Stable Matchings, Robust Solutions, and Distributive Lattices

Tung Mai¹ and Vijay V. Vazirani¹

¹University of California, Irvine

Abstract

We build on our recent paper [MV18a], which initiated the problem of finding stable matchings that are robust to errors in the input and gave a polynomial time algorithm for a special case in which the domain of errors was polynomially large. We had observed that this set of errors is too restrictive and had asked the question of extending the domain. In [MV18a] we had also initiated work on a new structural question regarding stable matchings, namely finding relationships between the lattices of solutions of two “nearby” instances (in this case, the given and the perturbed instances). In this paper, we make substantial progress on both fronts.

First, we extend the domain of errors to a super-exponentially large one, though we need to somewhat weaken the notion of “robust”. Underlying our polynomial time algorithms are new structural properties, of the type described above. A key to deriving these structural properties is a purely combinatorial proof of a generalization of Birkhoff’s Theorem for finite distributive lattices, derived here in the context of stable matching lattices. This proof yields crucial new notions, such as that of a meta-rotation, and is therefore of independent interest.

1 Introduction

The two topics, of stable matchings and the design of algorithms that produce robust solutions, have been studied intensively for decades and there are today several books on each of these topics, e.g., see [Knu97, GI89, Man13] and [CE06, BTEGN09]. Yet, at the intersection of these two topics lies a single work, namely [MV18a]. The contribution of that recent paper of ours was two-fold: First, it introduced the problem of finding a stable matching that is robust to errors in the input and gave a polynomial time algorithm for a special case. Second, it initiated work on a new structural question, namely finding relationships between the lattices of solutions of two “nearby” instances of stable matching (in this case, the given and the perturbed instances). In the current paper, we make substantial progress on both fronts.

The domain of errors considered in [MV18a] was too restrictive, it was polynomial sized, and we had asked the question of extending the domain. In the current paper, we extend the domain to a super-exponentially large one, though we need to somewhat weaken the notion of “robust”. Underlying our polynomial time algorithms are new structural properties, of the type described above, of lattices of stable matchings.

Over the last half century, the stable matching problem [GS62] has been the subject of intense
study from numerous different angles in many fields, including computer science, mathematics, operations research, economics and game theory, e.g., see [Knu97, GI89, Man13]. Over these decades, researchers have unearthed the deep and pristine combinatorial structure of this problem, which in turn has led to efficient algorithms for numerous questions studied about this problem. In 2012, this problem was a subject of the Nobel Prize in Economics, awarded to Roth and Shapley [RS12]. To the best of our knowledge, the issue of robust solutions had not been studied in the context of this problem until [MV18a] (see also Section 1.4), even though the design of algorithms that produce robust solutions is already a very well established field, especially as pertaining to optimization, e.g., see [CE06, BTEGN09].

The setting defined in [MV18a] was the following: Given an instance $A$ of stable matching on $n$ boys and $n$ girls, let $B$ be the instance that results after introducing one error from a domain $D$, chosen via a given discrete probability distribution. We had defined a robust stable matching as a matching that is stable for $A$ and maximizes the probability of being stable for $B$ as well. The domain, $D$, of errors was defined via an operation called shift. For a girl $g$, assume her preference list in instance $A$ is \{$\ldots, b_1, b_2, \ldots, b_k, b, \ldots$\}. Move up the position of $b$ so $g$’s list becomes \{$\ldots, b, b_1, b_2, \ldots, b_k, \ldots$\}, and let $B$ denote the resulting instance. An analogous operation is defined on a boy $b$’s list; again some girl $g$ on his list is moved up. For each girl and each boy, consider all possible such shifts to get the domain $D$; clearly, $|D| = O(n^3)$.

The setting of the current paper is: Let $A$ be as above and let $T$ denote the set of all possible instances, $B$, obtained by introducing one error of the following type in $A$: For any one girl or any one boy, arbitrarily permute of the preference list of the girl or the boy. Clearly $|T| = 2n(n!)$. Let $D \subset T$ be an arbitrary polynomial sized set. Define a fully robust stable matching to be a matching that is stable for $A$ and for each of the instances in $D$. Our main algorithmic result is:

**Theorem 1.** For the setting given above, there is a polynomial time algorithm for checking if there is a fully robust stable matching. If the answer is yes, the set of all such matchings form a sublattice of $L$ and our algorithm finds a partial order that generates this sublattice.

Terms used in Theorem 1 are standard but are also defined in the next section. The new structural properties which support this result are described in Section 1.2.

### 1.1 The lattice of stable matchings

Conway, see [Knu97], proved that the set of stable matchings of an instance forms a lattice, with the meet and join of two stable matchings being the operations of taking the boy-optimal choices and girl-optimal choices, respectively, of the two matchings. Knuth [Knu97] asked if every finite distributive lattice is isomorphic to the lattice arising from an instance of stable matching. A positive answer was provided by Blair [Bla84]; for a much better proof, see [GI89].

It is easy to see that the family of closed sets of a partial order, say $\Pi$, is closed under union and intersection and forms a distributive lattice, with join and meet being these two operations, respectively; let us denote it by $L(\Pi)$. Birkhoff’s theorem [Bir37], which has also been called the fundamental theorem for finite distributive lattices, e.g., see [Sta96], states that corresponding to any
finite distributed lattice, $\mathcal{L}$, there is a partial order, say $\Pi$, whose lattice of closed sets $L(\Pi)$ is isomorphic to $\mathcal{L}$, i.e., $\mathcal{L} \cong L(\Pi)$. We will say that $\Pi$ generates $\mathcal{L}$.

For the lattice of stable matchings, the partial order $\Pi$ defined in Birkhoff’s Theorem, has additional useful structural properties. First, its elements are rotations. A rotation takes $r$ matched boy-girl pairs in a fixed order, say $\{b_0g_0, b_1g_1, \ldots, b_{r-1}g_{r-1}\}$, and “cyclically” changes the mates of these $2r$ agents, see Section 2.3 for details. The number $r$, the $r$ pairs, and the order among the pairs are so chosen that when a rotation is applied to a stable matching containing all $r$ pairs, the resulting matching is also stable. Moreover, there is no valid rotation on any subset of these $r$ pairs, under any ordering. Hence, a rotation can be viewed as a minimal change to the current matching that results in a stable matching. Any boy–girl pair, $(b, g)$, belongs to at most one rotation. Consequently, the set $R$ of rotations underlying $\Pi$ satisfies $|R| = O(n^2)$, and hence, $\Pi$ is a succinct representation of $\mathcal{L}$; the latter can be exponentially large. $\Pi$ will be called the rotation poset for $\mathcal{L}$.

Second, the rotation poset helps traverse the lattice as follows. For any closed set $S$ of $\Pi$, the corresponding stable matching $M(S)$ can be obtained as follows: start from the boy-optimal matching in the lattice and apply the rotations in set $S$, in any topological order consistent with $\Pi$. The resulting matching will be $M(S)$. In particular, applying all rotations in $R$, starting from the boy-optimal matching, leads to the girl-optimal matching.

### 1.2 Overview of structural results

We start by giving a short overview of the structural facts proven in [MV18a]. Let $A$ and $B$ be two instances of stable matching over $n$ boys and $n$ girls, with sets of stable matchings $\mathcal{M}_A$ and $\mathcal{M}_B$, and lattices $\mathcal{L}_A$ and $\mathcal{L}_B$, respectively. Let $\Pi$ be the poset on rotations that is isomorphic to $\mathcal{L}_A$. It is easy to see that if $B$ is obtained from $A$ by changing the lists of only one gender, either boys or girls, but not both, then the matchings in $\mathcal{M}_A \cap \mathcal{M}_B$ form a sublattice in each of the two lattices (Proposition 2). [MV18a] further show that if $B$ is obtained by applying a shift operation, then $\mathcal{M}_{AB} = \mathcal{M}_A \setminus \mathcal{M}_B$ is also a sublattice of $\mathcal{L}_A$. Using this fact, they show that there is at most one rotation, $\rho_{in}$, that leads from $\mathcal{M}_A \cap \mathcal{M}_B$ to $\mathcal{M}_{AB}$ and at most one rotation, $\rho_{out}$ that leads from $\mathcal{M}_{AB}$ to $\mathcal{M}_A \cap \mathcal{M}_B$; moreover, these rotations can be efficiently found. Furthermore, for a closed set $S$ of $\Pi$, $M(S)$ is stable for instance $B$ iff $\rho_{in} \in S \Rightarrow \rho_{out} \in S$.

Our failure at extending the domain of errors in any straightforward manner led to the following abstract question. Suppose $B$ is such that $\mathcal{M}_A \cap \mathcal{M}_B$ and $\mathcal{M}_{AB}$ are both sublattices of $\mathcal{L}_A$, i.e., $\mathcal{M}_A$ is partitioned into two sublattices. Then is there a polynomial time algorithm for finding a matching in $\mathcal{M}_A \cap \mathcal{M}_B$? This is one of the two abstract questions we will answer in this paper and hence we will call it Case I (See Section 5).

Since we are dealing with arbitrary sublattices, say $\mathcal{L}'$, of a lattice, say $\mathcal{L}$, it would be useful to find a way of obtaining the partial order, $\Pi'$, which generates $\mathcal{L}'$ from the partial order, $\Pi$, that generates $\mathcal{L}$. A generalization of Birkhoff’s Theorem, proved within category theory [Wika] provides an avenue towards answering this question. In this paper, we provide a purely combinatorial proof in the setting of lattices of stable matching. The advantage being that it gives some crucial notions and insights, such as that the central notion of meta-rotation; just as rotations help traverse a lattice, meta-rotations help traverse a sublattice. An outline of our proof
is given in Section 1.3. We give the notion of a compression, which when applied to $\Pi$ yields another partial order $\Pi'$; it is formally defined in Definition 1. We prove that there is a one-to-one correspondence between all possible sublattices of $\mathcal{L}$ and all possible compressions of $\Pi$ such that if $L'$ corresponds to $\Pi'$, then $L(\Pi') = L'$.

Via the machinery developed above, we can answer the stated question, i.e., suppose lattice $\mathcal{L}_A$ is partitioned into two sublattices over $\mathcal{M}_A \cap \mathcal{M}_B$ and $\mathcal{M}_{AB}$, then how do you generate a matching in $\mathcal{M}_A \cap \mathcal{M}_B$? We prove that there exists a sequence of rotations $r_0, r_1, \ldots, r_{2k}, r_{2k+1}$ such that a closed set of $\Pi$ generates a matching in $M \in \mathcal{M}_A \cap \mathcal{M}_B$ iff it contains $r_{2i}$ but not $r_{2i+1}$ for some $0 \leq i \leq k$ (Proposition 8). Furthermore, this sequence of rotations can be found in polynomial time. However, so far this abstract fact has not yielded a concrete error pattern, beyond shift, that we can add to our domain.

Next, we address the case that $\mathcal{M}_{AB}$ is not a sublattice of $\mathcal{L}_A$. We start by proving that if $B$ is obtained by permuting the preference list of any one boy, then $\mathcal{M}_{AB}$ must be a join semi-sublattice of $\mathcal{L}_A$ (Lemma 24). Hence our second abstract case, which we will call Case II, is that lattice $\mathcal{L}_A$ is partitioned into a sublattice and a join semi-sublattice (see Section 6). For this case, we obtain a more elaborate characterization of compressions that generate the sublattice of $\mathcal{M}_A \cap \mathcal{M}_B$ (Theorem 9), and a more elaborate condition on rotations which is satisfied by a closed set of $\Pi$ iff the corresponding matching is in the sublattice (Proposition 10). Furthermore, we show how to efficiently find these rotations (Theorem 11), hence leading to an efficient algorithm for finding a matching in $\mathcal{M}_A \cap \mathcal{M}_B$.

Finally, consider the setting given in the Introduction, with $T$ being the super-exponential set of all possible erroneous instances obtained by permuting the preference list of one boy or one girl, and $D \subset T$ a polynomial sized set of instances which the algorithm needs to consider. We show that the set of all such matchings that are stable for $A$ and for each of the instances in $D$ forms a sublattice of $\mathcal{L}$ and we obtain the compression of $\Pi$ that generates this sublattice (Section 8.2). Each matching in this sublattice is a fully robust stable matching. Moreover, since we have obtained the poset generating it, we can go further: given a weight function on all boy-girl pairs, we can obtain, using the algorithm of [MV18b], a matching that optimizes (maximizes or minimizes) the weight among all fully robust stable matchings.

1.3 Our proof of the generalization of Birkhoff’s Theorem

As stated above, we will prove the generalization (Theorem 4) in the context of stable matching lattices; as remarked earlier, such lattices are as general as arbitrary finite distributive lattices. Let $\mathcal{L}$ be a stable matching lattice which is generated by poset $\Pi$. We first give one definition of compression, in Definition 1, which when applied to $\Pi$ yields another partial order $\Pi'$. Our proof involves showing that each compression $\Pi'$ of $\Pi$ generates a sublattice of $\mathcal{L}$ (Section 3.1), and corresponding to each sublattice $\mathcal{L}'$ of $\mathcal{L}$, there is a compression $\Pi'$ of $\Pi$ that generates $\mathcal{L}'$ (Section 3.2).

The second part is quite non-trivial. It involves first identifying the correct partition of the set of rotations of $\Pi$ by considering pairs of matchings, $M, M'$ in $\mathcal{L}'$ such that $M$ is a direct successor of the $M'$, and obtaining the set of rotations that takes us from $M'$ to $M$. This set will be a meta-rotation for $\Pi'$. Consider one such meta-rotation $X$. To obtain all predecessors of $X$ in $\Pi'$,
consider all paths that go from the boy-optimal matching in \( L \) to the girl-optimal matching by going through the lattice \( L' \). Find all meta-rotations that *always* occur before \( X \) does on all such paths. Then each of these meta-rotations precedes \( X \). These are the precedence relations between meta-rotations in \( \Pi' \).

**A second definition of compression:** We next present a different, equivalent, definition of compression (Section 4). This definition is in terms of a set of directed edges, \( E \), that needs to be added to \( \Pi \) to yield, after some prescribed operations, the desired partial order \( \Pi' \). Let \( L' \) be the sublattice generated by \( \Pi' \). Then we will say that edges \( E \) *define* \( L' \).

The advantage of this definition is that it is much easier to work with for the applications presented later. Its drawback is that several different sets of edges may yield the same compression. Therefore, there is no one-to-one correspondence between sublattices of \( L \) and the sets of edges that can be added to \( \Pi \) to yield compressions. Hence this definition is not suitable for proving the generalization of Birkhoff’s Theorem.

### 1.4 A matter of nomenclature

Assigning correct nomenclature to a new issue under investigation is clearly critical for ease of comprehension. In this context we wish to mention that very recently, Genc et. al. [GSOS17] defined the notion of an \((a,b)\)-supermatch as follows: this is a stable matching in which if any \( a \) pairs break up, then it is possible to match them all off by changing the partners of at most \( b \) other pairs, so the resulting matching is also stable. They showed that it is NP-hard to decide if there is an \((a,b)\)-supermatch. They also gave a polynomial time algorithm for a very restricted version of this problem, namely given a stable matching and a number \( b \), decide if it is a \((1,b)\)-supermatch. Observe that since the given instance may have exponentially many stable matchings, this does not yield a polynomial time algorithm even for deciding if there is a stable matching which is a \((1,b)\)-supermatch for a given \( b \).

Genc. et. al. [GSSO17] also went on to defining the notion of the most robust stable matching, namely a \((1,b)\)-supermatch where \( b \) is minimum. We would like to point out that “robust” is a misnomer in this situation and that the name “fault-tolerant” is more appropriate. In the literature, the latter is used to describe a system which continues to operate even in the event of failures and the former is used to describe a system which is able to cope with erroneous inputs, e.g., see the following pages from Wikipedia [Wikc, Wikb].

### 2 Preliminaries

#### 2.1 The stable matching problem

The stable matching problem takes as input a set of boys \( B = \{b_1, b_2, \ldots, b_n\} \) and a set of girls \( G = \{g_1, g_2, \ldots, g_n\} \); each person has a complete preference ranking over the set of opposite sex. The notation \( b_i < g \) indicates that girl \( g \) strictly prefers \( b_i \) to \( b_j \) in her preference list. Similarly, \( g_i < b \) indicates that the boy \( b \) strictly prefers \( g_j \) to \( g_i \) in his list.
A matching \( M \) is a one-to-one correspondence between \( B \) and \( G \). For each pair \( bg \in M \), \( b \) is called the partner of \( g \) in \( M \) (or \( M \)-partner) and vice versa. For a matching \( M \), a pair \( bg \not\in M \) is said to be blocking if they prefer each other to their partners. A matching \( M \) is stable if there is no blocking pair for \( M \).

### 2.2 The lattice of stable matchings

Let \( M \) and \( M' \) be two stable matchings. We say that \( M \) dominates \( M' \), denoted by \( M \preceq M' \), if every boy weakly prefers his partner in \( M \) to \( M' \). It is well known that the dominance partial order over the set of stable matchings forms a distributive lattice [GI89], with meet (greatest lower bound) and join (least upper bound) defined as follows. The meet of \( M \) and \( M' \), \( M \land M' \), is defined to be the matching that results when each boy chooses his more preferred partner (or equivalently, each girl chooses her less preferred partner) from \( M \) and \( M' \); it is easy to show that this matching is also stable. The join of \( M \) and \( M' \), \( M \lor M' \), is defined to be the matching that results when each boy chooses his less preferred partner (or equivalently, each girl chooses her more preferred partner) from \( M \) and \( M' \); this matching is also stable. These operations distribute, i.e., given three stable matchings \( M, M', M'' \),

\[
M \lor (M' \land M'') = (M \lor M') \land (M \lor M'') \quad \text{and} \quad M \land (M' \lor M'') = (M \land M') \lor (M \land M'').
\]

It is easy to see that the lattice must contain a matching, \( M_0 \), that dominates all others and a matching \( M_z \) that is dominated by all others. \( M_0 \) is called the boy-optimal matching, since in it, each boy is matched to his most favorite girl among all stable matchings. This is also the girl-pessimal matching. Similarly, \( M_z \) is the boy-pessimal or girl-optimal matching.

### 2.3 Rotations help traverse the lattice

A crucial ingredient needed to understand the structure of stable matchings is the notion of a rotation, which was defined by Irving [Irv85] and studied in detail in [IL86]. A rotation takes \( r \) matched pairs in a fixed order, say \( \{b_0g_0, b_1g_1, \ldots, b_{r-1}g_{r-1}\} \) and “cyclically” changes the mates of these \( 2r \) agents, as defined below, to arrive at another stable matching. Furthermore, it represents a minimal set of pairings with this property, i.e., if a cyclic change is applied on any subset of these \( r \) pairs, with any ordering, then the resulting matching has a blocking pair and is not stable. After rotation, the boys’ mates weakly worsen and the girls’ mates weakly improve. Thus one can traverse from \( M_0 \) to \( M_z \) by applying a suitable sequence of rotations (specified by the rotation poset defined below). Indeed, this is precisely the purpose of rotations.

Let \( M \) be a stable matching. For a boy \( b \) let \( s_M(b) \) denote the first girl \( g \) on \( b \)'s list such that \( g \) strictly prefers \( b \) to her \( M \)-partner. Let \( next_M(b) \) denote the partner in \( M \) of girl \( s_M(b) \). A rotation \( \rho \) exposed in \( M \) is an ordered list of pairs \( \{b_0g_0, b_1g_1, \ldots, b_{r-1}g_{r-1}\} \) such that for each \( i, 0 \leq i \leq r-1 \), \( b_{i+1} \) is \( next_M(b_i) \), where \( i + 1 \) is taken modulo \( r \). In this paper, we assume that the subscript is taken modulo \( r \) whenever we mention a rotation. Notice that a rotation is cyclic and the sequence of pairs can be rotated. \( M/\rho \) is defined to be a matching in which each boy not in a pair of \( \rho \) stays matched to the same girl and each boy \( b_i \) in \( \rho \) is matched to \( g_{i+1} = s_M(b_i) \). It can
be proven that $M/\rho$ is also a stable matching. The transformation from $M$ to $M/\rho$ is called the elimination of $\rho$ from $M$.

**Lemma 1** ([GI89], Theorem 2.5.4). Every rotation appears exactly once in any sequence of elimination from $M_0$ to $M_z$.

Let $\rho = \{b_0g_0, b_1g_1, \ldots, b_r-1g_{r-1}\}$ be a rotation. For $0 \leq i \leq r - 1$, we say that $\rho$ moves $b_i$ from $g_i$ to $g_{i+1}$, and moves $g_i$ from $b_i$ to $b_{i-1}$. If $g$ is either $g_i$ or is strictly between $g_i$ and $g_{i+1}$ in $b_i$'s list, then we say that $\rho$ moves $b_i$ below $g$. Similarly, $\rho$ moves $g_i$ above $b_i$ if $b$ is $b_i$ or between $b_i$ and $b_{i-1}$ in $g_i$'s list.

### 2.4 The rotation poset

A rotation $\rho'$ is said to precede another rotation $\rho$, denoted by $\rho' \prec \rho$, if $\rho'$ is eliminated in every sequence of eliminations from $M_0$ to a stable matching in which $\rho$ is exposed. If $\rho'$ precedes $\rho$, we also say that $\rho$ succeeds $\rho'$. If neither $\rho' \prec \rho$ nor $\rho' \succ \rho$, we say that $\rho'$ and $\rho$ are incomparable. Thus, the set of rotations forms a partial order via this precedence relationship. The partial order on rotations is called rotation poset and denoted by $\Pi$.

**Lemma 2** ([GI89], Lemma 3.2.1). For any boy $b$ and girl $g$, there is at most one rotation that moves $b$ to $g$, $b$ below $g$, or $g$ above $b$. Moreover, if $\rho_1$ moves $b$ to $g$ and $\rho_2$ moves $b$ from $g$ then $\rho_1 \prec \rho_2$.

**Lemma 3** ([GI89], Lemma 3.3.2). $\Pi$ contains at most $O(n^2)$ rotations and can be computed in polynomial time.

A closed set of a poset is a set $S$ of elements of the poset such that if an element is in $S$ then all of its predecessors are also in $S$. There is a one-to-one relationship between the stable matchings and the closed subsets of $\Pi$. Given a closed set $S$, the corresponding matching $M$ is found by eliminating the rotations starting from $M_0$ according to the topological ordering of the elements in the set $S$. We say that $S$ generates $M$ and that $\Pi$ generates the lattice $\mathcal{L}$ of all stable matchings of this instance.

Let $S$ be a subset of the elements of a poset, and let $v$ be an element in $S$. We say that $v$ is a minimal element in $S$ if there is no predecessors of $v$ in $S$. Similarly, $v$ is a maximal element in $S$ if it has no successors in $S$.

The Hasse diagram of a poset is a directed graph with a vertex for each element in poset, and an edge from $x$ to $y$ if $x \prec y$ and there is no $z$ such that $x \prec z \prec y$. In other words, all precedences implied by transitivity are suppressed.

### 2.5 Sublattice and Semi-sublattice

A sublattice $\mathcal{L}'$ of a distributive lattice $\mathcal{L}$ is subset of $\mathcal{L}$ such that for any two elements $x, y \in \mathcal{L}$, $x \lor y \in \mathcal{L}'$ and $x \land y \in \mathcal{L}'$ whenever $x, y \in \mathcal{L}'$. 


A join semi-sublattice $L'$ of a distributive lattice $L$ is subset of $L$ such that for any two elements $x, y \in L$, $x \vee y \in L'$ whenever $x, y \in L'$.

Similarly, meet semi-sublattice $L'$ of a distributive lattice $L$ is subset of $L$ such that for any two elements $x, y \in L$, $x \wedge y \in L'$ whenever $x, y \in L'$.

Note that $L'$ is a sublattice of $L$ iff $L'$ is both join and meet semi-sublattice of $L$.

**Proposition 2.** Let $A$ be an instance of stable matching and let $B$ be another instance obtained from $A$ by changing the lists of only one gender, either boys or girls, but not both. Then the matchings in $M_A \cap M_B$ form a sublattice in each of the two lattices.

**Proof.** It suffices to show that $M_A \cap M_B$ is a sublattice of $L_A$. Assume $|M_A \cap M_B| > 1$ and let $M_1$ and $M_2$ be two different matchings in $M_A \cap M_B$. Let $\vee_A$ and $\vee_B$ be the join operations under $A$ and $B$ respectively. Likewise, let $\wedge_A$ and $\wedge_B$ be the meet operations under $A$ and $B$.

By definition of join operation in Section 2.2, $M_1 \vee_A M_2$ is the matching obtained by assigning each boy to his less preferred partner (or equivalently, each girl to her more preferred partner) from $M_1$ and $M_2$ according to instance $A$. Without loss of generality, assume that $B$ is an instance obtained from $A$ by changing the lists of only girls. Since the list of each boy is identical in $A$ and $B$, his less preferred partner from $M_1$ and $M_2$ is also the same in $A$ and $B$. Therefore, $M_1 \vee_A M_2 = M_1 \vee_B M_2$. A similar argument can be applied to show that $M_1 \wedge_A M_2 = M_1 \wedge_B M_2$.

Hence, $M_1 \vee_A M_2$ and $M_1 \wedge_A M_2$ are both in $M_A \cap M_B$ as desired.

**Corollary 1.** Let $A$ be an instance of stable matching and let $B_1, \ldots, B_k$ be other instances obtained from $A$ each by changing the lists of only one gender, either boys or girls, but not both. Then the matchings in $M_A \cap M_{B_1} \cap \ldots \cap M_{B_k}$ form a sublattice in $M_A$.

**Proof.** Assume $|M_A \cap M_{B_1} \cap \ldots \cap M_{B_k}| > 1$ and let $M_1$ and $M_2$ be two different matchings in $M_A \cap M_{B_1} \cap \ldots \cap M_{B_k}$. Therefore, $M_1$ and $M_2$ are in $M_A \cap M_{B_i}$ for each $1 \leq i \leq k$. By Proposition 2, $M_A \cap M_{B_i}$ is a sublattice of $L_A$. Hence, $M_1 \vee_A M_2$ and $M_1 \wedge_A M_2$ are in $M_A \cap M_{B_i}$ for each $1 \leq i \leq k$. The claim then follows.

In Section 8.1, we show that for any instance $B$ obtained by permuting the preference list of one boy or one girl, $M_{AB}$ forms a semi-sublattice of $L_A$ (Lemma 24). In particular, if the list of a boy is permuted, $M_{AB}$ forms a join semi-sublattice of $L_A$, and if the list of a girl is permuted, $M_{AB}$ forms a meet semi-sublattice of $L_A$. In both cases, $M_A \cap M_B$ is a sublattice of $L_A$ according to Proposition 2, and the set of matchings in $M_A \cap M_B$ can be characterized in the same manner.

### 2.6 Robust Stable Matching

Let $A$ be a stable matching instance, and let $D$ be a discrete probability distribution over stable matching instances. A **robust stable matching** is a stable matching $M \in M_A$ maximizing the probability that $M \in M_A \cap M_B$, where $B \sim D$. We denote $x \succ^y y'$ if $y$ prefers $x$ to $y'$ with respect to instance $I$. When the probability is 1, $M$ is said to be a **fully robust stable matching**. In other words, $M \in M_B$ for all $B$ in the domain of $D$. 

8
3 A Generalization of Birkhoff’s Theorem

Let $\Pi$ be a finite poset. For simplicity of notation, in this paper we will assume that $\Pi$ must have two dummy elements $s$ and $t$; the remaining elements will be called proper elements and the term element will refer to proper as well as dummy elements. Further, $s$ precedes all other elements and $t$ succeeds all other elements in $\Pi$. A proper closed set of $\Pi$ is any closed set that contains $s$ and does not contain $t$. It is easy to see that the set of all proper closed sets of $\Pi$ form a distributive lattice under the operations of set intersection and union. We will denoted this lattice by $L(\Pi)$. Birkhoff’s Theorem states that every finite distributive lattice is isomorphic to the proper closed sets of some poset.

**Theorem 3.** (Birkhoff [Bir37]) Every finite distributive lattice $\mathcal{L}$ is isomorphic to $L(\Pi)$, for some finite poset $\Pi$.

Our proof of the generalization of Birkhoff’s Theorem deals with the sublattices of a finite distributive lattice. First, in Definition 1 we state the critical operation of compression of a poset.

**Definition 1.** Given a finite poset $\Pi$, first partition its elements; each subset will be called a meta-element. Define the following precedence relations among the meta-elements: if $x, y$ are elements of $\Pi$ such that $x$ is in meta-element $X$, $y$ is in meta-element $Y$ and $x$ precedes $y$, then $X$ precedes $Y$. Assume that these precedence relations yield a partial order, say $Q$, on the meta-elements (if not, this particular partition is not useful for our purpose). Let $\Pi'$ be any partial order on the meta-elements such that the precedence relations of $Q$ are a subset of the precedence relations of $\Pi'$. Then $\Pi'$ will be called a compression of $\Pi$. Let $A_s$ and $A_t$ denote the meta-elements of $\Pi'$ containing $s$ and $t$, respectively.

For examples of compressions see Figure 1. Clearly, $A_s$ precedes all other meta-elements in $\Pi'$ and $A_t$ succeeds all other meta-elements in $\Pi'$. Once again, by a proper closed set of $\Pi'$ we mean a closed set of $\Pi'$ that contains $A_s$ and does not contain $A_t$. Then the lattice formed by the set of all proper closed sets of $\Pi'$ will be denoted by $L(\Pi')$.

**Theorem 4.** [Wika] There is a one-to-one correspondence between the compressions of $\Pi$ and the sublattices of $L(\Pi)$. Furthermore, if a sublattice $\mathcal{L}'$ of $L(\Pi)$ corresponds to compression $\Pi'$, then $\mathcal{L}'$ is isomorphic to $L(\Pi')$.

We will prove Theorem 4 in the context of stable matching lattices; this is w.l.o.g. since stable matching lattices are as general as finite distributive lattices. In this context, the proper elements of partial order $\Pi$ will be rotations, and meta-elements are called meta-rotations. Let $\mathcal{L} = L(\Pi)$ be the corresponding stable matching lattice.

Clearly it suffices to show that:

- Given a compression $\Pi'$, $L(\Pi')$ is isomorphic to a sublattice of $\mathcal{L}$.
- Any sublattice $\mathcal{L}'$ is isomorphic to $L(\Pi')$ for some compression $\Pi'$.

These two proofs are given in Sections 3.1 and 3.2, respectively.
Figure 1: Two examples of compressions. Lattice $\mathcal{L} = L(P)$. $P_1$ and $P_2$ are compressions of $P$, and they generate the sublattices in $\mathcal{L}$, of red and blue elements, respectively.
3.1 $L(\Pi')$ is isomorphic to a sublattice of $L(\Pi)$

Let $I$ be a closed subset of $\Pi'$; clearly $I$ is a set of meta-rotations. Define $\text{rot}(I)$ to be the union of all meta-rotations in $I$, i.e.,

$$\text{rot}(I) = \{p \in A : A \text{ is a meta-rotation in } I\}.$$ 

We will define the process of elimination of a meta-rotation $A$ of $\Pi'$ to be the elimination of the rotations in $A$ in an order consistent with partial order $\Pi$. Furthermore, elimination of meta-rotations in $I$ will mean starting from stable matching $M_0$ in lattice $L$ and eliminating all meta-rotations in $I$ in an order consistent with $\Pi'$. Observe that this is equivalent to starting from stable matching $M_0$ in $L$ and eliminating all rotations in $\text{rot}(I)$ in an order consistent with partial order $\Pi$. This follows from Definition 1, since if there exist rotations $x, y$ in $\Pi$ such that $x$ is in meta-rotation $X$, $y$ is in meta-rotation $Y$ and $x$ precedes $y$, then $X$ must also precede $Y$. Hence, if the elimination of all rotations in $\text{rot}(I)$ gives matching $M_I$, then elimination of all meta-rotations in $I$ will also give the same matching.

Finally, to prove the statement in the title of this section, it suffices to observe that if $I$ and $J$ are two proper closed sets of the partial order $\Pi'$ then

$$\text{rot}(I \cup J) = \text{rot}(I) \cup \text{rot}(J) \quad \text{and} \quad \text{rot}(I \cap J) = \text{rot}(I) \cap \text{rot}(J).$$

It follows that the set of matchings obtained by elimination of meta-rotations in a proper closed set of $\Pi'$ are closed under the operations of meet and join and hence form a sublattice of $L$.

3.2 Any sublattice $\mathcal{L}'$ of $L$ is isomorphic to $L(\Pi')$, for a compression $\Pi'$ of $\Pi$

We will obtain compression $\Pi'$ of $\Pi$ in stages. First, we show how to partition the set of rotations of $\Pi$ to obtain the meta-rotations of $\Pi'$. We then find precedence relations among these meta-rotations to obtain $\Pi'$. Finally, we show $L(\Pi') = \mathcal{L}'$.

Notice that $L$ can be represented by its Hasse diagram $H(L)$. Each edge of $H(L)$ contains exactly one (not necessarily unique) rotation of $\Pi$. Then, by Lemma 1, for any two stable matchings $M_1, M_2 \in L$ such that $M_1 \prec M_2$, all paths from $M_1$ to $M_2$ in $H(L)$ contain the same set of rotations.

**Definition 2.** For $M_1, M_2 \in \mathcal{L}'$, $M_2$ is said to be an $\mathcal{L}'$-direct successor of $M_1$ iff $M_1 \prec M_2$ and there is no $M \in \mathcal{L}'$ such that $M_1 \prec M \prec M_2$. Let $M_1 \prec \ldots \prec M_k$ be a sequence of matchings in $\mathcal{L}'$ such that $M_{i+1}$ is an $\mathcal{L}'$-direct successor of $M_i$ for all $1 \leq i \leq k - 1$. Then any path in $H(L)$ from $M_1$ to $M_k$ containing $M_i$, for all $1 \leq i \leq k - 1$, is called an $\mathcal{L}'$-path.

Let $M_0'$ and $M_z'$ denote the boy-optimal and girl-optimal matchings, respectively, in $\mathcal{L}'$. For $M_1, M_2 \in \mathcal{L}'$ with $M_1 \prec M_2$, let $S_{M_1, M_2}$ denote the set of rotations contained on any $\mathcal{L}'$-path from $M_1$ to $M_2$. Further, let $S_{M_0, M_0'}$ and $S_{M_z', M_z'}$ denote the set of rotations contained on any path from $M_0$ to $M_0'$ and $M_z'$ to $M_z$, respectively in $H(L)$. Define the following set whose elements are sets of rotations.

$$S = \{S_{M_i, M_j} \mid M_j \text{ is an } \mathcal{L}'\text{-direct successor of } M_i, \text{ for every pair of matchings } M_i, M_j \text{ in } \mathcal{L}'\} \cup$$
\{S_{M_0,M_1}, S_{M_2,M_3}\}.

**Lemma 4.** \(S\) is a partition of \(\Pi\).

*Proof.* First, we show that any rotation must be in an element of \(S\). Consider a path \(p\) from \(M_0\) to \(M_2\) in the \(H(L)\) such that \(p\) goes from \(M_0\) to \(M_2\) via an \(L'\)-path. Since \(p\) is a path from \(M_0\) to \(M_2\), all rotations of \(\Pi\) are contained on \(p\) by Lemma 1. Hence, they all appear in the sets in \(S\).

Next assume that there are two pairs \((M_1,M_2) \neq (M_3,M_4)\) of \(L'\)-direct successors such that \(S_{M_1,M_2} \neq S_{M_3,M_4}\) and \(X = S_{M_1,M_2} \cap S_{M_3,M_4} \neq \emptyset\). The set of rotations eliminated from \(M_0\) to \(M_2\) is

\[ S_{M_0,M_2} = S_{M_0,M_1} \cup S_{M_1,M_2}. \]

Similarly,

\[ S_{M_0,M_4} = S_{M_0,M_3} \cup S_{M_3,M_4}. \]

Therefore,

\[
S_{M_0,M_2} \cup M_3 = S_{M_0,M_2} \cup S_{M_1,M_2} \cup S_{M_0,M_1},
\]

\[
S_{M_0,M_1} \cup M_4 = S_{M_0,M_1} \cup S_{M_3,M_4} \cup S_{M_0,M_1}.
\]

Let \(M = (M_2 \cup M_3) \cap (M_1 \cup M_4)\), we have

\[ S_{M_0,M} = S_{M_0,M_2} \cup S_{M_0,M_4} \cup X. \]

Hence,

\[ S_{M_0,M} \cup X = S_{M_0,M_1} \cup X. \]

Since \(X \subset S_{M_1,M_2}\) and \(S_{M_1,M_2} \cap S_{M_0,M_1} = \emptyset\), \(X \cap S_{M_0,M_1} = \emptyset\). Therefore,

\[ S_{M_0,M_1} \subset S_{M_0,M} \subset S_{M_0,M_2}, \]

and hence \(M_2\) is not a \(L'\)-direct successor of \(M_1\), leading to a contradiction. \(\square\)

We will denote \(S_{M_0,M} \) and \(S_{M_2,M_3}\) by \(A_s\) and \(A_t\), respectively. The elements of \(S\) will be the meta-rotations of \(\Pi'\). Next, we need to define precedence relations among these meta-rotations to complete the construction of \(\Pi'\). For a meta-rotation \(A \in S\), \(A \neq A_t\), define the following subset of \(L'\):

\[
\mathcal{M}^A = \{M \in L' \text{ such that } A \subseteq S_{M_0,M}\}.
\]

**Lemma 5.** For each meta-rotation \(A \in S\), \(A \neq A_t\), \(\mathcal{M}^A\) forms a sublattice \(L^A\) of \(L'\).

*Proof.* Take two matchings \(M_1, M_2\) such that \(S_{M_0,M_1}\) and \(S_{M_0,M_2}\) are supersets of \(A\). Then \(S_{M_0,M_1} \cup M_2 = S_{M_0,M_1} \cap S_{M_0,M_2}\) and \(S_{M_0,M_1} \cup M_2 = S_{M_0,M_1} \cup S_{M_0,M_2}\) are also supersets of \(A\). \(\square\)

Let \(M^A\) be the boy-optimal matching in the lattice \(L^A\). Let \(p\) be any \(L'\)-path from \(M_0\) to \(M^A\) and let \(\text{pre}(A)\) be the set of meta-rotations appearing before \(A\) on \(p\).

**Lemma 6.** The set \(\text{pre}(A)\) does not depend on \(p\). Furthermore, on any \(L'\)-path from \(M_0\) containing \(A\), each meta-rotation in \(\text{pre}(A)\) appears before \(A\).
Proof. Since all paths from $M_0'$ to $M^A$ give the same set of rotations, all $L'$-paths from $M_0'$ to $M^A$ give the same set of meta-rotations. Moreover, $A$ must appear last in the any $L'$-path from $M_0'$ to $M^A$; otherwise, there exists a matching in $L^A$ preceding $M^A$, giving a contradiction. It follows that $\text{pre}(A)$ does not depend on $p$.

Let $q$ be an $L'$-path from $M_0'$ that contains matchings $M', M \in L'$, where $M$ is an $L'$-direct successor of $M'$. Let $A$ denote the meta-rotation that is contained on edge $(M', M)$. Suppose there is a meta-rotation $A' \in \text{pre}(A)$ such that $A'$ does not appear before $A$ on $q$. Then $S_{M_0,M^A} = S_{M_0,M^A} \cap S_{M_0,M}$ contains $A$ but not $A'$. Therefore $M^A \wedge M$ is a matching in $L^A$ preceding $M^A$, giving is a contradiction. Hence all matchings in $\text{pre}(A)$ must appear before $A$ on all such paths $q$.

Finally, add precedence relations from all meta-rotations in $\text{pre}(A)$ to $A$, for each meta-rotation in $S \setminus \{A_1\}$. Also, add precedence relations from all meta-rotations in $S \setminus \{A_1\}$ to $A_1$. This completes the construction of $\Pi'$. Below we show that $\Pi'$ is indeed a compression of $\Pi$, but first we need to establish that this construction does yield a valid poset.

**Lemma 7.** $\Pi'$ satisfies transitivity and anti-symmetry.

Proof. First we prove that $\Pi'$ satisfies transitivity. Let $A_1, A_2, A_3$ be meta-rotations such that $A_1 \prec A_2$ and $A_2 \prec A_3$. We may assume that $A_3 \neq A_1$. Then $A_1 \in \text{pre}(A_2)$ and $A_2 \in \text{pre}(A_3)$. Since $A_1 \in \text{pre}(A_2)$, $S_{M_0,M^A_2}$ is a superset of $A_1$. By Lemma 5, $M^{A_1} \prec M^{A_2}$. Similarly, $M^{A_2} \prec M^{A_3}$. Therefore $M^{A_1} \prec M^{A_3}$, and hence $A_1 \in \text{pre}(A_3)$.

Next we prove that $\Pi'$ satisfies anti-symmetry. Assume that there exist meta-rotations $A_1, A_2$ such that $A_1 \prec A_2$ and $A_2 \prec A_1$. Clearly $A_1, A_2 \neq A_1$. Since $A_1 \prec A_2$, $A_1 \in \text{pre}(A_2)$. Therefore, $S_{M_0,M^A_2}$ is a superset of $A_1$. It follows that $M^{A_1} \prec M^{A_2}$. Applying a similar argument we get $M^{A_2} \prec M^{A_1}$. Now, we get a contradiction, since $A_1$ and $A_2$ are different meta-rotations.

**Lemma 8.** $\Pi'$ is a compression of $\Pi$.

Proof. Let $x, y$ be rotations in $\Pi$ such that $x \prec y$. Let $X$ be the meta-rotation containing $x$ and $Y$ be the meta-rotation containing $y$. It suffices to show that $X \in \text{pre}(Y)$. Let $p$ be an $L'$-path from $M_0$ to $M^Y$. Since $x \prec y$, $x$ must appear before $y$ in $p$. Hence, $X$ also appears before $Y$ in $p$. By Lemma 6, $X \in \text{pre}(Y)$ as desired.

Finally, the next two lemmas prove that $L(\Pi') = L'$.

**Lemma 9.** Any matching in $L(\Pi')$ must be in $L'$.

Proof. For any proper closed subset $I$ in $\Pi'$, let $M_I$ be the matching generated by eliminating meta-rotations in $I$. Let $J$ be another proper closed subset in $\Pi'$ such that $J = I \setminus \{A\}$, where $A$ is a maximal meta-rotation in $I$. Then $M_J$ is a matching in $L'$ by induction. Since $I$ contains $A$, $S_{M_0,M_I} \supset A$. Therefore, $M^A \prec M_I$. It follows that $M_I = M_J \cup M^A \in L'$.

**Lemma 10.** Any matching in $L'$ must be in $L(\Pi')$.
Proof. Suppose there exists a matching \( M \) in \( L' \) such that \( M \not\in L(\Pi') \). Then it must be the case that \( S_{M_0,M} \) cannot be partitioned into meta-rotations which form a closed subset of \( \Pi \). Now there are two cases.

First, suppose that \( S_{M_0,M} \) can be partitioned into meta-rotations, but they do not form a closed subset of \( \Pi' \). Let \( A \) be a meta-rotation such that \( S_{M_0,M} \supset A \), and there exists \( B \prec A \) such that \( S_{M_0,M} \not\supset B \). By Lemma 5, \( M \succ M_A \) and hence \( S_{M_0,M} \) is a superset of all meta-rotations in \( \text{pre}(A) \), giving a contradiction.

Next, suppose that \( S_{M_0,M} \) cannot be partitioned into meta-rotations in \( \Pi' \). Since the set of meta-rotations partitions \( \Pi \), there exists a meta-rotation \( X \) such that \( Y = X \cap S_{M_0,M} \) is a non-empty subset of \( X \). Let \( J \) be the set of meta-rotations preceding \( X \) in \( \Pi \).

\((M_J \lor M) \land M^X\) is the matching generated by meta-rotations in \( J \cup Y \). Obviously, \( J \) is a closed subset in \( \Pi' \). Therefore, \( M_J \in L(\Pi') \). By Lemma 9, \( M_J \in L' \). Since \( M, M^X \in L' \), \((M_J \lor M) \land M^X \in L'\) as well. The set of rotations contained on a path from \( M_J \) to \( (M_J \lor M) \land M^X \) in \( H(L) \) is exactly \( Y \). Therefore, \( Y \) can not be a subset of any meta-rotation, contradicting the fact that \( Y = X \cap S_{M_0,M} \) is a non-empty subset of \( X \).

\end{proof}

4 An Alternative View of Compression

In this section we give an alternative definition of compression of a poset; this will be used in the rest of the paper. We are given a poset \( \Pi \) for a stable matching instance; let \( \mathcal{L} \) be the lattice it generates. Let \( H(\Pi) \) denote the Hasse diagram of \( \Pi \). Consider the following operations to derive a new poset \( \Pi' \): Choose a set \( E \) of directed edges to add to \( H(\Pi) \) and let \( H_E \) be the resulting graph. Let \( H' \) be the graph obtained by shrinking the strongly connected components of \( H_E \); each strongly connected component will be a meta-rotation of \( \Pi' \). The edges which are not shrunk will define a DAG, \( H' \), on the strongly connected components. These edges give precedence relations among meta-rotation for poset \( \Pi' \).

Let \( \mathcal{L}' \) be the sublattice of \( \mathcal{L} \) generated by \( \Pi' \). We will say that the set of edges \( E \) defines \( \mathcal{L}' \). It can be seen that each set \( E \) uniquely defines a sublattice \( L(\Pi') \); however, there may be multiple sets that define the same sublattice. Observe that given a compression \( \Pi' \) of \( \Pi \), a set \( E \) of edges defining \( L(\Pi') \) can easily be obtained. See Figure 2 for examples of sets of edges which define sublattices.

**Proposition 5.** The two definitions of compression of a poset are equivalent.

**Proof.** Let \( \Pi' \) be a compression of \( \Pi \) obtained using the first definition. Clearly, for each meta-rotation in \( \Pi' \), we can add edges to \( \Pi \) so the strongly connected component created is precisely this meta-rotation. Any additional precedence relations introduced among incomparable meta-rotations can also be introduced by adding appropriate edges.

The other direction is even simpler, since each strongly connected component can be defined to be a meta-rotation and extra edges added can also be simulated by introducing new precedence constraints.
Figure 2: $E_1$ (red edges) and $E_2$ (blue edges) define the sublattices in Figure 1, of red and blue elements, respectively.

For a (directed) edge $e = uv \in E$, $u$ is called the tail and $v$ is called the head of $e$. Let $I$ be a closed set of $\Pi$. Then we say that:

- $I$ separates an edge $uv \in E$ if $v \in I$ and $u \notin I$.
- $I$ crosses an edge $uv \in E$ if $u \in I$ and $v \notin I$.

If $I$ does not separate or cross any edge $uv \in E$, $I$ is called a splitting set w.r.t. $E$.

**Lemma 11.** Let $L'$ be a sublattice of $L$ and $E$ be a set of edges defining $L'$. A matching $M$ is in $L'$ iff the closed subset $I$ generating $M$ does not separate any edge $uv \in E$.

**Proof.** Let $\Pi'$ be a compression corresponding to $L'$. By Theorem 4, the matchings in $L'$ are generated by eliminating rotations in closed subsets of $\Pi'$.

First, assume $I$ separates $uv \in E$. Moreover, assume $M \in L'$ for the sake of contradiction, and let $I'$ be the closed subset of $\Pi'$ corresponding to $M$. Let $U$ and $V$ be the meta-rotations containing $u$ and $v$ respectively. Notice that the sets of rotations in $I$ and $I'$ are identical. Therefore, $V \in I'$ and $U \notin I'$. Since $uv \in E$, there is an edge from $U$ to $V$ in $H'$. Hence, $I'$ is not a closed subset of $\Pi'$.

Next, assume that $I$ does not separate any $uv \in E$. We show that the rotations in $I$ can be partitioned into meta-rotations in a closed subset $I'$ of $\Pi'$. If $I$ cannot be partitioned into meta-rotations, there must exist a meta-rotation $A$ such that $A \cap I$ is a non-empty proper subset of $A$. Since $A$ consists of rotations in a strongly connected component of $H_E$, there must be an edge $uv$ from $A \setminus I$ to $A \cap I$ in $H_E$. Hence, $I$ separates $uv$. Since $I$ is a closed subset, $uv$ can not be an edge in $H$. Therefore, $uv \in E$, which is a contradiction. It remains to show that the set of meta-rotations partitioning $I$ is a closed subset of $\Pi'$. Assume otherwise, there exist meta-rotation $U \in I'$ and $V \notin I'$ such that there exists an edge from $U$ to $V$ in $H'$. Therefore, there exists $u \in U$, $v \in V$ and $uv \in E$, which is a contradiction. \hfill \Box

**Remark 6.** We may assume w.l.o.g. that the set $E$ defining $L'$ is minimal in the following sense:
There is no edge $uv \in E$ such that $uv$ is not separated by any closed set of $\Pi$. Observe that if there is such an edge, then $E \setminus \{uv\}$ defines the same sublattice $L'$. Similarly, there is no edge $uv \in E$ such that each closed set separating $uv$ also separates another edge in $E$.

**Definition 3.** W.r.t. an element $v$ in a poset $\Pi$, we define four useful subsets of $\Pi$:

\[
I_v = \{r \in \Pi : r < v\} \\
J_v = \{r \in \Pi : r \leq v\} = I_v \cup \{v\} \\
I'_v = \{r \in \Pi : r > v\} \\
J'_v = \{r \in \Pi : r \geq v\} = I'_v \cup \{v\}
\]

Notice that $I_v, J_v, \Pi \setminus I'_v, \Pi \setminus J'_v$ are all closed sets.

**Lemma 12.** Both $J_v$ and $\Pi \setminus J'_v$ separate $uv$ for each $uv \in E$.

**Proof.** Since $uv$ is in $E$, $u$ cannot be in $J_v$; otherwise, there is no closed subset separating $uv$, contradicting Remark 6. Hence, $J_v$ separates $uv$ for all $uv$ in $E$.

Similarly, since $uv$ is in $E$, $v$ cannot be in $J'_u$. Therefore, $\Pi \setminus J'_v$ contains $v$ but not $u$, and thus separates $uv$. $\Box$

## 5 Case I: The Lattice Can be Partitioned into Two Sublattices

In this section we will prove the following theorem:

**Theorem 7.** Let $L_1$ and $L_2$ be sublattices of $L$ such that $L_1$ and $L_2$ partition $L$. Then there exist sets of edges $E_1$ and $E_2$ defining $L_1$ and $L_2$ such that they form an alternating path from $t$ to $s$.

Again, we give a proof in the context of stable matchings. To prove the theorem, we let $E_1$ and $E_2$ be any two sets of edges defining $L_1$ and $L_2$, respectively. We will show that $E_1$ and $E_2$ can be adjusted so that they form an alternating path from $t$ to $s$, without changing the corresponding compressions.

**Lemma 13.** There must exist a path from $t$ to $s$ composed of edges in $E_1$ and $E_2$.

**Proof.** Let $R$ denote the set of vertices reachable from $t$ by a path of edges in $E_1$ and $E_2$. Assume by contradiction that $R$ does not contain $s$. Consider the matching $M$ generated by rotations in $\Pi \setminus R$. Without loss of generality, assume that $M \in L_1$. By Lemma 11, $\Pi \setminus R$ separates an edge $uv \in E_2$. Therefore, $u \in R$ and $v \in \Pi \setminus R$. Since $uv \in E_2$, $v$ is also reachable from $t$ by a path of edges in $E_1$ and $E_2$. $\Box$

Let $Q$ be a path from $t$ to $s$ according to Lemma 13. Partition $Q$ into subpaths $Q_1, \ldots, Q_k$ such that each $Q_i$ consists of edges in either $E_1$ or $E_2$ and $E(Q_i) \cap E(Q_{i+1}) = \emptyset$ for all $1 \leq i \leq k-1$. Let $r_i$ be the rotation at the end of $Q_i$ except for $i = 0$ where $r_0 = t$. Specifically, $t = r_0 \rightarrow r_1 \rightarrow \ldots \rightarrow r_k = s$ in $Q$. We will show that each $Q_i$ can be replaced by a direct edge from $r_{i-1}$ to $r_i$, and furthermore, all edges not in $Q$ can be removed.
Lemma 14. Let $Q_i$ consist of edges in $E_\alpha$ ($\alpha = 1$ or 2). $Q_i$ can be replaced by an edge from $r_i - 1$ to $r_i$ where $r_i - 1 r_i \in E_\alpha$.

Proof. A closed subset separating $r_i - 1 r_i$ must separate an edge in $Q_i$. Moreover, any closed subset must separate exactly one of $r_0 r_1, \ldots, r_k - 2 r_{k-1}, r_{k-1} r_k$. Therefore, the set of closed subsets separating an edge in $E_1$ (or $E_2$) remains unchanged. \hfill \Box

Lemma 15. Edges in $E_1 \cup E_2$ but not in $Q$ can be removed.

Proof. Let $e$ be an edge in $E_1 \cup E_2$ but not in $Q$. Suppose that $e \in E_1$. Let $I$ be a closed subset separating $e$. By Lemma 11, the matching generated by $I$ belongs to $L_2$. Since $e$ is not in $Q$ and $Q$ is a path from $t$ to $s$, $I$ must separate another edge $e'$ in $Q$. By Lemma 11, $I$ can not separate edges in both $E_1$ and $E_2$. Therefore, $e'$ must also be in $E_1$. Hence, the matching generated by $I$ will still be in $L_2$ after removing $e$ from $E_1$. The argument applies to all closed subsets separating $e$. \hfill \Box

By Lemma 14 and Lemma 15, $r_0 r_1, \ldots, r_k - 2 r_{k-1}, r_{k-1} r_k$ are all edges in $E_1$ and $E_2$ and they alternate between $E_1$ and $E_2$. Therefore, we have Theorem 7. An illustration of such a path is given in Figure 3(a).

Proposition 8. There exists a sequence of rotations $r_0, r_1, \ldots, r_{2k}, r_{2k+1}$ such that a closed subset generates a matching in $L_1$ iff it contains $r_{2i}$ but not $r_{2i+1}$ for some $0 \leq i \leq k$.

Figure 3: Examples of: (a) canonical path, and (b) bouquet.
6 Case II: The Lattice Can be Partitioned into a Sublattice and a Semi-Sublattice

Let \( \mathcal{L} \) be a distributive lattice that can be partitioned into a sublattice \( \mathcal{L}_1 \) and a semi-sublattice \( \mathcal{L}_2 \). We assume that \( \mathcal{L}_2 \) is a join semi-sublattice. By reversing the order of \( \mathcal{L} \), the case of meet semi-sublattice is identical. The next theorem, which generalizes Theorem 7, gives a sufficient characterization of a set of edges defining \( \mathcal{L}_1 \).

**Theorem 9.** There exists a set of edges \( E \) defining sublattice \( \mathcal{L}_1 \) such that:

1. The set of tails \( T_E \) of edges in \( E \) forms a chain in \( \Pi \).
2. There is no path of length two consisting of edges in \( E \).
3. For each \( r \in T_E \), let \( F_r = \{ v \in \Pi : rv \in E \} \).
   Then any two rotations in \( F_r \) are incomparable.
4. For any \( r_i, r_j \in T_E \) where \( r_i \prec r_j \), there exists a splitting set containing all rotations in \( F_{r_i} \cup \{ r_j \} \) and no rotations in \( F_{r_j} \cup \{ r_j \} \).

A set \( E \) satisfying Theorem 9 will be called a bouquet. For each \( r \in T_E \), let \( L_r = \{ rv \mid v \in F_r \} \). Then \( L_r \) will be called a flower. Observe that the bouquet \( E \) is partitioned into flowers. These notions are illustrated in Figure 3(b). The black path, directed from \( s \) to \( t \), is the chain mentioned in Theorem 4 and the red edges constitute \( E \). Observe that the tails of edges \( E \) lie on the chain. For each such tail, the edges of \( E \) outgoing from it constitute a flower.

Let \( E \) be an arbitrary set of edges defining \( \mathcal{L}_1 \). We will show that \( E \) can be modified so that the conditions in Theorem 9 are satisfied. Let \( S \) be a splitting set of \( \Pi \). In other words, \( S \) is a closed subset such that for all \( uv \in E \), either \( u, v \) are both in \( S \) or \( u, v \) are both in \( \Pi \setminus S \).

**Lemma 16.** There is a unique maximal rotation in \( T_E \cap S \).

**Proof.** Suppose there are at least two maximal rotations \( u_1, u_2, \ldots, u_k (k \geq 2) \) in \( T_E \cap S \). Let \( v_1, \ldots, v_k \) be the heads of edges containing \( u_1, u_2, \ldots, u_k \). For each \( 1 \leq i \leq k \), let \( S_i = J_{u_i} \cup J_{v_i} \) where \( j \) is any index such that \( j \neq i \). Since \( u_i \) and \( u_j \) are incomparable, \( u_j \not\in J_{u_i} \). Moreover, \( u_j \not\in J_{v_i} \) by Lemma 12. Therefore, \( u_j \not\in S_i \). It follows that \( S_i \) contains \( u_i \) and separates \( u_i, v_i \). Since \( S_i \) separates \( u_i, v_i \in E \), the matching generated by \( S_i \) is in \( \mathcal{L}_2 \) according to Lemma 11.

Since \( \bigcup_{i=1}^{k} S_i \) contains all maximal rotations in \( T_E \cap S \) and \( S \) does not separate any edge in \( E \), \( \bigcup_{i=1}^{k} S_i \) does not separate any edge in \( E \) either. Therefore, the matching generated by \( \bigcup_{i=1}^{k} S_i \) is in \( \mathcal{L}_1 \), and hence not in \( \mathcal{L}_2 \). This contradicts the fact that \( \mathcal{L}_2 \) is a semi-sublattice. \( \square \)

Denote by \( r \) the unique maximal rotation in \( T_E \cap S \). Let
\[
\begin{align*}
R_r &= \{ v \in \Pi : \text{there is a path from } r \text{ to } v \text{ using edges in } E \}, \\
E_r &= \{ uv \in E : u, v \in R_r \}, \\
G_r &= \{ R_r, E_r \}.
\end{align*}
\]
Note that $r \in R_r$. For each $v \in R_r$ there exists a path from $r$ to $v$ and $r \in S$. Since $S$ does not cross any edge in the path, $v$ must also be in $S$. Therefore, $R_r \subseteq S$.

**Lemma 17.** Let $u \in (T_E \cap S) \setminus R_r$ such that $u \succ x$ for $x \in R_r$. Then we can replace each $uv \in E$ with $rv$.

**Proof.** We will show that the set of closed subsets separating an edge in $E$ remains unchanged. Let $I$ be a closed subset separating $uv$. Then $I$ must also separate $rv$ since $r \succ v$.

Now suppose $I$ is a closed subset separating $rv$. We consider two cases:

- If $u \in I$, $I$ must contain $x$ since $u \succ x$. Hence, $I$ separates an edge in the path from $r$ to $x$.
- If $u \not\in I$, $I$ separates $uv$.

Keep replacing edges according to Lemma 17 until there is no $u \in (T_E \cap S) \setminus R_r$ such that $u \succ x$ for some $x \in R_r$.

**Lemma 18.** Let

$$X = \{ v \in S : v \succeq x \text{ for some } x \in R_r \}.$$

1. $S \setminus X$ is a closed subset.
2. $S \setminus X$ contains $u$ for each $u \in (T_E \cap S) \setminus R_r$.
3. $S \setminus X \cap R_r = \emptyset$.
4. $S \setminus X$ is a splitting set.

**Proof.** The lemma follows from the claims given below:

**Claim 1.** $S \setminus X$ is a closed subset.

**Proof.** Let $v$ be a rotation in $S \setminus X$ and $u$ be a predecessor of $v$. Since $S$ is a closed subset, $u \in S$. Notice that if a rotation is in $X$, all of its successor must be included. Hence, since $v \not\in X$, $u \not\in X$. Therefore, $u \in S \setminus X$. □

**Claim 2.** $S \setminus X$ contains $u$ for each $u \in (T_E \cap S) \setminus R_r$.

**Proof.** After replacing edges according to Lemma 17, for each $u \in (T_E \cap S) \setminus R_r$ we must have that $u$ does not succeed any $x \in R_r$. Therefore, $u \not\in X$ by the definition of $X$. □

**Claim 3.** $(S \setminus X) \cap R_r = \emptyset$.

**Proof.** Since $R_r \subseteq X$, $(S \setminus X) \cap R_r = \emptyset$. □
Claim 4. \( S \setminus X \) does not separate any edge in \( E \).

Proof. Suppose \( S \setminus X \) separates \( uv \in E \). Then \( u \in X \) and \( v \in S \setminus X \). By Claim 2, \( u \) can not be a tail vertex, which is a contradiction. \( \square \)

Claim 5. \( S \setminus X \) does not cross any edge in \( E \).

Proof. Suppose \( S \setminus X \) crosses \( uv \in E \). Then \( u \in S \setminus X \) and \( v \in X \). Let \( J \) be a closed subset separating \( uv \). Then \( v \in J \) and \( u \notin J \).

Since \( uv \in E \) and \( u \in S, u \in T_E \cap S \). Therefore, \( r \succ u \) by Lemma 16. Since \( J \) is a closed subset, \( r \notin J \).

Since \( v \in X \), \( v \succeq x \) for \( x \in R_r \). Again, as \( J \) is a closed subset, \( x \in J \).

Therefore, \( J \) separates an edge in the path from \( r \) to \( x \) in \( G_r \). Hence, all closed subsets separating \( uv \) must also separate another edge in \( E_r \). This contradicts the assumption made in Remark 6. \( \square \)

Lemma 19. \( E_r \) can be replaced by the following set of edges:

\[
E'_r = \{ rv : v \in R_r \}.
\]

Proof. We will show that the set of closed subsets separating an edge in \( E_r \) and the set of closed subset separating an edge in \( E'_r \) are identical.

Consider a closed subset \( I \) separating an edge in \( rv \in E'_r \). Since \( v \in R_r \), \( I \) must separate an edge in \( E \) in a path from \( r \) to \( v \). By definition, that edge is in \( E_r \).

Now let \( I \) be a closed subset separating an edge in \( uv \in E_r \). Since \( uv \in E \), \( u \in T_E \cap S \). By Lemma 16, \( r \succ u \). Thus, \( I \) must also separate \( rv \in E'_r \). \( \square \)

Proof of Theorem 9. To begin, let \( S_1 = \Pi \) and let \( r_1 \) be the unique maximal rotation according to Lemma 16. Then we can replace edges according to Lemma 17 and Lemma 19. After replacing, \( r_1 \) is the only tail vertex in \( G_{r_1} \). By Lemma 18, there exists a set \( X \) such that \( S_1 \setminus X \) does not contain any vertex in \( R_{r_1} \) and contains all other tail vertices in \( T_E \) except \( r_1 \). Moreover, \( S_1 \setminus X \) is a splitting set. Hence, we can set \( S_2 = S_1 \setminus X \) and repeat.

Let \( r_1, \ldots, r_k \) be the rotations found in the above process. Since \( r_i \) is the unique maximal rotation in \( T_E \cap S_i \) for all \( 1 \leq i \leq k \) and \( S_1 \supseteq S_2 \supseteq \ldots \supseteq S_k \), we have \( r_1 \succ r_2 \succ \ldots \succ r_k \). By Lemma 19, for each \( 1 \leq i \leq k \), \( E_{r_i} \) consists of edges \( r_i v \) for \( v \in R_{r_{i+1}} \). Therefore, there is no path of length two composed of edges in \( E \) and condition 2 is satisfied. Moreover, \( r_1, \ldots, r_k \) are exactly the tail vertices in \( T_E \), which gives condition 1.

Let \( r \) be a rotation in \( T_E \) and consider \( u, v \in F_r \). Moreover, assume that \( u \prec v \). A closed subset \( I \) separating \( rv \) contains \( v \) but not \( r \). Since \( I \) is a closed subset and \( u \prec v \), \( I \) contains \( u \). Therefore, \( I \) also separates \( ru \), contradicting the assumption in Remark 6. The same argument applies when \( v \prec u \). Therefore, \( u \) and \( v \) are incomparable as stated in condition 3.
Finally, let \( r_i, r_j \in T_E \) where \( r_i \prec r_j \). By the construction given above, \( S_j \supset S_{j-1} \supset \ldots \supset S_i \), \( R_{r_j} \subseteq S_j \setminus S_{j-1} \) and \( R_{r_i} \subseteq S_i \). Therefore, \( S_i \) contains all rotations in \( R_{r_i} \) but none of the rotations in \( R_{r_j} \), giving condition 4. \( \square \)

**Proposition 10.** There exists a sequence of rotations \( r_1 \prec \ldots \prec r_k \) and a set \( F_{r_i} \) for each \( 1 \leq i \leq k \) such that a closed subset generates a matching in \( L_1 \) if and only if whenever it contains a rotation in \( F_{r_i} \), it must also contain \( r_i \).

## 7 Algorithm for Finding a Bouquet

In this section, we give an algorithm for finding a bouquet. Let \( L \) be a distributive lattice that can be partitioned into a sublattice \( L_1 \) and a semi-sublattice \( L_2 \). Then given a poset \( \Pi \) of \( L \) and a membership oracle, which determines if a matching of \( L \) is in \( L_1 \) or not, the algorithm returns a bouquet defining \( L_1 \).

By Theorem 9, the set of tails \( T_E \) forms a chain \( C \) in \( \Pi \). The idea of our algorithm, given in Figure 4, is to find the flowers according to their order in \( C \). Specifically, a splitting set \( S \) is maintained such that at any point, all flowers outside of \( S \) are found. At the beginning, \( S \) is set to \( \Pi \) and becomes smaller as the algorithm proceeds. Step 2 checks if \( M_z \) is a matching in \( L_1 \) or not. If \( M_z \notin L_1 \), the closed subset \( \Pi \setminus \{t\} \) separates an edge in \( E \) according to Lemma 11. Hence, the first tail on \( C \) must be \( t \). Otherwise, the algorithm jumps to Step 3 to find the first tail. Each time a tail \( r \) is found, Step 5 immediately finds the flower \( L_r \) corresponding to \( r \). The splitting set \( S \) is then updated so that \( S \) no longer contains \( L_r \) but still contains the flowers that have not been found yet. Next, our algorithm continues to look for the next tail inside the updated \( S \). If no tail is found, it terminates.

First we prove a simple observation.
FINDNEXTTAIL($\Pi, S$):

**Input:** A poset $\Pi$, a splitting set $S$.

**Output:** The maximal tail vertex in $S$, or NULL if there is no tail vertex in $S$.

1. Compute the set $V$ of rotations $v$ in $S$ such that:
   - $\Pi \setminus I'_v$ generates a matching in $L_1$.
   - $\Pi \setminus J'_v$ generates a matching in $L_2$.
2. If $V \neq \emptyset$ and there is a unique maximal element $v$ in $V$: Return $v$.
   Else: Return NULL.

Figure 5: Subroutine for finding the next tail.

**Lemma 20.** Let $v$ be a rotation in $\Pi$. Let $S \subseteq \Pi$ such that both $S$ and $S \cup \{v\}$ are closed subsets. If $S$ generates a matching in $L_1$ and $S \cup \{u\}$ generates a matching in $L_2$, $v$ is the head of an edge in $E$. If $S$ generates a matching in $L_2$ and $S \cup \{u\}$ generates a matching in $L_1$, $v$ is the tail of an edge in $E$.

**Proof.** Suppose that $S$ generates a matching in $L_1$ and $S \cup \{u\}$ generates a matching in $L_2$. By Lemma 11, $S$ does not separate any edge in $E$, and $S \cup \{u\}$ separates an edge $e \in E$. This can only happen if $u$ is the head of $e$.

A similar argument can be given for the second case. \qed

**Lemma 21.** Given a splitting set $S$, FINDNEXTTAIL($\Pi, S$) (Figure 5) returns the maximal tail vertex in $S$, or NULL if there is no tail vertex in $S$.

**Proof.** Let $r$ be the maximal tail vertex in $S$.

First we show that $r \in V$. By Theorem 9, the set of tails of edges in $E$ forms a chain in $\Pi$. Therefore $\Pi \setminus I'_r$ contains all tails in $S$. Hence, $\Pi \setminus I'_r$ does not separate any edge whose tails are in $S$. Since $S$ is a splitting set, $\Pi \setminus I'_r$ does not separate any edge whose tails are in $\Pi \setminus S$. Therefore, by Lemma 11, $\Pi \setminus I'_r$ generates a matching in $L_1$. By Lemma 12, $\Pi \setminus J'_r$ must separate an edge in $E$, and hence generates a matching in $L_2$ according to Lemma 11.

By Lemma 20, any rotation in $V$ must be the tail of an edge in $E$. Hence, they are all predecessors of $r$ according to Theorem 9. \qed

**Lemma 22.** Given a tail vertex $r$ and a splitting set $S$ containing $r$, FINDFLOWER($\Pi, S$) (Figure 6) correctly returns $F_r$.

**Proof.** First we give two crucial properties of the set $Y$. By Theorem 9, the set of tails of edges in $E$ forms a chain $C$ in $\Pi$.

**Claim 1.** $Y$ contains all predecessors of $r$ in $C$.

**Proof.** Assume that there is at least one predecessor of $r$ in $C$, and denote by $r'$ the direct predecessor. It suffices to show that $r' \in Y$. By Theorem 9, there exists a splitting set $I$ such that $R_{r'} \subseteq I$
**Find**lower\((\Pi, S, r)\):

**Input:** A poset \(\Pi\), a tail vertex \(r\) and a splitting set \(S\) containing \(r\).

**Output:** The set \(\mathcal{F}_r = \{ v \in \Pi : rv \in E \} \).

1. Compute \(X = \{ v \in I_r : J_v \text{ generates a matching in } \mathcal{L}_1 \} \).
2. Let \(Y = \bigcup_{v \in X} J_v \).
3. If \(Y = \emptyset\) and \(M_0 \in \mathcal{L}_2\): Return \(\{s\} \).
4. Compute the set \(V\) of rotations \(v\) in \(S\) such that:
   - \(Y \cup I_v\) generates a matching in \(\mathcal{L}_1\).
   - \(Y \cup J_v\) generates a matching in \(\mathcal{L}_2\).
5. Return \(V\).

Figure 6: Subroutine for finding a flower.

\(R_r \cap I = \emptyset\). Let \(v\) be the maximal element in \(C \cap I\). Then \(v\) is a successor of all tail vertices in \(I\). It follows that \(I_v\) does not separate any edges in \(E\) inside \(I\). Therefore, \(v \in X\). Since \(J_v \subseteq Y\), \(Y\) contains all predecessors of \(r\) in \(C\).

**Claim 2.** \(Y\) does not contain any rotation in \(\mathcal{F}_r\).

**Proof.** Since \(Y\) is the union of closed subset generating matching in \(\mathcal{L}_1\), \(Y\) also generates a matching in \(\mathcal{L}_1\). By Lemma 11, \(Y\) does not separate any edge in \(E\). Since \(r \notin Y\), \(Y\) must not contain any rotation in \(\mathcal{F}_r\).

By Claim 1, if \(Y = \emptyset\), \(r\) is the last tail found in \(C\). Hence, if \(M_0 \in \mathcal{L}_2\), \(s\) must be in \(\mathcal{F}_r\). By Theorem 9, the heads in \(\mathcal{F}_r\) are incomparable. Therefore, \(s\) is the only rotation in \(C\). \textsc{FindFlower} correctly returns \(\{s\}\) in Step 3. Suppose such a situation does not happen, we will show that the returned set is \(\mathcal{F}_r\).

**Claim 3.** \(V = \mathcal{F}_r\).

**Proof.** Let \(v\) be a rotation in \(V\). By Lemma 20, \(v\) is a head of some edge \(e\) in \(E\). Since \(Y\) contains all predecessors of \(r\) in \(C\), the tail of \(e\) must be \(r\). Hence, \(v \in \mathcal{F}_r\).

Let \(v\) be a rotation in \(\mathcal{F}_r\). Since \(Y\) contains all predecessors of \(r\) in \(C\), \(Y \cup I_v\) can not separate any edge whose tails are predecessors of \(r\). Moreover, by Theorem 9, the heads in \(\mathcal{F}_r\) are incomparable. Therefore, \(I_v\) does not contain any rotation in \(\mathcal{F}_r\). Since \(Y\) does not contain any rotation in \(\mathcal{F}_r\), by the above claim, \(Y \cup I_v\) does not separate any edge in \(E\). It follows that \(Y \cup J_v\) generates a matching in \(\mathcal{L}_1\). Finally, \(Y \cup J_v\) separates \(rv\) clearly, and hence generates a matching in \(\mathcal{L}_2\). Therefore, \(v \in V\) as desired.

\(\square\)

**Theorem 11.** \textsc{FindBouquet}(\(\Pi\)), given in Figure 4, returns a set of edges defining \(\mathcal{L}_1\).
Proof. From Lemmas 21 and 22, it suffices to show that \( S \) is updated correctly in Step 6(b). To be precised, we need that

\[
S \setminus \bigcup_{u \in F \cup \{r\}} J_u'
\]

must still be a splitting set, and contains all flowers that have not been found. This follows from Lemma 18 by noticing that

\[
\bigcup_{u \in F \cup \{r\}} J_u' = \{ v \in \Pi : v \succeq u \text{ for some } u \in R_r \}.
\]

Clearly, a sublattice of \( L \) must also be a semi-sublattice. Therefore, \textsc{FindBouquet} can be used to find a canonical path described in Section 5.

### 8 Finding an Optimal Fully Robust Stable Matching

Consider the setting given in the Introduction, with \( D \) being the domain of all erroneous instances \( B \) under consideration. We show how to use the algorithm in Section 7 to find the poset generating all fully robust matchings w.r.t. \( D \), and then use this poset to obtain a fully robust matching maximizing (or minimizing) any given weight function.

#### 8.1 Studying semi-sublattices is necessary and sufficient

Let \( A \) be a stable matching instance, and \( B \) be an instance obtained by permuting the preference list of one boy or one girl. Lemma 23 gives an example of a permutation so that \( M_{AB} \) is not a sublattice of \( L_A \), hence showing that the case studied in Section 5 does not suffice to solve the problem at hand. On the other hand, for all such instances \( B \), Lemma 24 shows that \( M_{AB} \) forms a semi-sublattice of \( L_A \) and hence the case studied in Section 6 does suffice.

The next lemma pertains to the example given in Figure 7, in which the set of boys is \( B = \{a, b, c, d\} \) and the set of girls is \( G = \{1, 2, 3, 4\} \). Instance \( B \) is obtained from instance \( A \) by permuting girl 1’s list.

**Lemma 23.** \( M_{AB} \) is not a sublattice of \( L_A \).

**Proof.** \( M_1 = \{1a, 2b, 3d, 4c\} \) and \( M_2 = \{1b, 2a, 3c, 4d\} \) are stable matching with respect to instance \( A \). Clearly, \( M_1 \wedge_A M_2 = \{1a, 2b, 3c, 4d\} \) is also a stable matching under \( A \).

In going from \( A \) to \( B \), the positions of boys \( b \) and \( c \) are swapped in girl 1’s list. Under \( B \), \( 1c \) is a blocking pair for \( M_1 \) and \( 1a \) is a blocking pair for \( M_2 \). Hence, \( M_1 \) and \( M_2 \) are both in \( M_{AB} \). However, \( M_1 \wedge_A M_2 \) is a stable matching under \( B \), and therefore is it not in \( M_{AB} \). Hence, \( M_{AB} \) is not closed under the \( \wedge_A \) operation.
### Table

|     | b   | a   | c   | d   |
|-----|-----|-----|-----|-----|
| 1   |     | a   | b   | c   |
| 2   | a   | b   | c   | d   |
| 3   | d   | c   | a   | b   |
| 4   | c   | d   | a   | b   |

Girls’ preferences in A

|     |     | a   | b   | c   |
|-----|-----|-----|-----|-----|
| 1   |     | c   | a   | b   |
| 2   | c   | a   | b   |     |
| 3   |     | d   | c   | a   |
| 4   |     |     | d   | c   |

Girls’ preferences in B

|     |     | a   | b   | c   |
|-----|-----|-----|-----|-----|
| 1   |     |     |     |     |
| 2   | b   | 2   | 1   | 3   |
| 3   | d   | c   | a   | b   |
| 4   | c   | d   | a   | b   |

Boys’ preferences in both instances

Figure 7: An example in which $\mathcal{M}_{AB}$ is not a sublattice of $\mathcal{L}_A$.

**Lemma 24.** For any instance $B$ obtained by permuting the preference list of one boy or one girl, $\mathcal{M}_{AB}$ forms a semi-sublattice of $\mathcal{L}_A$.

**Proof.** Assume that the preference list of a girl $g$ is permuted. We will show that $\mathcal{M}_{AB}$ is a join semi-sublattice of $\mathcal{L}_A$. By switching the role of boys and girls, permuting the list of a boy will result in $\mathcal{M}_{AB}$ being a meet semi-sublattice of $\mathcal{L}_A$.

Let $M_1$ and $M_2$ be two matchings in $\mathcal{M}_{AB}$. Hence, neither of them are in $\mathcal{M}_B$. In other words, each has a blocking pair under instance $B$.

Let $b$ be the partner of $g$ in $M_1 \lor_A M_2$. Then $b$ must also be matched to $g$ in either $M_1$ or $M_2$ (or both). We may assume that $b$ is matched to $g$ in $M_1$.

Let $xy$ be a blocking pair of $M_1$ under $B$. We will show that $xy$ must also be a blocking pair of $M_1 \lor_A M_2$ under $B$. To begin, the girl $y$ must be $g$ since other preference lists remain unchanged. Since $xg$ is a blocking pair of $M_1$ under $B$, $x >_B g$. Similarly, $g >_x g'$ where $g'$ is the $M_1$-partner of $x$. Let $g''$ be the partner of $x$ in $M_1 \lor_A M_2$. Then $g' \geq_x g''$. It follows that $g >_x g''$. Since $x >_B g$ and $g >_x g''$, $xg$ must be a blocking pair of $M_1 \lor_A M_2$ under $B$.

**Proposition 12.** A set of edges defining the sublattice $\mathcal{L}'$, consisting of matchings in $\mathcal{M}_A \cap \mathcal{M}_B$, can be computed efficiently.

**Proof.** We have that $\mathcal{L}'$ and $\mathcal{M}_{AB}$ partition $\mathcal{L}_A$, with $\mathcal{M}_{AB}$ being a semi-sublattice of $\mathcal{L}_A$, by Lemma 24. Therefore, FindBouquet($\Pi$) finds a set of edges defining $\mathcal{L}'$ by Theorem 11.

By Lemma 3, the input $\Pi$ to FindBouquet can be computed in polynomial time. Clearly, a membership oracle checking if a matching is in $\mathcal{L}'$ or not can also be implemented efficiently.

Since $\Pi$ has $O(n^2)$ vertices (Lemma 3), any step of FindBouquet takes polynomial time.

**8.2 Optimizing fully robust stable matchings**

Finally, we will prove Theorem 1. Let $B_1, \ldots, B_k$ be polynomially many instances in the domain $D \subset T$, as defined in the Introduction. Let $E_i$ be the set of edges defining $\mathcal{M}_A \cap \mathcal{M}_{B_i}$ for all $1 \leq i \leq k$. By Corollary 1, $\mathcal{L}' = \mathcal{M}_A \cap \mathcal{M}_{B_1} \cap \ldots \cap \mathcal{M}_{B_k}$ is a sublattice of $\mathcal{L}_A$.

**Lemma 25.** $E = \bigcup_i E_i$ defines $\mathcal{L}'$. 

25
Proof. By Lemma 11, it suffices to show that for any closed subset $I$, $I$ does not separate an edge in $E$ iff $I$ generates a matching in $L'$.

$I$ does not separate an edge in $E$ iff $I$ does not separate any edge in $E_i$ for all $1 \leq i \leq k$ iff the matching generated by $I$ is in $M_A \cap M_{B_i}$ for all $1 \leq i \leq k$ by Lemma 11.

By Lemma 25, a compression $\Pi'$ generating $L'$ can be constructed from $E$ as described in Section 4. By Proposition 12, we can compute each $E_i$, and hence, $\Pi'$ efficiently. Clearly, $\Pi'$ can be used to check if a fully robust stable matching exists. To be precise, a fully robust stable matching exists iff there exists a proper closed subset of $\Pi'$. This happens iff $s$ and $t$ belong to different meta-rotations in $\Pi'$, an easy to check condition. Hence, we have Theorem 1.

We can use $\Pi'$ to obtain a fully robust stable matching $M$ maximizing $\sum_{b \in M} w_{bg}$ by applying the algorithm of [MV18b]. Specifically, let $H(\Pi')$ be the Hasse diagram of $\Pi'$. Then each pair $bg$ for $b \in B$ and $g \in G$ can be associated with two vertices $u_{bg}$ and $v_{bg}$ in $H(\Pi')$ as follows:

- If there is a rotation $r$ moving $b$ to $g$, $u_{bg}$ is the meta-rotation containing $r$. Otherwise, $u_{bg}$ is the meta-rotation containing $s$.

- If there is a rotation $r$ moving $b$ from $g$, $v_{bg}$ is the meta-rotation containing $r$. Otherwise, $v_{bg}$ is the meta-rotation containing $t$.

By Lemma 2 and the definition of compression, $u_{bg} \prec v_{bg}$. Hence, there is a path from $u_{bg}$ to $v_{bg}$ in $H(\Pi')$. We can then add weights to edges in $H(\Pi')$, as stated in [MV18b]. Specifically, we start with weight 0 on all edges and increase weights of edges in a path from $u_{bg}$ to $v_{bg}$ by $w_{bg}$ for all pairs $bg$. A fully robust stable matching maximizing $\sum_{bg \in M} w_{bg}$ can be obtained by finding a maximum weight ideal cut in the constructed graph. An efficient algorithm for the latter problem is given in [MV18b].

9 Discussion

The structural and algorithmic results introduced in this paper naturally lead to a number of new questions, such as finding robust (with respect to a probability distribution on the domain of errors) rather than fully robust stable matchings, extending to more than one error, improving the running time of our algorithm, extending to the stable roommate problem, incomplete preference lists, etc.

Considering the deep and pristine structure of stable matching, it will not be surprising if many of these questions do get settled satisfactorily in due course of time. As stated previously, our proof of the generalization of Birkhoff’s Theorem, and the new notions it yields, are of independent interest. Finally, considering the number of new and interesting matching-based markets being defined on the Internet, e.g., see [Rot16], it will not be surprising if new, deeper structural facts about stable matching lattices find suitable applications.
References

[Bir37] Garrett Birkhoff. Rings of sets. *Duke Mathematical Journal*, 3(3):443–454, 1937.

[Bla84] Charles Blair. Every finite distributive lattice is a set of stable matchings. *Journal of Combinatorial Theory, Series A*, 37(3):353–356, 1984.

[BTEGN09] A. Ben-Tal, L. El Ghaoui, and A.S. Nemirovski. *Robust Optimization*. Princeton Series in Applied Mathematics. Princeton University Press, October 2009.

[CE06] G. C. Calafiore and L. El Ghaoui. On distributionally robust chance-constrained linear programs. *Jour. of Optimization Theory and Applications*, 130(1), December 2006.

[GI89] Dan Gusfield and Robert W Irving. *The stable marriage problem: structure and algorithms*. MIT press, 1989.

[GS62] David Gale and Lloyd S Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.

[GSOS17] Begum Genc, Mohamed Siala, Barry O'Sullivan, and Gilles Simonin. Finding robust solutions to stable marriage. *arXiv preprint arXiv:1705.09218*, 2017.

[GSSO17] Begum Genc, Mohamed Siala, Gilles Simonin, and Barry O'Sullivan. On the complexity of robust stable marriage. In *International Conference on Combinatorial Optimization and Applications*, pages 441–448. Springer, 2017.

[IL86] Robert W Irving and Paul Leather. The complexity of counting stable marriages. *SIAM Journal on Computing*, 15(3):655–667, 1986.

[Irv85] Robert W Irving. An efficient algorithm for the “stable roommates” problem. *Journal of Algorithms*, 6(4):577–595, 1985.

[Knu97] Donald Ervin Knuth. *Stable marriage and its relation to other combinatorial problems: An introduction to the mathematical analysis of algorithms*. American Mathematical Soc., 1997.

[Man13] David Manlove. *Algorithmics of Matching Under Preferences*. World Scientific, 2013.

[MV18a] Tung Mai and Vijay V. Vazirani. Finding stable matchings that are robust to errors in the input. In arXiv, 2018.

[MV18b] Tung Mai and Vijay V. Vazirani. A natural generalization of stable matching solved via new insights into ideal cuts. In arXiv, 2018.

[Rot16] Alvin E. Roth. Al Roth’s game theory, experimental economics, and market design page, 2016. http://stanford.edu/~alroth/alroth.html#MarketDesign.

[RS12] Alvin E. Roth and Lloyd S. Shapley. The 2012 Nobel Prize in Economics, 2012. https://www.nobelprize.org/nobel_prizes/economic-sciences/laureates/2012/.

[Sta96] Richard Stanley. Enumerative combinatorics, vol. 1, wadsworth and brooks/cole, pacific grove, ca, 1986; second printing, 1996.
[Wika] Wikipedia. Birkhoff’s Representation Theorem. https://en.wikipedia.org/wiki/Birkhoff%27s_representation_theorem#Functoriality.

[Wikb] Wikipedia. Fault Tolerance. https://en.wikipedia.org/wiki/Fault_tolerance.

[Wiki] Wikipedia. Robustness (Computer Science). https://en.wikipedia.org/wiki/Robustness_(computer_science).