. Each hour represents a number except for:

- the stack at 1 o' clock represents the Ace
- the stack at 11 o' clock represent the Jack
- the stack at 12 o' clock represents the Queen
- a stack at a center that represents the King

We start by turning over the first card at the top of the stack at the center. The value of that drawn card will indicate which stack we will draw from next (so for example if the card we turned over is a 2, we will next turn over a card from the pile representing the number 2, if the card we turned over is an Ace, we will next turn over a card from the pile representing the Ace etc). Every time, the pile we will draw from next is the one that represents the value of the card that was drawn at the previous move.

The game ends when we attempt to draw a card from an empty stack. The empty stack that we arrive at can only be the one at the center representing the King: If it was a different one, the fact that we attempted to draw a card from it means that on our previous move we discovered a card with the number that pile represents. At the same time, the fact that, when we attempted to draw a card from that pile, it was already empty, indicates that we had already drawn a card from that pile 4 times, that is we had already found 4 cards with the value that pile represents, which is a contradiction. As a result, the game ends when we draw the 4th King.

We say that we have won the game when all cards are uncovered by the time the game ends, that is if the 4th King is the last card to be turned over. We want to calculate the probability of winning.

Theorem 2.12. *The probability of winning at the clock solitaire is* $\frac{1}{13}$.

Proof. In order to find the probability of winning we will use the principle of deferred decisions. In our case, that means that, instead of fixing a random placement of the entire set of cards in advance, we assume that, every time we need to turn over a card, we choose the card we are going to turn over uniformly at random from the set of cards that have not already been turned over. Moreover, we assume that if the game ends before all the cards are turned over, we "cheat" by turning over all the other remaining cards by an arbitrary rule. In our new way of describing the game, it is clear that the process of the game is equivalent to drawing, one by one, all the cards from a deck of 52 cards uniformly at random and we win if the last card is a King. In this game, the probability of winning is $\frac{51! \times 4}{52!} = \frac{1}{13}$. $\square$

Returning to the problem of stable matchings, the principle of deferred decisions means that for each man his random preference list is not chosen in advance from the $n!$ possible options but rather, each time he needs to make a proposal he chooses at random a woman he has not yet proposed to, to whom he addresses his proposal.

However, it still remains difficult to study the expected value $\mathbb{E}(N)$ of the number of proposals since, at any step, the random choice of a woman by a man depends at the previous proposals made by that man. To simplify the problem even more and eliminate that dependency we assume that the men are amnesiacs: when it is time for a man to choose a woman to propose to, he forgets the women towards whom he has already made an advance. Clearly, if a man proposes again to a woman he has already proposed to, that woman will reject him since she has already rejected him once. Moreover, the algorithm terminates once each woman is proposed to at least once. Thus, the output of the algorithm when the men are amnesiacs is the same as the output of the fundamental algorithm, with the only difference that there might have been made a redundant number of proposals. As a result, the expected value of the number of proposals made by the amnesiac men in the algorithm will constitute an upper bound for the expected value of the number of proposals in the fundamental algorithm.

Let's take the sequence of women the amnesiac men make proposals to with the order they are proposed to. (For example if the set of women is the set $\{a, b, c, d\}$ and we have the sequence $d, b, d, b, c, c, b, c, d, a$ that means that the first woman that received a proposal by a man was d, then b etc.) Then, as mentioned earlier, the algorithm terminates as soon as each woman receives at least one proposal, that is as soon as each woman appears on the sequence at least once. As a result, the total number of proposals needed to acquire a stable matching with amnesiac men is equal to the length of that sequence.

The calculation of the expected value of the length of such a sequence constitutes **the coupons collector's problem**:

Suppose there are n different coupons and each time we buy an envelope we obtain a coupon at random. Each coupon is equally likely to appear and we keep buying envelopes until we find all n different coupons. Let X be the random variable expressing the total number of envelopes needed to acquire at least one from each of the n different coupons. We are looking for $\mathbb{E}(X)$.

Theorem 2.13. *$\mathbb{E}(X) = nH_n$, where H_n is the sum of the first n terms of the harmonic series $\sum_{n=1}^{\infty} \frac{1}{n}$.*

Proof. Let X_i= the amount of envelopes needed to be bought, after we have collected i distinct coupons, in order to get a new, $i+1-th$ coupon, different from the ones we have already collected. So for example, X_0= the number of envelopes needed to collect the $1-st$ coupon=1, X_1= the number of envelopes that we bought, after we got the first coupon, in order to acquire the $2-nd$ distinct coupon etc. Then we have $X = X_0 + X_1 + ... + X_n$ and as a result

$$\mathbb{E}(X) = \mathbb{E}(X_0) + \mathbb{E}(X_1) + ... + \mathbb{E}(X_n) \tag{1}$$

Moreover, from the definition of X_i, it occurs that X_i follows the geometric distribution with parameter

$$p = \frac{n-i}{n} \tag{2}$$

Thus,

$$\mathbb{E}(X_i) = \frac{1}{p} = \frac{n}{n-i} \tag{3}$$

So from (1) and (3) we have that $\mathbb{E}(X) = \frac{n}{n} + \frac{n}{n-1} + ... + \frac{n}{1} = n(\frac{1}{n} + \frac{1}{n-1} + ... + \frac{1}{2} + 1)$=n$H_n$, where H_n is the sum of the first n terms of the harmonic series $\sum_{n=1}^{\infty} \frac{1}{n}$. $\square$

It is clear the the expected value of the number of envelopes needed to acquire all n distinct coupons is equal to the expected value of the number of proposals needed during the course of the algorithm when the men are amnesiacs. Hence, nH_n is an upper bound for the expected value of the number of proposals $\mathbb{E}(N)$ made during the course of the fundamental algorithm, where, in fact, $H_n = \ln n + \gamma + \frac{1}{2n} - \frac{1}{12n^2} + \varepsilon$, where $0 < \varepsilon < \frac{1}{120n^4}$ and γ is Euler's constant.

3 Generalizations and variations of the problem

3.1 Incomplete lists

We consider the problem of stable matchings when the men have not necessarily rated all women, thus their lists might be incomplete, and vice versa the women have not necessarily rated all men. We are looking for stable matchings with the extra condition that each person has to marry someone that is included in their lists of preferences.

In the case of complete lists, we saw that there is always at least one stable matching. However that is not the case with the incomplete lists. Consider the following example of incomplete lists:

$$\begin{array}{llll} m_1: & w_2 & w_1 & \\ m_2: & w_1 & & \\ m_3: & w_2 & w_1 & w_3 \end{array} \qquad \begin{array}{llll} w_1: & m_3 & m_2 & m_1 \\ w_2: & m_3 & m_1 & m_2 \\ w_3: & m_2 & m_3 & m_1 \end{array}$$

The only possible matching is $\{(m_2, w_1), (m_1, w_2), (m_3, w_3)\}$ but it is unstable because of w_2 and m_3. Thus the existence of at least one stable matching does not extend to incomplete lists.

We may convert the problem of incomplete lists to a problem of complete lists as follows:

We add a new man m' and a new woman w' such that m' is the last choice of w' and w' is the last choice of m'. Every woman rates m' last on her possibly incomplete list and then she classifies below m', in an arbitrary order, the rest of the men that were missing from her list. Similarly, every man rates w' last on his possibly incomplete list and then he classifies below w', in an arbitrary order, the women that were missing from his list.

Theorem 3.1. *There exists a stable matching for the incomplete system $\iff$ there exists a stable matching for the complete system where m' is married to w'.*

Proof. Let $M = \{m_1, ..., m_n\}$ the set of men and $W = \{w_1, ..., w_n\}$ set of women.

If there exists a stable matching M for the incomplete system then this stable matching along with the married couple (m', w') forms a stable matching N for the complete system: Obviously, there is no blocking pair of the form m_i-w_j (because then M would not be stable). As a result, if there was a blocking pair it would be of the form m_i-w' or m'-w_i. In the first case, that would mean that a man m_i would prefer w' to his spouse, which is a contradiction since we put w' at the end of the incomplete lists. Similarly, there could not be an unstable pair of the form m'-w_i.

Conversely, lets assume that there exists a stable matching N for the complete system where m' is married to w'. Then, the pairs (m_i, w_j) yielded from this stable matching constitute a stable matching of the incomplete system: For every pair (m_i, w_j) the spouse w_j of m_i is higher in his list of preference than w' (if she was lower than w', then, in the matching N, m_i and w' would prefer each other to their spouses which is a contradiction). Similarly, the spouse m_i of w_j is higher in her list of preferences than m'. Thus, the spouses of every person belong in the incomplete lists and we have a stable matching for the incomplete system. □

Theorem 3.2. *If there exists a stable matching with m' married to w' for the complete system, then, for all stable matchings of this system, m' is married to w'.*

Proof. The worst stable matching from the point of view of the women contains (m', w') (since m' is the last choice of w'). Thus that is also the best stable matching for the men. As a result, every other stable matching contains (m', w') (since in the best stable matching from the point of view of the men m' is married to his last choice w', he will be married to w' in every other stable matching as well). □

As a result, in order to decide on the existence of a stable matching for the incomplete system, it suffices to find one stable matching for the complete system. If, in this matching, m' is not married to w' then there does not exist a stable matching for the incomplete system. Otherwise, if in this matching m' is married to w', then this stable matching without the pair (m', w') is also a stable matching for the incomplete system.

3.2 Admitting n students to m universities

We now consider the problem of admitting n students to m universities. Each student has an order of preference for the universities and each university has an order of preference for the students. An admission is stable when there does not exist a student b and a university B such that b, not accepted to a university B, prefers B to his assigned university and university B prefers b to at least one of its admitted students.

Suppose that the $k-th$ university accepts n_k students and that, without loss of generality, $n = n_1+n_2+\cdots+n_m$. If the number of students surpasses the number of possible admissions by the universities then we create an imaginary university that is the last choice of all students and admits exactly the required number of students (the preference list of the new university is random). If the number of students is smaller than the total number of admissions of the universities we create imaginary students who are the last option for all universities (the preference lists of the new students are random).

To reduce that problem to the problem of stable matchings we replace the $k-th$ university with n_k $'$universities$'$ such that each one of them have the same preference list with $k-th$ university and they admit exactly one student. For each student, the n_k new universities are placed at their preference lists at a random order at the place of the previous $k-th$ university.

4 The set of stable matchings: Lattice structure and number of stable matchings

4.1 Lattice structure

We have seen so far that, for a specific stable marriage instance, more than 1 stable matchings may exist and that the Gale-Shapley algorithm gives the optimal matching for the men and, at the same time, the pessimal matching for the women. In this section we study the structure of the set of all stable matchings and, more specifically, we prove that the set of all stable matchings forms a distributive lattice. First, we present some definitions.

Definition 4.1. *For a given stable marriage M, we define the (man oriented) dominance relation as follows: We say that matching M dominates matching M', and we write $M \preceq M'$ if every man prefers M to M' or is indifferent between them, that is if every man in M has a partner at least as good as he had in M'.*

Definition 4.2. *We say that M strictly dominates M' and we write $M \prec M'$ if $M \preceq M'$ and $M \neq M'$.*

It is easy to show that this dominance relation is a partial order on the set $\mathscr{M}$ of all stable matchings.We will now show that the partially ordered set $(\mathscr{M}, \preceq)$ is, in fact, a distributive lattice.

Definition 4.3. *A lattice is a partially ordered set in which:*

1. *every pair of elements a, b has a greatest lower bound, called meet, denoted by $a \wedge b$. In other words, $a \wedge b \preceq a$, $a \wedge b \preceq b$ and for every other element element c such that $c \preceq a$, $c \preceq b$ we have that $c \preceq a \wedge b$.*

2. *every pair of elements a, b has a least upper bound, called join, denoted by $a \vee b$. In other words, $a \preceq a \vee b$, $b \preceq a \vee b$ and for every other element c such that $a \preceq c$, $b \preceq c$ we have that $a \vee b \preceq c$.*

Definition 4.4. *A distributive lattice is a lattice in which the distributive laws hold, that is*

1. *$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$ and*

2. *$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$.*

In order to prove that $\mathscr{M}$ is a distributive lattice, we present 2 lemmas that lead to the interpretation of the meet and join in $\mathscr{M}$.

Lemma 4.5. *For a given instance of the stable marriage problem, let M and M' be 2 (distinct) stable matchings. If every man is partnered with the best of his partners in M and M', the result is a stable matching.*

Proof. First we prove that it is indeed a matching: Assume that it was not a matching, that is there were 2 different men m and m' that received the same partner, say because $(m, w) \in M$ and $(m', w) \in M'$. Then m prefers M to M' and m' prefers M' to M, because we give each man the best of his partners in M and M'. Then, by applying the theorem 2.11 to the pair (m, w), w prefers M' to M. Similarly by applying the theorem 2.11 to the pair (m', w), w prefers M to M', a contradiction. Hence a matching, let's call it N, is formed. Now we prove that the matching is stable. Assume that there is a man m and a woman w such that they both prefer each other to their assigned partners in N. We have that $p_N(m)$ is the best amongst his partners in M and M' so, since he prefers w to $p_N(m)$, we have $wmp_M(m)$ and $wmp_{M'}(m)$. Moreover, $mwp_N(w)$. But $p_N(w)$ is either her partner in M or her partner in M'. In the first case, if $p_N(w) = p_M(w)$ then we have $mwp_M(w)$ but we also have $wmp_M(m)$ which is a contradiction, since M is stable. Similarly, if $p_N(w) = p_{M'}(w)$ then we have $mwp_{M'}(w)$ but we also have $wmp_{M'}(m)$, a contradiction since M' is stable. In either case we arrived at a contradiction, so N is stable. $\square$

Corollary 4.5.1. *If each man is given the best of his partners in any fixed set of stable matchings then a stable matching occurs. If the fixed set is the set of all stable matchings the stable matching that occurs is exactly the one yielded by the Gale-Shapley algorithm.*

Notation 4.6. *We denote by $M \wedge M'$ the stable matching obtained by assigning each man to his best partner between M and M'. Similarly we denote by $\wedge_{M \in S} M$ or $\wedge S$ the stable matching obtained by assigning each man to his best partner among his partners in all stable matchings in S.*

An immediate consequence of the theorem 2.11 and the way we defined $M \wedge M'$ is the following corollary:

Corollary 4.6.1. *In $M \wedge M'$ each woman is assigned to the worst of her partners in M and M'.*

Note that an easy consequence of this corollary is the fact that the man-optimal stable matching is also the woman-pessimal, as shown earlier in theorem 2.8.

Lemma 4.7. *For a given stable marriage instance let M and M' be 2 (distinct) stable matchings. If every man is partnered with the worst of his partners in M and M', the result is a stable matching.*

Proof. If each man is given the worst of his partners in M and M' then by theorem 2.11 we get that each woman is assigned to the best of her partners in M and M'. As a result, this lemma is just a restatement of lemma 4.5 with the roles of men and women interchanged. □

Notation 4.8. *We denote by $M \vee M'$ the stable matching obtained by assigning each man to his worst partner between M and M'. Similarly we denote by $\vee_{M \in S} M$ or $\vee S$ the stable matching obtained by assigning each man to his worst partner among his partners in all stable matchings in S.*

It is now easy to see that $M \wedge M'$ is the greatest lower bound for M and M' under the dominance relation and M $\vee$ M' is the least upper bound.

Theorem 4.9. *For a given instance of the stable marriage problem, the partial order $(\mathscr{M}, \preceq)$ is a distributive lattice, with $M \wedge M'$ being the meet of M and M' and $M \vee M'$ being the join of M and M'.*

Proof. By the definition of $M \wedge M'$ it is obvious that $M \wedge M' \preceq M$, $M \wedge M' \preceq M'$. Also if there is stable matching M^* such that $M^* \preceq M$ and $M^* \preceq M'$ then each man has in M^* a partner at least as good as in each of M and M'. As a result $M^* \preceq M \wedge M'$. Thus, $M \wedge M'$ is the greatest lower bound for M and M'. Similarly, we can prove that $M \vee M'$ is the least upper bound for M and M', establishing that $(\mathscr{M}, \preceq)$ is a lattice.

For the first distributive law let X, Y and Z be stable matchings and let $U = X \wedge (Y \vee Z)$, $V = (X \wedge Y) \vee (X \wedge Z)$. We shall prove that $U = V$. If $p_Y(m) = p_Z(m) = w$ then in both U and V m is assigned to whoever of $p_X(m)$ and w he prefers. If $p_Y(m) \neq p_Z(m)$ then we have the following cases:

- YmZ and ZmX (and as a result YmX): m is partnered with $p_Z(m)$ both in U and V
- YmZ and XmZ and either YmX or XmY: in both cases, m is partnered with $p_X(m)$ both in U and V
- ZmY and YmX (and as a result ZmX): m is partnered with $p_Y(m)$ both in U and V
- ZmY and XmY and either ZmX or XmZ: in both cases , m is partnered with $p_X(m)$ both in U and V

Hence, we proved that in every case each man has the same partner in U and V and therefore $U = V$. Similarly we can prove the second distributive law, establishing that $(\mathscr{M}, \preceq)$ is a distributive lattice.

$\square$

4.2 Number of stable matchings

As we have seen, it is possible that more than 1 stable matchings exist for a given instance of the stable marriage problem. In this subsection, we will show that the number of stable matchings can grow exponentially with the number of the instance. As a result, it is possible that an algorithm that finds all the stable matchings may need exponentially many steps to do so. Furthermore, if we seek a stable matching with a specific property, then a brute force algorithm examining all the stable matchings will be of exponential time complexity in the worst case.

Below, we will give a lower bound for the number of stable matchings in an instance of size n.

Lemma 4.10. *Given stable marriage instances of sizes m and n with x and y stable matchings respectively, there is a stable marriage instance of size mn with at least $max(xy^m, yx^n)$ stable matchings.*

Proof. Suppose that the men and the women in the given instances are denoted by a_1, a_2,..., a_m, c_1,c_2,...,c_m and b_1,b_2,...,b_n, d_1,d_2,..,d_n respectively. Now consider the instance of size mn in which:

- the men are labeled (a_i,b_j) for $i = 1, ..., m$ and $j = 1, ..., n$
- the women are labeled (c_i,d_j) for $i = 1, ..., m$ and $j = 1, ..., n$
- the man (a_i,b_j) prefers (c_k,d_l) to $(c_{k'}, d_{l'})$ if b_j prefers d_l to $d_{l'}$ or if $l = l'$ and a_i prefers c_k to $c_{k'}$
- the woman (c_i,d_j) prefers (a_k,b_l) to $(a_{k'}, b_{l'})$ if d_j prefers b_l to $b_{l'}$ or if $l = l'$ and c_i prefers a_k to $a_{k'}$

Let M_1,.., M_n be any sequence of (not necessarily distinct) stable matchings of the problem instance of size m and M a stable matching of the problem instance of size n. There are $x^n y$ possible choices for M_1,...,M_n and M. We will now show that the mapping: $(a_i,b_j) \mapsto (p_{M_j}(a_i), p_M(b_j))$
gives a stable matching in the problem instance of size mn.

It is a matching because both M and M_j are matchings. Suppose it is not stable, meaning that there is a man (a, b) and a woman (c, d) that prefer each other to their assigned spouses. Then among the following conditions, we must have either one of 1 and 2 along with either one of 3 and 4:

1. $dbp_M(b)$
2. $d = p_M(b)$ and $cap_{M_j}(a)$
3. $bdp_M(d)$
4. $b = p_M(d)$ and $acp_{M_j}(c)$

1 and 4 cannot be satisfied at the same time, same with 2 and 3, because they are incompatible. Moreover, the combination of 1 with 3 is rejected because then M would not be stable and the combination of 2 with 4 is also rejected because then M_j would not be stable. As a result, all possible combinations are rejected and the matching is stable. So we have now found an instance of size mn with at least yx^n stable matchings. Similarly, by interchanging the roles of the original instances we can find an instance with at least xy^m stable matchings. $\square$

Theorem 4.11. *For each $n \geq 0$, where n is a power of 2, there is a stable marriage instance of size n with at least 2^{n-1} stable matchings.*

Proof. We prove it by induction in the power of 2. For $n = 2^0 = 1$ the problem of instance 1 obviously has 1 stable matching. Suppose that the theorem holds for problems of instance $n = 2^k$. We will prove that it holds for problems of instance $n = 2^{k+1}$. We apply lemma 4.10 for a problem instance of size $n = 2^k$ which has at least $y = 2^{n-1} = 2^{2^k-1}$ stable matchings (by induction hypothesis) and the following instance of size $m = 2$ which has $x = 2$ stable matchings since both possible matchings are stable:

men's preference lists			women's preference lists		
m_1 :	w_1	w_2	w_1 :	m_2	m_1
m_2 :	w_2	w_1	w_2 :	m_1	m_2

As a result, there exists an instance of size $mn = 2^k \cdot 2 = 2^{k+1}$ with at least $max(xy^m, yx^n) = max(2 \cdot (2^{2^k-1})^2, 2^{2^k-1} \cdot 2^{2^k}) = max(2^{2^{k+1}-1}, 2^{2^{k+1}-1}) = 2^{2^{k+1}-1}$ stable matchings, as required.

$\square$

5 The complexity of counting stable matchings

We have seen so far that, given an instance of a stable marriage problem, we can find in polynomial time a stable matching. We have also seen that more than one stable matchings can exist and in the section above we proved that the number of stable matchings can grow exponentially with the number of the instance. In this section, we will show that, in fact, the problem of determining the number of stable matchings in a given stable marriage instance, is $\#P-$complete. In order to prove that, it suffices to prove that it is in class $\#P$ and that there is a parsimonious transformation from a known $\#P-$complete problem such that the instance of the original problem and the corresponding instance of the stable marriage problem have the same number of solutions.

The fact that it is in class $\#P$ is easy:

Theorem 5.1. *: Determining the number of stable matchings in a given stable marriage instance problem is in $\#P$.*

Proof. Given a stable marriage problem instance and a specific matching it is easy to check in polynomial time if that matching is stable. □

In order to demonstrate the needed transformation from a known $\#P-$complete problem, we will be using, without proof, the following theorem of Provan and Ball [10]:

Theorem 5.2. *: Determining the number of antichains in a poset is $\#P-$complete.*

We now state our main theorem:

Theorem 5.3. *Given a poset $(P, \leq)$ with n elements, there exists an instance I of the stable marriage problem, constructible from $(P, \leq)$ in time polynomial in n, such that the stable matchings of I are in one to one correspondence with the antichains of $(P, \leq)$.*

In order to describe the transformation and prove that it has the required property, we will be introducing a series of new concepts and theorems, gaining at the same time further insight into the structure of the set of stable matchings for a general stable marriage instance. We start by introducing the concept of shortlists, followed by a number of properties.

Definition 5.4. *We define the shortlists to be the lists obtained from the original preference lists by a number of deletions as follows: We remove m from w's list and w from m's if and only if w receives a proposal from someone she likes better than m during the Gale-Shapley algorithm.*

We present some properties that are either immediate or explicit or implicit consequences of the results presented in section 2:

1. m appears on w's shortlist if and only if w appears on m's.
2. If w does not appear on m's shortlist, then there is no stable matching in which m and w are partners.
3. w appears first on m's shortlist if and only if m appears last on w's.
4. If every man is paired with the first woman on his shortlist then the resulting matching is stable, and it is the male optimal solution, meaning that no man can have a better partner than he does in this matching.

5.1 Rotations

In this section we will introduce the concept of a rotation, which plays an important role in establishing the relationship between the structure of a stable marriage instance and that of an associated partially ordered set.

In the male optimal solution of the problem, every man is assigned to the first woman on his shortlist. If we want to generate a different stable matching, some of the men must sacrifice their optimal partners.

Suppose for example that man m sacrifices his optimal partner, woman w. Then the best partner that m can have in the new stable matching is the second woman on his shortlist, say woman v. If v was partnered with a man k in the male optimal matching, man k would be the worst possible partner for v, meaning that, compared to k, m is an improvement for v in the new stable matching. In the new stable matching, k would sacrifice v for someone worse. Repeating the same argument for the second woman in man's k shortlist, and so on, a chain of forced sacrifices is generated. Because the number of men and women is finite that procedure can end in one of the following ways:

(i) the chain eventually cycles, so we have a finite sequence $m_0, m_1, ..., m_{r-1}$ of men such that the second woman on m_i's shortlist is the first one in that of m_{i+1} (where the subscripts are taken *modulo* r) or

(ii) the chain reaches a man whose shortlist contains just one woman so the procedure ends.

In case (i), we refer to such a cycle as a rotation relative to the shortlists. The terminology arises because, as we shall see, the partners can be rotated one place without destroying stability. A more precise definition of a rotation will be given later below.

Definition 5.5. *For a given stable marriage problem, a set of reduced preference lists is a set of preference lists obtainable from the originals by zero or more deletions, such that:*

(i) no list is empty

(ii) woman w is absent from man m's list $\iff$ man m is absent from w's list.

Notation 5.6. *Relative to such a set $\mathscr{A}$ of reduced preference lists, we denote by $first_{\mathscr{A}}(x)$, $second_{\mathscr{A}}(x)$, $last_{\mathscr{A}}(x)$ the first, second and last person respectively on x's list. When it is obvious which set of lists we are talking about, we shall omit the subscript $\mathscr{A}$. Of course, $second_{\mathscr{A}}(x)$ may be undefined if x's reduced preference list in $\mathscr{A}$ has only one entry.*

Definition 5.7. *A set $\mathscr{A}$ of reduced preference lists is called stable if, for each man m and each woman w,*

(i) $w = first_{\mathscr{A}}(m) \iff m = last_{\mathscr{A}}(w)$

(ii) w is absent from m's list $\iff wr(w, m) > wr(w, last_{\mathscr{A}}(w))$

A stable set of reduced preferences list will be referred to as a stable set for short.

Lemma 5.8. *The shortlists form a stable set.*

Proof. Property (i) was noted above. For property (ii), by the definition of shortlists, w is absent from m's list if and only if, during the course of the Gale-Shapley algorithm, w received a proposal from someone better than m, meaning that the last proposal held by w is also by a man better than m, which happens if and only if $wr(w, m) > wr(w, last(w))$ □

Lemma 5.9. *If, relative to a stable set, each man is partnered with $first(m)$, then the result is a stable matching.*

Proof. First we prove that the result is indeed a matching: if it wasn't, that is if there were 2 distinct men m_1, m_2 such that $first(m_1) = first(m_2) = w$ then by property (i) of stable sets we have $m_1 = last(w) = m_2$, a contradiction. The matching is also stable: If there was a man m that prefered a woman w to his spouse $first(m)$ then, since $mr(m, w) < mr(m, first(m))$, w is absent from m's list on that stable set. Hence, by property (ii) of stable sets we have $wr(w, m) > wr(w, last(w))$ meaning that w prefers her spouse $last(w)$ to m so there can be no instability. □

The stable matching obtained from a stable set $\mathscr{A}$ as described in lemma 5.9 will be referred to as the stable matching corresponding to $\mathscr{A}$.

Observation 5.10. *In the stable matching S corresponding to a stable set $\mathscr{A}$, each man is partnered with the first woman on his preference list relative to $\mathscr{A}$, so, by property (i) of definition 5.7 of stable sets, each woman is partnered with the last man on her preference list. As a result, $(last(w), w) \in S$ for each woman w.*

Lemma 5.11. *If, for some stable set,*

(i) $mr(m, first(m)) < mr(m, w)$ *and*

(ii) w *is absent from* m*'s list,*

then there is no stable matching in which w and m are partners.

Proof. Suppose that there is a stable matching S' such that $(m, w) \in S'$. By lemma 5.9 $(m, first(m)) \in S$ for some stable matching S. Then from (i) m prefers S to S'. We also know that $(last(w), w) \in S$ by observation 5.10. Moreover, since w is absent from m's list, we have that $wr(w, m) > wr(w, last(w))$ from the definition of stable sets. As a result, w prefers S (where she is partnered with $last(w)$) to S' (where she is partnered with m), a contradiction by theorem 2.11. □

Definition 5.12. *An ordered sequence $(m_0, w_0), ..., (m_{r-1}, w_{r-1}), r \geq 2$ of man/woman pairs forms a rotation in a stable marriage instance if, relative to some stable set $\mathscr{A}$, $w_{i+1} = first(m_{i+1}) = second(m_i)$ for each i $(0 \leq i \leq r-1$, where $i+1$ is taken modulo r). The rotation is said to be exposed in $\mathscr{A}$. Clearly, m_i is in w_{i+1}'s list in $\mathscr{A}$ and vice versa.*

Lemma 5.13. *Let $\mathscr{A}$ be a stable set, and let S be the corresponding stable matching. If S' is another stable matching in which man m has a worse partner than $first_{\mathscr{A}}(m)$ (who is his partner in S) then there is a rotation exposed in $\mathscr{A}$ all of whose male members have worse partners in S' than in S.*

Proof. If $first(m)$ is the only entry in m's list, then by lemma 5.11 there is no such stable matching S' so there is nothing to prove. Otherwise, we form the following sequence $\{(m_i, w_i)\}$, where:

(i) $m_0 = m$

(ii) $w_i = first(m_i)$, i=0,1,...

(iii) $m_{i+1} = last(second(m_i))$, $i = 0, 1, ...$ (so $second(m_i) = first(m_{i+1}) = w_{i+1}$)

Since $w_0 = first(m_0)$, then $(m_0, w_0) \in S$. Moreover, m_0's partner in S' is $second(m_0) = first(m_1) = w_1$ or someone worse. As a result, for S' to be stable, w_1's partner in S' must be m_0 or someone better. Therefore her partner in S' is better than $last(w_1) = m_1$ (who is her partner in S). Hence, by theorem 2.11, m_1's partner in S' is someone worse than $w_1 = first(m_1)$. Therefore, by lemma 5.11, $second(m_1)$ is defined. Repeating the same argument we can show that $second(m_i)$ is defined for all i and that all the m_i have worse partners in S' than in S.

Now, since the number of men and women is finite, the sequence $\{(m_i, w_i)\}$ must cycle eventually. So let's suppose that $m_0, ..., m_{s-1}$ are all distinct but $m_s = m_t$ for some t, $0 \leq t \leq s-2$ (we cannot have $m_s = m_{s-1}$ as $second(m_i) = first(m_{i+1})$). Then, $(m_t, w_t), ..., (m_{s-1}, w_{s-1})$ forms a rotation that is exposed in $\mathscr{A}$ and has the required property. □

For a given man m and a given stable set $\mathscr{A}$, we call the rotation obtained by m in the way that was described in Lemma 5.13 the rotation generated by m. Clearly, if m is himself in a rotation exposed in $\mathscr{A}$ then this is the rotation generated by m. The following corollaries are immediate results of Lemma 5.13.

Corollary 5.13.1. *If $\mathscr{A}$ is a stable set with a corresponding stable matching S, then either:*

(i) at least one rotation is exposed in $\mathscr{A}$ or

(ii) no man can have a worse partner in any stable matching than he has in S (and as a result S is the female optimal solution).

Corollary 5.13.2. *If $(m_0, w_0), ..., (m_{r-1}, w_{r-1})$ is a rotation and if, in some stable matching S, some fixed m_j has a partner worse than w_j, then each of the m_i have an $S-$partner worse than w_i.*

Definition 5.14. *Suppose that $p = (m_0, w_0), ..., (m_{r-1}, w_{r-1})$ is a rotation that is exposed in some stable set $\mathscr{A}$. If, for each i ($0 \leq i \leq r-1$, $i+1$ taken modulo r) all successors of m_i are deleted from w_{i+1}'s list in $\mathscr{A}$ and w_{i+1} is removed from the corresponding men's list, we say that the rotation has been eliminated.*

Observation 5.15. *Since $w_{i+1} = first(m_{i+1}) \iff m_{i+1} = last(w_{i+1})$, we have that m_{i+1} is a successor of m_i in w_{i+1}'s list. As a result m_{i+1} is* *deleted from w_{i+1}'s list and $w_{i+1} = first(m_{i+1})$ is deleted from m_{i+1}'s list during rotation elimination.*

Observation 5.16. *We have $second(m_{i+1}) = first(m_{i+2}) = w_{i+2}$ is deleted from the lists of all men that are successors of m_{i+1} in w_{i+2}'s list, so she is not removed from m_{i+1}'s list.*

Lemma 5.17. *If a rotation is eliminated from a stable set $\mathscr{A}$ in which it is exposed, then the resulting set $\mathscr{B}$ of lists is also stable.*

Proof. We have to prove properties (i) and (ii) of stable sets.

(i)
- If m is not in the rotation, then
  $w = first_{\mathscr{B}}(m) \iff$
  $\iff w = first_{\mathscr{A}}(m)$, since $first_{\mathscr{A}}(m)$ is not removed,
  $\iff m = last_{\mathscr{A}}(w)$, since $\mathscr{A}$ is a stable set,
  $\iff m = last_{\mathscr{B}}(w)$, since $last_{\mathscr{A}}(w)$ is not removed.

- If m is in the rotation and $w = first_{\mathscr{A}}(m)$ then from observations 5.15 and 5.16 we have that $first_{\mathscr{A}}(m)$ is removed from m's list but $second_{\mathscr{A}}(m)$ is not. As a result,
  $w = first_{\mathscr{B}}(m) \iff$
  $\iff w = second_{\mathscr{A}}(m)$, since $first_{\mathscr{A}}(m)$ is removed but $second_{\mathscr{A}}(m)$ is not,
  $\iff m = last_{\mathscr{B}}(w)$, since all successors of m are removed from $second_{\mathscr{A}}(m) = w$'s list during rotation elimination

(ii) w is absent from m's new list $\iff$
$\iff$ (w was already absent from m's old list) or
(w is removed from m's list during rotation elimination) $\iff$
$\iff (wr(w, m) > wr(w, last_{\mathscr{A}}(w))$ or
$(wr(w, last_{\mathscr{A}}(w)) > wr(w, m) > wr(w, last_{\mathscr{B}}(w)) \iff$
$\iff wr(w, m) > wr(w, last_{\mathscr{B}}(w))$.

□

We will now establish a one-to-one relationship between stable sets and stable matchings for any given instance of the stable marriage problem.

Lemma 5.18. *For a given instance of the stable marriage problem, there is a one-to-one correspondence between the stable matchings and the stable sets. Furthermore, each stable set can be obtained from the set of shortlists by a sequence of zero or more rotation eliminations.*

Proof. Given a stable set, a unique stable matching can be constructed as in lemma 5.9 (that is, relative to the stable set, each man is partnered with $first(m)$).
On the other hand, given a stable matching $S = \{(m_0, w_0), ..., (m_{n-1}, w_{n-1})\}$ we construct a stable set as follows. We start from the set $\mathscr{M}$ of shortlists, which of course has a corresponding stable matching M, the male optimal solution. If $M \neq S$ then there is a woman w_i such that $w_i \neq first_{\mathscr{M}}(m_i)$, in fact $mr(m_i, first_{\mathscr{M}}(m_i)) < mr(m_i, w_i)$. From lemma 5.11, w_i is in m_i's shortlist. Moreover, m_i's partner in S is worse than his partner in M so by lemma 5.13 the exposed rotation p generated by m_i is such that all of its male members have worse partners in S than in M. Assume now that p is eliminated to obtain, by lemma 5.17 a new stable set N. Then, again, by lemma 5.11, w_i remains in m_i's list and vice versa for all i. We now repeat the same process relative to the new set of lists, as many times as needed until $w_i = first(m_i)$ for all i. We have then created a new stable set to which the stable matching S corresponds, and it will have been obtained by a sequence of zero or more eliminations from the set of shortlists. $\square$

Lemma 5.19. *In a given stable marriage instance, no pair (m, w) can belong to 2 different rotations.*

Proof. Suppose that the pair (m, w) belongs to 2 different rotations p_1 and p_2. Since the rotations are different, suppose that the pair (m', w') belongs to p_1 but not to p_2. We will show that these assumptions lead to a contradiction.

If $\mathscr{A}$ is a stable set in which p_2 is exposed then $w = first_{\mathscr{A}}(m)$. Let $v = first_{\mathscr{A}}(m')$ the partner of m' in the corresponding stable matching S. Then, if $mr(m', v) > mr(m', w')$, Corollary 5.13.2 applied to the rotation p_1 would force m to have an $S-$partner worse than w, which is a contradiction. As a result $mr(m', v) \leq mr(m', w')$.

Now let $\mathscr{T}$ be the stable set, and T the corresponding stable matching, obtained by eliminating p_2 from $\mathscr{A}$. Since $(m, w) \in p_2$ and $\mathscr{T}$ arises from the elimination of p_2, we have, from observation 5.15 that $w = first_{\mathscr{A}}(m)$ is removed from m's list so in T m has a partner worse than w. As a result, applying corollary 5.13.2 to the rotation p_1 and the stable matching T, we deduce that, since m has a $T-$partner worse than w, then m' should also have a $T-$partner worse than w'.
We consider 2 cases:

Case 1: $(m', v) \in p_2$: since $(m', w') \notin p_2$ we have that $v \neq w'$ and thus, $mr(m', v) < mr(m', w')$. However, applying Lemma 5.11 to the stable set

$\mathscr{A}$, we have that, in $\mathscr{A}$, w' is present in the list of m': since $mr(m', first_{\mathscr{A}}(m')) = mr(m', v) < mr(m', w')$ if w' was not present in the list of m' then Lemma 5.11 would lead to a contradiction. Furthermore, since $(m', w') \notin p_2$ the presence of w' in the list of m' is not affected by the elimination of p_2 so in T, m' has a partner at least as good as w' which is a contradiction.

Case 2: $(m', v) \notin p_2$: Then the elimination of p_2 from $\mathscr{A}$ does not affect the presence of v in the list of m'. As a result (m', v) is a pair in the matching T, a contradiction since m' should have a $T-$partner worse than w.

□

Lemma 5.20. *Suppose* (m, w) *belongs to a rotation. Then:*

(i) (m, w) *belongs to some stable matching;*

(ii) *in a stable set obtained from the shortlists by a sequence of rotation eliminations, w is absent from m's list if and only if the rotation containing* (m, w) *has been eliminated.*

Proof. *(i)* There must be a stable set in which the rotation containing (m, w) is exposed. The matching obtained from that stable set includes the pair (m, w).

(ii) If the rotation containing (m, w) has been eliminated, then w is removed from m's list. For the other direction, suppose we have a stable set obtained from the shortlists by a sequence of rotation eliminations such that w is absent from m's list. We will show that the rotation containing (m, w) has been eliminated. When a rotation p is eliminated, entries may disappear from m's list in one of the following ways:

(a) the first entry in m's list will disappear if $(m, first(m)) \in p$.

(b) one or more entries may disappear as a result of the deletion of m from the lists of one or more women.

If w were to disappear from m's list by method (b) then immediately after the conditions of lemma 5.11 are satisfied and as a result m and w would not be partners in a stable matching, which is a contradiction. As a result the only way for w to disappear from m's list is from method (a), meaning that the rotation containing (m, w) has been eliminated.

□

Given a rotation p of a stable marriage instance, there may be several stable sets in which p is exposed. By lemma 5.18, any such stable set is obtained from the set of shortlists by zero or more rotation eliminations.

Definition 5.21.

- *If no stable set in which a rotation p is exposed can be obtained from the set of shortlists without eliminating rotation τ, then we say τ is a predecessor of p and we write $\tau < p$. The relation $<$ is anti symmetric, transitive and reflexive and therefore defines a partial order on the set of rotations. We call this the rotation poset for that stable marriage instance.*
- *We say that τ is an immediate predecessor of p if $\tau < p$ and there is no σ such that $\tau < \sigma < p$*
- *We say that a subset of the poset is a closed subset if it is closed under predecessors.*

Note 5.22. *An alternative representation for the poset is in the form of a directed acyclic graph such that there is one node for every element of the poset and an arc from the node representing rotation τ to the node representing rotation p iff τ is an immediate predecessor of p.*

Definition 5.23.

- *An antichain in a poset $(P, \leq)$ is a subset A of P such that there are no elements τ, p in A such that $\tau < p$.*
- *For an antichain A, we define the closure A^* of A as follows $A^* = \{\tau \in P : \tau < p$ for some $p \in A\}$. Obviously, A^* is a closed subset of P.*
- *For any closed subset C of P, there is a unique antichain A such that $A^* = C$. We call this the spanning antichain of C.*

Theorem 5.24. *For any stable marriage instance, there is a one-to-one correspondence between the stable matchings for that instance and the antichains of its rotation poset.*

Proof. First, we show that every antichain corresponds to a stable matching and different antichains yield different stable matchings. Given an antichain A, A^* is closed: If $\tau \in A^*$ then $\tau < p$ for some $p \in A$. If $\sigma < \tau < p$ then $\sigma \in A^*$ so A^* is closed under predecessors. As a result, starting from the set of shortlists, all rotations in A^* can be eliminated one by one, producing a stable set with its corresponding stable matching.

If two different sets of rotations are eliminated, then from lemma 5.19 and lemma 5.20, the resulting stable sets are different so the corresponding stable matchings are also different. Moreover, $A \neq B \implies A^* \neq B^*$ so, since different rotations result to different stable matchings, it follows that different antichains yield different stable matchings.

Now we prove that every stable matching corresponds to an antichain and different stable matchings correspond to different antichains. By lemma 5.18, any stable matching corresponds to a stable set so every stable matching arises from the shortlists via a sequence of rotation eliminations. The set $\mathscr{R}$ of rotations concerned must be closed: a rotation cannot be eliminated before it is exposed and it cannot be exposed before all of its predecessors have been eliminated. As a result, $\mathscr{R}$ has a unique spanning antichain. $\square$

5.2 $\#P-$completeness of counting stable matchings

We are now ready to complete the proof of the $\#P-$completeness of the problem as we can now prove theorem 5.3 by presenting a transformation and then proving that it has the needed property.

First we construct a directed acyclic graph from the poset as follows: We get one node for each element of P and there is an arc from node u to node v iff u is an immediate predecessor of v, that is $u \leq v$ and $\nexists\ s \in P$ such that $u \leq s \leq v$. Moreover, two extra nodes are included: one called the source, which has an outgoing arc towards every node representing a minimal element of the poset and one called the sink, which has an incoming arc from every node representing a maximal element of the poset.

The arcs of this graph are numbered arbitrarily $1, .., t$. Furthermore, each node, apart from the source and the sink, is labelled with an ordered subset of $1, 2, ..., t$, of size ≥ 2, which is the subset consisting of the numbers of the arcs incident to or from that node. The numbers in the subset are ordered in an arbitrary but fixed order, say increasing order.

In order to construct a stable marriage problem instance from that digraph, each node, with the exception of the source and the sink, is processed according to the rules that will be described below. In order to ensure that no node is processed before one of its predecessors the nodes are processed according to a topological order, that is a one-to-one mapping f from the nodes of the digraph onto the set $\{0, ...n-1\}$ such that if node u is a predecessor of node v then $f(u) \leq f(v)$, essentially meaning that no node is processed before all of its predecessors have been processed. Such a topological ordering of the nodes of an acyclic directed graph can be found in time polynomial in the number of the nodes. During this node processing phase, partial ordered lists are constructed for the men in natural (first to last) order and for the women in reverse (last to first) order. When the processing of the nodes is over, the preference lists are completed by adding arbitrarily all the absentees after those already present. The rules for the node processing from which the partial preference lists are constructed are the following:

- The stable marriage instance involves t men and t women (that is one man and one woman for each one of the arcs of the graph). The first woman on man i's list is woman i and the first man on woman i's list is man i (for $i = 1, .., t$).

- After the first k nodes ($k \geq 0$) of the graph have been processed according to the topological order, we notate the woman most recently appended to man i's list as $w(k, i)$.

- Node $k + 1$, labelled $\{a_0, ..., a_{r-1}\}$, is processed as follows:
  For each i ($0 \leq i \leq r - 1$) woman $w(k, a_{i+1})$ is appended to man a_i's list after any women already present and man a_i is appended to woman $w(k, a_{i+1})$'s list ahead of any men already present.
  More specifically, $w(k, a_{i+1})$ is placed on man a_i's list right after woman $w(k, a_i)$ (since that is the woman last appended to a_i's list after having processed k nodes). Hence,

$$mr(a_i, w(k, a_{i+1})) = 1 + mr(a_i, w(k, a_i)) \tag{4}$$

  Moreover, since $w(k, a_{i+1})$ is the last woman appended on man a_{i+1}'s list after having processed k nodes, a_{i+1} is also the last man appended to $w(k, a_{i+1})$'s list. As a result, man a_i is placed right before man a_{i+1} on $w(k, a_{i+1})$'s list. Thus,

$$wr(w(k, a_{i+1}), a_{i+1}) = 1 + wr(w(k, a_{i+1}), a_i) \tag{5}$$

  where $0 \leq i \leq r - 1$ and $i + 1$ is taken *modulo* r.

It is clear that this whole construction is completed in polynomial time. In order to prove that this construction has the required property, we first have to prove that it does indeed give a stable marriage instance of size t.

Lemma 5.25. *The construction described above does yield a stable marriage instance of size t.*

Proof. All we have to do is to show that during the processing of the nodes, no woman can be appended twice to some man's list (and thus no man can be appended twice to some woman's list).

For a given arc i in the graph, denote by $init(i)$ and $term(i)$ the initial and terminal nodes respectively for the arc i. According to the rules of nodes processing stated above, woman i is first involved when man i appears on some node's label list, that is when node $start(i)$ is processed, where $start(i)$ is $init(i)$, except if $init(i)$ is the source, in which case $start(i)$ is $term(i)$ (because the source does not take part in the nodes processing). Specifically, at this point she is appended to the list of man i_1, where i_1 immediately precedes i in the label set of node $start(i)$.

Since so far she is the last woman appended to man i_1's preference list, by the rules of the construction of the preference lists stated above, in order for her to be appended to some other man's list, the list of man i_1 must occur again in the processing of the nodes. In other words, arc i_1 must appear again in some label set. Since i_1 is included in the label set of $start(i)$ then either $start(i) = term(i_1)$ or $start(i) = init(i_1)$. In the first case, since the nodes are processed in topological order, $init(i_1)$ is already processed so the list of man i_1 does not appear again and as a result woman i appears on no other lists. In the second case, assuming that $term(i_1)$ is not the sink (because the sink does not participate in the node processing procedure), when $term(i_1)$ is processed, woman i is appended to the list of man i_2, where i_2 is in the label set of node $term(i_1)$. Similarly as before, for i to be appended in some other man's list, i_2 must be included in the label set of some other node, meaning that $term(i_1) = init(i_2)$ (for if $term(i_1) = term(i_2)$ then i_2 would not be included in any of the label sets of the nodes that have not yet been processed). Moreover, i would not be included in the label of some other node if $term(i_2)$ was the sink, as the sink does not participate in the node processing procedure. As a result, proceeding in the manner described above, we obtain a sequence i_1,i_2,...,i_s ($s \geq 2$) of men such that:

1. $start(i) = init(i_1)$
2. $init(i_j) = term(i_{j-1})$, for $j = 2, ..., s-1$
3. $stop(i_s) = term(i_{s-1})$, where $stop(i_s)$ is defined to be $term(i_s)$ unless $term(i_s)$ is the sink, in which case $stop(i_s)$ is defined to be $init(i_s)$.
4. during the processing of the nodes, woman i is appended to the lists of men i,i_1,...,i_s and to no others.

By 2., arcs i_1,...,i_{s-1} form a directed path in the acyclic graph and so are all distinct or else a cycle would be created.

By 1., node i is distinct from i_1,...,i_{s-1}: By the way we defined i_1 we have that $i \neq i_1$. Moreover, if $i = i_k$ for some $2 \leq k \leq s-1$ then:

- either $\mathit{start}(i) = \mathit{init}(i)$ so $\mathit{init}(i_1) = \mathit{init}(i) = \mathit{init}(i_k) = \mathit{term}(i_{k-1})$
- or $\mathit{start}(i) = \mathit{term}(i)$ so $\mathit{init}(i_1) = \mathit{term}(i) = \mathit{term}(i_k)$

In both cases, a cycle is created, which is a contradiction.

Similarly, by 3., node i_s is distinct from $i_1, ..., i_{s-1}$: By the way we defined the sequence, $i_s \neq i_{s-1}$. If $i_s = i_k$ for some $2 \leq k \leq s-2$ then:

- either $\mathit{stop}(i_s) = \mathit{term}(i_s)$ so $\mathit{term}(i_{s-1}) = \mathit{term}(i_s) = \mathit{term}(i_k) = \mathit{init}(i_{k+1})$
- or $\mathit{stop}(i_s) = \mathit{init}(i_s)$ so $\mathit{term}(i_{s-1}) = \mathit{init}(i_s) = \mathit{init}(i_k)$

In both cases a cycle is created, which is a contradiction.

Finally, $i \neq i_s$: i_1,...,i_{s-1} is a path from $\mathit{init}(i_1) = \mathit{start}(i)$ to $\mathit{term}(i_{s-1}) = \mathit{stop}(i_s)$ and if $i = i_s$ then we have the following possibilities:

- $\mathit{stop}(i_s) = \mathit{init}(i_s)$ and $\mathit{start}(i) = \mathit{term}(i)$ meaning that $\mathit{init}(i)$ is the source and $\mathit{term}(i_s)$ is the sink but $\mathit{term}(i_s) = \mathit{term}(i)$ so that would mean that $\mathit{init}(i)$ is the source and $\mathit{term}(i)$ is the sink, a contradiction.
- $\mathit{stop}(i_s) = \mathit{init}(i_s)$ and $\mathit{start}(i) = \mathit{init}(i)$ so $\mathit{term}(i_s) = \mathit{term}(i)$ is the sink but $\mathit{term}(i)$ would be the sink iff $\mathit{init}(i) = \mathit{init}(i_1)$ was representing a maximal element of the poset, which is a contradiction since $\mathit{init}(i_1)$ is a predecessor of $\mathit{term}(i_1)$.
- $\mathit{stop}(i_s) = \mathit{term}(i_s) = \mathit{term}(i)$ and $\mathit{start}(i) = \mathit{term}(i)$ (meaning that $\mathit{init}(i)$ is the source) and so we have a path from $\mathit{term}(i)$ to $\mathit{term}(i)$ which is a contradiction since the graph is acyclic.
- $\mathit{stop}(i_s) = \mathit{term}(i_s) = \mathit{term}(i)$ and $\mathit{start}(i) = \mathit{init}(i)$ so we would have a path i_1,...,i_{s-1} from $\mathit{init}(i)$ to $\mathit{term}(i)$ which is a contradiction since $\mathit{init}(i)$ is an immediate predecessor of $\mathit{term}(i)$.

so we necessarily also have that i$\neq$ i_s. □

Lemma 5.26. *For the stable marriage instance constructed above, the set of shortlists contains precisely the men and women appended before and during the processing of the nodes.*

Lemma 5.27. *(i) If* $\{a_0,\ldots,a_{r-1}\}$ *is the label set of the node numbered* k *in the chosen topological ordering, then the stable marriage instance constructed above contains a rotation* $p_k = (a_0, b_0), \ldots, (a_{r-1}, b_{r-1})$ *where* $b_i = w(k-1, a_i)$*. This rotation has as its predecessors precisely those rotations* p_j *for which node* j *is a predecessor of node* k*. Furthermore, when* p_k *is eliminated from any stable set in which it is exposed, only* b_i *is removed from* a_i*'s list, for each* i *(*$0 \le i \le r-1$*), and no woman is removed from any other man's list.*

(ii) There are no rotations other than those described in (i)

Proof. *(i)* We will prove this through induction. We first observe that this statement is true for the node numbered 1 in the chosen topological ordering: When this node was processed during the construction of the stable marriage instance, for each i, woman $b_{i+1} = w(0, a_{i+1})$ was appended to man a_i's list which previously had only 1 woman. More specifically,

$$second_{\mathscr{M}}(a_i) = mp(a_i, 2) = w(0, a_{i+1}) = mp(a_{i+1}, 1) = first_{\mathscr{M}}(a_{i+1}),$$

where $\mathscr{M}$ is the set of shortlists. So from the above we have that $b_{i+1} = first_{\mathscr{M}}(a_{i+1}) = second_{\mathscr{M}}(a_i)$ which shows that rotation p_1 as stated in (i) is exposed in $\mathscr{M}$. Moreover, since $\mathscr{M}$ is the set of shortlists, rotation p_1 has no predecessors, exactly like the node 1.

Furthermore, when p_1 is eliminated, from observation 5.15, we have that $b_i = w(0, a_i) = mp(a_i, 1)$ is removed from a_i's list. Moreover, we have that $wr(b_i, a_{i-1}) = wr(w(0, a_i), a_{i-1}) = wr(w(0, a_i), a_i) - 1 = wr(b_i, a_i) - 1 \implies wr(b_i, a_{i-1}) = wr(b_i, a_i) - 1$, meaning that there are no other men between a_{i-1} and a_i in b_i's list so, by definition of rotation elimination, only a_i is deleted from b_i's list and, as a result, only b_i is removed from a_i's list and no woman is removed from any other men's list.

We now assume that (i) is true for all nodes numbered up to k and we will prove that it must also be true for the node numbered $k+1$. Starting from the set of shortlists $\mathscr{M}$, for each node j that is a predecessor of $k+1$, there is by the induction hypothesis, a rotation p_j that may be eliminated, provided that the procedure is done in topological order. Let $\mathscr{L}$ be the stable set obtained from $\mathscr{M}$ after this set of rotation eliminations. Let's consider the preference list of a_i in $\mathscr{L}$. Since a_i is in the label of node $k+1$ we have 2 possible cases:

Case 1: Node $k+1$ is $init(a_i)$ or node $k+1$ is $term(a_i)$ and $init(a_i)$ is the source. Then a_i in not in any label set of any predecessor j of the node $k+1$ (the source is not taken into consideration as it does not take part in the processing). As a result, we have that a_i is not in any of the respective rotations p_j and that after the processing of the first k nodes the only woman on a_i's list is the first woman on his list. Thus after the sequence of rotation eliminations, no woman has been removed from a_i's list. Therefore, $first_{\mathscr{L}}(a_i) = first_{\mathscr{M}}(a_i) = w(k, a_i)$

Case 2: Node $k+1$ is $term(a_i)$ and $init(a_i)$ is not the source. Then node $init(a_i)$ is a predecessor of $term(a_i)$ and a_i is also in the label of the node $init(a_i)$. As a result, starting from the set of shortlists $\mathscr{M}$, by eliminating rotation $p_{init(a_i)}$, by induction hypothesis, only woman $first_{\mathscr{M}}(a_i)$ is removed from man a_i's list. Moreover, since a_i is in the label of one of the predecessors of node $k+1$, then after the processing of the first k nodes, there are 2 women in a_i's list so $w(k, a_i)$ is the second woman on a_i's shortlist. Hence, $first_{\mathscr{L}}(a_i) = second_{\mathscr{M}}(a_i) = w(k, a_i)$ since $first_{\mathscr{M}}(a_i)$ was deleted from a_i's list during rotation elimination.

So, in both cases we gave $first_{\mathscr{L}}(a_i) = w(k, a_i)$. Furthermore, since from the construction of the stable marriage instance we have

$$mr(a_i, w(k, a_{i+1})) = 1 + mr(a_i, w(k, a_i))$$

then, in both cases,

$$second_{\mathscr{L}}(a_i) = w(k, a_{i+1}) = first_{\mathscr{L}}(a_{i+1})$$

Therefore, by the definition of a rotation, p_{k+1} specified as in (i) is a rotation exposed in $\mathscr{L}$.

Now let's assume that there is a rotation p_j that has not been eliminated, where node j is a predecessor of node $k+1$. Then there is a rotation p_m, where node m is an immediate predecessor of node $k+1$, such that j is a predecessor of m so since p_j has not been eliminated, then p_m has not been eliminated either. As a result, there is an a_i in the label set of node $k+1$ such that node m is $init(a_i)$ and $k+1$ is $term(a_i)$ and $p_m = p_{init(a_i)}$ has not been eliminated. Hence,

$$first_{\mathscr{L}}(a_i) = first_{\mathscr{M}}(a_i) \neq w(k, a_i)$$

so the rotation p_{k+1} is not exposed, meaning that p_j has to be eliminated so it is a predecessor of p_{k+1}. As a result, rotation p_{k+1} has as predecessors precisely all of the rotations p_j for which node j is a predecessor of node k. Finally, when p_{k+1} is eliminated, then, since $wr(b_i, a_{i-1}) = wr(b_i, a_i) - 1$ for each i we have that the only entries removed from the men's lists are the b_i from the list of a_i, for $0 \leq i \leq r-1$.

(ii) If w is not in m's shortlist, then by the way we defined a rotation, (m, w) cannot be in a rotation. Moreover, if $w = last_{\mathscr{M}}(m)$ for the set $\mathscr{M}$ of shortlists then (m, w) cannot be in a rotation: If it was in a rotation exposed in $\mathscr{M}$, then $last_{\mathscr{M}}(m) = w = first_{\mathscr{M}}(m) \implies$ the only woman on m's shortlist is w and the only man on w's shortlist is m, which is a contradiction since $second(m)$ must be defined for all m in a rotation. Similarly it can not be in a rotation exposed in another stable set $\mathscr{L}$ since then we would have $last_{\mathscr{L}}(m) = w = first_{\mathscr{L}}(m)$.

All the other pairs (m, w) are in one of the rotations p_k described above and since, from lemma 5.19 no pair can belong to 2 different rotations, then there are no rotations other than those described in (i).

□

Proof of Theorem 5.3. From the given construction and lemma 5.25 we have an instance of the stable marriage problem constructible from $(P, \leq)$ in time polynomial in n. From Theorem 5.24 the stable matchings of the constructed instance are in a one-to-one correspondence with the antichains of its rotation poset. By Lemma 5.27, the rotations are in a one-to-one correspondence with the nodes of the graph, where the nodes of the graph are in a one-to-one correspondence with the elements of the poset P, and the partial order in the set of rotations coincides with the partial order in the original poset P. As a result, the antichains of the rotations poset are in a one-to-one correspondence with the antichains of poset $(P, \leq)$ so the stable matchings are in a one-to-one correspondence with the antichains of $(P, \leq)$. □

Corollary 5.27.1. *Determining the number of stable matchings for an instance of a stable marriage problem is* $\#P-$*complete.*

References

[1] D. Gale and L. S. Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.

[2] Michael R. Garey and David S. Johnson. *Computers and intractability: A guide to the theory of NP-completeness.* W.H. Freeman & Co., 1979.

[3] Dan Gusfield and Robert W. Irving. *The Stable Marriage Problem: Structure and Algorithms.* The MIT Press, 1989.

[4] Robert W. Irving and Paul Leather. The complexity of counting stable marriages. *SIAM Journal on Computing*, 15(3):655–667, 1986.

[5] Jon Kleinberg and Eva Tardos. *Algorithm Design.* Pearson Education, 2006.

[6] Anany Levitin. *Introduction to the Design and Analysis of Algorithms, 3rd edition.* Pearson Education, 2012.

[7] D. J. McVitie and L. B. Wilson. The stable marriage problem. *Communications of the ACM*, 14(7):486–492, 1971.

[8] Cristopher Moore and Stephan Mertens. *The Nature of Computation.* Oxford University Press, 2011.

[9] Rajeev Motwani and Prabhakar Raghavan. *Randomized Algorithms.* Cambridge University Press, 1995.

[10] J. Scott Provan and Michael O. Ball. The complexity of counting cuts and of computing the probability that a graph is connected. *SIAM Journal on Computing*, 12(4):777–788, 1983.

[11] Carla P. Gomes, Ashish Sabharwal, Bart Selman. Frontiers in artificial intelligence and applications. In Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsch, editors, *Handbook of satisfiability*, chapter 20. Model Counting. IOS Press, 2009.

[12] Michael Sipser. *Theory of Computation, 2nd edition.* Thomson Course Technology, 2006.

[13] Thomas Cormen, Charles Leiserson, Ronald Rivest, Clifford Stein. *Introduction to algorithms, 3rd edition.* The MIT Press, 2009.

[14] Donald E. Knuth (translated by Martin Goldstein). *Stable Marriage and its Relation to Other Combinatorial Problems*. American Mathematical Society, 1938.