Two-Sided Matching Markets: Models, Structures, and Algorithms

Xuan Zhang

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy
under the Executive Committee
of the Graduate School of Arts and Sciences

COLUMBIA UNIVERSITY

2022

Abstract

Two-Sided Matching Markets: Models, Structures, and Algorithms

Xuan Zhang

Two-sided matching markets are a cornerstone of modern economics. They model a wide range of applications such as ride-sharing (Wang, Agatz, and Erera, 2018), online dating (Hitsch, Hortaçsu, and Ariely, 2010), job positioning (Kelso Jr and Crawford, 1982), school admissions (Abdulkadiroğlu and Sönmez, 2003), and many more. In many of those markets, monetary exchange does not play a role. For instance, the New York City public high school system is free of charge. Thus, the decision on how eighth-graders are assigned to public high schools must be made using concepts of fairness rather than price. There has been therefore a huge amount of literature, mostly in the economics community, defining various concepts of fairness in different settings and showing the existence of matchings that satisfy these fairness conditions. Those concepts have enjoyed wide-spread success, inside and outside academia (see, e.g., Manlove, 2013). However, finding such matchings is as important as showing their existence. Moreover, it is crucial to have fast (i.e., polynomial-time) algorithms as the size of the markets grows. In many cases, modern algorithmic tools must be employed to tackle the intractability issues arising from the big data era.

The aim of my research is to provide mathematically rigorous and provably fast algorithms to find solutions that extend and improve over a well-studied concept of fairness in two-sided markets known as *stability*. This concept was initially employed by the *National Resident Matching Program* in assigning medical doctors to hospitals, and is now widely used, for instance, by cities in the US for assigning students to public high schools (Abdulkadiroğlu and Sönmez, 2003) and

by certain refugee agencies to relocate asylum seekers (Nguyen, Nguyen, and Teytelboym, 2019). In the classical model, a stable matching can be found efficiently using the renowned *deferred acceptance* algorithm by Gale and Shapley (1962). However, stability by itself does not take care of important concerns that arose recently, some of which were featured in national newspapers (Harris and Fessenden, 2017; Shapiro, 2019a; Shapiro, 2021). Some examples are: how can we make sure students get admitted to the best school they deserve, and how can we enforce diversity in a cohort of students?

By building on known and new tools from Mathematical Programming, Combinatorial Optimization, and Order Theory, my goal is to provide fast algorithms to answer questions like those above, and test them on real-world data.

In Chapter 1, I introduce the stable matching problem and related concepts, as well as its applications in different markets.

In Chapter 2, we investigate two extensions introduced in the framework of school choice that aim at finding an assignment that is more favorable to students – *legal assignments* (Morrill, 2016) and the *Efficiency Adjusted Deferred Acceptance Mechanism* (EADAM) (Kesten, 2010) – through the lens of classical theory of stable matchings. We prove that the set of legal assignments is exactly the set of stable assignments in *another* instance. Our result implies that essentially all optimization problems over the set of legal assignments can be solved within the same time bound needed for solving it over the set of stable assignments. We also give an algorithm that obtains the assignment output of EADAM. Our algorithm has the same running time as that of the deferred acceptance algorithm, hence largely improving in both theory and practice over known algorithms.

In Chapter 3, we introduce a property of distributive lattices, which we term as *affine representability*, and show its role in efficiently solving linear optimization problems over the elements of a distributive lattice, as well as describing the convex hull of the characteristic vectors of the lattice elements. We apply this concept to the stable matching model with path-independent quota-filling choice functions, thus giving efficient algorithms and a compact polyhedral description for this model. Such choice functions can be used to model many complex real-world decision rules that are not captured by the classical model, such as those with diversity concerns. To the best of our knowledge, this model generalizes all those for which similar results were known, and our paper is the first that proposes efficient algorithms for stable matchings with choice functions, beyond classical extensions of the Deferred Acceptance algorithm.

In Chapter 4, we study the *discovery program* (DISC), which is an affirmative action policy used by the New York City Department of Education (NYC DOE) for specialized high schools; and explore two other affirmative action policies that can be used to minimally modify and improve the discovery program: the minority reserve (MR) and the joint-seat allocation (JSA) mechanism. Although the discovery program is beneficial in increasing the number of admissions for disadvantaged students, our empirical analysis of the student-school matches from the 12 recent academic years (2005-06 to 2016-17) shows that about 950 in-group blocking pairs were created each year amongst disadvantaged group of students, impacting about 650 disadvantaged students every year. Moreover, we find that this program usually benefits *lower-performing* disadvantaged students more than *top-performing* disadvantaged students (in terms of the ranking of their assigned schools), thus unintentionally creating an incentive to under-perform. On the contrary, we show, theoretically by employing choice functions, that (i) both MR and JSA result in no in-group blocking pairs, and (ii) JSA is weakly group strategy-proof, ensures that at least one disadvantaged is not worse off, and when reservation quotas are carefully chosen then no disadvantaged student is worse-off. We show that each of these properties is not satisfied by DISC. In the general setting, we show that there is no clear winner in terms of the matchings provided by DISC, JSA and MR, from the perspective of disadvantaged students. We however characterize a condition for markets, that we term *high competitiveness*, where JSA dominates MR for disadvantaged students. This condition is verified, in particular, in certain markets when there is a higher demand for seats than supply, and the performances of disadvantaged students are significantly lower than that of advantaged students. Data from NYC DOE satisfy the high competitiveness condition, and for this dataset our empirical results corroborate our theoretical predictions, showing the superiority of JSA. We believe that the discovery program, and more generally affirmative action mechanisms, can be changed for the better by implementing the `JSA` mechanism, leading to incentives for the top-performing disadvantaged students while providing many benefits of the affirmative action program.

Table of Contents

List of Figures

List of Tables

Acknowledgements

Looking back at the five years of my PhD life, I feel very content. The acknowledgement section ended up being the hardest part of the thesis. The thought of putting an official period to these amazing five years brings tears to my eyes.

First and foremost, I would like to thank my advisor, Yuri Faenza, for his patience and support in guiding me throughout the PhD journey. I feel very lucky to have Yuri as my advisor. I learned a lot from Yuri, not only how to do research, but also how to be a good mentor and collaborator. I am particularly influenced by his positive attitudes towards and his unconditional trust in research. And as always, I really enjoyed all the random conversations we had during meetings.

I am very grateful to my dissertation committee members, Jay Sethuraman, Yash Kanoria, Christian Kroer, and Amin Saberi, for their valuable feedbacks which greatly improve this dissertation. I am also very thankful for all my other collaborators, Professor Daniel Bienstock and Professor Swati Gupta, as well as my fellow PhD students Shatian Wang and Judy Gan. I have learned so much from working with them. I also want to thank the Cheung Kong Graduate School of Business (CKGSB) for their fellowship support, and New York City Department of Education (NYC DOE) for compiling and providing me with the real-world data.

I would like to express my gratitude for all the staffs at the department for making our stay smooth and for organizing so many memorable events. I truly missed the Karaoke, the escape rooms, and many other fun gatherings we had prior to the pandemics.

Huge thanks to all the friends inside as well as outside the department: Yunhao Tang, Shatian Wang, Ruizhe Jia, Steven Yin, Lingyi Zhang, Judy Gan, Shuoguang Yang, Raghav Singal, Kumar

Goutam, Apurv Shukla, Rachitesh Kumar, Harsh Sheth, Sudeep Raja, Jacob Bergquist, Noemie Perivier, Weichi Yao, Wenying Deng, Yirou (Chris) Li, Shuting Shen, Jenny Wang, and many others. I feel particularly appreciative for those who were there for me during the harsh time of 2020. I am grateful for Tom Chen, my ex-husband, who supported me a lot during the first three (or maybe three and a half) years of my PhD. Special thank to Yunhao Tang for being my buddy for everything and always hearing my complaints. And a very special mention to Harsh Sheth for such a wonderful last year.

Finally, I would like to thank my family for always supporting and believing in me. I would not be where I am if it were not for them.

To my parents

Introduction

Since the seminal publication by Gale and Shapley (1962), the concept of stability in matching markets has been widely studied by the optimization community. With minor modifications, the one-to-many version of Gale and Shapley's original stable assignment model, together with their renowned *deferred acceptance* algorithm, is currently employed by the National Resident Matching Program (Roth, 1984b) for assigning medical residents to hospitals in the US, and by many school districts in the US for assigning eighth-graders to public high schools (Abdulkadiroğlu and Sönmez, 2003).

However, the traditional model does not capture features that have become of crucial importance both inside and outside academia. For instance, in the school choice problem, public schools are often perceived as commodities and only students' welfare matters. Hence, enforcing stability implies a loss of *efficiency* for the students. Abdulkadiroğlu, Pathak, and Roth (2009) demonstrated the magnitude of such efficiency loss with empirical data from the New York City school system, showing that over 4,000 eighth-graders could have improved their assignments if stability constraints were relaxed. Striving to regain this loss in welfare for the students, many models and mechanisms have been proposed in the literature (Abdulkadiroğlu, Che, and Yasuda, 2015; Erdil and Ergin, 2008; Kesten, 2010; Troyan, Delacrétaz, and Kloosterman, 2020; Morrill, 2016). However, results are oftentimes not algorithmic, which significantly hinders their applicability.

Moreover, there is a growing attention to models that can increase diversity in school cohorts (Nguyen and Vohra, 2019; Tomoeda, 2018). Such constraints cannot be represented in the original model, since admission decisions with diversity concerns cannot be captured by a strict preference list. To model these selection rules, instead of ranking individual potential partners, each agent is endowed with a *choice function* that selects a cohort they prefer the best from a given set of potential candidates. Besides school choice problem with diversity concerns, the stable matching model with choice functions has been widely used in many other markets, such as the *staffing* problem and the *course allocation* problem (see, e.g., Echenique and Yenmez, 2015; Aygün and Turhan, 2016; Kamada and Kojima, 2015 for more applications).

One particular class of application of the stable matching model with choice functions are mechanisms with affirmative actions in the context of school choice. Due to disparity in opportunities, some groups are underrepresented in education at different levels, and these include ethnic minorities who suffer discrimination historically, immigrants who are English learners, and students from households with lower socioeconomic status (see, e.g., Ashkenas, Park, and Pearce, 2017; Boschma and Brownstein, 2016; Quinn Capers et al., 2017). Affirmative action policies help remedy the situation by favoring these underrepresented disadvantaged groups in some way, often by giving them priority for or exclusive access to certain seats. However, it is well-known in the literature, through both theoretical and empirical results, that some affirmative action mechanisms do not automatically guarantee to improve the welfare of disadvantaged students, the purported beneficiary (Kojima, 2012; Hafalir, Yenmez, and Yildirim, 2013). Thus, it is important to have a deep understanding of the many affirmative action mechanisms implemented in practice, both theoretically and empirically.

The goal of my thesis is to have a better structural, algorithmic, and empirical understanding of these aforementioned extensions of the stable assignment problem. My research approaches the problem from three broadly categorized directions. The first direction focuses on deriving structural properties of the models, and the second one focuses on developing efficient algorithms for optimizing over the set of stable matchings using the structural results. While these two directions concern the set of stable matchings, the last direction studies the theoretical and empirical properties of one particular stable matching (the student-optimal one) under different mechanisms. In the remainder of the introduction, let me informally summarize my results.

Chapter 2: Legal assignments and fast EADAM with consent (Faenza and Zhang, 2022). Two concepts proposed in the literature which aim to regain the welfare loss for the students due to enforcing the stability constraints are: *legal matchings* (Ehlers and Morrill, 2020) and *Efficiency Adjusted Deferred Acceptance Mechanism* (EADAM) *with consent* (Kesten, 2010). Both define fairness via relaxation of the stability requirement while protecting school districts from potential lawsuits.

Structurally, we show that for any instance, there is *another* instance whose set of stable matchings is the same as the set of legal matchings in the original instance (see Theorem 2.3). Moreover, we provide an efficient algorithm that returns this (legalized) instance, using the *distributive lattice* structure of legal matchings and the concept of *rotations* (i.e., special alternating cycles). Thus, to solve an optimization problem over legal matchings, one can resort to the broad literature on algorithms developed for the same problem on the set of stable matchings.

A key component of our procedure to build the legalized instance is an algorithm that finds the student-optimal legal matching (see Algorithm 2.1). By using the structural results of legal matchings, our algorithm is faster than the previously known algorithms (Kesten, 2010; Tang and Yu, 2014) both theoretically and in practice (see Figure 2.1).

A modification of our algorithm can be used to find the outcome of EADAM with consent (see Algorithm 2.3). Computational experiments show that our algorithm performs roughly 80 times faster than current versions of EADAM (Tang and Yu, 2014) on instances whose size is similar to that of the NYC school system, hence making these algorithms for the first time of practical relevance (see Figure 2.2).

Results from this chapter are contained in Faenza and Zhang (2022), which is published in *Operations Research*.

Chapter 3: Stable matchings and choice functions (Faenza and Zhang, 2021). Stable assignment models with choice functions were first studied in Roth (1984a) and Kelso Jr and Crawford (1982). Roth (1984a) generalized the deferred acceptance algorithm by Gale and Shapley (1962) to find the student- and school-optimal stable matchings when choice functions are assumed to be *substitutable* and *consistent* (or equivalently, *path-independent*). When choice functions are further assumed to be *quota-filling*, stable matchings are known to form a distributive lattice (Alkan, 2001).

Using concepts from abstract algebra, *rings of sets* in particular, we present an algorithm (see Algorithm 3.7) that obtains a poset of *rotations* (i.e., special trading cycles) whose set of *upper sets* are in bijection with stable matchings (the existence of such poset is due to *Birkhoff's representation theorem* Birkhoff (1937)) for the choice function model. In particular, although the size of the set of stable matchings can be exponential in the number of agents in the market, the size of the rotation poset is always polynomial (see Theorem 3.4).

The structure results then provide us the tools to efficiently solve optimization problems over the set of stable matchings under any linear objectives (see Corollary 3.5). Linear objectives can be used to model a matching that is, e.g., aforementioned firm- or worker-optimal, egalitarian, or profit-maximal. More specifically, we show that the problem of optimizing a linear function over stable matchings under the choice function model can be solved by optimizing a linear function over upper sets of the rotation poset. It is known that the latter problem can be solved as a minimum cut problem (Picard, 1976).

Moreover, the structure results allow us to draw connection between the stable matching polytope (i.e., convex hull of stable matchings) and the *order polytope* (Stanley, 1986) (i.e., the convex hull of upper sets of a poset). We show that a compact polyhedral description of the stable matching polytope under the choice function model exists (see Theorem 3.6). This enables the possibility of solving optimization problems over the set of stable matching via linear programming.

To the best of our knowledge, the model we studied here generalizes all models from the literature for which similar results were known, and our work is the first that proposes efficient algorithms for stable matchings with choice functions, beyond the deferred acceptance algorithm.

More generally, we extend our results to a broad class of problems in combinatorial optimization whose feasible regions can be seen as distributive lattices (see, e.g., Garg, 2020) that satisfy a property which we term as *affine representability*.

An extended abstract of the work (Faenza and Zhang, 2021) was published in the *proceedings of Integer Programming and Combinatorial Optimization (IPCO)* 2021 and the journal version of the manuscript is currently under minor revision at *Mathematical Programming*.

Chapter 4: Discovering Opportunities in New York City's Discovery Program: an Analysis of Affirmative Action Mechanisms (Faenza, Gupta, and Zhang, 2022a). Discovery program (DISC) is an affirmative action policy used by the New York City Department of Education (NYC DOE) for specialized high schools. It has been instrumental in increasing the number of admissions for disadvantaged students. However, rigorous mathematical analyses of the program are lacking, even though they are critical in ensuring the welfare of disadvantaged students, given that certain forms of affirmative action may hurt disadvantaged students, the purported beneficiary, as shown by Kojima (2012) and Hafalir, Yenmez, and Yildirim (2013).

Our empirical analysis shows that the discovery program creates a large number of in-group blocking pairs (see Figure 4.1). In addition, we find that this program unintentionally creates an incentive for disadvantaged students to under-perform. These drawbacks of the discovery program are also verified theoretically. Hence, we explore other affirmative action mechanisms, with the goal of proposing practical modifications to how the discovery program is implemented, while alleviating the above-mentioned drawbacks.

The two alternative quota-based mechanisms we consider are minority reserve (MR), which was first studied by Hatfield and Milgrom (2005) and joint seat allocation (JSA), which is an abstract version of the mechanism used by the Joint Seat Allocation Authority for administering admissions to Indian Institutes of Technology (JoSAA, 2020). We show that, property-wise, JSA is similar to MR, given that they both (1) are strategy-proof, (2) are fair to disadvantaged students (i.e., no in-group blocking pairs), (3) guarantee that not all disadvantaged student is worse off, and (4) ensure that no disadvantaged students is worse off when the reservation quotas are "carefully" chosen (i.e., a *smart reserve*). See Table 4.1 for a summary. These results suggest that the discovery program could benefit by replacing the current implementation with either minority reserve or joint seat allocation, but at the same time call for a direct comparison of those mechanisms.

We show that all three mechanisms are incomparable, even under some pretty restrictive yet common hypothesis: (1) schools rank students in the same order; and/or (2) reservation quotas being a smart reserve. However, we also provide a novel ex-post condition which guarantees that `JSA` weakly dominates `MR` for disadvantaged students and show that this condition is verified by our data from NYC DOE. Roughly speaking, the high competitiveness condition is satisfied when the demand for seats (i.e., number of students) is much larger than the supply, and when disadvantaged students are performing systematically worse than advantaged students. See Theorem 4.20 for the formal statement.

Our results suggest that the many drawbacks of the current implementation of the discovery program can be corrected by following the *joint seat allocation* mechanism. This modification is powerful, yet it requires *minimal* modification: there is essentially no change in terms of what students and schools should report to the DOE, and there is no change in terms of the algorithm. In fact, to implement the `JSA` mechanism, one only needs to compute an equivalent instance where students' preference lists are expanded to be over reserved and general seats at schools, so that the matching we desire to obtain can be easily recovered from the matching obtained under the classical stable matching model on this equivalent instance. See Section 4.4.3 for details.

On the technical aspect, to compare these three mechanisms, we devise alternative formulations so that their assignment outputs can be obtained from the *same* algorithm applied to *different* input instances. There are two approaches by which we can obtain such a reformulation. This first approach is to employ *choice functions*, which allows us to directly use properties of choice functions and the generalized deferred acceptance algorithm (Roth, 1984a). The second approach is to expand students' original preferences over schools to preferences over reserved and general seats at schools. This allows us to deduce interesting properties of the mechanisms (e.g., strategy-proofness), by leveraging on classical results on stable matchings.

Chapter 1: The Stable Assignment Problem

In this chapter, we introduce the stable assignment problem and its related concepts. We defer the specific mathematical notations to individual chapters as they are slightly different and are adapted for each chapter.

An instance of the *stable assignment* (or *stable matching*) problem has the following components. The first is a bipartite graph, where the two sides of the vertices represent two sides of the markets, and the edges represent the set of *acceptable* partners. Some examples for the two sides of the market are men and women, students and schools, and firms and workers. Secondly, every agent in the market has a strict ordering of his or her acceptable partners (i.e., its neighbors in the graph), and this strict ordering is usually referred to as *preference list*. However, sometimes, when the agent is a school, preference list is also called *priority order*, since it indicates students' priorities at the school. Lastly, every agent has a *quota*, which signifies the maximum number of partners they can have.

Depending on the market, the quotas of the agents can be either singular or plural. For instance, in the case of the *marriage* market where two sides of the market are men and women, every agent has a quota of one; in the case of *school choice*, students have a quota of one, and schools have quotas that reflect their admission capacity; in the case of *labor market*, firms and workers can both have quotas that are more than one. These illustrate three types of market: *one-to-one*, *one-to-many*, and *many-to-many*. For the following, we introduce the concepts for the most general many-to-many market, and we refer to the two sides of the markets as firms and workers.

Stability. An *assignment* (or a *matching*) of an instance is a collection of acceptable firm-worker pairs (or edges) so that the number of pairs every agent is incident to is at most his or her quota. An assignment is *stable* if there is no unmatched acceptable firm and worker pair where both *prefer* each other to their assignment. Here, a firm f is said to prefer a worker w to its assignment if one of the following two cases is true: (1) the firm still has empty positions (i.e., its number of assigned workers is less than its quota); or (2) there exists a worker that is assigned to the firm, but the firm prefers w to this worker. The concept where a worker is said to prefer a firm can be similarly defined. The goal of the stable assignment problem is to find an assignment that is stable.

Every instance has at least one stable assignment, and the proof is based on an algorithm, called the *deferred acceptance algorithm*, which is guaranteed to terminate and output a stable assignment (Gale and Shapley, 1962; Roth, 1984a). The algorithm has one side of the market proposing to the other side. In the following, we describe the *worker-proposing* version. Due to symmetry, the *firm-proposing* version can be easily deduced.

Deferred acceptance algorithm. The worker-proposing algorithm starts with every worker unmatched and then runs in *rounds*. At every round, every worker proposes to his or her most preferred firms that have not rejected him or her, up to his or her quota. Then every firm *temporarily* accepts, among all the workers that proposed to it in the current round, its most preferred workers, also up to its quota, and rejects the rest. The algorithm terminates when there is no rejection, and all temporary acceptances becomes *permanent*. The assignment output consists of all the permanent acceptances.

In the stable assignment output by the worker-proposing algorithm, every worker is matched to the *best team* of firms (s)he can have in any stable assignments (Gale and Shapley, 1962; Roth, 1984a). That is, if a worker is asked to choose firms, up to his or her quota, among those assigned to him or her either under the output stable assignment or under any other stable assignment, (s)he will choose exactly the same set of firms as (s)he is assigned to under the output stable assignment. Hence, this stable assignment is called the *worker-optimal* stable assignment. Similarly, the firm-proposing deferred acceptance algorithm outputs a stable assignment that is *firm-optimal*. These two extreme stable assignments might coincide, when the instance has only one stable assignment.

The fact that one stable matching is optimal for one side of the market has crucial implications for real-world applications. For instance, for the National Resident Matching Program where medical interns are matched to hospitals, the hospital-proposing algorithm was replaced by the worker-proposing algorithm in the 1990s to increase the welfare of medical students; in school choice, since schools are usually considered as public commodities, deferred acceptance is always executed with students proposing.

Structure of stable assignments. As our previous discussion implies, in general, an instance may have many other stable assignments other than the firm- and worker-optimal ones. Interestingly, as observed by Knuth (1976) for the one-to-one case and later proved for the many-to-many case by Roth (1984a), the set of stable assignments form a *distributive lattice* under a natural ordering relation. The two extreme stable assignments, worker-optimal and firm-optimal, correspond to the minimum and maximum elements of the lattice.

Birkhoff's representation theorem (Birkhoff, 1937) associates to each distributive lattice an associate poset so that there is a bijection between elements of the distributive lattice and the family of *upper sets* of the poset. In the special case of stable assignments, this associated poset is called the *rotation poset*, where rotations are certain trading cycles (i.e., symmetric difference of certain pairs of stable matchings). This, together with the fact that optimization problem over upper sets of posets can be solved efficiently (Picard, 1976), is the backbone of many fast algorithms for optimizing over the set of stable matchings (see, e.g., Gusfield, 1987; Irving, Leather, and Gusfield, 1987), including our results in Chapter 2 and Chapter 3.

Choice functions. Note that in the description of the worker-proposing deferred acceptance algorithm, the firms to which workers propose to, and the workers who firms temporarily accept, both depend on their preference lists and quotas. Equivalently, one can encode these *decision rules* via *choice functions*, which is a function which picks a team that each agent prefers the best from a given set of potential partners. This leads to a generalization of the stable assignment problem, which is referred to as the stable assignment problem under choice functions, where agents in the market are endowed with choice functions instead of quotas and preference lists. We study such models in Chapter 3 and Chapter 4.

Such a generalization allows the stable assignment model to be used for applications with more complex decision rules, which preference lists and quotas alone do not capture. Many such markets have been studied in the literature (see, e.g., Echenique and Yenmez, 2015; Aygün and Turhan, 2016; Kamada and Kojima, 2015) and here we highlight some examples: in the case of labor market, firms might want to hire a group of employees that is as diverse as possible; in the case of school choice, schools might be required by their districts to admit a cohort of students whose demographic composition resembles that of the district; in the case of *course allocation* where two sides of the market are students and courses, students might desire to select a group of courses that covers as many topics as possible.

In the stable matching literature, choice functions are often assumed to be *substitutable* and *consistent*, as they are necessary and sufficient conditions to guarantee the existence of stable matchings (see, e.g., Aygün and Sönmez, 2013; Hatfield and Milgrom, 2005; Roth, 1984a). Informally speaking, substitutability states that whenever an agent is selected from a pool of candidates, (s)he will also be selected from a subset of the candidates; and consistency is also called "irrelevance of rejected contracts", which means removing rejected candidates from the input will not change the output. The formal definitions can be found in Section 3.2.2. The existence result is similarly proved by an algorithm, which can be viewed as a generalization of the deferred acceptance (Roth, 1984a). Although substitutability and consistency imply existence, when only these two conditions are assumed, the set of stable matchings might not form a distributive lattice (Blair, 1988).

Chapter 2: Legal Assignments and fast EADAM with consent via classical theory of stable matchings

2.1 Introduction

One of the most important applications of matching theory, the school choice problem, considers the assignment of high school students to public schools. After the pioneering work of Abdulkadiroğlu and Sönmez (2003), many school districts, such as New York City and Boston, adopted the student-optimal stable mechanism for its *fairness* (no priority violation or stability) and *strategy-proofness* (for students). The mechanism asks students to report their (strict) preferences of the schools and schools to report their priorities[1] (preferences with ties) over the students. It then randomly breaks ties in the latter to obtain an instance of the stable assignment problem and performs Gale-Shapley's deferred acceptance algorithm to obtain the student-optimal stable assignment. Gale-Shapley's algorithm embodies many desirable qualities an algorithm can have: it is simple, elegant, runs in time linear in the size of the instance, and outputs an assignment that satisfies the aforementioned strong properties. In our simulations, on random instances of the size of the New York City school system, it terminates on average in less than 3 minutes (see Figure 2.2).

In this setting, schools are often perceived as commodities, and only students' welfare matters. Hence, enforcing stability implies a loss of efficiency. Abdulkadiroğlu, Pathak, and Roth (2009) demonstrate the magnitude of such efficiency loss with empirical data from the New York City school system, where over 4,000 eighth graders in their sample could improve their assignments if stability constraints were relaxed. Striving to regain this loss in welfare for the students, many alternative concepts and mechanisms have therefore been introduced and extensively studied

[1] Priorities are preferences with ties, as schools usually rank students based on categorical information such as demographics, test scores, etc.

(see e.g. Abdulkadiroğlu, Che, and Yasuda, 2015; Erdil and Ergin, 2008; Kesten, 2010; Troyan, Delacrétaz, and Kloosterman, 2020; Morrill, 2016).

Those mechanisms lead to solutions outside the well-structured set of stable assignments. As a consequence, ad-hoc structural studies and algorithms must be presented. Unfortunately, properties of the former and performance of the latter rarely match theory of and algorithms for stable assignments (Kesten, 2010; Tang and Yu, 2014). For instance, Kesten's *Efficiency Adjusted Deferred Acceptance Mechanism* (*EADAM*) (Kesten, 2010) (one of the main focuses of this chapter), in our experiments, cannot terminate after 24 hours of computation, on average, on random instances of similar size as the New York City high school system. This algorithmic inefficiency harms the applicability of such mechanisms to real-world instances, especially if policy designers want to run them multiple times either as a subroutine in a more complex mechanism (Ashlagi and Nikzad, 2016; Erdil and Ergin, 2008).

The goal of this chapter is to show how certain concepts, introduced in the literature to regain the loss of welfare caused by stability constraints, can be fully understood through the lens of classical theory of stable assignments. Moreover, we show that this better understanding leads to theoretically and practically faster algorithms, as well as extensions and new connections within this classical theory. We believe that our results can stimulate further applications of those concepts, as well as future theoretical research. The two topics that we study in depth are *legal assignments* (Morrill, 2016) and *EADAM with consent* (Kesten, 2010). Let us therefore introduce them next.

Legal Assignments. Legality gives an alternative interpretation of fairness, in an attempt to eliminate the tension between stability and efficiency. The stability condition prohibits, in the assignment chosen, the existence of a student-school pair that prefer each other to their assigned partners. Such pairs are called *blocking pairs*. Therefore, stability makes sure that no student is harmed, and thus no student has the justification to take legal action against the public school system. However, Morrill observed that legal standing, as interpreted by the United States Supreme Court, is not exactly the same as prohibiting blocking pairs (Morrill, 2016). Specifically, in order for a student to have legal standing, not only must he be harmed (i.e., forming a blocking pair with a school), this harm also must be *redressable*. That is, there must be an assignment that is accepted as feasible under which the student is assigned to the school.

With this interpretation, an institution is safe from legal actions if the set $\mathcal{L}$ of assignments that are considered feasible has the property that if a student-school pair blocks an assignment from $\mathcal{L}$, then this pair is not matched in any assignment from $\mathcal{L}$ (*internal stability*). On the other hand, in order to justify the exclusion of an assignment M from the set $\mathcal{L}$, there must be a pair that blocks M and is matched in some assignment from $\mathcal{L}$ (*external stability*). Following Morrill, we call a set $\mathcal{L}$ with those properties *legal*. Note that every legal set contains the set of stable assignments. We illustrate this concept with an example.

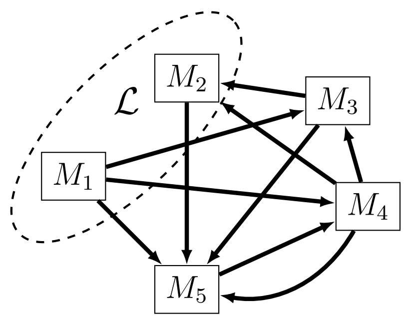
Example 2.1. Here and throughout the chapter, one side of the partition is called *students* and the other is called *schools*. In this example, we also assume that each school can admit at most one student. Consider the instance with preference lists given below.

student 1:	A	B	C		school A:	2	3	1
student 2:	B	A			school B:	1	2	
student 3:	A	C			school C:	3	1	

Below, we list all five maximal matchings. Note that it is sufficient to consider only maximal matchings because if a matching is not maximal, it cannot be in a legal set. We also list the blocking pairs each matching admits. In this instance, M_1 is the only stable matching.

#	maximal matching	blocking pairs
M_1	$1B, 2A, 3C$	$\emptyset$
M_2	$1A, 2B, 3C$	$3A$
M_3	$1B, 3A$	$2A$
M_4	$1C, 2B, 3A$	$1B$
M_5	$1C, 2A$	$1B, 2B, 3C$

We now construct a digraph below with each maximal matching as a vertex. We add an arc (M, M') if and only if matching M *blocks* matching M' where we say a matching M *blocks* M' if M contains an edge that is a blocking pair for M'.



By the definition of legality, we claim that, in this instance, the set $\mathcal{L} = \{M_1, M_2\}$ (circled above) is a legal set since it satisfies both internal and external stability. This is because M_1 and M_2 do not block each other, while all other matchings are blocked by at least one of M_1 and M_2. Although M_2 is not stable, the harm due to blocking pair $(3, A)$ is not redressable given that student 3 is not matched to school A in any matching from the set $\mathcal{L}$. Moreover, every student is weakly better off in M_2 compared to in M_1. $\triangle$

From the example, one can see that relaxing stability to legality allows us to extend the set of feasible assignments, while maintaining a certain level of fairness. As we will show in Section 2.10, the increase in the size of the feasible set can be very significant. Morrill (2016) observed that, in the present setting, legality coincides with the concept of *von Neumann-Morgenstern sta-*

bility (Von Neumann and Morgenstern, 1953) in game theory under an appropriate definition of dominance. This has been investigated in the one-to-one case by Wako (2010). Morrill (2016) also showed that every one-to-many instance has a unique legal set $\mathcal{L}$. Moreover, assignments in $\mathcal{L}$ form a lattice under the classical dominance relation. By standard arguments, this implies the existence of a student-optimal legal assignment, which Morrill (2016) showed is *Pareto-efficient* for students and can be found using Kesten's EADAM (Kesten, 2010).

EADAM with consent. Recall that EADAM stands for *Efficiency Adjusted Deferred Acceptance Mechanism*. As the name implies, it aims at regaining the efficiency lost due to stability constraints. EADAM again achieves efficiency improvement without creating legal concerns, and it does so by obtaining students' consent to allow for certain blocking pairs. More specifically, starting from the student-optimal stable assignment, EADAM iteratively asks for certain students' consent to allow the removal of certain schools from their preference lists, and then re-runs Gale-Shapley's algorithm. This removes the possibility that certain student-school pairs act as blocking pairs. We defer a detailed description to later sections, and illustrate here a number of properties (showed by Kesten (2010) and Tang and Yu (2014)) that make EADAM attractive for school choice.

If a student is asked to give consent, whether he consents or not, his assignment will not change and thus, no student has the incentive to not consent, and no student is harmed under EADAM. Moreover, EADAM outputs an assignment that is *constrained efficient*. That is, this assignment does not violate any nonconsenting students' priorities (i.e., no nonconsenting student is part of a blocking pair), but any other assignment that is weakly preferred by all students does. When all students consent, the output is therefore Pareto-efficient.

Although EADAM is not strategy-proof (i.e., a student can misstate his preference list in order to be assigned to a better school), Kesten (2010) remarked that violation of strategy-proofness does not necessarily imply easy manipulability in practice (see, e.g., Roth and Peranson, 1999), as agents usually do not have complete information about the preferences of other agents in the market and are thus unlikely to engage in potentially profitable strategic behaviors (Roth and Rothblum, 1999). Kesten (2010) also proved that any mechanism that improves over the student-optimal stable mechanism either violates some nonconsenting students' priority or is not strategy-proof.

Although both EADAM and legal assignments have been further analyzed and extended by several authors (see, e.g., Dur, Gitmez, and Yılmaz, 2015; Ehlers and Morrill, 2020; Troyan, Delacrétaz, and Kloosterman, 2020; Afacan, Alioğulları, and Barlo, 2017; Tang and Zhang, 2017), our knowledge of those two concepts is far from complete. In particular, the knowledge that legal assignments form a lattice gives little information on how to exploit it for algorithmic purposes, e.g., how to find the legal assignment that maximizes some linear profit function[2]. Moreover, little is known on how to exploit the structure of legal assignments to obtain the output of EADAM when not all students consent, since the assignment output by the algorithm may not be legal.

2.1.1 Main Results

Our first contribution addresses the structure of legal assignments. We prove in Section 2.3 that the set of legal assignments coincides with the set of stable assignments in a subinstance of the original one. That is, we can describe the set of legal assignments exactly as a set of stable assignments in a subinstance. We also show, by building on our approach and on results by Ehlers and Morrill (2020), that legal assignments coincide with the set of stable assignments in a subinstance for the more general case where school preferences are represented by substitutable and consistent choice functions. We defer details to Section 2.3 and to the appendix.

As our second contribution, in Section 2.7 we show how to obtain the aforementioned subinstance in time linear in the number of edges of the input. Hence, in order to solve an optimization problem over the set of legal assignments (e.g., to find the already mentioned school-optimal, or other assignments of interest such as the *egalitarian*, *profit-optimal*, *minimum regret*), one can resort to the broad literature on algorithms developed for the same problem on the set of stable

[2] Indeed, even though Birkhoff's representation theorem (Birkhoff, 1937) implies that there is a bijection between the elements of a distributive lattice and the closed sets of an associated poset, it is not clear how to use this information algorithmically. A typical example are *strongly stable matchings*, which have been known for a long time to form a distributive lattice (Manlove, 2002), but only recently was this structure exploited for algorithmic purposes (Kunysz, Paluch, and Ghosal, 2016). See Chapter 3 for sufficient conditions on algorithmic exploitation of Birkhoff's theorem.

assignments (see e.g., Manlove (2013) for a collection of those results). Since the worst-case running time of those algorithms is at least linear in the number of the edges, the complexity of the related problems over the set of legal assignments does not exceed their complexity over the set of stable assignments. To achieve this second contribution, we rely on the concept of meta-rotations (Bansal, Agrawal, and Malhotra, 2007a) and develop a symmetric pair of algorithms in Section 2.6, which we name `student-rotate-remove` and `school-rotate-remove`, that respectively find the school-optimal and student-optimal legal assignments.

Our third contribution is a fast algorithm for EADAM with consent. Algorithmic results above imply that, when all students consent, EADAM can be implemented as to run with the same time complexity as that of Gale-Shapley's, which is linear in the number of edges of the input. However, when only some students consent, the output of EADAM may no longer be legal (see Example 2.30). We show in Section 2.8 how to modify `school-rotate-remove` to produce the output of EADAM, again within the same time bound as Gale-Shapley's. Hence, for two-sided matching markets, if one were to switch from the currently widely used deferred acceptance mechanism to EADAM, the computational time required to obtain a solution would not significantly increase. Computational tests on random instances performed in Section 2.8.4 confirm that our algorithms run significantly faster in practice.

As our last contribution, we show that when relaxing stability to legality, we can greatly increase the number of feasible matchings. We show one-to-one instances that have only one stable matching, but exponentially many (in the number of agents) legal matchings. This is achieved by an exploration of the connection between *Latin marriages* introduced by Benjamin, Converse, and Krieger (1995) and legal matchings. We defer details to Section 2.10.

Our algorithm implementations for (1) finding student-optimal and school-optimal legal assignments and obtaining the legal subinstance; and (2) EADAM with consent can be found online[3].

[3](1). `https://github.com/xz2569/LegalAssignments`. (2). `https://github.com/xz2569/FastEADAM`.

2.1.2 Literature Review

There is a vast amount of literature on mechanism design for the school choice problem, balancing the focus among strategy-proofness, efficiency, and stability. From a theoretical perspective, Ergin (2002) shows that under certain acyclicity conditions on the priority structure, the student-optimal stable assignment is also Pareto-efficient for the students. Kesten (2010) interprets these cycles as sets of interrupting pairs (see Section 2.8.1 for a formal definition) and proposes EADAM, which improves efficiency by obtaining students' consent to waive their priorities.

Extending upon Kesten's framework, many researchers offer new perspectives. Tang and Yu (2014) propose a simplified version of EADAM, which repeatedly runs Gale-Shapley's algorithm after fixing the assignments of underdemanded schools. Bando (2014) shows an algorithm which iteratively runs Gale-Shapley's algorithm after fixing the assignments of the set of last proposers. Bando (2014) also shows that when restricting to the one-to-one setting, his algorithm finds the student-optimal matching in the *von Neumann-Morgenstern (vNM) stable set*. *vNM stable set* is a concept proposed by Von Neumann and Morgenstern (1953) for cooperative games. The definition of vNM stable set requires an irreflexive dominance relation among outcomes in the set.

For the stable assignment problem, the definition of legal assignments in Morrill (2016) corresponds to vNM stable set under the dominance relation *dom*, where assignment M_1 *dom* M_2 if M_1 blocks M_2. Under this dominance relation, results from Ehlers (2007) and Wako (2010) show existence and uniqueness of the vNM stable set in the one-to-one setting. Morrill (2016) further proves the existence and uniqueness results in the one-to-many setting, as well as the fact that the vNM stable set has a lattice structure. Morrill (2016) is superseded by Ehlers and Morrill (2020), where the concept of legality and the above-mentioned results are generalized to the setting where schools' preferences are specified by substitutable choice functions that satisfy the law of aggregate demand. To the best of our understanding, results from Ehlers and Morrill (2020) do not have any implication in the stable assignment setting, other than those that already follow from Morrill (2016), mentioned above. Interestingly, Ehlers and Morrill (2020) also investigate a different dominance relation *dom*′ (which they call "vNM-blocks") and observe that *dom* and *dom*′ lead to different vNM stable sets.

Wako (2010) presents an algorithm that finds the man- and woman-optimal matchings in the vNM stable set (under the dominance relation *dom* defined above) in the one-to-one case, and shows that the vNM stable set coincides with the set of stable matchings in another instance. When restricted to the one-to-one case, our algorithms from Section 2.6 essentially projects to that of Wako (2010), as Wako (2010) also obtains, e.g., the woman-optimal legal matching by starting from the woman-optimal stable matching and iteratively finding rotations and eliminating edges. However, our approach is different because, unlike Wako (2010), we show that legal assignments are stable assignments in a subinstance before and independently of the algorithm for finding them. Even when restricted to the one-to-one case, this allows for a more direct derivation and, we believe, a more intuitive understanding of the algorithm, and a simpler and shorter proof overall. Moreover, as Wako (2010) points out, his results do not have either structural or algorithmic implications for the vNM stable set in the one-to-many setting, and he actually poses as an open question to construct an algorithm to produce such assignments.

Our results answer this open question and allow us to also characterize legal assignments in the more general setting of Ehlers and Morrill (2020). We remark that, although there is a standard reduction from one-to-many instances to one-to-one instances (Gusfield and Irving, 1989; Roth and Sotomayor, 1990) such that the set of stable assignments of the former and the set of stable matchings of the latter correspond, this one-to-one mapping fails for the set of legal assignments (see Example 2.4). So we need to directly tackle the one-to-many setting.

2.2 Model and Notations

For $n \in \mathbb{N}$, we denote by $[n]$ the set $\{1, \ldots, n\}$. All (di)graphs in this chapter are simple. All paths and cycles in (di)graphs are therefore uniquely determined by the sequence of nodes they traverse, and are denoted using this sequence, e.g., $a_0, b_0, a_1, b_1, \cdots$. The edge connecting two nodes a, b in an undirected graph is denoted by ab. For a graph G, we denote by $V(G)$ and $E(G)$ its set of vertices and set of edges respectively. For $v \in V(G)$, we let $\deg_G(v)$ denote the degree of v (i.e., the number of adjacent vertices of v) in G. For a graph $G(V, E)$ and $F \subseteq E$, we denote by $G[F] := G(V, F)$. A *singleton* of a graph is a node of degree 0. For sets S, S', $S \triangle S'$ denotes their symmetric difference. A *sink* of a digraph is a node of outdegree 0.

An instance of the *stable assignment* problem is a triple $(G, <, \mathbf{q})$ with $G = (A \cup B, E)$, where G is a bipartite graph with bipartition (A, B), $<$ denotes the set $\{<_v\}_{v \in A \cup B}$, with $<_v$ being a strict ordering of the neighbors of v in G, and $\mathbf{q} = \{q_b\}_{b \in B} \in \mathbb{N}^B$ denotes the maximum number of vertices in A that can be assigned to each $b \in B$. q_b is called the *quota* of b. Elements of A are referred to as *students* and elements of B are referred to as *schools*. For $x, y, y' \in A \cup B$ with $xy, xy' \in E$ we say x *strictly prefers* y to y' if $y >_x y'$, and we say that x *(weakly) prefers* y to y' and write $y \geq_x y'$ if $y >_x y'$ or $y = y'$. For all $xy \in E$, we assume $y >_x \emptyset$. That is, the fact that ab is an edge in E means that a prefers to be assigned to b than to be unassigned and b prefers to accept a than to accept fewer than q_b students. When $\mathbf{q}$ is the vector all of 1's, we speak of an instance of the *stable marriage* problem, and denote it by $(G, <)$. In this case, elements of A are referred to as *men* and elements of B are referred to as *women*.

An *assignment* M for an instance $(G, <, \mathbf{q})$ is a collection of edges of G such that: at most one edge of M is incident to a for each $a \in A$; at most q_b edges of M are incident to b for each $b \in B$. For $x \in A \cup B$, we write $M(x) = \{y : xy \in M\}$. When $M(x) = \{y\}$, we often think of $M(x)$ as an element instead of a set and write $M(x) = y$. For $ab \in E$ and an assignment M, we call ab a *blocking pair* for M if student a prefers school b to his currently assigned school (i.e., $b >_a M(a)$) and school b either has empty seats (i.e., $|M(b)| < q_b$) or it prefers student a to someone who is currently occupying a seat at school b (i.e., $a >_b a'$ for some $a' \in M(b)$). In this case, we say that ab *blocks* M, and similarly, we say that M' blocks M for every assignment M' containing edge ab[4]. An assignment is *stable* if it is not blocked by any edge of G.

We let $\mathcal{M}(G, \mathbf{q})$ be the set of all assignments of $(G, <, \mathbf{q})$, and let $\mathcal{S}(G, <, \mathbf{q})$ be the set of all stable assignments of $(G, <, \mathbf{q})$. For a subgraph G' of G, we denote by $(G', <, \mathbf{q})$ the stable assignment instance whose preference lists are those induced by $<$ on G' and quotas are those

[4]This notion of an assignment blocking another assignment is not standard, and is adopted from (Morrill, 2016).

obtained by restricting $\mathbf{q}$ to nodes in G'. We say that a student-school pair (a, b) is a *stable pair* if there exists a stable matching $M \in \mathcal{S}(G, <, \mathbf{q})$ in which student a is assigned to school b. In such cases, we also say that a is a *stable partner* of b.

Every instance has at least one stable assignment. Algorithms proposed by Gale and Shapley (1962) output special stable assignments. The following theorem collects results from Gale and Shapley (1962) and Gusfield and Irving (1989).

Theorem 2.2. *The student-proposing Gale-Shapley's algorithm outputs a stable assignment M_0 that is optimal for the students: for any stable assignment $M \in \mathcal{S}(G, <, \mathbf{q})$, every student a prefers M_0 to M (i.e., $M_0(a) \geq_a M(a)$). Similarly, the school-proposing Gale-Shapley's algorithm outputs a stable assignment M_z that is optimal for the schools. Moreover, M_z is student-pessimal: for any stable assignment $M \in \mathcal{S}(G, <, \mathbf{q})$, every student a prefers M to M_z (i.e., $M(a) \geq_a M_z(a)$).*

2.3 Legal Assignments are Stable Assignments in Disguise

For an instance $(G, <, \mathbf{q})$ of the stable assignment problem and a set of assignments $\mathcal{M}' \subseteq \mathcal{M}(G, \mathbf{q})$, define $\mathcal{I}(\mathcal{M}')$ as the set of assignments that are blocked by some assignment from $\mathcal{M}'$. We say a set of assignments $\mathcal{M}'$ has the *legal property* if no assignment from $\mathcal{M}'$ is blocked by any assignment from $\mathcal{M}'$ (*internal stability*), and every assignment not in $\mathcal{M}'$ is blocked by some assignment from $\mathcal{M}'$ (*external stability*). These two requirements can be summarized as a fixed-point condition: $\mathcal{I}(\mathcal{M}') = \mathcal{M}(G, \mathbf{q}) \setminus \mathcal{M}'$. In this case, we say that $(\mathcal{M}', \mathcal{I}(\mathcal{M}'))$ is a *legal partition* of $\mathcal{M}(G, \mathbf{q})$. We show the following theorem.

Theorem 2.3. *Let $(G, <, \mathbf{q})$ be an instance of the stable assignment problem. There exists a unique set of assignments $\mathcal{L} \subseteq \mathcal{M}(G, \mathbf{q})$ that has the legal property. This set coincides with the set of stable assignments in $(G_L, <, \mathbf{q})$, where G_L is a subgraph of G induced by all and only edges that are in some assignment from $\mathcal{L}$. That is,*

$$E(G_L) = \bigcup\{M : M \in \mathcal{S}(G_L, <, \mathbf{q})\} = \bigcup\{M : M \in \mathcal{L}\}$$

As observed by Gusfield and Irving (1989) and Roth and Sotomayor (1992), a stable assignment instance $(G, <, \mathbf{q})$ can be transformed into a stable marriage instance $(H_G, <_G)$ via the following well-known reduction so that there is a one-to-one correspondence between stable assignments in $(G, <, \mathbf{q})$ and stable matchings in $(H_G, <_G)$. For each school $b \in B$, create q_b copies $b^1, \ldots, b^{q_b}$ of b, and replace b in the preference list of each adjacent $a \in A$ by the q_b copies in exactly this order. The preference list of each b^i is identical to the preference list of b. We call these copies *seats* of the schools and denote their collection by B_H. With this reduction, we can construct a map $\pi : \mathcal{M}(G, \mathbf{q}) \to \mathcal{M}(H_G, \mathbf{1})$ that induces a bijection between $\mathcal{S}(G, <, \mathbf{q})$ and $\mathcal{S}(H_G, <_G, \mathbf{1})$. Given $M \in \mathcal{M}(G, \mathbf{q})$, assume for some $b \in B$, $M(b) = \{a_1, \ldots, a_j\}$ and $a_1 >_b a_2 >_b \cdots >_b a_j$. Define $\pi(M)(b^i) = a_i$ for $i \in [j]$ and $\pi(M)(b^i) = \emptyset$ for $i = j+1, \ldots, q_b$. For the sake of shortness, we often abbreviate $M_H = \pi(M)$.

One could think of proving Theorem 2.3 by showing the (simpler) results for the instance $(H_G, <_G)$, and then deducing the set of legal assignments of $(G, <, \mathbf{q})$ from the set of legal matchings of $(H_G, <_G)$. Unfortunately, the bijection between stable assignments and stable matchings does not extend to the legal setting, as we demonstrate in Example 2.4 below.

Example 2.4. Consider an instance with 4 students and 2 schools, each with 2 seats. Let a_i, b_i, b_i^j represent students, schools, and seats respectively. The preference lists are given as follows. Since it is clear whose preference list we are referring to, the subscript in $>$ is dropped.

$$
\begin{array}{llll}
a_1: & b_1 > b_2 & \qquad b_1: & a_3 > a_4 > a_2 > a_1 \\
a_2: & b_2 > b_1 & \qquad b_2: & a_2 > a_4 > a_3 > a_1 \\
a_3: & b_2 > b_1 & & \\
a_4: & b_1 > b_2 & &
\end{array}
$$

Since all preference lists are complete, we can restrict our attention to the 6 assignments where all students are matched. One can easily verify that $M = \{a_1b_1, a_2b_2, a_3b_2, a_4b_1\}$ is the only stable assignment, and all other assignments are blocked by some edge in M. Thus, $\mathcal{L} = \{M\}$. Now, consider the reduced stable marriage instance. The preference lists can be expanded:

$$\begin{array}{llll} a_1: & b_1^1 > b_1^2 > b_2^1 > b_2^2 & b_1^1: & a_3 > a_4 > a_2 > a_1 \\ a_2: & b_2^1 > b_2^2 > b_1^1 > b_1^2 & b_1^2: & a_3 > a_4 > a_2 > a_1 \\ a_3: & b_2^1 > b_2^2 > b_1^1 > b_1^2 & b_2^1: & a_2 > a_4 > a_3 > a_1 \\ a_4: & b_1^1 > b_1^2 > b_2^1 > b_2^2 & b_2^2: & a_2 > a_4 > a_3 > a_1 \end{array}$$

The corresponding matchings and their blocking pairs are:

	matchings	blocking pairs	classification
M_{1H}	$\{a_1b_1^2, a_2b_1^1, a_3b_2^2, a_4b_2^1\}$	$\underline{a_2b_2^1}, a_2b_2^2, a_4b_1^1, a_4b_1^2$	illegal
M_{2H}	$\{a_1b_1^2, a_2b_2^1, a_3b_1^1, a_4b_2^2\}$	$a_4b_1^2$	legal
M_{3H}	$\{a_1b_1^2, a_2b_2^1, a_3b_2^2, a_4b_1^1\}$	none	stable
M_{4H}	$\{a_1b_2^2, a_2b_1^2, a_3b_1^1, a_4b_2^1\}$	$\underline{a_2b_2^1}, a_2b_2^2, \underline{a_3b_2^2}, a_4b_1^2$	illegal
M_{5H}	$\{a_1b_2^2, a_2b_1^2, a_3b_2^1, a_4b_1^1\}$	$\underline{a_2b_2^1}, a_2b_2^2$	illegal
M_{6H}	$\{a_1b_2^2, a_2b_2^1, a_3b_1^1, a_4b_1^2\}$	$\underline{a_3b_2^2}$	illegal

M_{3H} is the only stable matching. All other matchings except for M_{2H} are blocked by some edge in M_{3H} (underlined). Hence, one easily verifies that $\{M_{2H}, M_{3H}\}$ has the legal property and $\pi^{-1}(M_{3H}) = M$ but $\pi^{-1}(M_{2H}) \neq M$. $\triangle$

The proof of Theorem 2.3 presented in this section relies on the following result (Theorem 2.5) by Morrill (2016). In Section 2.4, we present an alternative proof of Theorem 2.3, which is self-contained and does not rely on Theorem 2.5.

Theorem 2.5. *Let $(G, <, \mathbf{q})$ be an instance of the stable assignment problem. There exists a unique set of assignments $\mathcal{L} \subseteq \mathcal{M}(G, \mathbf{q})$ that satisfies the legal property.*

For a stable assignment instance $(G, <, \mathbf{q})$, we denote by $\mathcal{L}(G, <, \mathbf{q}) \subseteq \mathcal{M}(G, <, \mathbf{q})$ the unique set of assignments that satisfies the legal property. Moreover, we call $\mathcal{L}(G, <, \mathbf{q})$ the set of *legal assignments* of instance $(G, <, \mathbf{q})$. We say that an edge $e \in E(G)$ is *legal* if it is contained in some assignment from $\mathcal{L}(G, <, \mathbf{q})$, and is *illegal* otherwise.

For the rest of the section, we fix a stable assignment instance $(G, <, \mathbf{q})$ with $G = (A \cup B, E)$ and let $\mathcal{M} := \mathcal{M}(G, \mathbf{q})$ be the set of assignments, $\mathcal{L} := \mathcal{L}(G, <, \mathbf{q})$ be the set of legal assignments, $\mathcal{I} := \mathcal{M} \setminus \mathcal{L}$ be the set of illegal assignments, $\bar{E} := \bigcup\{M : M \in \mathcal{L}\}$ be the set of legal edges, and $G_L := G[\bar{E}]$ be the subgraph with the illegal edges removed. Next lemmas show that illegal edges can be removed from G without modifying the set of legal assignments.

Lemma 2.6. *Let e be an illegal edge. Removing the edge e from the instance does not change the set of legal assignments. That is, $\mathcal{L} = \mathcal{L}(\widetilde{G}, <, \mathbf{q})$, where $\widetilde{G} := G[E \setminus \{e\}]$.*

Proof. Let $\mathcal{M}^e := \{M \in \mathcal{M} : e \in M\}$ be the set of assignments that contain the illegal edge e and let $\widetilde{\mathcal{M}} := \mathcal{M}(\widetilde{G}, \mathbf{q})$ be the set of all assignments of the instance $(\tilde{G}, <, \mathbf{q})$. Note that $\widetilde{\mathcal{M}} = \mathcal{M} \setminus \mathcal{M}^e$. Moreover, all legal assignments are assignments of the instance $(\tilde{G}, <, \mathbf{q})$ since e is an illegal edge: $\mathcal{L} \subseteq \widetilde{\mathcal{M}}$. Hence, $(\mathcal{L}, \widetilde{\mathcal{M}} \setminus \mathcal{L})$ is a partition of $\widetilde{\mathcal{M}}$. We show next that it is also a legal partition. To see this, first note that any two assignments $M_1, M_2 \in \mathcal{L}$ do not block each other, since $\mathcal{L}$ is the set of legal assignments of the original instance. Next, consider any assignment $M' \in \widetilde{\mathcal{M}} \setminus \mathcal{L}$. Then, M' is an illegal assignment in the original instance and must be blocked by some assignment in $\mathcal{L}$. Thus, together with the uniqueness of the legal partition given by Theorem 2.5, we conclude that $\mathcal{L}(\widetilde{G}, <, \mathbf{q}) = \mathcal{L}$. ■

Lemma 2.7. *The set of legal assignment does not change after removing all illegal edges. That is, $\mathcal{L} = \mathcal{L}(G_L, <, \mathbf{q})$.*

Proof. Let $e_1, e_2, \cdots, e_k$ be an ordering of the illegal edges and for $i = 1, 2, \cdots, k$, let $G^i := G[E \setminus \{e_1, e_2, \cdots, e_i\}]$ be a sequence of subgraphs obtained after successively removing illegal edges. Observe that $G^k = G_L$. By Lemma 2.6, we have $\mathcal{L}(G^1, <, \mathbf{q}) = \mathcal{L}$ and thus, by definition of illegal edges, edges $e_2, \cdots, e_k$ remain illegal in instance $(G^1, <, \mathbf{q})$. Therefore, applying Lemma 2.6 again to $(G^1, <, \mathbf{q})$, we have $\mathcal{L}(G^2, <, \mathbf{q}) = \mathcal{L}(G^1, <, \mathbf{q}) = \mathcal{L}$. Iterating the process, we can conclude that $\mathcal{L}(G_L, <, \mathbf{q}) = \mathcal{L}(G^k, <, \mathbf{q}) = \mathcal{L}(G^{k-1}, <, \mathbf{q}) = \cdots = \mathcal{L}$. ■

Lemma 2.8. *Once all illegal edges have been removed, the set of stable assignments coincides with the set of legal assignments:* $\mathcal{S}(G_L, <, \mathbf{q}) = \mathcal{L}(G_L, <, \mathbf{q})$.

Proof. The direction $\mathcal{S}(G_L, <, \mathbf{q}) \subseteq \mathcal{L}(G_L, <, \mathbf{q})$ is clear, since a stable assignment is not blocked by any other assignment. For the other direction, let $M \in \mathcal{L}(G_L, <, \mathbf{q})$ be a legal assignment of the instance $(G_L, <, \mathbf{q})$. Then M is not blocked by any assignment in $\mathcal{L}(G_L, <, \mathbf{q})$ due to internal stability. Since every edge in $E(G_L)$ appears in at least one assignment in $\mathcal{L}(G_L, <, \mathbf{q})$, M admits no blocking pair in G_L and thus is stable in the instance $(G_L, <, \mathbf{q})$. This concludes the proof. ■

Proof of Theorem 2.3. Immediately from Theorem 2.5, Lemma 2.7, and Lemma 2.8. ■

The approach developed in this section can be extended to the more general setting studied in Ehlers and Morrill (2020), where schools' preferences are represented by certain choice functions. In particular, Theorem 2.3 also holds in this setting. We defer details to Appendix A.1.

We have shown that legal assignments are stable assignments in G_L. Since there might be an exponential number of legal assignments, one cannot expect to construct G_L efficiently by explicitly listing all the legal assignments. Instead, the main tool we use is an efficient mechanism in identifying legal and illegal edges, which is developed in Section 2.6. Before we introduce this algorithm, we need some more properties of stable assignments, which we introduce in Section 2.5.

2.4 Self-Contained Proof of Theorem 2.3

In this section, we present a proof of Theorem 2.3 that builds only on "classical" concepts of stable matchings introduced in Chapter 1. Recall that $\mathcal{M} := \mathcal{M}(G, \mathbf{q})$ denotes the set of all assignments of the instance $(G, <, \mathbf{q})$ and for a set of assignments $\mathcal{M}' \subseteq \mathcal{M}$, $\mathcal{I}(\mathcal{M}')$ is defined as the set of assignment that are blocked by some assignment from $\mathcal{M}'$. The proof of Theorem 2.3 relies on the study of the fixed points of a function $\mathcal{L}$ applied to $\mathcal{S} := \mathcal{S}(G, <, \mathbf{q})$, where $\mathcal{L}(\mathcal{M}')$ is defined as the set of assignments that are *not* blocked by *any* assignment in $\mathcal{M} \setminus \mathcal{I}(\mathcal{M}')$. That is, $\mathcal{L}(\mathcal{M}') := \mathcal{M} \setminus \mathcal{I}\big(\mathcal{M} \setminus \mathcal{I}(\mathcal{M}')\big)$[5].

[5]We adapt the function $\mathcal{L}$ from a similar operator introduced in Morrill (2016).

In order to get some intuition about the $\mathcal{L}$ operator, let $\mathcal{L}^0 := \mathcal{S}$ and iteratively define $\mathcal{L}^i := \mathcal{L}(\mathcal{L}^{i-1})$ for $i \in \mathbb{N}$. In addition, let $\mathcal{I}^i := \mathcal{I}(\mathcal{L}^i)$. Clearly, any set of assignments that satisfies the legal property must contain $\mathcal{S}$, and as a result, must not contain any assignment from $\mathcal{I}^0$. Therefore, if an assignment is only blocked by assignments in $\mathcal{I}^0$, it is contained in every set of assignments that satisfies the legal property. All such assignments, together with those in $\mathcal{L}^0$, give exactly $\mathcal{L}^1$. Similarly, we can enlarge $\mathcal{L}^1$ to $\mathcal{L}^2$, etc. As Morrill (2016) also observed, the sequence $\mathcal{L}^0, \mathcal{L}^1, \mathcal{L}^2, \cdots$ converges.

Lemma 2.9. *There exists $k \in \mathbb{N}$ such that $\mathcal{L}^k = \mathcal{L}^{k+1}$.*

Proof. We show by induction on i that $\mathcal{L}^i \subseteq \mathcal{L}^{i+1}$, which concludes the proof since the set of assignments is finite. For the base case, clearly $\mathcal{L}^0 = \mathcal{S} \subseteq \mathcal{L}^1$. For the inductive step, fix $i \in \mathbb{N}$. Since $\mathcal{L}^{i-1} \subseteq \mathcal{L}^i$, we deduce $\mathcal{I}^{i-1} \subseteq \mathcal{I}^i$ and therefore, $\mathcal{I}(\mathcal{M} \setminus \mathcal{I}^i) \subseteq \mathcal{I}(\mathcal{M} \setminus \mathcal{I}^{i-1})$. Hence,

$$\mathcal{L}^{i+1} = \mathcal{L}(\mathcal{L}^i) = \mathcal{M} \setminus \mathcal{I}(\mathcal{M} \setminus \mathcal{I}^i) \supseteq \mathcal{M} \setminus \mathcal{I}(\mathcal{M} \setminus \mathcal{I}^{i-1}) = \mathcal{L}^i.$$

■

Example 2.10. Consider the instance given in Example 2.1. We have $\mathcal{L}^0 = \mathcal{S} = \{M_1\}$. Since M_1 blocks M_3, M_4, and M_5, we have $\mathcal{I}^0 = \mathcal{I}(\mathcal{L}^0) = \{M_3, M_4, M_5\}$. Note that M_2 is only blocked by M_3 and M_4, both of which are in $\mathcal{I}^0$. Thus, $M_2 \in \mathcal{L}(\mathcal{L}^0)$ and $\mathcal{L}^1 = \{M_1, M_2\}$. Repeating the process again, we can see that $\mathcal{I}^1 := \mathcal{I}(\mathcal{L}^1) = \{M_3, M_4, M_5\}$ and $\mathcal{L}^2 = \{M_1, M_2\}$. The sequence thereafter stabilizes. △

Fixed points of $\mathcal{L}$ are stable assignments. We want therefore to study the set to which the sequence $\mathcal{L}^0, \mathcal{L}^1, \mathcal{L}^2, \cdots$ stabilizes. One key observation is that every fixed point of $\mathcal{L}$ coincides with the set of stable assignments in some subinstance of the original problem. Although simple, this observation has dramatic consequences. In particular, we can now rely on all the structural knowledge on stable assignments. This distinguishes our approach from that of Morrill (2016), where properties of legal assignments are developed independently from those of stable assignments.

Lemma 2.11. *Assume $\mathcal{M}_0 \subseteq \mathcal{M}$ is a set of assignments that satisfies $\mathcal{L}(\mathcal{M}_0) = \mathcal{M}_0$. Then, $\mathcal{M}_0 = \mathcal{S}(G', <, \mathbf{q})$, where $G' := G[E']$ and $E' := \bigcup\{M : M \in \mathcal{M} \setminus \mathcal{I}(\mathcal{M}_0)\}$.*

Proof. Let $M \in \mathcal{M} \setminus \mathcal{M}_0$ be an assignment not in $\mathcal{M}_0$. Then, $M \notin \mathcal{L}(\mathcal{M}_0)$ and thus, there is an assignment $M' \in \mathcal{M} \setminus \mathcal{I}(\mathcal{M}_0)$ and an edge $e \in M'$ that blocks M. Note that $e \in E'$ by definition of E'. Therefore, $M \notin \mathcal{S}(G', <, \mathbf{q})$. This shows $\mathcal{S}(G', <, \mathbf{q}) \subseteq \mathcal{M}_0$. Conversely, let $M \in \mathcal{M} \setminus \mathcal{S}(G', <, \mathbf{q})$ be an assignment of the original instance that is not stable in $(G', <, \mathbf{q})$. If M is also an assignment in the subinstance (i.e., $M \in \mathcal{M}(G', \mathbf{q})$), then M is blocked by some edge $e \in E'$. This means that an assignment in $\mathcal{M} \setminus \mathcal{I}(\mathcal{M}_0)$ blocks M, implying that $M \notin \mathcal{L}(\mathcal{M}_0)$ and thus, $M \notin \mathcal{M}_0$. If M is not an assignment in the subinstance (i.e., $M \notin \mathcal{M}(G', \mathbf{q})$), then M contains an edge that is not in E'. This implies $M \in \mathcal{I}(\mathcal{M}_0)$ and thus $M \notin \mathcal{L}(\mathcal{M}_0) = \mathcal{M}_0$. ■

Assignments that do not block each other. Besides properties of stable assignments, we will also use properties of assignments that do not block each other. Those facts are established in the next two lemmas. They can be seen as an extension of the "opposition of interest" property: if a and b are partners in a stable assignment M, then they cannot both strictly prefer another stable assignment M' to M Gusfield and Irving, 1989, Theorem 1.3.1.

Lemma 2.12. *Consider an instance of stable marriage problem $(G, <)$ with $G = G(A \cup B, E)$. Let $M, M' \in \mathcal{M}(G, \mathbf{1})$ be two matchings of the instance. We say an edge $ab \in M \cup M'$ is* irregular *if both a and b strictly prefer M to M' or both strictly prefer M' to M. Suppose M does not block M' and M' does not block M. Then:*

1) *there are no irregular edges;*

2) *$G[M \triangle M']$ is a disjoint union of singletons and cycles;*

3) *a node is matched in M if and only if it is matched in M'.*

Proof. 1) Assume a_1b_1 is an irregular edge and assume wlog both endpoints strictly prefer M to M'. Then $a_1b_1 \in M'$, because otherwise M blocks M'. Starting from $i = 2$, iteratively define

$a_i = M(b_{i-1})$ and $b_i = M'(a_i)$. Repeatedly using the assumption that M and M' do not block each other, we deduce that, for all $i \geq 2$, a_i strictly prefers M' to M, and vice versa b_i strictly prefers M to M'. Moreover, $a_i \neq \emptyset$ and $b_i \neq \emptyset$. Since M and M' are matchings, there exists $\ell \geq 2$ such that $a_\ell = a_1$. Hence, $a_1 = a_\ell$ strictly prefers M' to M, which is a contradiction.

2) Note that the degree of each node in $G[M \triangle M']$ is at most 2. Suppose the thesis does not hold, then $G[M \triangle M']$ contains a path, say wlog $a_1, b_1, a_2, b_2, \cdots$, whose endpoints have degree 1 in $G[M \triangle M']$. Assume wlog that $a_1 b_1 \in M'$. Since a_1 is unmatched in M, a_1 strictly prefers M' to M. In addition, since $a_1 b_1 \in M'$ does not block M, b_1 strictly prefers M to M'. We can iterate and conclude, similarly to part 1), that all nodes a_i strictly prefer M' to M, and vice versa all nodes b_i strictly prefer M to M'. Suppose first that $a_k b_k$ is the last edge of the path. Then b_k strictly prefers M' to M as $a_k b_k \in M'$ and $M(b_k) = \emptyset$, which is a contradiction. Similarly, if instead the last edge is $b_k a_{k+1}$, a_{k+1} strictly prefers M as a_{k+1} is unmatched in M'. This is again a contradiction.

3) Immediately from 2). ■

Lemma 2.13. *Let $(G(A \cup B, E), <, \mathbf{q})$ be an instance of the stable assignment problem. Let $M, M' \in \mathcal{M}(G, \mathbf{q})$ be two assignments such that M does not block M' and M' does not block M, Let $a \in A$ be a student matched in M. Then a is matched in M'. Let therefore $b = M(a)$, $\overline{b} = M'(a)$. If $b >_a \overline{b}$, then there exists $\overline{a} \in M(\overline{b})$ such that $a >_{\overline{b}} \overline{a}$ and $\overline{b} >_{\overline{a}} M'(\overline{a})$.*

Proof. Let $M_H = \pi(M)$ and $M'_H = \pi(M')$, where π is the mapping defined in Section 2.3. In order to show that a is matched in M', we first prove that M'_H and M_H do not block each other. It then follows from Lemma 2.12, part 3) and the definition of mapping π that a is matched in M'. Assume therefore by contradiction that there exists $\bar{a}b^i \in M'_H$ that blocks M_H. That means $b^i >_{\bar{a}} M_H(\bar{a})$ and $\bar{a} >_{b^i} M_H(b^i)$. If $M_H(\bar{a}) = b_1^j$ and $b_1 \neq b$, then $b >_{\bar{a}} M(\bar{a})$ and $\bar{a} >_b a'$ for some $a' \in M(b)$. Therefore, $\bar{a}b \in M'$ blocks M, which is a contradiction. So assume $M_H(\bar{a}) = b^j$ for some $j \in [q_b]$. Since $b^i >_{\bar{a}} b^j$, we have $j > i$ by construction of $(H_G, <_G)$. Then, by the definition of mapping π, $M_H(b^i) >_b M_H(b^j) = \bar{a}$, which is again a contradiction.

To show the second statement, let $\overline{b}^i := M'_H(a)$ and $b^\ell := M_H(a)$. Because of what is shown above, we can apply Lemma 2.12, part 2) and conclude that there exists a cycle $C = a, b^\ell, ..., \overline{b}^i$

in $H_G[M'_H \triangle M_H]$, and this cycle has no irregular edges. Since $b^\ell >_a \bar{b}^i$, (i) all nodes from $A \cap C$ strictly prefer M_H to M'_H, and vice-versa (ii) all nodes from $B_H \cap C$ strictly prefer M'_H to M_H. Recall that B_H is the collection of seats in the reduced instance $(H_G, <_G)$. Let $\bar{b}^j \in C$ be a seat of school $\bar{b}$ such that the node of $C \cap B_H$ that immediately precedes $\bar{b}^j$ in C is not a seat of $\bar{b}$, while all nodes that follow $\bar{b}^j$ in $C \cap B_H$ are seats of $\bar{b}$. Note that $\bar{b}^j$ is well-defined, since $b \neq \bar{b}$ and C terminates with $\bar{b}^i$ (hence possibly $j = i$) Let $\bar{a} := M_H(\bar{b}^j)$, i.e., $C = a, b^\ell, \cdots, \bar{a}, \bar{b}^j, \cdots, \bar{b}^i$. Note that by choice of $\bar{b}_j$, $M'(\bar{a}) \neq \bar{b}$. By (i) above, $\bar{b}^j = M_H(\bar{a}) >_{\bar{a}} M'_H(\bar{a})$ and therefore, $\bar{b} >_{\bar{a}} M'(\bar{a})$ as required. Moreover, $\bar{a} = M_H(\bar{b}^j) <_{\bar{b}^j} M'_H(\bar{b}^j) \leq_{\bar{b}^j} M'_H(\bar{b}^i) = a$, where the strict preference follows from (ii) and the non-strict one follows from the definition of mapping π and (i). Hence, $a >_{\bar{b}} \bar{a}$, as required. ■

Edges of G_L. For $k \in \mathbb{N}$ that satisfies Lemma 2.9, we let $\mathcal{L} := \mathcal{L}^k$. Let G_L be the subgraph of G induced by edges $\bigcup\{M : M \in \mathcal{M} \setminus \mathcal{I}(\mathcal{L})\}$[6]. Using Gale-Shapley's algorithm and the structural properties deduced so far, we next characterize edges of G_L as all and only edges used by some assignment in $\mathcal{L}$.

Lemma 2.14. $E(G_L) = \bigcup\{M : M \in \mathcal{L}\}$.

Proof. The containment relationship $E(G_L) \supseteq \bigcup\{M : M \in \mathcal{L}\}$ is clear from definition. So it suffices to show $E(G_L) \subseteq \bigcup\{M : M \in \mathcal{L}\}$. Assume by contradiction that there exists an edge $ab \in E(G_L) \setminus \bigcup\{M : M \in \mathcal{L}\}$. Let $M \in \mathcal{M} \setminus \mathcal{I}(\mathcal{L})$ be an assignment such that $ab \in M$. Let M_0 and M_z be the stable assignment output by the student- and school-proposing Gale-Shapley's algorithm in G_L, respectively. Since $\mathcal{L} = \mathcal{L}(\mathcal{L})$, we have $M_0, M_z \in \mathcal{L}$ by Lemma 2.11. By construction, $ab \notin M_0 \cup M_z$. In the following, when talking about a specific execution of Gale-Shapley's algorithm, we say that a *rejects* b if during the execution, a rejects the proposal by b, possibly after having temporarily accepted it. We distinguish three cases.

Case a): $b >_a M_0(a) =: \bar{b}$. By the choice of M, we know that M_0 and M do not block each other (in either $(G, <, \mathbf{q})$ or $(G_L, <, \mathbf{q})$, as the preference lists are consistent), since both are

[6]Note that this definition of G_L is different from the one given in the main body of the paper. However, results from this section imply that the two definitions coincide.

assignments in $\mathcal{M} \setminus \mathcal{I}(\mathcal{L})$. Note that this case contains all and only the edges of G_L that have been rejected by some (equivalently, any) execution of the student-proposing Gale-Shapley's algorithm on G_L. Among all those edges, let ab be the one that is last rejected by some execution of the algorithm. Apply Lemma 2.13 (with $M = M$ and $M' = M_0$) and conclude that there exists $\overline{a} \in M(\overline{b})$ such that $\overline{b} >_{\overline{a}} M_0(\overline{a})$. This implies that $\overline{b}$ rejected $\overline{a}$ during the execution of Gale-Shapley's in consideration. Hence, when a proposes to $\overline{b}$, either $\overline{a}$ still has to be rejected by $\overline{b}$, or it has been rejected before. In the latter case, when a proposes to $\overline{b}$, $\overline{b}$ has her quota filled and rejects some other student. Hence the following events happen in this order during the execution of Gale-Shapley's algorithm: a is rejected by b; a proposes to $\overline{b}$; $\overline{b}$ rejects a student. This contradicts our assumption that ab is the last rejected edge.

Case b): $M_0(a) >_a b >_a M_z(a)$. First, we want to show that there exists a stable assignment M' such that $M'(a) >_a b$ and $a' >_b a$ for all $a' \in M'(b)$. Note that Lemma 2.19 implies $s_{M_0}(a) \neq \emptyset$. Apply a (possibly empty) sequence of exposed rotations from M_0 to obtain a stable assignment $M' \in \mathcal{S}(G_L, <, \mathbf{q})$ such that $M'(a) >_a b >_a s_{M'}(a) \geq_a M_z(a)$. By definition of $s_{M'}(a)$, we must have $a' >_b a$ for all $a' \in M'(b)$. Now, by choice of M, M and M' do not block each other. We can therefore apply Lemma 2.13 (with the roles of M and M' inverted) and conclude that there exists $\overline{a} \in M'(b)$ with $a >_b \overline{a}$, which is a contradiction.

Case c): $M_z(a) >_a b$. Using Lemma 2.13 (with $M = M_z$ and $M' = M$) we deduce that there exists $\bar{a} \in M_z(b)$ such that $a >_b \bar{a}$ and $b >_{\bar{a}} M(\bar{a}) =: \bar{b}$. Hence, in some (equivalently, any) iteration of the school-proposing Gale-Shapley's algorithm, a rejects b. Since this is the last case that still needs to be considered, we may assume edges $E(G_L) \setminus \bigcup\{M : M \in \mathcal{L}\}$ are exactly those rejected by some execution of the school-proposing Gale-Shapley's algorithm. Among all such edges, take ab that is the last rejected by some execution. Applying Lemma 2.13 again (with $a = \overline{a}$, $M = M_z$, $M' = M$), we know $\overline{a} >_{\overline{b}} a'$ for some $a' \in M_z(\overline{b})$. Recall that $M_z(\overline{a}) = b \neq \overline{b}$. This implies that $\overline{a}$ rejected $\overline{b}$ during the execution of Gale-Shapley's in consideration. Hence, when b proposes to $\overline{a}$, either $\overline{b}$ still has to be rejected by $\overline{a}$, or it has been rejected before. In the latter case, when b proposes to $\overline{a}$, $\overline{a}$ rejects the school it temporarily accepted. Hence, the following events happen during the considered execution in this order: a rejects b; b proposes to $\overline{a}$; $\overline{a}$ rejects a school, contradicting the choice of ab. ■

Concluding the proof. Once the previous facts have been established, concluding the proof of Theorem 2.3 is quite straightforward.

Lemma 2.15. *$\mathcal{L}$ has the legal property. That is, $\mathcal{I}(\mathcal{L}) = \mathcal{M} \setminus \mathcal{L}$.*

Proof. Clearly $\mathcal{I}(\mathcal{L}) \subseteq \mathcal{M}\setminus\mathcal{L}$. Now take $M \in \mathcal{M}\setminus\mathcal{I}(\mathcal{L})$. Then $M \subseteq \bigcup\{M' : M' \in \mathcal{M}\setminus\mathcal{I}(\mathcal{L})\} = E(G_L)$. Hence, M is an assignment of G_L not blocked by any assignment from $\mathcal{L} = \mathcal{S}(G_L, <, \mathbf{q})$, where the last equality holds by Lemma 2.11. By Lemma 2.14, M is not blocked by any edge in $E(G_L)$, and we conclude that $M \in \mathcal{S}(G_L, <, \mathbf{q}) = \mathcal{L}$. ■

Because of Lemma 2.15, we say that $(\mathcal{L}, \mathcal{I} := \mathcal{M} \setminus \mathcal{L})$ is a *legal partition* of $\mathcal{M}$.

Lemma 2.16. *$\mathcal{L}$ is the unique subset of $\mathcal{M}$ with the legal property.*

Proof. Assume by contradiction that there exists a set $\mathcal{L}' \subseteq \mathcal{M}$, $\mathcal{L}' \neq \mathcal{L}$ with the legal property. Let $\mathcal{I}' := \mathcal{M} \setminus \mathcal{L}'$. If $\mathcal{L} \subsetneq \mathcal{L}'$, we must have $\mathcal{I}' \subsetneq \mathcal{I}$. Take any $M \in \mathcal{I} \setminus \mathcal{I}'$, it must be blocked by some assignment in $\mathcal{L} \subsetneq \mathcal{L}'$. However, $M \in \mathcal{L}'$, contradicting the assumption that $\mathcal{L}'$ has the legal property. Similarly, we cannot have $\mathcal{L}' \subsetneq \mathcal{L}$. Thus, sets $\mathcal{A} := \{M : M \in \mathcal{I} \cap \mathcal{L}'\}$ and $\mathcal{B} := \{M : M \in \mathcal{L} \cap \mathcal{I}'\}$ are both non-empty. In addition, let $\mathcal{C} := \mathcal{L} \cap \mathcal{L}'$. It is also non-empty because all stable assignments are contained in any set with the legal property. In particular, $\mathcal{L}^0 \subseteq \mathcal{C}$. Note that every assignment in $\mathcal{B}$ is blocked by some assignment from $\mathcal{A}$. Moreover, (†) no assignments from $\mathcal{A} \cup \mathcal{B}$ can be blocked by any assignment from $\mathcal{C}$. Now take the first $i \in \mathbb{N}$ such that $\mathcal{L}^i \cap \mathcal{B} \neq \emptyset$, and note that $i \geq 1$. Let $M \in \mathcal{L}^i \cap \mathcal{B}$. All assignments blocking M must be contained in $\mathcal{I}(\mathcal{L}^{i-1})$. Thus, we can pick $M' \in \mathcal{I}(\mathcal{L}^{i-1}) \cap \mathcal{A}$. Hence, M' is blocked by some assignment in $\mathcal{L}^{i-1} \subseteq \mathcal{C}$ (containment relation due to the choice of i), contradicting (†). ■

Proof of Theorem 2.3. Immediately from Lemmas 2.11, 2.14, 2.15, and 2.16. ■

2.5 The Structure of Stable Assignments

In this section, we recall known results on structural properties of stable assignments and their algorithmic consequences. Throughout the section, we fix a stable assignment instance $(G, <, \mathbf{q})$, with $G = (A \cup B, E)$. Given two assignments $M, M' \in \mathcal{M}(G, \mathbf{q})$, we say M (*weakly*) *dominates* M', and write $M \succeq M'$, if every student a prefers M to M': $M(a) \geq_a M'(a), \forall a \in A$. If moreover $M \neq M'$, we say that M *strictly dominates* M' and write $M \succ M'$. An assignment $M \in \mathcal{M}(G, \mathbf{q})$ is said to be *Pareto-efficient* (for students) if there is no other assignment $M' \in \mathcal{M}(G, \mathbf{q})$ such that M' dominates M. The following fact is well-known (see, e.g., Gusfield and Irving, 1989).

Theorem 2.17. *The set of stable assignments $\mathcal{S}(G, <, \mathbf{q})$ endowed with the dominance relation $\succeq$ forms a distributive lattice. In particular, there exists stable assignments M_0 and M_z such that $M_0 \succeq M \succeq M_z$ for all stable assignment $M \in \mathcal{S}(G, <, \mathbf{q})$ (it is possible that $M_0 = M_z$). M_0 and M_z are called the* student-optimal *and* school-optimal *stable assignment, respectively.*

Note that the student-optimal (resp. school-optimal) stable assignment coincides with the one output by Gale-Shapley's algorithm with students (resp. schools) proposing, as described in Theorem 2.2. Hence, the notation describing those assignments coincide.

Next, we introduce the concept of rotations in the one-to-many setting. Informally speaking, a rotation exposed in a stable assignment M is a certain M-alternating cycle C such that $M \triangle C$ is again a stable assignment. C has the property that every agent from one side of the bipartition prefers M to $M \triangle C$, while every agent from the other side prefers $M \triangle C$ to M. We can interpret a rotation as a cycle of un-matches and re-matches with one side getting better and the other side getting worse. Hence, rotations provide a mechanism to generate one stable assignment from another, moving along the distributive lattice formed by the set of stable assignments.

Because of the different role played by the two sides of the bipartition, we distinguish between *school-* and *student-rotations*. In the following, we present them jointly by choosing X to be one side of the bipartition and Y the other[7]. We extend the quota vector $\mathbf{q}$ to students by letting $q_a = 1$ for each student $a \in A$.

For a stable assignment $M \in \mathcal{S}(G, <, \mathbf{q})$ and agent $x \in X$, let $s_M(x)$ be the first agent $y \notin M(x)$ on x's preference list such that y prefers x to one of y's partners (i.e., $x >_y x'$ for some $x' \in M(y)$). If $y := s_M(x)$ exists, we must have that x prefers all of x's partners over agent y (i.e., $y' >_x y$ for all $y' \in M(x)$) since M is stable. If moreover $|M(y)| = q_y$, define $next_M(x)$ to be the *least preferred* partner of agent y among all current partners of y. That is, $next_M(x) \in M(y)$ and for all $x' \in M(y)$, $x' \geq_y next_M(x)$. If otherwise $|M(y)| < q_y$, then define $next_M(x) = \emptyset$.

Given distinct $x_0, \dots, x_{r-1} \in X$ and $y_0, \dots, y_{r-1} \in Y$, a cycle $y_0, x_0, y_1, x_1, \dots, y_{r-1}, x_{r-1}, y_0$ of G is an X-*rotation exposed* in M if $s_M(x_i) = y_{i+1}$ and $next_M(x_i) = x_{i+1}$ for all $i = 0, \dots, r-1$ (here and later, indices are taken modulo r). Note that $x_i y_i \in M$ for all $i = 0, \dots, r-1$. Let D_X be the digraph with vertices $X \cup Y \cup \{\emptyset\}$, and arcs (x, y) and (y, x') if and only if $s_M(x) = y$ and $next_M(x) = x'$. We call D_X the X-*rotation digraph* (of M) and denote by $A(D_X)$ the set of arcs of D_X. If $s_M(x)$ does not exist for some agent $x \in X$, then x is a sink in D_X. Thus, note that sinks in D_X are either agents in X or $\emptyset$. One easily observes that X-rotations exposed in M are in one-to-one correspondence with directed cycles in D_X.

Let $\rho := y_0, x_0, \cdots, y_{r-1}, x_{r-1}$ be an X-rotation exposed in M. The *elimination* of ρ maps stable assignment M to the assignment $M' := M/\rho$ where $M'(x) = M(x)$ for every agent x who is not in the rotation (i.e., $x \in X \setminus \rho$) and $M'(x_i) = (M(x_i) \setminus \{y_i\}) \cup \{y_{i+1}\}$ for $i = 0, 1, \cdots, r-1$. Note that the mapping is well-defined since it is easy to check that M' is an assignment.

An X-rotation (digraph) is called a student- or school- rotation (digraph) respectively when X is the set of students or schools. When it is clear whether we are referring to students or schools, we drop the prefix. See Example 2.23 for an illustration of rotations and rotation digraphs in the context of our algorithm.

The following lemmas (Bansal, Agrawal, and Malhotra, 2007a) extend classical results on rotations in the one-to-one setting to our one-to-many setting. They show that the set of stable

[7] It is worth noticing that the definition of both student-rotations and school-rotations can be simplified, but in different ways. However, in order to keep the treatment compact, we give a unique presentation encompassing both.

assignments is complete and closed under the elimination of exposed rotations.

Lemma 2.18. *Let $M \in \mathcal{S}(G, <, \mathbf{q})$ be a stable assignment, ρ be an X-rotation exposed in M, and $M' = M/\rho$ be the assignment obtained after eliminating ρ from M. Then M' is stable in $(G, <, \mathbf{q})$ (i.e., $M' \in \mathcal{S}(G, <, \mathbf{q})$). Moreover, M strictly dominates M' (i.e., $M \succ M'$) if X is the set of students and M is strictly dominated by M' (i.e., $M' \succ M$) if X is the set of schools. If there is no X-rotation exposed in M, M is the Y-optimal stable assignment. In addition, every stable assignment can be generated by a sequence of X-rotation eliminations, starting from the X-optimal stable assignment, and every such sequence contains the same set of X-rotations.*

Lemma 2.19. *$xy \in E$ is a stable pair if and only if: (i) either x is assigned to y in the Y-optimal stable assignment or, (ii) for some X-rotation $y_0, x_0, y_1, x_1, \cdots, y_{r-1}, x_{r-1}$ exposed in some stable assignment, we have $x = x_i$ and $y = y_i$ for some $i \in \{0, \ldots, r-1\}$.*

We refer to Baïou and Balinski (2004) for further results on the stable assignment model.

For an instance $(G, <, \mathbf{q})$, we denote by $\mathcal{R}(G, <, \mathbf{q})$ the set of student-rotations exposed in some of its stable assignments, and by $\mathcal{SR}(G, <, \mathbf{q})$ the set of school-rotations exposed in some of its stable assignments.

Lemma 2.20. *$|\mathcal{R}(G, <, \mathbf{q})| = |\mathcal{SR}(G, <, \mathbf{q})|$. There is a bijection $\sigma : \mathcal{R}(G, <, \mathbf{q}) \rightarrow \mathcal{SR}(G, <, \mathbf{q})$ between the set of student-rotations and the set of school-rotations such that for each stable assignment $M \in \mathcal{S}(G, <, \mathbf{q})$ and student-rotation $\rho \in \mathcal{R}(G, <, \mathbf{q})$ exposed in M, we have $M = (M/\rho)/\sigma(\rho)$.*

2.6 Algorithms for Student- and School-Optimal Legal Assignments

Because of Theorem 2.3 and Theorem 2.17, the concepts of student- and school-optimal legal assignments are well-defined. In this section, we show efficient routines for finding them. Throughout the section, we again fix a stable assignment instance $(G, <, \mathbf{q})$ with $G = (A \cup B, E)$. We denote by $M_0^{\mathcal{L}}$ and $M_z^{\mathcal{L}}$ the student-optimal and school-optimal legal assignments, respectively.

Algorithm 2.1 X-`rotate-remove` to find the Y-optimal legal assignment

Input: $(G(A \cup B, E), <, \mathbf{q})$

Find the Y-optimal stable assignment M_Y of $(G, <, \mathbf{q})$ via Gale-Shapley's algorithm.
Let $G^0 := G$ and $M^0 := M_Y$.
Set $i = 0$ and let D^0 to be the X-rotation digraph of M^0 in $(G^0, <, \mathbf{q})$.
while D^i still has an arc **do**
    Find (i) arcs $(x', y), (y, x) \in A(D^i)$ where x is a sink in D^i or (ii) a cycle C^i of D^i.
    if (i) is found **then**
        Define G^{i+1} from G^i by removing $x'y$, and set $M^{i+1} = M^i$.
    else if (ii) is found **then**
        Let ρ^i be the corresponding X-rotation. Set $M^{i+1} = M^i/\rho^i$, and $G^{i+1} = G^i$.
    end if
    Set $i = i + 1$ and let D^i to be the X-rotation digraph of M^i in $(G^i, <, \mathbf{q})$.
end while

Output: M^i.

Suppose first we want to find the student-optimal legal assignment. The basic idea of the algorithm is the following: at each iteration, a legal assignment M and a set of edges identified as illegal are taken as input, and one of the following three cases will happen: either (i) the set of edges identified as illegal is expanded; or (ii) a legal assignment M' that strictly dominates M (i.e., $M' \succ M$) is produced; or (iii) M is certified as the student-optimal legal assignment. If we are in case (i), then we can safely remove the newly found illegal edge (because of Lemma 2.6) and proceed to the next iteration. If we are in case (ii), we replace M with M', and proceed to the next iteration. If we are in case (iii), we halt the algorithm and output the current assignment.

In order to distinguish between cases (i), (ii), and (iii) above, we rely on properties of the rotation digraph. In the following, X can again be either the set of students or the set of schools.

Lemma 2.21. *Let $M \in \mathcal{S}(G, <, \mathbf{q})$ be a stable assignment. If $x \in X \cup \{\emptyset\}$ is a sink in the X-rotation digraph D_X of M and $(x', y), (y, x) \in A(D_X)$ for some $x' \in X$ and $y \in Y$, then $x'y$ is an illegal edge.*

The proof of Lemma 2.21 builds on Lemma 2.13 and on the following fact (see, e.g., Gusfield and Irving, 1989).

Lemma 2.22. *Let $M, M' \in \mathcal{S}(G, <, \mathbf{q})$ be two stable assignments such that $M \succeq M'$. Then, for every school $b \in B$, $a' >_b a$ for all $a \in M(b) \setminus M'(b)$ and $a' \in M'(b) \setminus M(b)$.*

Proof of Lemma 2.21. We prove the result with X being the set of students, and thus let $x' = a'$, $y = b$, and $x = a$. The other case can be shown similarly. By Theorem 2.3, $M \in \mathcal{S}(G, <, \mathbf{q}) \subseteq \mathcal{L}(G, <, \mathbf{q}) = \mathcal{S}(G_L, <, \mathbf{q})$, where G_L is the subgraph of G with only the edges that appear in some legal assignment. In $(G_L, <, \mathbf{q})$, consider any sequence of student-rotations, $\rho_1, \rho_2, \cdots, \rho_k$, whose elimination from M gives the school-optimal legal assignment $M_z^{\mathcal{L}}$. The existence of such a sequence follows from Lemma 2.18. Let $M^i = M/\rho_1/\cdots/\rho_i$ for $i \in [k]$ and let $M^0 = M$. If $a = \emptyset$, by definition of student-rotations, we have $b \notin \rho_i$ for all $i \in [k]$. Now consider the case where $a \neq \emptyset$. Since $M^i \succeq M^j$ for all $i \leq j$, by Lemma 2.22, we have $s_{M^i}(a) = \emptyset$ and $next_{M^i}(a') = a$ for $i = 0, \ldots, k$. We again conclude $b \notin \rho_i$ for all $i \in [k]$. Thus, we deduce $M(b) = M_z^{\mathcal{L}}(b)$. Now assume by contradiction that $a'b \in M'$ for some legal assignment M'. First note that $M' \succeq M_z^{\mathcal{L}}$ because $M_z^{\mathcal{L}}$ is the school-optimal legal assignment and thus the school-optimal stable assignment in $(G_L, <, \mathbf{q})$ due to Theorem 2.3. Also note that M and M' do not block each other given that both are legal assignments. Moreover, since $M(a') >_{a'} b = M'(a')$ by stability of M, we can apply Lemma 2.13 (with $a = a'$, $\overline{b} = b$) and conclude that there exist $\overline{a} \in M(b)$ such that $b >_{\overline{a}} M'(\overline{a})$. However, $M' \succeq M_z^{\mathcal{L}}$ implies $M'(\overline{a}) \geq_{\overline{a}} M_z^{\mathcal{L}}(\overline{a})$ and $\overline{a} \in M(b)$ implies $M_z^{\mathcal{L}}(\overline{a}) = M(\overline{a}) = b$. Hence, $M'(\overline{a}) \geq_{\overline{a}} b$, which is a contradiction. ■

Hence, if the algorithm finds a sink fulfilling the properties of Lemma 2.21 in the school-rotation digraph, we are in case (i) above. If the school-rotation digraph has a directed cycle, eliminating the corresponding school-rotation from M brings us to case (ii)[8]. Lastly, if D_B has no arc, we conclude that we are in case (iii). The initial iteration starts with the set of identified illegal edges being empty, and M being the student-optimal stable assignment. The algorithm that finds the school-optimal legal assignment proceeds similarly, with a legal assignment M' that is dominated by M (i.e., $M \succ M'$) generated in case (ii).

[8]If D_B has both a sink and a directed cycle, the algorithm is free to choose between the two cases.

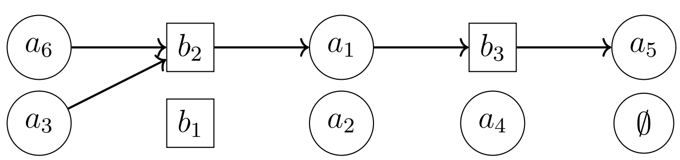
A formal description of our algorithm is given in Algorithm 2.1. Its correctness is shown in the proof of Theorem 2.24. We illustrate the algorithm in Example 2.23.

Example 2.23. We apply `student-rotate-remove` and `school-rotate-remove` to the following instance with 6 students and 3 schools, where each school has a quota of 2. In this and all following examples, when it is clear whose preference list we are referring to, the subscript in $>$ is dropped.

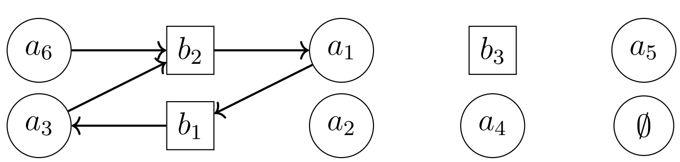
$$
\begin{array}{llll}
a_1: & \boxed{b_2} > b_3 > b_1 & \qquad b_1: & a_1 > \boxed{a_4} > \boxed{a_3} > a_5 > a_2 > a_6 \\
a_2: & b_1 > \boxed{b_2} > b_3 & \qquad b_2: & a_3 > \boxed{a_2} > a_6 > \boxed{a_1} > a_5 > a_4 \\
a_3: & b_3 > \boxed{b_1} > b_2 & \qquad b_3: & \boxed{a_6} > a_1 > \boxed{a_5} > a_2 > a_4 > a_3 \\
a_4: & \boxed{b_1} > b_2 > b_3 & & \\
a_5: & \boxed{b_3} > b_2 > b_1 & & \\
a_6: & b_1 > \boxed{b_3} > b_2 & &
\end{array}
$$

The student- and school-optimal stable assignments coincide, and are given by $\{a_1b_2,\ a_2b_2,\ a_3b_1,\ a_4b_1,\ a_5b_3,\ a_6b_3\}$ (squared entries above). This is the M^0 for both algorithms.

Student-Rotate-Remove. On a_1's preference list, b_3 is the first school after $M^0(a_1)$. In addition, b_3 prefers a_1 to a_5, who is b_3's least preferred student among $M^0(b_3)$. Thus, $s_{M^0}(a_1) = b_3$ and $next_{M^0}(a_1) = a_5$. After working out $s_{M^0}(\cdot)$ and $next_{M^0}(\cdot)$ of all the students, we have the rotation digraph D^0 for the first iteration of `student-rotate-remove`:

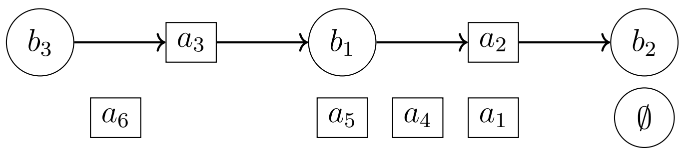


Here, we find a case (i) with $x' = a_1$, $y = b_3$, and $x = a_5$. So we set $M^1 = M^0$, remove $x'y = a_1b_3$ from the instance, and update the rotation digraph D^1 for the next iteration:

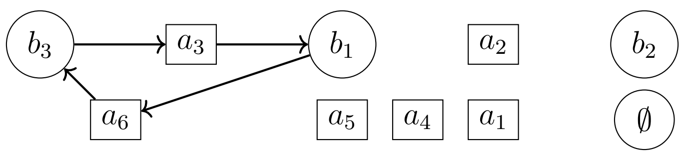


Now, we have a case (ii), with the corresponding student-rotation $\rho^1 = b_2, a_1, b_1, a_3$. Eliminating ρ^1 from M^1, we have $M^2 = M^1/\rho^1 = \{a_1b_1, a_2b_2, a_3b_2, a_4b_1, a_5b_3, a_6b_3\}$. In the next iteration, the rotation digraph D^2 only contains sinks. Thus, the algorithm terminates and output M^2 as the school-optimal legal assignment.

School-Rotate-Remove. The first student on b_1's preference list that prefers b_1 to his assigned school under M^0 is a_2. Thus, $s_{M^0}(b_1) = a_2$ and $next_{M^0}(b_1) = b_2$. After working out $s_{M^0}(\cdot)$ and $next_{M^0}(\cdot)$ of all the schools, we have the rotation digraph D^0 for the first iteration:



Here, we find a case (i) with $x' = b_1$, $y = a_2$ and $x = b_2$. So we set $M^1 = M^0$, remove $x'y = a_2b_1$ from the instance, and update the rotation digraph D^1 for the next iteration:



Now, we have a case (ii), with the corresponding school-rotation $\rho^1 = a_6, b_3, a_3, b_1$. Eliminating ρ^1 from M^1, we have $M^2 = M^1/\rho^1 = \{a_1b_2, a_2b_2, a_3b_3, a_4b_1, a_5b_3, a_6b_1\}$. In the next iteration, the rotation digraph D^2 only contains sinks. Thus, the algorithm terminates and output M^2 as the student-optimal legal assignment. $\triangle$

2.6.1 Correctness of Algorithm 2.1

We deduce the correctness of Algorithm 2.1 using the lattice structure of the legal assignments

Theorem 2.24. *Algorithm 2.1 finds the Y-optimal legal assignment.*

Proof. We focus on the statement with Y being the set of students, the other follows analogously. We first show, by induction on the iteration i of the algorithm, that $M^i \in \mathcal{S}(G^i, <, \mathbf{q})$ and $\mathcal{L}(G^i, <$

$, \mathbf{q}) = \mathcal{L}$. This is obvious for $i = 0$. Assume the claim is true for $i - 1 \geq 0$ and consider iteration i. If the condition at Step 6 is satisfied, $M^i = M^{i-1}$ is unchanged and the edge removed from G^{i-1} is illegal by Lemma 2.21. Hence, $M^i = M^{i-1} \in \mathcal{S}(G^{i-1}, <, \mathbf{q}) \subseteq \mathcal{S}(G^i, <, \mathbf{q})$ and $\mathcal{L}(G^i, <, \mathbf{q}) = \mathcal{L}(G^{i-1}, <, \mathbf{q}) = \mathcal{L}$ by induction and Lemma 2.6. If conversely the condition at Step 8 is satisfied, then ρ^{i-1} is a school-rotation exposed in M^{i-1}, and $M^i = M^{i-1}/\rho^{i-1} \in \mathcal{S}(G^{i-1}, <, \mathbf{q})$ by induction and Lemma 2.18. Moreover, since $G^i = G^{i-1}$, we have $\mathcal{S}(G^i, <, \mathbf{q}) = \mathcal{S}(G^{i-1}, <, \mathbf{q})$ and $\mathcal{L}(G^i, <, \mathbf{q}) = \mathcal{L}(G^{i-1}, <, \mathbf{q}) = \mathcal{L}$.

In order to conclude the proof, observe that at the end of the algorithm, the school-rotation digraph – call it D^* – only has sinks. We first claim that the assignment output – call it M^* – strictly dominates every assignment in $\mathcal{M}(G^*, \mathbf{q})$, where G^* is the graph at the end of the algorithm. Assume by contradiction that there is $M \in \mathcal{M}(G^*, \mathbf{q})$ and a student a such that $b := M(a) >_a M^*(a)$. Then $s_{M^*}(b)$ exists by definition, contradicting the fact that b is a sink in D^* (it is possible that $s_{M^*}(b) \neq a$, as there may be other nodes that precede a in b's list and have the required property, but it is a contradiction regardless). By what we proved above, we know that $\mathcal{L} = \mathcal{L}(G^*, <, \mathbf{q}) \subseteq \mathcal{M}(G^*, \mathbf{q})$. By Theorem 2.3 and Theorem 2.17, legal assignments form a lattice with respect to the partial order $\succeq$. Hence, M^* is the student-optimal legal assignment. ■

Note that the previous theorem in particular implies that the output of Algorithm 2.1 is unique, regardless of how we choose between Step 6 and Step 8 at each iteration, when multiple possibilities are present.

2.6.2 Time Complexity

A straightforward implementation of Algorithm 2.1 requires the construction of a rotation digraph at each iteration. However, this is computationally expensive. Instead of obtaining the complete rotation digraph at each iteration, we only *locally* build and update a directed path of the rotation digraph until a cycle or a sink is found. Together with suitable data structures, we can achieve the time complexity of $O(|E|)$.

Theorem 2.25. *Algorithm 2.1 can be implemented as to run in time $O(|E|)$.*

The full details of our implementation and the proof of Theorem 2.25 are included in Appendix A.2.1 and Appendix A.2.2.

2.7 An $O(|E|)$ Algorithm for Computing G_L

Throughout the section, we fix an instance $(G, <, \mathbf{q})$ with $G = (A \cup B, E)$ and abbreviate the set of stable assignments as $\mathcal{S} := \mathcal{S}(G, <, \mathbf{q})$. We start with a preliminary fact. Recall that we denote by $\mathcal{R}(G, <, \mathbf{q})$ and $\mathcal{SR}(G, <, \mathbf{q})$ the set of student-rotations and school-rotations exposed in some stable assignment of $(G, <, \mathbf{q})$, respectively. Let G_L be the subgraph of G that includes all and only edges in some legal assignments in $\mathcal{L}(G, <, \mathbf{q})$, as defined in Theorem 2.3.

Lemma 2.26. *Let e be an illegal edge of $(G, <, \mathbf{q})$ and let $\widetilde{G} = G[E \setminus \{e\}]$. Deleting edge e does not remove any element from either the set of student-rotations or the set of school-rotations: $\mathcal{R}(G, <, \mathbf{q}) \subseteq \mathcal{R}(\widetilde{G}, <, \mathbf{q})$ and $\mathcal{SR}(G, <, \mathbf{q}) \subseteq \mathcal{SR}(\widetilde{G}, <, \mathbf{q})$.*

Proof. Fix a stable assignment $M \in \mathcal{S}$. Since $\mathcal{S} \subseteq \mathcal{S}(\widetilde{G}, <, \mathbf{q})$, M is also a stable assignment of $(\widetilde{G}, <, \mathbf{q})$. First consider any student-rotation $\rho \in \mathcal{R}(G, <, \mathbf{q})$ exposed in M. We want to show that ρ is also exposed in M in $(\widetilde{G}, <, \mathbf{q})$. Assume $\rho = b_0, a_0, b_1, a_1, \cdots, b_{r-1}, a_{r-1}$. By Lemma 2.19, edges $a_i b_{i+1}$ and $a_{i+1} b_{i+1}$ for all $i = 0, 1, \cdots, r-1$, are stable and therefore legal. Hence, all such edges are in $E(\widetilde{G})$, implying that $b_{i+1} = s_M(a_i)$ and $next_M(a_i) = a_{i+1}$ hold in $(\widetilde{G}, <, \mathbf{q})$ as well. Thus, ρ is exposed in M in $(\widetilde{G}, <, \mathbf{q})$ and as desired. Therefore, $\rho \in \mathcal{R}(\widetilde{G}, <, \mathbf{q})$ and $\mathcal{R}(G, <, \mathbf{q}) \subseteq \mathcal{R}(\widetilde{G}, <, \mathbf{q})$. A similar argument shows $\mathcal{SR}(G, <, \mathbf{q}) \subseteq \mathcal{SR}(\widetilde{G}, <, \mathbf{q})$. ■

Theorem 2.27. *The subgraph G_L can be found in time $O(|E|)$.*

Proof. By Theorem 2.3 and Lemma 2.19, $E(G_L)$ is given by all and only edges in the student-optimal legal assignment $M_0^{\mathcal{L}}$, plus all pairs $a_i b_{i+1}$ for some student-rotation $\rho = b_0, a_0, \ldots, a_k \in \mathcal{R}(G_L, <, \mathbf{q})$. By Lemma 2.18, there exists exactly one set $\mathcal{R}_1$ of student-rotations whose elimination leads from $M_0^{\mathcal{L}}$ to the student-optimal stable assignment M_0; one set $\mathcal{R}_2$ leading from M_0 to the school-optimal stable assignment M_z; and one set $\mathcal{R}_3$ leading from M_z to the school-optimal legal assignment $M_z^{\mathcal{L}}$. By Lemma 2.18, $\mathcal{R}_1 \cup \mathcal{R}_2 \cup \mathcal{R}_3$ is the set of all rotations $\mathcal{R}(G_L, <, \mathbf{q})$ of

$(G_L, <, \mathbf{q})$. We argue that $\mathcal{R}_3$ is computed during the execution of `student-rotate-remove`. Indeed, throughout the algorithm, a sequence of rotations is found and eliminated, leading from M_z to $M_z^{\mathcal{L}}$. Each of these is exposed in some stable assignment in an instance that contains all legal edges. Hence, by repeated application of Lemma 2.26, those rotations form set $\mathcal{R}_3$. They can be computed in time $O(|E|)$ by Theorem 2.25. By Lemma 2.18 and repeated applications of Lemma 2.26, $\mathcal{R}_2$ coincides with the set $\mathcal{R}(G, <, \mathbf{q})$, which can be computed in time $O(|E|)$ by classical algorithms, see, e.g., Gusfield and Irving (1989). `school-rotate-remove` computes in time $O(|E|)$, again by Theorem 2.25, the set of school-rotations $\mathcal{SR}_1$ whose sequential elimination starting from M_0 leads to $M_0^{\mathcal{L}}$. By Lemma 2.20, the set $\mathcal{R}_1$ can be obtained from $\mathcal{SR}_1$ via the bijection σ. Consider a student-rotation $\rho = b_0, a_0, b_1, a_1, \cdots, a_{r-1} \in \mathcal{R}_1$. Since its corresponding school-rotation $\sigma(\rho) \in \mathcal{SR}_1$ can be obtained simply as $a_0, b_1, a_1, \cdots, a_{r-1}, b_0$, computing $\mathcal{R}_1$ from $\mathcal{SR}_1$ takes time $O(|E|)$. This concludes the proof. ■

2.8 An $O(|E|)$ Algorithm for EADAM with Consent

In this section, we first formally introduce EADAM with consent (Kesten, 2010). Then in Section 2.8.2 we show that a fast implementation of EADAM can be achieved by a suitable modification of our `school-rotate-remove` algorithm. The proof relies on a simplified and outcome-equivalent version of EADAM introduced by Tang and Yu (2014). Thus, we defer the proof as well as a formal introduction of simplified EADAM to Section 2.9. Together with Theorem 2.25, this implies the following.

Theorem 2.28. *EADAM with consent on a stable assignment instance $(G(A \cup B, E), <, \mathbf{q})$ can be implemented as to run in time $O(|E|)$.*

We also compare our algorithm with previous versions of EADAM through computational experiments. In Section 2.8.4, the theoretical advantage of `student-rotate-remove` is verified computationally on random instances.

Algorithm 2.2 Kesten's EADAM

Input: $(G(A \cup B, E), <, \mathbf{q})$, consenting students $\overline{A} \subseteq A$

1: Let $G^0 = G$, $i = 0$.
2: Run student-proposing Gale-Shapley's algorithm on $(G^i, <, \mathbf{q})$ to obtain assignment M^i.
3: **while** there is a consenting interrupter **do**
4: Identify the maximum k' such that there exists a consenting interrupter at step k'.
5: Let E' be the set of all interrupting pairs ab at step k' such that a is consenting.
6: Define G^{i+1} from G^i by removing edges in E'. Set $i = i + 1$.
7: Run student-proposing Gale-Shapley's algorithm on $(G^i, <, \mathbf{q})$ to obtain assignment M^i.
8: **end while**

Output: M^i

2.8.1 Kesten's EADAM

Recall that Gale-Shapley's algorithm (with students proposing) is executed in successive steps. During each step, every student that is currently unmatched applies to the first school in his preference list that he has not yet applied to, and gets either temporarily accepted or rejected. A student a is called an *interrupter* (*for school* b, *at step* k') if: a is temporarily accepted by school b at some step $k < k'$; a is rejected by school b at step k'; and there exists a student that is rejected by school b during steps $k, k+1, \cdots, k'-1$. In such case, we will also call ab an *interrupting pair* (*at step* k'). Informally speaking, an interrupter is a student who, by applying to school b, interrupts a desirable assignment between school b and another student at no gain to himself. Removing such interruptions is crucial in neutralizing their adverse effects on the outcome. Demonstration of these concepts can be found in Example 2.30.

Kesten's EADAM takes as input an instance $(G, <, \mathbf{q})$ with $G = (A \cup B, E)$ and a set $\overline{A} \subseteq A$ of students which we call *consenting*. Each iteration of EADAM starts by running Gale-Shapley's algorithm from scratch. It then removes from the graph certain interrupting pairs involving consenting interrupters. The algorithm terminates when there are no interrupting pairs whose corresponding interrupters are consenting students.

Details of Kesten's algorithm can be found in Algorithm 2.2 and an illustration of the algorithm can be found later in Example 2.30.

The following theorem collects some results from Kesten (2010) and Tang and Yu (2014), demonstrating the transparency of the consenting incentives and some attractive properties of EADAM's output. Recall that an assignment M is *constrained efficient* if it does not violate any nonconsenting students' priorities[9], but any other assignment M' that dominated M does.

Theorem 2.29. *Under Kesten's EADAM:*

1. *The assignment of a student does not change whether he consents or not. That is, for any student $a \in A$ and any set of consenting students $\bar{A} \subseteq A$, if M and M' are the outputs of EADAM on inputs $\{(G, <, \mathbf{q}), \overline{A} \setminus \{a\}\}$ and $\{(G, <, \mathbf{q}), \overline{A}\}$ respectively, then $M(a) = M'(a)$.*

2. *The output is Pareto-efficient when all students consent and is constrained efficient otherwise.*

Example 2.30. Each school in this example has a quota of 1. Their preference lists are given below. All students are consenting except for a_3.

$$
\begin{array}{llll}
a_1: & b_1 > b_2 > b_3 > b_4 & b_1: & a_4 > a_2 > a_1 > a_3 \\
a_2: & b_1 > b_2 > b_3 > b_4 & b_2: & a_2 > a_3 > a_1 > a_4 \\
a_3: & b_3 > b_2 > b_4 > b_1 & b_3: & a_1 > a_4 > a_3 > a_2 \\
a_4: & b_3 > b_1 > b_2 > b_4 & b_4: & a_3 > a_1 > a_2 > a_4
\end{array}
$$

Gale-Shapley's algorithm: The student-proposing Gale-Shapley's algorithm outputs the assign-

[9] A student a's priority is violated at assignment M if there is a school b such that ab is a blocking pair of M.

ment $M^0 = \{a_1b_3, a_2b_2, a_3b_4, a_4b_1\}$. Steps of the algorithm are given below:

step	b_1	b_2	b_3	b_4
1	$\cancel{a_1}, \mathbf{a_2}$		$\cancel{a_3}, \mathbf{a_4}$	
2		$\cancel{a_1}, \mathbf{a_3}$		
3			$a_1, \cancel{\mathbf{a_4}}$	
4	$\cancel{\mathbf{a_2}}, a_4$			
5		$a_2, \cancel{\mathbf{a_3}}$		
6				a_3

Iteration #1: From the steps of Gale-Shapley's algorithm, one can identify all interrupting pairs. For instance, a_2 proposes to b_1 at step 1. This causes a_1 to be rejected by b_1. However, a_2 is later rejected by b_1 at step 4. Thus, by definition, a_2b_1 is an interrupting pair at step 4.

In total, there are three interrupting pairs, a_3b_2, a_2b_1, a_4b_3, from the last step to the first. The last interrupting pair of a consenting interrupter is a_2b_1, given that a_3 is not a consenting student. Thus, $k' = 4$. Since there is only one interrupting pair at step $k' = 4$, EADAM simply removes a_2b_1 from the instance. On the new instance, EADAM re-runs Gale-Shapley's algorithm, and the resulting assignment is $M^1 = \{a_1b_1, a_2b_2, a_3b_4, a_4b_3\}$.

Iteration #2: One can check that there are no interrupting pairs, thus no consenting interrupters. Hence, EADAM terminates and outputs assignment M^1.

Note that using tools developed in previous sections, one can show that a_2b_1, the first edge that is removed by EADAM, is actually a legal edge and the assignment output of EADAM, M^1, is not a legal assignment. $\triangle$

2.8.2 `School-Rotate-Remove with Consent`

Morrill (2016) showed that when all students consent, the output of EADAM is the student-optimal legal assignment. Hence, `school-rotate-remove` can be employed to find this assignment in time $O(|E|)$ (see, Theorem 2.25). However, as Example 2.30 shows, when only some stu-

Algorithm 2.3 `school-rotate-remove with consent`

Input: $(G(A \cup B, E), <, \mathbf{q})$, consenting students $\overline{A} \subseteq A$

Find the student-optimal stable assignment M_0 of $(G, <, \mathbf{q})$ via Gale-Shapley's algorithm.
Let $G^0 := G$ and $M^0 := M_0$.
Set $i = 0$ and let D^0 to be the school-rotation digraph of M^0 in $(G^0, <, \mathbf{q})$.
while D^i still has an arc **do**
    Find (i) arcs (b', a) and $(a, b) \in A(D^i)$ where b is a sink in D^i, or (ii) a cycle C^i of D^i.
    if (i) is found **then**
        Define G^{i+1} from G^i by removing ab', and set $M^{i+1} = M^i$.
        if $a \notin \overline{A}$ **then**
            Remove from G^{i+1} edges $a'b'$ for all a' such that $a >_{b'} a'$.
        end if
    else if (ii) is found **then**
        Let ρ^i be the corresponding school-rotation. Set $M^{i+1} = M^i/\rho^i$, and $G^{i+1} = G^i$.
    end if
    Set $i = i + 1$ and let D^i to be the school-rotation digraph of M^i in $(G^i, <, \mathbf{q})$.
end while

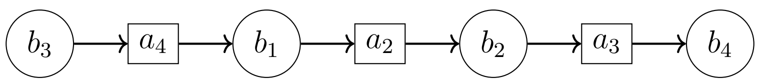
Output: M^i

dents consent, EADAM may output an assignment that is not legal. We show in this section how to suitably modify `school-rotate-remove` in order to obtain the assignment output of EADAM for any given set of consenting students, without sacrificing the running time.

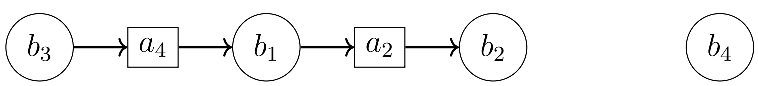
In `school-rotate-remove`, the key idea is to reroute arcs that point to students who are assigned to sinks in the rotation digraph. This allows us to identify school-rotations in the underlying legalized instance $(G_L, <, \mathbf{q})$. Assume for example that $(b', a), (a, b) \in A(D_B)$, and b is a sink. Upon such rerouting, a's priority might be violated. In particular, if b' successfully participates in a school-rotation after the rerouting, then ab' will be a blocking pair for the new assignment. Hence, under the EADAM framework, if a is not consenting, we can no longer freely reroute arcs pointing to a. In fact, in order to respect a's priority (i.e., to avoid ab' becoming a blocking pair), b' cannot be assigned to any student a' such that $a >_{b'} a'$. This means that the arc coming out of b' cannot be rerouted to any other student, essentially marking b' a sink.

A detailed description of our algorithm is presented in Algorithm 2.3. Throughout the rest of the section, we call school-rotations simply rotations. As in Algorithm 2.1, when both cases (i) and (ii) are present at Step 5 of some iteration, we are free to choose between Step 6 and Step 11. These choices do not affect the final assignment output. We formalize this statement in Theorem 2.40. A step-by-step application of our algorithm on the instance from Example 2.30 is outlined in Example 2.31.

Example 2.31. Consider the instance given in Example 2.30. From the student-optimal stable assignment $M^0 := \{a_1b_3, a_2b_2, a_3b_4, a_4b_1\}$, we can construct the rotation digraph as below. Note that in this graph and in the following, some isolated nodes are not included.



Iteration #1: Since b_4 is a sink, we remove edge a_3b_2 as in Step 7, in the hope of rerouting the arc coming out of b_2. However, because a_3 is not consenting, we have to additionally remove edges a_1b_2 and a_4b_2 as in Step 9. This completely removes the possibilities of rerouting, essentially making b_2 a sink, as seen in the rotation digraph of the updated instance:



Iteration #2: Now, b_2 is a sink. Since its assigned student a_2 is consenting, the algorithm simply removes edge a_2b_1 in Step 7, resulting in the following updated rotation digraph:

Iteration #3: We can now eliminate the rotation (i.e., trading schools between a_1 and a_4), and update the assignment to be $\{a_1b_1, a_2b_2, a_3b_4, a_4b_3\}$. After the assignment update, the new rotation digraph only contains sinks, and thus the algorithm terminates.

This final assignment coincides with the assignment output of EADAM. △

In our rotation-based algorithm, the students from whom we seek consent are those who are assigned to schools corresponding to sinks, and thus they are not in any directed cycles in the current and subsequent rotation digraphs. Therefore, there is a clear separation between the students from whom we ask for consent and those participating in Pareto-improvement cycles (i.e., school-rotations). This is consistent with the result in Theorem 2.29, part 1 that students have no incentive to not consent.

Theorem 2.32. *For any given input, the outputs of Algorithm 2.3 and Algorithm 2.2 coincide.*

We defer the proof of Theorem 2.32 to Section 2.9, and we remark here that the proof of Theorem 2.32 is different (and quite harder) than the proof of Theorem 2.24. Indeed, for the latter, we can build on the fact that legal assignments form a lattice, while in the former we do not have such a well-behaved structural result at our disposal. Hence, a careful analysis of the algorithms is needed.

2.8.3 Fast Implementation of `School-Rotate-Remove with Consent`

The fast implementation is a modification of that of Algorithm 2.1. Therefore, we defer the proof of Lemma 2.33 to Appendix A.2.4. An example demonstrating the implementation can also be found in Appendix A.2.3.

Lemma 2.33. *Algorithm 2.3 can be implemented as to run in time $O(|E|)$.*

Proof of Theorem 2.28. It follows immediately from Theorem 2.32 and Lemma 2.33. ■

2.8.4 Computational Experiments

Since Gale-Shapley's algorithm on stable assignment instances can be implemented to run in time $O(|E|)$ (see Manlove, 2013; Gusfield and Irving, 1989), the original EADAM (Kesten, 2010) runs in time $O(|E|^2)$ because it runs Gale-Shapley's routine at most $|E|$ times. A simplified version of EADAM (Tang and Yu, 2014), for which the details are presented in the appendix, runs in time $O(|E||V|)$ because it runs Gale-Shapley's routine at most $|V|$ times. We remark that although mechanism design, rather than computational complexity, is the primary interest of Kesten's paper, computational efficiency is nevertheless crucial in putting the mechanism into practice, especially for large markets such as the New York school system. In fact, Tang and Yu (2014) mentioned computational tractability as one of their contributions.

One major advantage of our `school-rotate-remove with consent` is that instead of repeatedly running Gale-Shapley's algorithm, we update the assignment locally using the structural results (lattice structure and rotations) of stable assignments. Our algorithm runs in time $O(|E|)$ as shown in Lemma 2.33.

To further demonstrate the computational advantage of our algorithm, we randomly generated instances of varying sizes, and recorded the running time of all three algorithms. The running time of Gale-Shapley's algorithm is also recorded as a benchmark. The number of students in our instances ranges from 500 to $30,000$, and the corresponding number of schools ranges from 5 to 300. For each instance size, 100 instances $(G, <, \mathbf{q})$ are obtained by randomly generating $<$ and $\mathbf{q}$. For each student a, the preference list $<_a$ is defined by a random permutation of B. The preference lists of schools are similarly defined. The quota of each school is randomly selected between 50 and 150 uniformly. Note that in this set of simulations, students and schools have complete preference ranking of the opposite side. That is, in all instances, G is a complete bipartite graph. We also conduct another set of simulations (details later) with incomplete preference lists. We tested scenarios where each student is randomly determined to be consenting with probability 10%, 30%, 50%, 80%, and 100%. The experiments were carried out on a computing node with 1 core and 4GB RAM.

A visual representation of the running times of different algorithms can be found in Figure 2.1. The shaded areas are 95% confidence intervals of each algorithm for given instance sizes. Our algorithm performs significantly faster than the simplified EADAM and dramatically faster than the original EADAM, with the differences being especially pronounced when all students consent.

The New York City school district has approximately $90,000$ students applying to 700 public high school programs every year, where students can list up to 12 schools in their application (Narita, 2016). We further conducted computational experiments whose instance sizes are similar to those of New York City. We compared our algorithm with simplified EADAM on random instances generated similarly as previously described. However, in this set of simulations, we fix instance size with $|A| = 90,000$ and $|B| = 700$. Moreover, the quota of each school is selected uniformly at random from integers between $\lceil 0.5 \times \mu \rceil$ and $\lceil 1.5 \times \mu \rceil$ where $\mu = \lceil \frac{|A|}{|B|} \rceil$. In generating $<$, for every student a, $<_a$ is obtained by truncating the random permutation such that only the top 12 schools are listed; for every school b, $<_b$ is obtained by restricting the random permutation to students who have b in their preference lists. Graph G can be deduced from the preference lists. Results of our experiments are summarized in Figure 2.2. The difference in computational time is noticeably different from all levels of consenting percentages. In particular, when all students consent, `school-rotate-remove` takes approximately 3 minutes, whereas simplified EADAM takes on average 4 hours and its run time has a much higher variance.

2.9 Proof of Outcome-Equivalence

The goal of this section is to prove Theorem 2.32. The proof consists of three steps: first, in Section 2.9.1, we show that all executions of Algorithm 2.3 give the same output; then, in Section 2.9.2, we introduce an outcome-equivalent version of EADAM, called *Simplified EADAM*, from Tang and Yu (2014); and lastly, in section 2.9.3, we show that the output of Algorithm 2.3 and that of the Simplified EADAM coincide.

2.9.1 Uniqueness of the Output of Algorithm 2.3

From now on, fix the input $(G(A \cup B, E), <, \mathbf{q}), \overline{A})$ to Algorithm 2.3. An *execution* of Algorithm 2.3 on the input is an ordered collection of *iterations*, where iteration i denotes the i-th repetition of the *while* loop from Step 4. Hence, in iteration i, Algorithm 2.3 takes from the previous iteration graph G^{i-1}, assignment M^{i-1}, and rotation digraph D^{i-1} and creates G^i, M^i, and D^i. For each iteration i, we let I^i be the cycle found in D^{i-1} or the pair of arcs $(b', a), (a, b)$ with b being a sink found in D^{i-1} (depending on whether the **if** condition at Step 6 is satisfied). We identify an execution $\mathcal{E}$ of Algorithm 2.3 by the sequence $\mathcal{E} = (I^1, I^2, \ldots)$. Note that in particular, for each iteration i, I^i is a subgraph of D^{i-1}. Let $G^i_{\mathcal{E}}$, $M^i_{\mathcal{E}}$, and $D^i_{\mathcal{E}}$ denote G^i, M^i, and D^i under execution $\mathcal{E}$. The collection of all possible executions is denoted by $\mathbb{E}$. We start with several useful observations.

Lemma 2.34. *Let $\mathcal{E} \in \mathbb{E}$. $\mathcal{E}$ contains a finite number k of iterations. Moreover, for every $i \in [k]$, we have $M^i_{\mathcal{E}} \succeq M^{i-1}_{\mathcal{E}}$.*

Proof. At each iteration i, either case (i) or case (ii) is found. For case (i), we have $M^i_{\mathcal{E}} = M^{i-1}_{\mathcal{E}}$ and some edges are removed from $G^{i-1}_{\mathcal{E}}$. For case (ii), a school-rotation exposed in $M^{i-1}_{\mathcal{E}}$ is eliminated and thus, we have $M^i_{\mathcal{E}} \succ M^{i-1}_{\mathcal{E}}$. This proves immediately the second thesis. The first thesis follows from the fact that the number of edges and the number of assignments of the instance are both finite. ■

The definition of I^i and Lemma 2.34 implies that for every $\mathcal{E} = (I^1, I^2, \ldots, I^k) \in \mathbb{E}$ and $i, j \in [k]$, if $i \neq j$, then $I^i \neq I^j$.

Lemma 2.35. *Let $\mathcal{E}_1 = (I^1_1, I^2_1, \cdots, I^{k_1}_1)$ and $\mathcal{E}_2 = (I^1_2, I^2_2, \cdots, I^{k_2}_2)$ be two executions such that $\{I^1_1, I^2_1, \cdots, I^k_1\} = \{I^1_2, I^2_2, \cdots, I^k_2\}$ for some $k \leq \min(k_1, k_2)$. Then, $M^k_{\mathcal{E}_1} = M^k_{\mathcal{E}_2}$ and $G^k_{\mathcal{E}_1} = G^k_{\mathcal{E}_2}$.*

Proof. Note that both executions start from the same assignment M_0. Let $\rho_1, \rho_2, \cdots, \rho_\ell$ be all the rotations eliminated in the first k iterations of execution $\mathcal{E}_1$ and let $\rho'_1, \rho'_2, \cdots, \rho'_{\ell'}$ be those of execution $\mathcal{E}_2$. Then, since the first k iterations of these two executions coincide, we have $\{\rho_1, \rho_2, \cdots, \rho_\ell\} = \{\rho'_1, \rho'_2, \cdots, \rho'_{\ell'}\}$ and thus, matching $M^k_{\mathcal{E}_1} = M_0/\rho_1/\rho_2/\cdots/\rho_\ell$ and matching $M^k_{\mathcal{E}_2} = M_0/\rho'_1/\rho'_2/\cdots/\rho'_{\ell'}$ coincide. Similarly, let $\widetilde{E}_1$ and $\widetilde{E}_2$ be the set of edges removed in the first k iterations of $\mathcal{E}_1$ and $\mathcal{E}_2$ respectively. Again, because $\{I^1_1, I^2_1, \cdots, I^k_1\} = \{I^1_2, I^2_2, \cdots, I^k_2\}$, we have $\widetilde{E}_1 = \widetilde{E}_2$ and thus, $G^k_{\mathcal{E}_1} = G^k_{\mathcal{E}_2}$. ■

Lemma 2.36. *Let $\mathcal{E} \in \mathbb{E}$ and assume $\mathcal{E} = (I^1, I^2, \cdots, I^k)$. Assume at some iteration $j < k$, rotation digraph $D^{j-1}_{\mathcal{E}}$ contains a cycle I' and $I' \neq I^j$. Then, I' must also be a subgraph of $D^j_{\mathcal{E}}$. Moreover, there exists a unique $i \in \{j+1, \cdots, k\}$ such that $I' = I^i$.*

Proof. Since every vertex in the rotation digraph $D_{\mathcal{E}}^{j-1}$ has outdegree at most 1, I' and I^j must be vertex-disjoint. Thus, for every vertex $x \in I'$, we have $M_{\mathcal{E}}^{j-1}(x) = M_{\mathcal{E}}^{j}(x)$ and for all $b \in I' \cap B$, we have $\{a : ab \in E(G_{\mathcal{E}}^{j-1})\} = \{a : ab \in E(G_{\mathcal{E}}^{j})\}$. Since $M_{\mathcal{E}}^{j} \succeq M_{\mathcal{E}}^{j-1}$ by Lemma 2.34, we can conclude that $s_{M_{\mathcal{E}}^{j-1}}(b) = s_{M_{\mathcal{E}}^{j}}(b)$ for all $b \in I' \cap B$. Hence, I' is a subgraph of $D_{\mathcal{E}}^{j}$. The second thesis follows from repeated application of the first thesis and the termination criterion of Algorithm 2.3, and uniqueness holds because for $\ell_1, \ell_2 \in [k]$, if $\ell_1 \neq \ell_2$, then $I^{\ell_1} \neq I^{\ell_2}$ as implied by Lemma 2.34. ■

Lemma 2.37. *Let $\mathcal{E} \in \mathbb{E}$ and assume $\mathcal{E} = (I^1, I^2, \cdots, I^k)$. Assume at some iteration $j < k$, rotation digraph $D_{\mathcal{E}}^{j-1}$ contains a pair of arcs $I' = (b', a), (a, b)$ with b being a sink of $D_{\mathcal{E}}^{j-1}$ and $I' \neq I^j$. Then, I' must also be a subgraph of $D_{\mathcal{E}}^{j}$ with b being a sink in $D_{\mathcal{E}}^{j}$. Moreover, there must exists a unique $i \in \{j+1, \cdots, k\}$ such that $I' = I^i$.*

Proof. Since every vertex in the rotation digraph $D_{\mathcal{E}}^{j-1}$ has outdegree at most 1, we must have $b' \notin I^j$. Note that if I^j is a directed cycle, then I^j and I' are vertex-disjoint, otherwise, it is possible to have $(a, b) \in I^j$. Nevertheless, we have that for $x \in \{b', b\}$, $M_{\mathcal{E}}^{j-1}(x) = M_{\mathcal{E}}^{j}(x)$ and $\{a : ax \in E(G_{\mathcal{E}}^{j-1})\} = \{a : ax \in E(G_{\mathcal{E}}^{j})\}$. Since $M_{\mathcal{E}}^{j} \succeq M_{\mathcal{E}}^{j-1}$ by Lemma 2.34, we can conclude that $s_{M_{\mathcal{E}}^{j-1}}(b') = s_{M_{\mathcal{E}}^{j}}(b')$ and thus, I' is a subgraph of $D_{\mathcal{E}}^{j}$ and b is a sink in $D_{\mathcal{E}}^{j}$. The second thesis follows as in the proof of Lemma 2.36. ■

Lemma 2.38. *Let $\mathcal{E} = (I^1, I^2, \cdots, I^k) \in \mathbb{E}$ and assume for some iteration $j \in \{2, 3, \cdots, k\}$, I^j is a subgraph of $D_{\mathcal{E}}^{j-2}$ and if $I^j = (b', a), (a, b)$, we also have b being a sink of $D_{\mathcal{E}}^{j-2}$. Then, $\mathcal{E}' := (I^1, I^2, \cdots, I^{j-2}, I^j, I^{j-1}, I^{j+1} \cdots, I^k) \in \mathbb{E}$.*

Proof. Let $\bar{M}^i$, $\bar{G}^i$, and $\bar{D}^i$ be the assignment, graph, and rotation digraph after the first i iterations of $\mathcal{E}'$. For $i \leq j-2$, they are well defined and $\bar{M}^i = M_{\mathcal{E}}^{i}$, since the first $j-2$ iterations of $\mathcal{E}$ and $\mathcal{E}'$ are exactly the same. Thus, I^j is either a case (i) at Step 6 or a case (ii) at Step 11 of $\bar{D}^{j-2} = D_{\mathcal{E}}^{j-2}$. Therefore, $\bar{M}^i$, $\bar{G}^i$, and $\bar{D}^i$ are also well-defined for $i = j-1$. Now, because of Lemma 2.36 and Lemma 2.37 with $I' = I^{j-1}$, they are also well-defined for $i = j$. Lastly, since the first j iterations of $\mathcal{E}$ and $\mathcal{E}'$ coincide, we know $\bar{M}^j = M_{\mathcal{E}}^{j}$ and $\bar{G}^j = G_{\mathcal{E}}^{j}$ due to Lemma 2.35. Since all iterations after the j^{th} iteration are exactly the same and in the same order in $\mathcal{E}$ and $\mathcal{E}'$, we also know that for all $i > j$, $\bar{M}^i = M^i_{\mathcal{E}}$ and $\bar{G}^i = G^i_{\mathcal{E}}$. This concludes the proof. ■

Corollary 2.39. *Let $\mathcal{E} = (I^1, I^2, \cdots, I^k) \in \mathbb{E}$ and assume that for some iterations $j_1, j_2 \in [k]$ with $j_1 < j_2$, I^{j_2} is a subgraph of $D^{j_1-1}_{\mathcal{E}}$, and if $I^{j_2} = (b', a), (a, b)$, we also have b being a sink of $D^{j_1-1}_{\mathcal{E}}$. Then, $\mathcal{E}' := (I^1, I^2, \cdots, I^{j_1-1}, I^{j_2}, I^{j_1}, \cdots, I^{j_2-1}, I^{j_2+1} \cdots, I^k) \in \mathbb{E}$.*

Proof. Note that $I^{j_2} \neq I^i$ for all $i \neq j_2$. Because of Lemma 2.36 and Lemma 2.37, I^{j_2} is a subgraph of $D^{j_2-2}_{\mathcal{E}}$and if $I^{j_2} = (b', a), (a, b)$, we also have b being a sink of $D^{j_2-2}_{\mathcal{E}}$. Thus, due to Lemma 2.38, $\mathcal{E}_1 = (I^1, \cdots, I^{j_2-2}, I^{j_2}, I^{j_2-1}, I^{j_2+1}, \cdots, I^k) \in \mathbb{E}$. Repeatedly applying the argument and moving the iteration I^{j_2} to earlier steps, we can conclude that $\mathcal{E}' \in \mathbb{E}$. ■

For two executions $\mathcal{E}_1 = (I^1_1, I^2_1, \cdots, I^{k_1}_1)$ and $\mathcal{E}_2 = (I^1_2, I^2_2, \cdots, I^{k_2}_2)$, let $C(\mathcal{E}_1, \mathcal{E}_2)$ denote the largest $i \leq \min(k_1, k_2)$ such that $(I^1_1, I^2_1, \cdots, I^i_1) = (I^1_2, I^2_2, \cdots, I^i_2)$. We can now prove the following theorem.

Theorem 2.40. *The output of Algorithm 2.3 is unique.*

Proof. Assume by contradiction that there are two executions $\mathcal{E}_1 = (I^1_1, I^2_1, \cdots, I^{k_1}_1) \in \mathbb{E}$ and $\mathcal{E}_2 = (I^1_2, I^2_2, \cdots, I^{k_2}_2) \in \mathbb{E}$ such that $M^{k_1}_{\mathcal{E}_1} \neq M^{k_2}_{\mathcal{E}_2}$. Also assume that among all executions that output distinct assignments, $\mathcal{E}_1$ and $\mathcal{E}_2$ are the ones with the largest value $C(\mathcal{E}_1, \mathcal{E}_2)$. Let $j := C(\mathcal{E}_1, \mathcal{E}_2) + 1$. That is, we assume $I^i_1 = I^i_2$ for all $i \in [j-1]$, but $I^j_1 \neq I^j_2$.

By construction, we know $G^{j-1}_{\mathcal{E}_1} = G^{j-1}_{\mathcal{E}_2}$ and $M^{j-1}_{\mathcal{E}_1} = M^{j-1}_{\mathcal{E}_2}$. Thus, I^j_2 is also a subgraph of $D^{j-1}_{\mathcal{E}_1}$ and if $I^j_2 = (b', a), (a, b)$, we also have b being a sink of $D^{j-1}_{\mathcal{E}_1}$. Thus, due to Lemma 2.36 and Lemma 2.37, there must exist a unique $\ell > j$ such that $I^j_2 = I^\ell_1$. Therefore, we can apply Corollary 2.39 on $\mathcal{E}_1$ with $j_2 = \ell$ and $j_1 = j$ and conclude that

$$\mathcal{E}'_1 := (I^1_1, I^2_1, \cdots, I^{j-1}_1, I^\ell_1, I^j_1, \cdots, I^{\ell-1}_1, I^{\ell+1}_1 \cdots, I^{k_1}_1) \in \mathbb{E}.$$

Because of Lemma 2.35, we know $M^{k_1}_{\mathcal{E}_1} = M^{k_1}_{\mathcal{E}'_1}$ and thus, $M^{k_1}_{\mathcal{E}'_1} \neq M^{k_2}_{\mathcal{E}_2}$. However, $C(\mathcal{E}'_1, \mathcal{E}_2) = j > C(\mathcal{E}_1, \mathcal{E}_2)$, which is a contradiction to the choice of $\mathcal{E}_1$ and $\mathcal{E}_2$. ■

Algorithm 2.4 simplified EADAM

Input: $(G(A \cup B, E), <, \mathbf{q})$, consenting students $\overline{A} \subseteq A$

Let $G^0 = G$ and $i = 0$.
repeat
    Run student-proposing Gale-Shapley's algorithm on $(G^i, <, \mathbf{q})$ to obtain assignment M^i.
    Identify underdemanded schools B^i in M^i and their assigned students $A^i := \cup_{b \in B^i} M^i(b)$.
    Set $G^{i+1} = G^i$.
    for $a \in A^i$ **do**
        for $b \in B$ such that $ab \in E(G^{i+1})$ and $b >_a M^i(a)$ **do**
            remove edge ab from G^{i+1}.
            if $a \notin \overline{A}$ **then**
                remove edges $a'b$ from G^{i+1} for all $a' \in A$ such that $a >_b a'$.
            end if
        end for
    end for
    Set $i = i + 1$.
until $B^{i-1} = B$

Output: M^{i-1}

2.9.2 Simplified EADAM

In this section, we introduce a simplified and outcome-equivalent version of EADAM by Tang and Yu (2014). The key concept exploited by Tang and Yu (2014) is that of *underdemanded* schools at an assignment M. A school $b \in B$ is underdemanded in M if there is no student a that strictly prefers b to $M(a)$. Tang and Yu (2014) observe that if a student a is assigned to an underdemanded school at the student-optimal stable assignment M_0, then a is not *Pareto-improvable*. That is, if an assignment M' dominates M_0, it must be that $M_0(a) = M'(a)$. With this key observation, they develop the simplified algorithm and show that it is output-equivalent to Kesten's original algorithm. A formal description of their algorithm is presented in Algorithm 2.4 and an example is given in Example 2.42 for the same instance from Example 2.30.

Simplified EADAM takes as input an instance and a list of *consenting* students, and similarly to Kesten's original algorithm, it iteratively re-runs Gale-Shapley's procedure. In each iteration, it identifies underdemanded schools and fixes their assignments via deletion of edges (see Line 8 of Algorithm 2.4). If a non-consenting student is matched to an underdemanded school, more edges are removed from the instance in order to respect his priorities (see Line 10 of Algorithm 2.4). The following theorem is proved in Tang and Yu (2014).

Theorem 2.41. *For any given input, the outputs of Algorithm 2.2 and Algorithm 2.4 coincide.*

Note that the running time of Algorithm 2.4 is $O(|V||E|)$ because it runs Gale-Shapley's routine at most $|V|$ times.

Example 2.42. Consider the instance given in Example 2.30.

Iteration #1: The iteration starts with the student-proposing Gale-Shapley's algorithm, whose steps can be found in Example 2.30. Since b_4 never rejects any students in the execution, no student strictly prefers b_4 to his current assignment, and thus, b_4 is an underdemanded school in M^0. In fact, b_4 is the only underdemanded school. Hence, $B^0 = \{b_4\}$ and $A^0 = \{a_3\}$. Simplified EADAM then settles assignment a_3b_4 by removing edges a_3b_3 and a_3b_2 from the instance, as in Step 8 of Algorithm 2.4. In addition, since a_3 is not consenting, edges b_2a_1, b_2a_4, and b_3a_2 are removed to respect his priority at school b_3 and b_2, as in Step 10 of the algorithm.

Iteration #2: Re-running Gale-Shapley's algorithm on the updated instance:

step	b_1	b_2	b_3	b_4
1	$\cancel{a_1}, a_2$		a_4	a_3
2			$a_1, \cancel{a_4}$	
3	$\cancel{a_2}, a_4$			
4		a_2		

The resulting assignment is $M^1 = \{a_1b_3, a_2b_2, a_3b_4, a_4b_1\}$. b_2 is an additional underdemanded school in M^1 and its assigned student a_2 is consenting. So simplified EADAM simply fixes assignment a_2b_2 by removing edge a_2b_1 from the instance.

Iteration #3: The algorithm then runs Gale-Shapley's algorithm again on the updated instance:

step	b_1	b_2	b_3	b_4
1	a_1	a_2	a_4	a_3

The resulting assignment is $M^2 = \{a_1b_1,\ a_2b_2,\ a_3b_4,\ a_4b_3\}$. Now, all schools are underdemanded. Hence, the algorithm terminates and outputs assignment M^2, which is equivalently to the assignment output of Kesten's EADAM that we obtained in Example 2.30. $\diamondsuit$ $\triangle$

2.9.3 Equivalence between `School-Rotate-Remove with Consent` and Simplified EADAM

The goal of this section is to show that our Algorithm 2.3 is outcome-equivalent to Algorithm 2.4, which together with Theorem 2.41 implies Theorem 2.32.

The following lemmas show an interesting connection between underdemanded schools and sinks in rotation digraphs (of any stable assignment).

Lemma 2.43. *Consider the school-rotation digraph D_B associated with an instance $(G, <, \mathbf{q})$ at a stable assignment $M \in \mathcal{S}(G, <, \mathbf{q})$. A school $b \in B$ is a sink in D_B if and only if it is an underdemanded school in M.*

Proof. Let a be b's least preferred student among $M(b)$. The fact that b is a sink implies that $S_M(b)$ does not exist. That is, $M(a') >_{a'} b, \forall a' \notin M(b)$. This concludes the proof for the "only if" direction. The other direction is clear from the construction of the rotation digraph. ■

Fix an instance $(G(A \cup B, E), <, \mathbf{q})$ of the stable assignment problem. For a subgraph H of G, we denote by `output`(Algorithm 2.3, H) the output of Algorithm 2.3 on instance $(H, <, \mathbf{q})$. Note that this output is uniquely defined due to Theorem 2.40. Similarly, we denote by `output`(Algorithm 2.4, H) the output of Algorithm 2.4 on instance $(H, <, \mathbf{q})$.

Recall that the i-th iteration of Algorithm 2.3 takes in subgraph G^{i-1} and a stable assignment M^{i-1} of $(G^{i-1}, <, \mathbf{q})$ to construct a subgraph G^i and a stable assignment M^i of $(G^i, <, \mathbf{q})$.

Lemma 2.44. *Consider an execution of Algorithm 2.3. If $b \in B$ is a sink in D^i for some iteration i, then it remains a sink in D^j for all later iterations $j \geq i$. Moreover, if $a \in M^i(b)$, then $M^i(a) = M^j(a)$ for all $j \geq i$.*

Proof. The first part follows from the observation that $M^j \succeq M^i$ for all iterations $j \geq i$. For any $a \in M^i(b)$, since b is a sink in D^j for all $j \geq i$, (a, b) is not part of a directed cycle of D^j for any $j \geq i$. Thus, the assignment of a remains unchanged for all iterations $j \geq i$. ■

Lemma 2.45. *Consider an execution of Algorithm 2.3. If H is constructed at some iteration i (i.e., $H = G^i$), then* `output`(Algorithm 2.3, H) = `output`(Algorithm 2.3, G).

Proof. Consider the execution of Algorithm 2.3 where after graph H is constructed, the algorithm always enters case (ii) defined in Step 5, which is to eliminate rotations, until the rotation digraph contains no cycles. Let $j \geq i$ be the smallest index such that the rotation digraph D^j has no cycles. Note that $G^j = H$ by our choices of the execution and of j. By Lemma 2.18, M^j is the student-optimal stable assignment of $(H, <, \mathbf{q})$. Hence,

`output`(Algorithm 2.3, G) = `output`(Algorithm 2.3, G^j) = `output`(Algorithm 2.3, H),

concluding the proof. ■

Next lemma shows that certain edges can be removed from the input graph without changing the output of Algorithm 2.3. For a subset of edges $F \subseteq E$, we let $G \setminus F$ be the subgraph of G with vertices V and edges $E \setminus F$. For any student-school pair $ab \in E$, define

$$R(ab) = \begin{cases} \{ab\} & \text{if } a \in \bar{A}, \\ \{a'b \in E : a \geq_b a'\} & \text{if } a \notin \bar{A}. \end{cases}$$

Lemma 2.46. *Let b be a sink in the initial rotation graph D^0 of Algorithm 2.3, a be a student such that $M^0(a) = b$, and $b' >_a b$. Then,* `output`(Algorithm 2.3, G) = `output`(Algorithm 2.3, $G \setminus R(ab')$).

Proof. For the following, fix an execution of Algorithm 2.3. Since b is a sink in D^0, it follows from Lemma 2.44 that for all iterations j, b remains a sink in D^j and $M^j(a) = M^0(a) = b$. Since $b' >_a b$, by stability, we have $a' >_{b'} a$ for all $a' \in M^0(b')$. Thus, the student-school pair ab' is removed during the execution. Assume that ab' is removed during the $(j+1)$-th iteration, that is, $ab' \in E(G^j) \setminus E(G^{j+1})$. Note that ab' is removed due to one of the following two cases.

Case 1. ab' is removed in Step 9. There is therefore a non-consenting student $\bar{a} \notin \bar{A}$ such that: in iteration $j+1$, $(b', \bar{a}), (\bar{a}, M^j(\bar{a})) \in A(D^j)$ are selected as case (i) of Step 5, $M^j(\bar{a})$ is a sink in D^j, and $\bar{a} >_{b'} a$. Hence, in iteration $j+1$, the set of removed edges includes $R(ab')$. Notice that for any $a'b' \in R(ab')$, (b', a') does not appear in any of the digraphs $D^0, \ldots, D^j$ and moreover, the construction of $D^0, \ldots, D^j$ does not depend on whether $a'b'$ is present in subgraphs $G^0, \ldots, G^j$. Hence, there is an execution of Algorithm 2.3 on the input graph $G \setminus R(ab')$ which constructs subgraphs $G^0 \setminus R(ab'), \ldots, G^j \setminus R(ab'), G^{j+1}$ in exactly this order. Using Theorem 2.40 and Lemma 2.45, we conclude:

$$\texttt{output}(\text{Algorithm 2.3}, G \setminus R(ab')) = \texttt{output}(\text{Algorithm 2.3}, G^{j+1}) = \texttt{output}(\text{Algorithm 2.3}, G),$$

as required.

Case 2. ab' is removed in Step 7. That is, $(b', a), (a, b) \in A(D^j)$ are selected as case (i) of Step 5 in iteration $j+1$. Then, $R(ab') = E(G^j) \setminus E(G^{j+1})$. Because of Theorem 2.40 and the fact (discussed above) that (a, b) is an edge and b is a sink throughout the execution, we can assume that j is the smallest integer so that $(b', a) \in A(D^j)$. Then, similarly to the previous case, we have that the construction of rotation digraphs $D^0, \ldots, D^{j-1}$ is independent from whether an edge from $R(ab')$ is in graphs $G^0, \ldots, G^{j-1}$. Hence, there is an execution of Algorithm 2.3 on the input graph $G \setminus R(ab')$ which generates subgraphs $G^0 \setminus R(ab'), \ldots, G^{j-1} \setminus R(ab'), G^j \setminus R(ab') = G^{j+1}$ in exactly this order. Therefore, again by Theorem 2.40 and Lemma 2.45, we can conclude:

$$\texttt{output}(\text{Algorithm 2.3}, G \setminus R(ab')) = \texttt{output}(\text{Algorithm 2.3}, G^j) = \texttt{output}(\text{Algorithm 2.3}, G).$$

■

We are now ready to show that `school-rotate-remove with consent` is outcome-equivalent to simplified EADAM.

Theorem 2.47. *For any given input, the outputs of Algorithm 2.3 and Algorithm 2.4 coincide.*

Proof. We show `output`(Algorithm 2.3, G) = `output`(Algorithm 2.4, G) by induction on the number of edges removed, denoted as k, by Algorithm 2.4 on input graph G.

The base case is when $k = 0$. All schools are therefore underdemanded in the student-optimal stable assignment M^0 of $(G, <, \mathbf{q})$ and thus, `output`(Algorithm 2.4, G) = M^0. By Lemma 2.43, all nodes of B are sinks in the rotation digraph associated to M^0. We conclude that `output`(Algorithm 2.3, G) $= M^0 =$ `output`(Algorithm 2.4, G), as required.

Now, let $k \geq 1$ and suppose the statement is true for all instances on which Algorithm 2.4 removes at most $k-1$ edges. Define

$$Q := \{ab \in E : a \in A, M^0(a) \text{ underdemanded in } M^0, b >_a M^0(a)\} = \{a_1b_1, a_2b_2, a_3b_3, \ldots, a_kb_k\}.$$

During the first iteration of Algorithm 2.4 on input graph G, all and only the edges from the set $R := \cup_{ab \in Q} R(ab)$ are removed. Notice that $|R| \geq 1$, because otherwise $k = 0$. Let G^1 be the graph constructed by Algorithm 2.4 at the end of the first iteration, i.e., $G^1 = G \setminus R$. Observe that `output`(Algorithm 2.4, G) = `output`(Algorithm 2.4, G^1) and moreover, the number of edges removed by Algorithm 2.4 on input graph G^1 are $|R| > 0$ fewer than those removed by Algorithm 2.4 on input graph G. We have:

$$\begin{array}{rcll}
\texttt{output}(\text{Algorithm 2.4, } G) & = & \texttt{output}(\text{Algorithm 2.4, } G^1) & \text{(by what just argued)} \\
& = & \texttt{output}(\text{Algorithm 2.3, } G^1) & \text{(by induction hypothesis)} \\
& = & \texttt{output}(\text{Algorithm 2.3, } G \setminus R) & \text{(by construction)} \\
& = & \texttt{output}(\text{Algorithm 2.3, } G \setminus \cup_{i=1}^{k-1} R(a_ib_i)) & \text{(by Lemma 2.43, 2.46)} \\
& = & \ldots & \\
& = & \texttt{output}(\text{Algorithm 2.3, } G), &
\end{array}$$

as required. ■

Proof of Theorem 2.32. Immediately from Theorem 2.41 and Theorem 2.47. ■

2.10 Legal Matchings and Latin Marriages

In this section, we restrict to one-to-one instances. For an instance $(G, <)$ of the stable marriage problem, let $\mathcal{S}(G, <)$ and $\mathcal{L}(G, <)$ denote the set of stable matchings and legal matchings respectively. In addition, we call $(G_L, <)$ the *legalized instance* of $(G, <)$, where G_L is the subgraph of G defined as in Theorem 2.3. We say an instance $(G, <)$ is *legal* if $G_L = G$.

An $n \times n$ matrix is a *Latin square* if each row and each column is a permutation of numbers $1, 2, \cdots, n$. Given an instance $(G, <)$ of the stable marriage problem with complete lists, we call the position of a in the preference list of b the *rank* of a in b's list. Following the work of Benjamin, Converse, and Krieger (1995), we say an instance $(G, <)$ with $|A| = |B| = n$ is *Latin* if there exists a Latin square Q with n rows indexed by elements of A and n columns indexed by elements of B such that, for each row a and column b, $Q(a, b)$ is the rank of b in a's list, and $n + 1 - Q(a, b)$ is the rank of a in b's list. We call such Q the *Latin ranking matrix*. See Example 2.51 for an example of a Latin ranking matrix and its associated stable marriage instance. In this section, we prove the following.

Theorem 2.48. *Let $(G, <)$ be a Latin instance. Then, $G_L = G$ and there exists an instance $(G', <')$ with an additional man $\tilde{a}$ and an additional woman $\tilde{b}$ such that $|\mathcal{S}(G', <')| = 1$ and $\mathcal{L}(G', <') = \{M \cup \{\widetilde{ab}\} : M \in \mathcal{S}(G, <)\}$.*

Benjamin, Converse, and Krieger (1995) construct, for each even n, a Latin instance $(G, <)$ with n men and n women such that $|\mathcal{S}(G, <)| = \omega(2^n)$ and in the man-optimal stable matching, each man is given his favorite partner. Hence, Theorem 2.48 implies that for each odd n, there is an instance $(G', <')$ with n men and n women such that $|\mathcal{S}(G', <')| = 1$ and $|\mathcal{L}(G', <')| = \omega(2^n)$ – that is, it has one stable matching but exponentially many legal matchings. Moreover, proofs of our construction for $(G', <')$ shows that the man-optimal legal matching in $\mathcal{L}(G', <')$ assigns to each man from G his favorite partner, while the stable matching in $\mathcal{S}(G', <')$ assigns to each man from G his second least favorite partner (see Lemma 2.52). Note that, up to a different constant in the basis, the asymptotic ratio between the quantities $|\mathcal{L}(G, <)|$ and $|\mathcal{S}(G, <)|$ cannot be increased, as it has been recently shown that there exists an absolute constant $c > 1$ such that every instance of the stable marriage problem with n men and n women has $O(c^n)$ stable matchings (Karlin, Gharan, and Weber, 2018).

We believe that future investigations of the relationship between Latin instances and legal matchings may provide further advancement on a question by Knuth (1976). In his seminal, Knuth asks for a characterization of instances that maximize $|\mathcal{S}(G, <)|$ for each value of $|A| = |B| = n \in \mathbb{N}$. While an asymptotic upper bound follows from the work cited above by Karlin, Gharan, and Weber (2018), the characterization of these instances is unsolved even for reasonable small sizes. Note that, for each $n \in \mathbb{N}$, there is always a legal instance achieving the maximum, as for any instance $(G, <)$, we have $|\mathcal{S}(G_L, <)| = |\mathcal{L}(G_L, <)| \geq |\mathcal{S}(G, <)|$.

The theorem below (Benjamin, Converse, and Krieger, 1995) gives a necessary and sufficient condition for a matching to be stable in a Latin instance.

Theorem 2.49. *Let M be a matching of the instance defined by a Latin ranking matrix Q. M is stable if and only if there do not exist row a and column b such that $Q(M(b), b) > Q(a, b) > Q(a, M(a))$ or $Q(M(b), b) < Q(a, b) < Q(a, M(a))$.*

The following lemma shows that every Latin instance is legal.

Lemma 2.50. *Let $(G, <)$ be a Latin instance. Then $G_L = G$.*

Proof. Assume Q is the Latin ranking matrix of instance $(G, <)$ and $Q \in \mathbb{Z}^{n \times n}$. For $i \in [n]$, let $M^i = \{ab : Q(a, b) = i\}$. By definition of Latin squares, M^i is a matching. By construction, for any row a and column b, $Q(M^i(b), b) = i = Q(a, M^i(a))$. Therefore, M^i must be stable and thus legal due to Theorem 2.49. Since $\bigcup_{i \in [n]} M^i = E(G)$, by Theorem 2.3, $G_L = G$. ∎

As we will show next, the set of stable matchings of a Latin instance can be "masked" into the set of legal matchings of an auxiliary instance with only one more man and one more woman, such that the auxiliary instance has only one stable matching. The construction is as follows: given a Latin instance $(G(A \cup B, E), <)$, construct an auxiliary instance $(G'(A' \cup B', E'), <')$, where $A' = A \cup \{\widetilde{a}\}$, $B' = B \cup \{\widetilde{b}\}$, $E' = A' \times B'$, and $<'$ is defined as follows:

(i) every $a \in A$ ranks $\widetilde{b}$ in the last position, and $<'_a$ restricted to B is exactly $<_a$.

(ii) $\widetilde{a}$ has an arbitrary ranking of B' as long as $\widetilde{b}$ is the least preferred.

(iii) every $b \in B$ ranks $\widetilde{a}$ in the second place, and $<'_b$ restricted to A is exactly $<_b$.

(iv) $\widetilde{b}$ has an arbitrary ranking of A' as long as $\widetilde{a}$ is ranked first.

An example of our construction can be found in Example 2.51.

Example 2.51. Consider the following Latin ranking matrix Q and the associated instance $(G, <)$.

$$\begin{array}{cc} & \begin{array}{cccc} b_1 & b_2 & b_3 & b_4 \end{array} \\ \begin{array}{c} a_1 \\ a_2 \\ a_3 \\ a_4 \end{array} & \begin{pmatrix} \boxed{1} & 2 & 3 & 4 \\ 2 & 1 & \boxed{4} & 3 \\ 3 & \boxed{4} & 1 & 2 \\ 4 & 3 & 2 & \boxed{1} \end{pmatrix} \end{array}$$

$$\begin{array}{ll} a_1: & b_1 > b_2 > b_3 > b_4 \\ a_2: & b_2 > b_1 > b_4 > b_3 \\ a_3: & b_3 > b_4 > b_1 > b_2 \\ a_4: & b_4 > b_3 > b_2 > b_1 \end{array} \qquad \begin{array}{ll} b_1: & a_4 > a_3 > a_2 > a_1 \\ b_2: & a_3 > a_4 > a_1 > a_2 \\ b_3: & a_2 > a_1 > a_4 > a_3 \\ b_4: & a_1 > a_2 > a_3 > a_4 \end{array}$$

Consider the matching $M = \{a_1b_1, a_2b_3, a_3b_2, a_4b_4\}$, which corresponds to the boxed cells in the Latin ranking matrix. M is not stable because as one can check, a_3b_1 is a blocking pair. Equivalently, we can apply Theorem 2.49 on the Latin ranking matrix with $a = a_3, b = b_1$ and conclude that M is not stable. In particular, we have $Q(M(b), b) = 1 < Q(a, b) = 3 < Q(a, M(a)) = 4$.

One can check that $(G, <)$ has 10 stable matchings. Now consider the auxiliary instance $(G', <'$). Note that its preference lists are exactly those given in Example A.1 with $a_5 = \tilde{a}$ and $b_5 = \tilde{b}$. $(G', <')$. The auxiliary instance has only one stable matching, which is $\{a_1b_4$, a_2b_3, a_3b_2, a_4b_1, $\widetilde{a}\widetilde{b}\}$, but its legalized instance $(G'_L, <'_L)$ has 10 stable matchings. △

Before concluding the proof of Theorem 2.48, we first show the following facts.

Lemma 2.52. *Given a Latin instance* $(G, <)$ *with* $G = (A \cup B, E)$*, define* $(G', <')$ *as above. Then* $|\mathcal{S}(G', <')| = 1$ *and each man from* A *is given his second least favorite partner (with respect to* $<'$*) in the unique stable matching of* $(G', <')$*.*

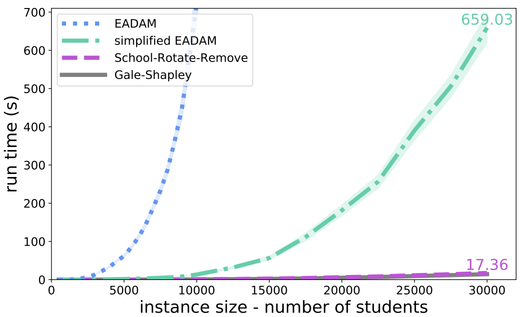
Proof. Let $M \in \mathcal{S}(G', <')$ be a stable matching in the auxiliary instance. We first show $M(\widetilde{b}) = \widetilde{a}$. Assume by contradiction that $M(\widetilde{b}) = a$ for some $a \in A$. Let b be a's least preferred partner in B. Then $b >'_a \widetilde{b} = M(a)$ by construction. By the symmetric nature of Latin instances, a must be b's most preferred partner in A, which means $a >'_b M(b)$. But then ab is a blocking pair of M, contradicting stability. Next, we want to show every woman in B is matched to her most preferred man. Assume by contradiction that the claim is not true for some $b \in B$. Then $\widetilde{a} >'_b M(b)$. Since $b >'_{\widetilde{a}} \widetilde{b}$ by construction, $\widetilde{a}b$ blocks M, which again contradicts stability. Hence, $\mathcal{S}(G', <')$ contains exactly one stable matching, namely the one where every woman is matched to her most preferred man according to $<'$. That is, every man $a \in A$ is given his second least favorite partner with respect to $<'$. ■

Lemma 2.53. *Let* $(G, <)$ *and* $(G', <')$ *be as before with* $G = (A \cup B, E)$ *and* $G' = (A' \cup B', E')$*. Then,* $\mathcal{L}(G', <') = \{M \cup \{\widetilde{a}\widetilde{b}\} : M \in \mathcal{S}(G, <)\}$*.*

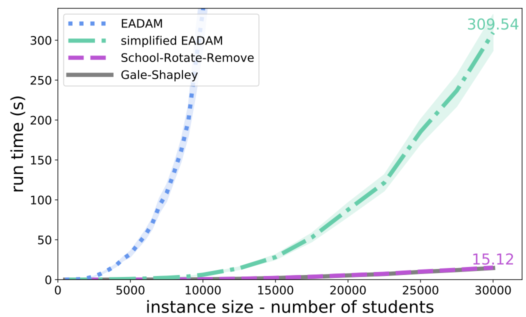
Proof. Let M_0 be the only stable matching of $(G', <')$. Since every woman in B' is matched to her most preferred man in A' as shown in the proof of Lemma 2.52, M_0 is also the woman-optimal legal matching of $\mathcal{L}(G', <')$. In addition, since $\widetilde{b}$ is the least preferred woman of every man by construction of G', $\widetilde{b}$ is a sink in the woman-rotation digraph of M_0 and remains a sink throughout the execution of `woman-rotate-remove`. Thus, $\widetilde{a}$ is matched to $\widetilde{b}$ in the man-optimal legal matching of $\mathcal{L}(G', <')$. Hence, $\widetilde{a}\widetilde{b} \in M$ for all $M \in \mathcal{L}(G', <')$ and according to Theorem 2.3, all edges in $\widetilde{E} := \{a\widetilde{b} : a \in A\} \cup \{\widetilde{a}b : b \in B\}$ (i.e., edges that are adjacent to exactly one of $\tilde{a}$ and $\tilde{b}$) are illegal. By Lemma 2.6, we have $\mathcal{L}(G', <') = \mathcal{L}(G'[E' \setminus \tilde{E}], <') = \{M \cup \{\widetilde{a}\widetilde{b}\} : M \in \mathcal{L}(G, <)\}$, where the last equality is because $E \setminus \widetilde{E} = E(G) \cup \{\widetilde{a}\widetilde{b}\}$. Finally, by Lemma 2.50, we have $\mathcal{L}(G, <) = \mathcal{S}(G, <)$ and thus, $\mathcal{L}(G', <') = \{M \cup \{\widetilde{a}\widetilde{b}\} : M \in \mathcal{S}(G, <)\}$. ■

Proof of Theorem 2.48. Immediately implied by Lemmas 2.50, 2.52 and 2.53. ■

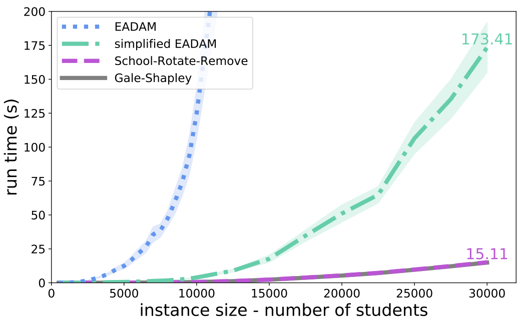
2.11 Figures



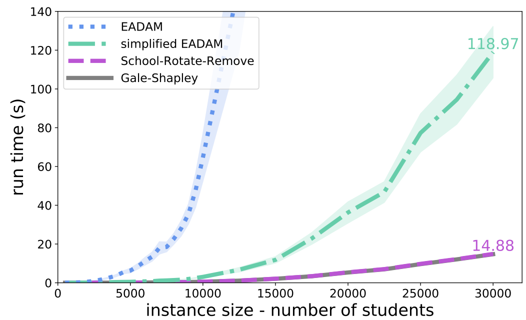
(a) all students consent



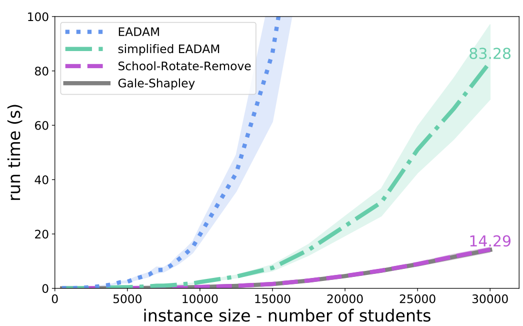
(b) 80% students consent



(c) 50% students consent

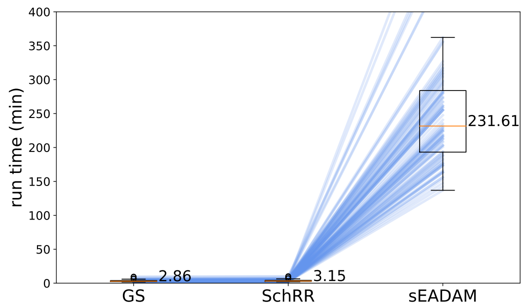


(d) 30% students consent

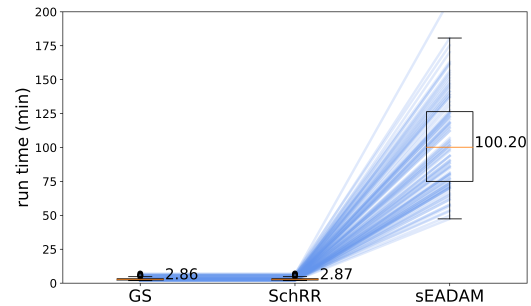


(e) 10% students consent

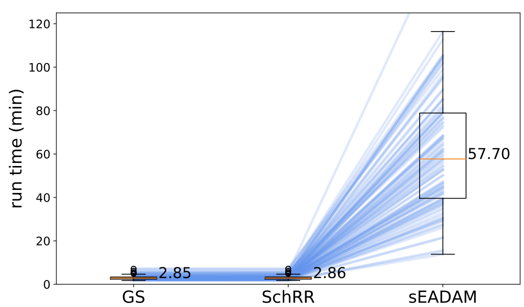
Figure 2.1: Comparing EADAM, simplified EADAM, and `school-rotate-remove with consent` in the one-to-many setting on random instances of varying sizes. Average run time of simplified EADAM and `school-rotate-remove with consent` included for the largest instance in our experiment.



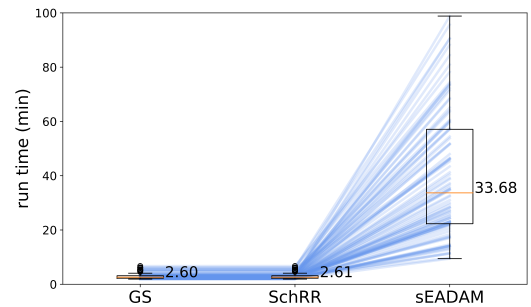
(a) all students consent



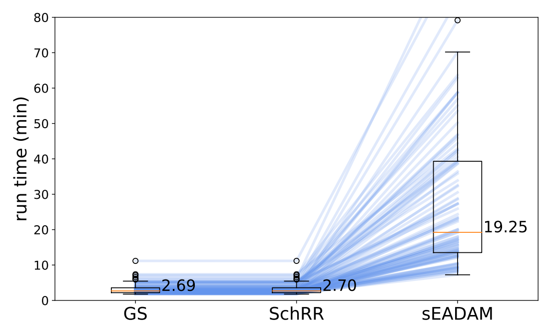
(b) 80% students consent



(c) 50% students consent



(d) 30% students consent



(e) 10% students consent

Figure 2.2: Comparing simplified EADAM (sEADAM) and `school-rotate-remove with consent` (SchRR) on random instances whose sizes are similar to those of the New York City school system. Run times of Gale-Shapley's algorithm (GS) are included as a benchmark. Run time of Kesten's original algorithm is not included because most instances fail to finish within 24 hours. Each line represents one instance. Box plots and averages of run times are included for each algorithm.

.

Chapter 3: Affinely representable lattices, stable matchings, and choice functions

3.1 Introduction

In this chapter, matching markets have two sides, which we call firms F and workers W. Although successful, the stable assignment model does not capture features that have become of crucial importance both inside and outside academia. For instance, there is growing attention to models that can increase diversity in school cohorts (Nguyen and Vohra, 2019; Tomoeda, 2018). Such constraints cannot be represented in the original model, or its one-to-many or many-to-many generalizations, since admission decisions with diversity concerns cannot be captured by a strict preference list.

To model these and other markets, instead of ranking individual potential partners, each agent $a \in F \cup W$ is endowed with a *choice function* $\mathcal{C}_a$ that picks a team she prefers the best from a given set of potential partners. See, e.g., Echenique and Yenmez (2015), Aygün and Turhan (2016), and Kamada and Kojima (2015) for more applications of models with choice functions. Models with choice functions were first studied in Roth (1984a) and Kelso Jr and Crawford (1982) (see Section 3.1.2). *Mutatis mutandis*, one can define a concept of stability in this model as well (for this and the other technical definition mentioned below, see Section 3.2). Two classical assumptions on choices functions are *substitutability* and *consistency*, under which the existence of stable matchings is guaranteed (Hatfield and Milgrom, 2005; Aygün and Sönmez, 2013). Clearly, existence results are not enough for applications (and for optimizers). Interestingly, little is known about efficient algorithms in models with choice functions. Only extensions of the classical Deferred Acceptance algorithm for finding the one-side optimal matching have been studied for this model (Roth, 1984a; Chambers and Yenmez, 2017).

The goal of this chapter is to study algorithms for optimizing a linear function w over the set of stable matchings in models with choice functions, where w is defined over firm-worker pairs. Such questions are classical in combinatorial optimization, see, e.g., Schrijver (2003) (and Manlove (2013) for problems on matching markets). We focus on two models. The first model (CM-MODEL) assumes that all choice functions are substitutable, consistent, and cardinal monotone. The second model (CM-QF-MODEL) additional assumes that for one side of the market, choice functions are also *quota-filling*. Both models generalize all classical models where agents have strict preference lists, on which results for the question above were known. For these models, Alkan (2002) has shown that stable matchings form a distributive lattice. As we argue next, this is a fundamental property that allows us to solve our optimization problem efficiently.

3.1.1 Our contributions and techniques

We give here a high-level description of our approach and results. For the standard notions of posets, distributive lattices, and related definitions see Section 3.2.1. All sets considered in this chapter are finite.

Let $\mathcal{L} = (\mathcal{X}, \succeq)$ be a distributive lattice, where the elements of $\mathcal{X}$ are distinct subsets of a base set E and $\succeq$ is a partial order on $\mathcal{X}$. We refer to $S \in \mathcal{X}$ as an *element* (of the lattice). Birkhoff's theorem (Birkhoff, 1937) implies that we can associate[1] to $\mathcal{L}$ a poset $\mathcal{B} = (Y, \succeq^\star)$ such that there is a bijection $\psi : \mathcal{X} \to \mathcal{U}(\mathcal{B})$, where $\mathcal{U}(\mathcal{B})$ is the family of *upper sets* of $\mathcal{B}$. $U \subseteq Y$ is an upper set of $\mathcal{B}$ if $y \in U$ and $y' \succeq^\star y$ for some $y' \in Y$ implies $y' \in U$. We say therefore that $\mathcal{B}$ is a *representation poset* for $\mathcal{L}$ with the *representation function* ψ. See Example 3.2 below. $\mathcal{B}$ may contain much fewer elements than the lattice $\mathcal{L}$ it represents, thus giving a possibly "compact" description of $\mathcal{L}$.

The representation poset $\mathcal{B}$ and the representation function ψ are univocally defined per Birkhoff's theorem. Moreover, the representation function ψ satisfies that for $S, S' \in \mathcal{X}$, $S \succeq S'$ if and only if $\psi(S) \subseteq \psi(S')$. Although $\mathcal{B}$ explains how elements of $\mathcal{X}$ are related to each other with respect

[1] The result proved by Birkhoff is actually a bijection between the families of lattices and posets, but in this chapter we shall not need it in full generality.

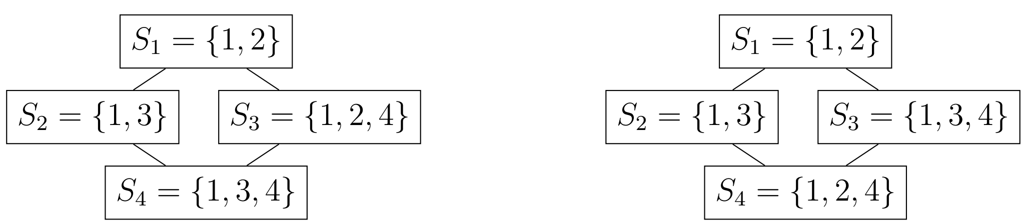
to $\succeq$, it does not contain any information on which items from E are contained in each lattice element. We introduce therefore Definition 3.1. For $S \in \mathcal{X}$ and $U \in \mathcal{U}(\mathcal{B})$, we write $\chi^S \in \{0,1\}^E$ and $\chi^U \in \{0,1\}^Y$ to denote their characteristic vectors, respectively.

Definition 3.1. Let $\mathcal{L} = (\mathcal{X}, \succeq)$ be a distributive lattice on a base set E and $\mathcal{B} = (Y, \succeq^\star)$ be a representation poset for $\mathcal{L}$ with representation function ψ. $\mathcal{B}$ is an *affine representation* of $\mathcal{L}$ if there exists an affine function $g : \mathbb{R}^Y \to \mathbb{R}^E$ such that $g(\chi^U) = \chi^{\psi^{-1}(U)}$, for all $U \in \mathcal{U}(\mathcal{B})$. In this case, we also say that $\mathcal{B}$ *affinely represents* $\mathcal{L}$ via function g and that $\mathcal{L}$ is *affinely representable*.

We observe that, in Definition 3.1, we can always assume $g(u) = Au + x^0$, where $A \in \{0, \pm 1\}^{E \times Y}$ and x^0 is the characteristic vector of the maximal element of $\mathcal{L}$. Indeed, $g(\chi^\emptyset) = x^0$. Moreover, for every $y \in \mathcal{B}$, there is $U, U' \in \mathcal{U}(\mathcal{B})$ such that $U' = U \setminus \{y\}$. Hence, letting a^y be the column of A corresponding to y, we have

$$a^y = g(\chi^U) - g(\chi^{U'}) = \chi^{\psi^{-1}(U)} - \chi^{\psi^{-1}(U')} \in \{0, \pm 1\}^E.$$

Example 3.2. Consider first the distributive lattice $\mathcal{L} = (\mathcal{X}, \succeq)$ whose Hasse diagram is given in the Figure 3.1a, with base set $E = \{1, 2, 3, 4\}$.



(a) Lattice affinely representable (b) Lattice not affinely representable

Figure 3.1: Lattices for Example 3.2.

The representation poset $\mathcal{B} = (Y, \succeq^\star)$ of $\mathcal{L}$ contains two non-comparable elements, y_1 and y_2. The representation function ψ maps S_i to U_i for $i \in [4]$ with $U_1 = \emptyset$, $U_2 = \{y_1\}$, $U_3 = \{y_2\}$, and $U_4 = \{y_1, y_2\}$. That is, $\mathcal{U}(\mathcal{B}) = \{U_1, U_2, U_3, U_4\}$. One can think of y_1 as the operation of adding $\{3\}$ and removing $\{2\}$, and y_2 as the operation of adding $\{4\}$. $\mathcal{B}$ affinely represents $\mathcal{L}$ via the function $g(\chi^U) = A\chi^U + \chi^{S_1}$ where

$$A = \begin{pmatrix} 0 & 0 \\ -1 & 0 \\ 1 & 0 \\ 0 & 1 \end{pmatrix}, \text{ as } \begin{array}{lclclcl} g(\chi^{U_1})^\intercal & = & (0,0,0,0) & + & (1,1,0,0) & = (1,1,0,0) & = (\chi^{S_1})^\intercal; \\ g(\chi^{U_2})^\intercal & = & (0,-1,1,0) & + & (1,1,0,0) & = (1,0,1,0) & = (\chi^{S_2})^\intercal; \\ g(\chi^{U_3})^\intercal & = & (0,0,0,1) & + & (1,1,0,0) & = (1,1,0,1) & = (\chi^{S_3})^\intercal; \\ g(\chi^{U_4})^\intercal & = & (0,-1,1,1) & + & (1,1,0,0) & = (1,0,1,1) & = (\chi^{S_4})^\intercal. \end{array}$$

Next consider the distributive lattice $\mathcal{L}'$ whose Hasse diagram is presented in Figure 3.1b. Note that the same poset $\mathcal{B}$ represents $\mathcal{L}'$ with the same representation function ψ. Nevertheless, $\mathcal{L}'$ is not affinely representable. If it is and such a function $g(\chi^U) = A\chi^U + \chi^{S_1}$ exists, then since $(\chi^{U_1} + \chi^{U_4})^\intercal = (1,1) = (\chi^{U_2} + \chi^{U_3})^\intercal$, we must have

$$\chi^{S_1} + \chi^{S_4} = (\chi^{S_1} + A\chi^{U_1}) + (\chi^{S_1} + A\chi^{U_4}) = (\chi^{S_1} + A\chi^{U_2}) + (\chi^{S_1} + A\chi^{U_3}) = \chi^{S_2} + \chi^{S_3}.$$

However, this is clearly not the case as $(\chi^{S_1} + \chi^{S_4})^\intercal = (2,2,0,1)$ but $(\chi^{S_2} + \chi^{S_3})^\intercal = (2,0,2,1)$. $\triangle$

As we show next, affine representability allows one to efficiently solve linear optimization problems over elements of a distributive lattice. In particular, it generalizes properties that are at the backbone of algorithms for optimizing a linear function over the set of stable matchings in the marriage model and its one-to-many and many-to-many generalizations (see, e.g., Irving, Leather, and Gusfield, 1987; Bansal, Agrawal, and Malhotra, 2007b). For instance, in the marriage model, the base set E is the set of potential pairs of agents from two sides of the market, $\mathcal{X}$ is the set of stable matchings, and for $S, S' \in \mathcal{X}$, we have $S \succeq S'$ if every firm prefers its partner in S to its partner in S'. Elements of its representation poset are certain (trading) cycles, called *rotations*.

Lemma 3.3. *Suppose we are given a poset $\mathcal{B} = (Y, \succeq^\star)$ that affinely represents a lattice $\mathcal{L} = (\mathcal{X}, \succeq)$ with representation function ψ. Let $w : E \to \mathbb{R}$ be a linear function over the base set E of $\mathcal{L}$. Then the problem $\max\{w^\intercal \chi^S : S \in \mathcal{X}\}$ can be solved in time $\texttt{min-cut}(|Y| + 2)$, where $\texttt{min-cut}(k)$ is the time complexity required to solve a minimum $s-t$ cut problem with nonnegative weights in a digraph with k nodes.*

Proof. Let $g(u) = Au + x^0$ be the affine function from the definition of affine representability. We have:

$$\max_{S \in \mathcal{X}} w^\intercal \chi^S = \max_{U \in \mathcal{U}(\mathcal{B})} w^\intercal g(\chi^U) = \max_{U \in \mathcal{U}(\mathcal{B})} w^\intercal (A\chi^U + x^0) = w^\intercal x^0 + \max_{U \in \mathcal{U}(\mathcal{B})} (w^\intercal A)\chi^U.$$

Our problem boils down therefore to the optimization of a linear function over the upper sets of $\mathcal{B}$. It is well-known that the latter problem is equivalent to computing a minimum $s - t$ cut in a digraph with $|Y| + 2$ nodes (Picard, 1976). ■

We want to apply Lemma 3.3 to the CM-QF-MODEL model. Observe that a choice function may be defined on all the (exponentially many) subsets of agents from the opposite side. We avoid this computational concern by modeling choice functions via an oracle model. That is, choice functions can be thought of as agents' private information. The complexity of our algorithms will therefore be expressed in terms of $|F|$, $|W|$, and the time required to compute the choice function $\mathcal{C}_a(X)$ of an agent $a \in F \cup W$, where the set X is in the domain of $\mathcal{C}_a$. The latter running time is denoted by `oracle-call` and we assume it to be independent of a and X. Our first result is the following.

Theorem 3.4. *The distributive lattice $(\mathcal{S}, \succeq)$ of stable matchings in the* CM-MODEL *is affinely representable. Its representation poset $(\Pi, \succeq^\star)$ has $O(|F||W|)$ elements. This representation poset, as well as its representation function ψ and affine function $g(u) = Au + x^0$, can be computed in time $O(|F|^3|W|^3 \texttt{oracle-call})$ for the* CM-QF-MODEL. *Moreover, matrix A has full column rank.*

In Theorem 3.4, we assumed that operations, such as comparing two sets and obtaining an entry from the set difference of two sets, take constant time. If this is not the case, a factor mildly polynomial in $|F| \cdot |W|$ needs to be added to the running time. Observe that Theorem 3.4 is the union of two statements. First, the distributive lattice of stable matchings in the CM-MODEL is affinely representable. Second, this representation and the corresponding functions ψ and g can be found efficiently for the CM-QF-MODEL. Those two results are proved in Section 3.3 and

Section 3.4, respectively. Combining Theorem 3.4, Lemma 3.3 and algorithms for $\texttt{min-cut}(\cdot)$, we obtain the following.

Corollary 3.5. *The problem of optimizing a linear function over the set of stable matchings in the* CM-QF-MODEL *can be solved in time* $O(|F|^3|W|^3\texttt{oracle-call})$.

Since algorithms for solving $\texttt{min-cut}(k)$ in time sub-cubic in k are known (see, e.g., Cheriyan, Hagerup, and Mehlhorn, 1996), the bottleneck in the running time of Corollary 3.5 is given by the operations that construct the poset. As an interesting consequence of studying a distributive lattice via the poset that affinely represents it, one immediately obtains a linear description of the convex hull of the characteristic vectors of elements of the lattice (see Section 3.5). In contrast, most stable matching literature (see Section 3.1.2) has focused on deducing linear descriptions for special cases of our model via ad-hoc proofs, independently of the lattice structure.

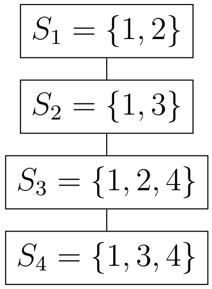
Theorem 3.6. *Let* $\mathcal{L} = (\mathcal{X}, \succeq)$ *be a distributive lattice and* $\mathcal{B} = (Y, \succeq^\star)$ *be a poset that affinely represents it via function* $g(u) = Au + x^0$. *Then the extension complexity of* $\text{conv}(\mathcal{X}) := \text{conv}\{\chi^S : S \in \mathcal{X}\}$ *is* $O(|Y|^2)$. *If moreover A has full column rank, then* $\text{conv}(\mathcal{X})$ *has* $O(|Y|^2)$ *facets.*

Theorem 3.4 and Theorem 3.6 imply the following description of the stable matching polytope $\text{conv}(\mathcal{S})$, i.e., the convex hull of the characteristic vectors of stable matchings.

Corollary 3.7. $\text{conv}(\mathcal{S})$ *has* $O(|F|^2|W|^2)$ *facets in the* CM-MODEL.

We next give an example of a lattice represented via a non-full-column rank matrix A.

Example 3.8. Consider the distributive lattice given in Figure 3.2a. It can be represented via the poset $\mathcal{B} = (Y, \succeq^\star)$ that contains three elements y_1, y_2, and y_3 where $y_1 \succeq^\star y_2 \succeq^\star y_3$. The upper sets of $\mathcal{B}$ are $\mathcal{U}(\mathcal{B}) = \{\emptyset, \{y_1\}, \{y_1, y_2\}, \{y_1, y_2, y_3\}\}$. In addition, $\mathcal{B}$ affinely represents $\mathcal{L}$ via the function $g(\chi^U) = A\chi^U + \chi^{S_1}$, where A is given in Figure 3.2b. Matrix A clearly does not have full column rank. $\triangle$



(a)

$$A = \begin{pmatrix} 0 & 0 & 0 \\ -1 & 1 & -1 \\ 1 & -1 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

(b)

Figure 3.2: Affine representation with non-full-column-rank matrix A

When the matrix A from Theorem 3.6 has full column rank, one can build on the (simple) irredundant linear description known for the order polytope (Stanley, 1986) to obtain an irredundant description in the original space for $\mathrm{conv}(\mathcal{X})$. We illustrate this idea in Section 3.5.1, by deriving a minimal description of the stable matching polytope in the one-to-one stable marriage model with strict preference lists (Rothblum, 1992; Vate, 1989), which coincides with the one given in Eirinakis et al. (2014).

Lastly, in Section 3.6, we discuss alternative ways to represent choice functions, dropping the oracle-model assumption. Interestingly, we show that choice functions in the CM-MODEL (i.e., substitutable, consistent, and cardinal monotone) do not have polynomial-size representation because the number of possible choice functions in such a model is doubly-exponential in the size of acceptable partners.

3.1.2 Relationship with the literature

Gale and Shapley (1962) introduced the one-to-one stable marriage (SM-MODEL) and the one-to-many stable admission model (SA-MODEL), and presented an algorithm which finds a stable matching. McVitie and Wilson (1971) proposed the break-marriage procedure that allows us to find the full set of stable matchings. Irving, Leather, and Gusfield (1987) presented an efficient algorithm for the maximum-weighted stable matching problem with weights over pairs of agents, utilizing the fact stable matchings form a distributive lattice (Knuth, 1976) and that its representation poset – an affine representation following our terminology – can be constructed efficiently via the concept of rotations (Irving and Leather, 1986). The above-mentioned structural and algorithm results were shown for its many-to-many generalization (MM-MODEL) by Baïou and Balinski (2000), and Bansal, Agrawal, and Malhotra (2007b). A complete survey of results on these models can be found, e.g., in Gusfield and Irving (1989) and Manlove (2013).

For models with substitutable and consistent choice functions, Roth (1984a) proved that stable matchings always exist by generalizing the algorithm presented in Gale and Shapley (1962). Blair (1988) proved that stable matchings form a lattice, although not necessarily distributive. Alkan (2001) showed that if choice functions are further assumed to be quota-filling, the lattice is distributive. Results on (non-efficient) enumeration algorithms for certain choice functions appeared in Martínez et al. (2004).

It is then natural to investigate whether algorithms from Bansal, Agrawal, and Malhotra (2007b) and Irving and Leather (1986) can be directly extended to construct the representation poset in the CM-QF-MODEL or the more general CM-MODEL. However, their definition of rotation and techniques rely on the fact that there is a strict ordering of partners, which is not available with choice functions. This, for instance, leads to the fact that the symmetric difference of two stable matchings that are adjacent in the Hasse Diagram of the lattice is a simple cycle, which is not always true in the CM-MODEL (see Example 3.27). We take then a more fundamental approach by showing a carefully defined ring of sets is isomorphic to the set of stable matchings, and thus we can construct the rotation poset following a maximal chain of the stable matching lattice. This approach conceptually follows the one by Gusfield and Irving (1989) for the SM-MODEL and leads to a generalization of the break-marriage procedure from McVitie and Wilson (1971). Again, proofs in Gusfield and Irving (1989) and McVitie and Wilson (1971) heavily rely on the strict ordering of partners, while we need to tackle the challenge of not having one.

Besides the combinatorial perspective, another line of research focuses on the polyhedral aspects. Linear descriptions of the convex hull of the characteristic vectors of stable matchings are provided for the SM-MODEL (Vate, 1989; Rothblum, 1992; Roth, Rothblum, and Vande Vate, 1993), the SA-MODEL (Baïou and Balinski, 2000), and the MM-MODEL (Fleiner, 2003). In this chapter, we provide a polyhedral description for the CM-QF-MODEL, by drawing connections between the order polytope (i.e., the convex hull of the characteristic vectors of upper sets of a poset) and Birkhoff's representation theorem of distributive lattices. A similar approach has been proposed in Aprile, Cevallos, and Faenza (2018): their result can be seen as a specialization of Theorem 3.4 to the SM-MODEL.

3.2 Basics

3.2.1 Posets, lattices, and distributivity

A set X endowed with a partial order relation $\geq$, denoted as $(X, \geq)$, is called a *partially ordered set (poset)*. When the partial order $\geq$ is clear from context, we often times simply use X to denote the poset $(X, \geq)$. Let $a, a' \in X$, if $a' > a$, we say a' is a *predecessor* of a in poset $(X, \geq)$, and a is a *descendant* of a' in poset $(X, \geq)$. If moreover, there is no $b \in X$ such that $a' > b > a$, we say that a' an *immediate predecessor* of a in poset $(X, \geq)$ and that a is an *immediate descendant* of a' in poset $(X, \geq)$. If $a \not\geq a'$ and $a' \not\geq a$, we say a and a' are *incomparable*.

For a subset $S \subseteq X$, an element $a \in X$ is said to be an *upper bound* (resp. *lower bound*) of S if for all $b \in S$, $a \geq b$ (resp. $b \geq a$). An upper bound (resp. lower bound) a' of S is said to be its *least upper bound* or *join* (resp. *greatest lower bound* or *meet*), if $a \geq a'$ (resp. $a' \geq a$) for each upper bound (resp. lower bound) a of S.

A *lattice* is a poset for which every pair of elements has a join and a meet, and for every pair those are unique by definition. Thus, two binary operations are defined over a lattice: join and meet. A lattice is *distributive* where the operations of join and meet distribute over each other.

For $n \in \mathbb{N}$, we denote by $[n]$ the set $\{1, \cdots, n\}$. Two lattices are said to be *isomorphic* if there is a structure-preserving mapping between them that can be reversed by an inverse mapping. Such a structure-preserving mapping is called an *isomorphism* between the two lattices.

3.2.2 The firm-worker models

Let F and W denote two disjoint finite sets of agents, say firms and workers respectively. Associated with each firm $f \in F$ is a *choice function* $\mathcal{C}_f : 2^{W(f)} \rightarrow 2^{W(f)}$ where $W(f) \subseteq W$ is the set of *acceptable* partners of f and $\mathcal{C}_f$ satisfies the property that for every $S \subseteq W(f)$, $\mathcal{C}_f(S) \subseteq S$. Similarly, a choice function $\mathcal{C}_w : 2^{F(w)} \rightarrow 2^{F(w)}$ is associated to each worker w. We assume that for every firm-worker pair (f, w), $f \in F(w)$ if and only if $w \in W(f)$. We let $\mathcal{C}_W$ and $\mathcal{C}_F$ denote the collection of firms' and workers' choice functions respectively. A *matching market* (or an instance) is a tuple $(F, W, \mathcal{C}_F, \mathcal{C}_W)$. Following Alkan (2002), we define below the properties of *substitutability*, *consistency*, and *cardinal monotonicity (law of aggregate demand)* for choice function $\mathcal{C}_a$ of an agent a.

Definition 3.9 (Substitutability)**.** An agent a's choice function $\mathcal{C}_a$ is substitutable if for any set of partners S, $b \in \mathcal{C}_a(S)$ implies that for all $T \subseteq S$, $b \in \mathcal{C}_a(T \cup \{b\})$.

Definition 3.10 (Consistency)**.** An agent a's choice function $\mathcal{C}_a$ is consistent if for any sets of partners S and T, $\mathcal{C}_a(S) \subseteq T \subseteq S$ implies $\mathcal{C}_a(S) = \mathcal{C}_a(T)$.

Definition 3.11 (Cardinal monotonicity)**.** An agent a's choice function $\mathcal{C}_a$ is cardinal monotone if for all sets of partners $S \subseteq T$, we have $|\mathcal{C}_a(S)| \leq |\mathcal{C}_a(T)|$.

Intuitively, substitutability implies that if an agent is selected from a set of candidates, she will also be selected from a smaller subset; consistency is also called "irrelevance of rejected contracts"; and cardinal monotonicity implies that the size of the image of the choice function is monotone with respect to set inclusion.

Aizerman and Malishevski (1981) showed that a choice function is substitutable and consistent if and only if it is *path-independent*.

Definition 3.12 (Path-independence)**.** An agent a's choice function $\mathcal{C}_a$ is path-independent if for any sets of partners S and T, $\mathcal{C}_a(S \cup T) = \mathcal{C}_a\big(\mathcal{C}_a(S) \cup T\big)$.

We next prove a few properties of path-independent choice functions.

Lemma 3.13. *Let $\mathcal{C} : 2^A \to 2^A$ be a path-independent choice function and let $A_1, A_2 \subseteq A$. If $\mathcal{C}(A_1 \cup \{a\}) = \mathcal{C}(A_1)$ for every $a \in A_2 \setminus A_1$, then $\mathcal{C}(A_1 \cup A_2) = \mathcal{C}(A_1)$.*

Proof. Assume $A_2 \setminus A_1 = \{a_1, a_2, \cdots, a_t\}$. Then, by repeated application of the path independence property,

$$\begin{aligned}\mathcal{C}(A_1 \cup A_2) &= \mathcal{C}(A_1 \cup \{a_1, a_2, \cdots, a_t\}) = \mathcal{C}(\mathcal{C}(A_1 \cup \{a_1\}) \cup \{a_2, \cdots, a_t\}) \\ &= \mathcal{C}(\mathcal{C}(A_1) \cup \{a_2, \cdots, a_t\}) = \mathcal{C}(A_1 \cup \{a_2, a_3, \cdots, a_t\}) = \cdots = \mathcal{C}(A_1).\end{aligned}$$

■

Corollary 3.14. *Let $\mathcal{C} : 2^A \to 2^A$ be a path-independent choice function and let $A_1, A_2 \subseteq A$. If $a \notin \mathcal{C}(A_1 \cup \{a\})$ for every $a \in A_2 \setminus A_1$, then $\mathcal{C}(A_1 \cup A_2) = \mathcal{C}(A_1)$.*

Proof. By the consistency property of $\mathcal{C}$, $a \notin \mathcal{C}(A_1 \cup \{a\})$ implies $\mathcal{C}(A_1 \cup \{a\}) = \mathcal{C}(A_1)$. Lemma 3.13 then applies directly. ■

Lemma 3.15. *Let $\mathcal{C} : 2^A \to 2^A$ be a path-independent choice function and let $A_1, A_2 \subseteq A$, $a \in A$. Assume $\mathcal{C}(A_1 \cup A_2) = A_1$ and $a \in \mathcal{C}(A_1 \cup \{a\})$. Then, $a \in \mathcal{C}(A_2 \cup \{a\})$.*

Proof. By path-independence, we have that $\mathcal{C}(A_1 \cup A_2 \cup \{a\}) = \mathcal{C}(\mathcal{C}(A_1 \cup A_2) \cup \{a\}) = \mathcal{C}(A_1 \cup \{a\})$ and thus $a \in \mathcal{C}(A_1 \cup A_2 \cup \{a\})$. Also, by path-independence, we have $\mathcal{C}(A_1 \cup A_2 \cup \{a\}) = \mathcal{C}\big(\mathcal{C}(A_1 \setminus \{a\}) \cup \mathcal{C}(A_2 \cup \{a\})\big)$. Since $a \notin \mathcal{C}(A_1 \setminus \{a\})$, it must be that $a \in \mathcal{C}(A_2 \cup \{a\})$. ■

Recall that a *matching* μ is a mapping from $F \cup W$ to $2^{F \cup W}$ such that for all $w \in W$ and $f \in F$, (1) $\mu(w) \subseteq F(w)$; (2) $\mu(f) \subseteq W(f)$; and (3) $w \in \mu(f)$ if and only if $f \in \mu(w)$. A matching can also be viewed as a collection of firm-worker pairs. That is, $\mu \equiv \{(f, w) : f \in F, w \in \mu(f)\}$. Thus, we use $(f, w) \in \mu$, $w \in \mu(f)$, and $f \in \mu(w)$ interchangeably. We say a matching μ is *individually rational* if for every agent a, $\mathcal{C}_a(\mu(a)) = \mu(a)$. An acceptable firm-worker pair $(f, w) \notin \mu$ is called a *blocking pair* if $w \in \mathcal{C}_f(\mu(f) \cup \{w\})$ and $f \in \mathcal{C}_w(\mu(w) \cup \{f\})$, and when such pair exists, we say μ is *blocked by* the pair or the pair blocks μ. A matching μ is *stable* if it is individually rational and it admits no blocking pairs. If f is matched to w in some stable matching, we say that (f, w) is a *stable pair* and that f (resp. w) is a *stable partner* of w (resp. f). We denote by $\mathcal{S}(\mathcal{C}_F, \mathcal{C}_W)$ the set of stable matchings in the market $(F, W, \mathcal{C}_F, \mathcal{C}_W)$, and when the market is clear from the context we abbreviate $\mathcal{S} := \mathcal{S}(\mathcal{C}_F, \mathcal{C}_W)$.

Alkan (2002) showed the following.

Theorem 3.16 (Alkan, 2002)**.** *Consider a matching market $(F, W, \mathcal{C}_F, \mathcal{C}_W)$ and assume $\mathcal{C}_F$ and $\mathcal{C}_W$ are substitutable, consistent, and cardinal monotone. Then $\mathcal{S}(\mathcal{C}_F, \mathcal{C}_W)$ is a distributive lattice under the partial order relation $\succeq$ where $\mu_1 \succeq \mu_2$ if for all $f \in F$, $\mathcal{C}_f(\mu_1(f) \cup \mu_2(f)) = \mu_1(f)$. The join (denoted by $\vee$) and meet (denoted by $\wedge$) operations of the lattice are defined component-wise. That is, for all $f \in F$:*

$$
\begin{aligned}
(\mu_1 \vee \mu_2)(f) &\coloneqq \mu_1(f) \vee \mu_2(f) \coloneqq \mathcal{C}_f(\mu_1(f) \cup \mu_2(f)), \\
(\mu_1 \wedge \mu_2)(f) &\coloneqq \mu_1(f) \wedge \mu_2(f) \\
&\qquad\qquad\quad \coloneqq \big(\big(\mu_1(f) \cup \mu_2(f)\big) \setminus (\mu_1 \vee \mu_2)(f)\big) \cup \big(\mu_1(f) \cap \mu_2(f)\big).
\end{aligned}
$$

Moreover, $\mathcal{S}(\mathcal{C}_F, \mathcal{C}_W)$ satisfies the polarity *property: $\mu_1 \succeq \mu_2$ if and only if $\mathcal{C}_w(\mu_1(w)) \cup \mu_2(w)) = \mu_2(w)$ for every worker $w \in W$.*

Because of the lattice structure, the firm- and worker-optimal stable matchings are well-defined, and we denote them respectively by μ_F and μ_W. In addition, Alkan (2002) showed two properties, which we call *concordance* (Alkan, 2002, Proposition 7) and *equal-quota* (Alkan, 2002, Proposition 6), satisfied by the family of sets of partners under all stable matchings for every agent a. Let $\Phi_a \coloneqq \{\mu(a) : \mu \in \mathcal{S}(\mathcal{C}_F, \mathcal{C}_W)\}$. Then for all $S, T \in \Phi_a$,

$$S \cap T \subseteq S \vee T \tag{concordance}$$

and

$$|S| = |T| =: \bar{q}_a. \tag{equal-quota}$$

Instead of cardinal monotonicity, in an earlier paper, Alkan (2001) considers a more restrictive property of choice functions, called *quota-filling*.

Definition 3.17 (Quota-filling)**.** An agent a's choice function $\mathcal{C}_a$ is quota-filling if there exists $q_a \in \mathbb{N}$ such that for any set of partners S, $|\mathcal{C}_a(S)| = \min(q_a, |S|)$. We call q_a the *quota* of agent a.

Intuitively, quota-filling means that an agent has a number of positions and she tries to fill as many of these positions as possible. Note that quota-filling implies cardinal monotonicity. Let q_a denote the quota of each agent $a \in F \cup W$.

Our results from Section 3.3 assume path-independence (i.e., substitutability and consistency) and cardinal monotonicity. In Section 3.4, we will restrict our model by replacing cardinal monotonicity with quota-filling for one side of the market. These two models are what we call the CM-MODEL and the CM-QF-MODEL, respectively.

3.2.3 MC-representation for path-independent choice functions

We now introduce an alternative, equivalent description of choice functions for the model studied in this chapter that we will use in examples throughout the chapter, and investigate more in detail in Section 3.6.

Aizerman and Malishevski (1981) showed that a choice function $\mathcal{C}_a$ is path-independent if and only if there exists a finite sequence of $p(\mathcal{C}_a) \in \mathbb{N}$ preference relations over acceptable partners, denoted as $\{\geq_{a,i}\}_{i \in [p(\mathcal{C}_a)]}$ indexed by i, such that for every subset of acceptable partners S, $\mathcal{C}_a(S) = \cup_{i \in [p(\mathcal{C}_a)]} \{x^*_{a,i}\}$, where $x^*_{a,i} = \max(S, \geq_{a,i})$ is the maximum element[2] of S according to $\geq_{a,i}$. We call this sequence of preference relations the *Maximizer-Collecting representation* (MC-representation) of choice function $\mathcal{C}_a$. Note that for distinct $i_1, i_2 \in [p(\mathcal{C}_a)]$, it is possible to have $x^*_{a,i_1} = x^*_{a,i_2}$.

Conceptually, one can view the MC-representation as follows: a firm is a collection of *positions*, each of which has its own preference relation; a worker is a collection of *personas*, each of whom also has his or her own preference relation. Each firm hires the best candidate for each

[2]If $S = \emptyset$, then $\max(S, \geq_{a,i})$ is defined to be $\emptyset$.

position, and the same candidate can be hired for two positions if (s)he is the best for both. A symmetric statement holds for workers and personas.

Remark 3.18. We would like to again highlight the differences between MC-representation of choice functions and the representation, in the MM-MODEL, by a single preference list $\geq_a$ together with a quota q_a. In particular, in the MM-MODEL, $\mathcal{C}_a(S) = \cup_{i\in[q_a]}\{\tilde{x}_{a,i}\}$, where $\tilde{x}_{a,i} = \max(S \setminus \{\tilde{x}_{a,j} : j \in [i-1]\}, \geq_a)$. Note that here for distinct $i_1, i_2 \in [q_a]$, $\tilde{x}_{a,i_1} \neq \tilde{x}_{a,i_2}$ unless both are $\emptyset$.

3.3 Affine representability of the stable matching lattice

In this section, we show that the distributive lattice of stable matchings in the model by Alkan (2002) is affinely representable. An algorithm to construct an affine representation is given in Section 3.4 where we additionally impose the quota-filling property upon choice functions of agents in one side of the markets. The proof of this section proceeds as follows. First, we show in Section 3.3.1 that the lattice of stable matchings $(\mathcal{S}, \succeq)$ is isomorphic to a lattice $(\mathcal{P}, \subseteq)$ belonging to a special class, that is called *ring of sets*. In Section 3.3.2, we then show that ring of sets are always affinely representable. In Section 3.3.3, we show a poset $(\Pi, \succeq^\star)$ representing $(\mathcal{S}, \succeq)$. Lastly, in Section 3.3.4, we show how to combine all those results and "translate" the affine representability of $(\mathcal{P}, \subseteq)$ to the affine representability of $(\mathcal{S}, \succeq)$, concluding the proof.

3.3.1 Isomorphism between the stable matching lattice and a ring of sets

A family $\mathcal{H} = \{H_1, H_2, \cdots, H_k\}$ of subsets of a *base set* B is a *ring of sets* over B if $\mathcal{H}$ is closed under set union and set intersection (Birkhoff, 1937). Note that a ring of sets is a distributive lattice with the partial order relation $\subseteq$, and the join and meet operations corresponds to set intersection and set union, respectively. An example of a ring of sets is given in Example 3.32.

In this and the following section, we fix a matching market $(F, W, \mathcal{C}_F, \mathcal{C}_W)$ and assume that $\mathcal{C}_F$ and $\mathcal{C}_W$ are path-independent and cardinal monotone (i.e., the framework of Alkan (2002)). Let $\phi(a)$ denote the set of stable partners of agent a. That is, $\phi(a) := \{b : b \in \mu(a) \text{ for some } \mu \in \mathcal{S}\}$.

For a stable matching μ, define

$$P_f(\mu) \coloneqq \{w \in \phi(f) : w \in \mathcal{C}_f(\mu(f) \cup \{w\})\},$$

and define the *P-set* of μ as

$$P(\mu) \coloneqq \{(f, w) : f \in F, w \in P_f(\mu)\}.$$

The goal of this section is to show the following theorem, which gives a representation of the stable matching lattice as a ring of sets. Let $\mathcal{P}(\mathcal{C}_F, \mathcal{C}_W)$ denote the set $\{P(\mu) : \mu \in \mathcal{S}(\mathcal{C}_F, \mathcal{C}_W)\}$, and we often abbreviate $\mathcal{P} \coloneqq \mathcal{P}(\mathcal{C}_F, \mathcal{C}_W)$.

Theorem 3.19. *Assume $\mathcal{C}_F$ and $\mathcal{C}_W$ are path-independent and cardinal monotone. Then,*

(1) the mapping $P : \mathcal{S} \to \mathcal{P}$ is a bijection;

(2) $(\mathcal{P}, \subseteq)$ is isomorphic to $(\mathcal{S}, \succeq)$. That is, for two stable matchings $\mu_1, \mu_2 \in \mathcal{S}$, we have $\mu_2 \succeq \mu_1$ if and only if $P(\mu_2) \subseteq P(\mu_1)$. Moreover, $P(\mu_1 \vee \mu_2) = P(\mu_1) \cap P(\mu_2)$ and $P(\mu_1 \wedge \mu_2) = P(\mu_1) \cup P(\mu_2)$. In particular, $(\mathcal{P}, \subseteq)$ is a ring of sets over the base set $\{(f, w) : f \in F, w \in \phi(f)\}$.

Remark 3.20. An isomorphism between the lattice of stable matchings and a ring of sets (also called P-set) is proved in the SM-MODEL by Gusfield and Irving (1989) as well. However, they define $P(\mu) := \{(f, w) : f \in F, w \geq_f \mu(f)\}$, hence including firm-worker pairs that are not stable. We show in Example 3.27 that in our more general setting, the P-set by Gusfield and Irving (1989) is not a ring of sets. As a consequence, while in their model the construction of the P-set for a given stable matching is immediate, in ours it is not, since we need to know first which pairs are stable.

Lemma 3.21. *Let μ_1 and μ_2 be two stable matchings such that $\mu_2 \succeq \mu_1$. Then, $P_f(\mu_2) \subseteq P_f(\mu_1)$ for every firm f.*

Proof. Since $\mu_2 \succeq \mu_1$, we have that $\mathcal{C}_f(\mu_2(f) \cup \mu_1(f)) = \mu_2(f)$. The claim then follows from Lemma 3.15. ■

Lemma 3.22. *Let μ_1 be a stable matching such that $w \in P_f(\mu_1)$ for some firm f and worker w. Then, there exists a stable matching μ_2 such that $\mu_2 \succeq \mu_1$ and $w \in \mu_2(f)$.*

Proof. By definition of $P_f(\mu_1)$, we know there exists a stable matching μ_1' such that $w \in \mu_1'(f)$. Let $\mu_2 := \mu_1 \vee \mu_1'$. We want to show that $w \in \mu_2(f)$. If $w \in \mu_1(f)$, then the claim follows due to the concordance property. So assume $w \notin \mu_1(f)$ and also assume by contradiction that $w \notin \mu_2(f)$. Then, we must have $w \in (\mu_1 \wedge \mu_1')(f)$ by definition of the meet. Since $\mu_1 \succeq \mu_1 \wedge \mu_1'$, we have $\mathcal{C}_f(\mu_1(f) \cup (\mu_1 \wedge \mu_1')(f)) = \mu_1(f)$. However, applying path-independence and consistency, we have

$$\begin{aligned}\mathcal{C}_f(\mu_1(f) \cup (\mu_1 \wedge \mu_1')(f)) &= \mathcal{C}_f\big(\mathcal{C}_f(\mu_1(f) \cup (\mu_1 \wedge \mu_1')(f) \setminus \{w\}) \cup \{w\}\big) \\ &= \mathcal{C}_f(\mu_1(f) \cup \{w\}) \neq \mu_1(f),\end{aligned}$$

which is a contradiction. ■

Lemma 3.23. *Let μ_1 and μ_2 be two stable matchings such that $\mu_2 \succeq \mu_1$. Assume $w \in P_f(\mu_1) \setminus P_f(\mu_2)$ for some firm f. Then, there exists a stable matching $\bar{\mu}_1$ with $\mu_2 \succeq \bar{\mu}_1 \succeq \mu_1$ such that $w \in \bar{\mu}_1(f)$.*

Proof. By Lemma 3.22, there exists a stable matching $\bar{\mu}_2 \succeq \mu_1$ such that $w \in \bar{\mu}_2(f)$. Let $\bar{\mu}_1 := \bar{\mu}_2 \wedge \mu_2$ and we claim that $\bar{\mu}_1$ is the desired matching. First, by definition of meet, we have $\mu_2 \succeq \bar{\mu}_1 \succeq \mu_1$. Since $w \notin P_f(\mu_2)$, by the contrapositive of the substitutability property, we have $w \notin \mathcal{C}_f(\mu_2(f) \cup \bar{\mu}_2(f))$, which implies that $w \notin (\mu_2 \vee \bar{\mu}_2)(f)$. Therefore, $w \in \bar{\mu}_1(f)$, again by the definition of meet. ■

Lemma 3.24. *Let μ_1 and μ_2 be two stable matchings. Then,*

$$P(\mu_1 \vee \mu_2) = P(\mu_1) \cap P(\mu_2) \quad \text{and} \quad P(\mu_1 \wedge \mu_2) = P(\mu_1) \cup P(\mu_2).$$

Proof. Fix a firm f, and we want to show $P_f(\mu_1 \vee \mu_2) = P_f(\mu_1) \cap P_f(\mu_2)$ and $P_f(\mu_1 \wedge \mu_2) = P_f(\mu_1) \cup P_f(\mu_2)$. If $\mu_1(f) = \mu_2(f)$, then the claim is obviously true. Thus, for the following, we assume $\mu_1(f) \neq \mu_2(f)$. We first show that $P_f(\mu_1 \vee \mu_2) \subseteq P_f(\mu_1) \cap P_f(\mu_2)$. Since $\mu_1 \vee \mu_2 \succeq \mu_1, \mu_2$, the claim follows from Lemma 3.21. Next, we show that $P_f(\mu_1 \vee \mu_2) \supseteq P_f(\mu_1) \cap P_f(\mu_2)$. If $P_f(\mu_1) \cap P_f(\mu_2) = \emptyset$, then the claim follows trivially. So we assume $P_f(\mu_1) \cap P_f(\mu_2) \neq \emptyset$ and let $w \in P_f(\mu_1) \cap P_f(\mu_2)$. By Lemma 3.22, there exists a stable matching $\bar{\mu}_1$ such that $\bar{\mu}_1 \succeq \mu_1$ and $w \in \bar{\mu}_1(f)$. Similarly, there exists a stable matching $\bar{\mu}_2$ such that $\bar{\mu}_2 \succeq \mu_2$ and $w \in \bar{\mu}_2(f)$. Consider the stable matching $\bar{\mu}_1 \vee \bar{\mu}_2$. Because of the concordance property, $w \in (\bar{\mu}_1 \vee \bar{\mu}_2)(f)$. In addition, by transitivity of $\succeq$, we have that $\bar{\mu}_1 \vee \bar{\mu}_2 \succeq \mu_1, \mu_2$ and thus $\bar{\mu}_1 \vee \bar{\mu}_2 \succeq \mu_1 \vee \mu_2$ by minimality of $\mu_1 \vee \mu_2$. Hence, by Lemma 3.21, $w \in P_f(\mu_1 \vee \mu_2)$. This concludes the first part of the thesis.

For the second half, we first show $P_f(\mu_1 \wedge \mu_2) \subseteq P_f(\mu_1) \cup P_f(\mu_2)$. Let $w \notin P_f(\mu_1) \cup P_f(\mu_2)$, we want to show that $w \notin P_f(\mu_1 \wedge \mu_2)$. Assume by contradiction that $w \in P_f(\mu_1 \wedge \mu_2)$. $w \notin P_f(\mu_1) \cup P_f(\mu_2)$ implies $w \notin \mu_1(f)$ and $w \notin \mu_2(f)$ and thus, $w \notin (\mu_1 \wedge \mu_2)(f)$. By Lemma 3.23, for both $i \in \{1, 2\}$, there exists a stable matching $\bar{\mu}_i$ such that $\mu_i \succeq \bar{\mu}_i \succeq \mu_1 \wedge \mu_2$ and $w \in \bar{\mu}_i(f)$. Note that $\mu_1 \wedge \mu_2 \succeq \bar{\mu}_1 \wedge \bar{\mu}_2 \succeq \mu_1 \wedge \mu_2$, where the first relation holds because $\mu_i \succeq \bar{\mu}_i$ for both $i \in \{1, 2\}$, and the second relation holds because $\bar{\mu}_1, \bar{\mu}_2 \succeq \mu_1 \wedge \mu_2$. Hence, $\bar{\mu}_1 \wedge \bar{\mu}_2 = \mu_1 \wedge \mu_2$. However, by applying the meet operator $\wedge$ over $\bar{\mu}_1$ and $\bar{\mu}_2$, we have $w \in (\bar{\mu}_1 \wedge \bar{\mu}_2)(f) = (\mu_1 \wedge \mu_2)(f)$, which is a contradiction.

Lastly, we show $P_f(\mu_1 \wedge \mu_2) \supseteq P_f(\mu_1) \cup P_f(\mu_2)$. Let $w \in P_f(\mu_1) \cup P_f(\mu_2)$ and wlog assume $w \in P_f(\mu_1)$. Since $\mu_1 \succeq \mu_1 \wedge \mu_2$, by Lemma 3.21, we have $w \in P_f(\mu_1 \wedge \mu_2)$. ■

Lemma 3.25. *Let μ_1 and μ_2 be two stable matchings such that $\mu_2 \succ \mu_1$ and assume that $\mu_1(f) \neq \mu_2(f)$ for some $f \in F$. Then, $P_f(\mu_1) \neq P_f(\mu_2)$.*

Proof. Assume by contradiction that $P_f(\mu_1) = P_f(\mu_2)$. Let $w \in \mu_1(f) \setminus \mu_2(f)$. w exists because $\mu_1(f) \neq \mu_2(f)$ and $|\mu_1(f)| = |\mu_2(f)|$ due to the equal-quota property. Since the stable matching lattice $(\mathcal{S}, \succeq)$ has the polarity property as shown in Theorem 3.16, we have that $\mathcal{C}_w(\mu_1(w) \cup \mu_2(w)) = \mu_1(w)$ and thus, by substitutability, we have $f \in \mathcal{C}_w(\mu_2(w) \cup \{f\})$. On the other hand, $w \in \mu_1(f)$ implies that $w \in P_f(\mu_1) = P_f(\mu_2)$. Since $w \notin \mu_2(f)$, this means that (f, w) is a blocking pair of μ_2, which contradicts the stability assumption. ■

Lemma 3.26. *Let μ_1 and μ_2 be two distinct stable matchings and assume that $\mu_1(f) \neq \mu_2(f)$ for some $f \in F$. Then, $P_f(\mu_1) \neq P_f(\mu_2)$.*

Proof. Assume by contradiction that $P_f(\mu_1) = P_f(\mu_2)$. Then, we have $P_f(\mu_1 \vee \mu_2) = P_f(\mu_1 \wedge \mu_2)$ by Lemma 3.24. However, $\mu_1(f) \neq \mu_2(f)$ implies that $(\mu_1 \vee \mu_2)(f) \neq (\mu_1 \wedge \mu_2)(f)$, which contradicts Lemma 3.25 since $\mu_1 \vee \mu_2 \succ \mu_1 \wedge \mu_2$. ■

Proof of Theorem 3.19. For (1), note that the mapping P is onto by definition. It is therefore a bijection since it is also injective as shown in Lemma 3.26. Next, we show (2). One direction of the first statement is shown in Lemma 3.21. Conversely, if $P(\mu_2) \subseteq P(\mu_1)$, then by Lemma 3.24, $P(\mu_1 \vee \mu_2) = P(\mu_1) \cap P(\mu_2) = P(\mu_2)$. Hence, by Lemma 3.26, we have $\mu_1 \vee \mu_2 = \mu_2$ and thus, $\mu_2 \succeq \mu_1$. The second statement of (2) follows from Lemma 3.24. The third follows from the second and the fact that stable matchings form a distributive lattice (Theorem 3.16). ■

Example 3.27. Consider the following instance with 4 firms and 5 workers. Agents' choice functions are given below in their MC-representations. For instance, the first position of firm f_1 prefers w_1 the most and prefers w_2 the least.

$f_1:$	$\geq_{f_1,1}:$	$w_1\ w_5\ w_3\ w_4\ w_2$	$w_1:$	$\geq_{w_1,1}:$	$f_3\ f_1\ f_2\ f_4$
	$\geq_{f_1,2}:$	$w_2\ w_5\ w_4\ w_3\ w_1$	$w_2:$	$\geq_{w_2,1}:$	$f_2\ f_1\ f_3\ f_4$
	$\geq_{f_1,3}:$	$w_1\ w_2\ w_3\ w_4\ w_5$	$w_3:$	$\geq_{w_3,1}:$	$f_1\ f_3\ f_2\ f_4$
$f_2:$	$\geq_{f_2,1}:$	$w_4\ w_2\ w_1\ w_3\ w_5$	$w_4:$	$\geq_{w_4,1}:$	$f_1\ f_2\ f_3\ f_4$
$f_3:$	$\geq_{f_3,1}:$	$w_3\ w_1\ w_2\ w_4\ w_5$	$w_5:$	$\geq_{w_5,1}:$	$f_4\ f_1\ f_2\ f_3$
$f_4:$	$\geq_{f_4,1}:$	$w_5\ w_1\ w_2\ w_3\ w_4$			

There are four stable matchings in this instance:

$$\mu_F = (f_1, w_1), (f_1, w_2), (f_2, w_4), (f_3, w_3), (f_4, w_5);$$
$$\mu_1 = (f_1, w_1), (f_1, w_4), (f_2, w_2), (f_3, w_3), (f_4, w_5);$$
$$\mu_2 = (f_1, w_2), (f_1, w_3), (f_2, w_4), (f_3, w_1), (f_4, w_5);$$
$$\mu_W = (f_1, w_3), (f_1, w_4), (f_2, w_2), (f_3, w_1), (f_4, w_5).$$

Note that μ_1 and μ_2 are not comparable. Their corresponding P-sets are

$$P(\mu_F) = (f_1, w_1), (f_1, w_2), (f_2, w_4), (f_3, w_3), (f_4, w_5);$$
$$P(\mu_1) = (f_1, w_1), (f_1, w_2), (f_1, w_4), (f_2, w_2), (f_2, w_4), (f_3, w_3), (f_4, w_5);$$
$$P(\mu_2) = (f_1, w_1), (f_1, w_2), (f_1, w_3), (f_2, w_4), (f_3, w_1), (f_3, w_3), (f_4, w_5);$$
$$P(\mu_W) = (f_1, w_1), (f_1, w_2), (f_1, w_3), (f_1, w_4), (f_2, w_2), (f_2, w_4), (f_3, w_1), (f_3, w_3), (f_4, w_5).$$

One can easily check that the claims given in Lemma 3.24 are true. Note that if we follow the definition given in Gusfield and Irving (1989) and include the pair (f_1, w_5) in $P(\mu_1)$ and $P(\mu_2)$. Then Lemma 3.24 no longer holds since $w_5 \notin P_{f_1}(\mu_F) = P_{f_1}(\mu_1 \vee \mu_2)$. $\triangle$

3.3.2 Affine representability of rings of sets via the posets of minimal differences

We now recall (mostly known) facts about posets representing rings of sets, and observe that the *affine* representability of rings of sets easily follows from those.

Fix a ring of sets $(\mathcal{H}, \subseteq)$ over a base set B, and let H_0 and H_z denote respectively the unique minimal and maximal elements of $\mathcal{H}$. That is, for all $H \in \mathcal{H}$, we have $H_0 \subseteq H \subseteq H_z$. For $a \in H_z$, let $H(a)$ denote the unique inclusion-wise minimal set among all sets in $\mathcal{H}$ that contain a, where uniqueness follows from the fact that $\mathcal{H}$ is closed under set intersection. That is,

$$H(a) := \bigcap \{H \in \mathcal{H} : a \in H\}.$$

In addition, define the set $\mathcal{I}(\mathcal{H})$ of the *irreducible* elements of $\mathcal{H}$ as follows

$$\mathcal{I}(\mathcal{H}) \coloneqq \{H \in \mathcal{H} : \exists\, a \in H_z \text{ s.t. } H = H(a)\}.$$

Since $\mathcal{I}(\mathcal{H})$ is a subset of $\mathcal{H}$, we can view $\mathcal{I}(\mathcal{H})$ as a poset under the set containment relation.

For $H \in \mathcal{I}(\mathcal{H})$, let $K(H) \coloneqq \{a \in H_z : H(a) = H\}$ denote the *centers* of H. Note that $K(H_0) = H_0$. Define $\mathcal{D}(\mathcal{H})$ as the set of centers of irreducible elements of $\mathcal{H}$ without the set H_0. Formally,

$$\mathcal{D}(\mathcal{H}) \coloneqq \{K(H) : H \in \mathcal{I}(\mathcal{H}), H \neq H_0\}.$$

Immediately from the definition of centers, we obtain the following.

Lemma 3.28. *Let $a \in B$. There is at most one $K_1 \in \mathcal{D}(\mathcal{H})$ such that $a \in K_1$. In particular, $|\mathcal{D}(\mathcal{H})| = O(|B|)$.*

For $K_1 \in \mathcal{D}(\mathcal{H})$, let $I(K_1)$ denote the irreducible element from $\mathcal{I}(\mathcal{H})$ such that $K(I(K_1)) = K_1$. Let $\sqsupseteq$ be a partial order over the set $\mathcal{D}(\mathcal{H})$ that is inherited from the set containment relation of the poset $\mathcal{I}(\mathcal{H})$. That is, for $K_1, K_2 \in \mathcal{D}(\mathcal{H})$, we have $K_1 \sqsupseteq K_2$ if and only if $I(K_1) \subseteq I(K_2)$.

Theorem 3.29 (Birkhoff, 1937)**.** *Let $(\mathcal{H}, \subseteq)$ be a ring of sets. Then, $(\mathcal{D}(\mathcal{H}), \sqsupseteq)$ is a representation poset for $(\mathcal{H}, \subseteq)$ with representation function $\psi_{\mathcal{H}}$, where $\psi_{\mathcal{H}}^{-1}(\bar{\mathcal{D}}) = \bigcup\{K_1 : K_1 \in \bar{\mathcal{D}}\} \cup H_0$ for any upper set $\bar{\mathcal{D}}$ of $(\mathcal{D}(\mathcal{H}), \sqsupseteq)$, and H_0 is the minimal element of $\mathcal{H}$.*

Lemma 3.28 and Theorem 3.29 directly imply the following.

Theorem 3.30. *Let $(\mathcal{H}, \subseteq)$ be a ring of sets over base set B. Then, $(\mathcal{D}(\mathcal{H}), \sqsupseteq)$ affinely represents $(\mathcal{H}, \subseteq)$ via affine function $g(u) = Au + x^0$, where x^0 is the characteristic vector of the minimal element of $\mathcal{H}$, and $A \in \{0,1\}^{B \times \mathcal{D}(\mathcal{H})}$ has columns χ^{K_1} for each $K_1 \in \mathcal{D}(\mathcal{H})$. Moreover, A has full column rank.*

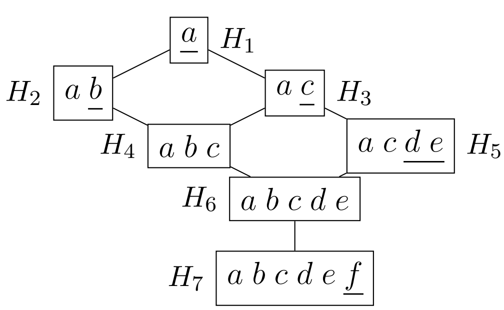
Proof. Because of the representation function $\psi_{\mathcal{H}}$ given in Theorem 3.29, it is clear that $g(\chi^U) = \chi^{\psi_{\mathcal{H}}^{-1}(U)}$ for every upper set $U \in \mathcal{U}((\mathcal{D}(\mathcal{H}), \sqsupseteq))$. Note that every row of A has at most one non-zero entry due to Lemma 3.28, and every column of A contains at least one non-zero entry by definition. Therefore, A has full column rank. ■

Lemma 3.31. *Let $(\mathcal{H}, \subseteq)$ be a ring of sets with minimal element H_0, and let $H \in \mathcal{H}$. If $H = \bigcup\{K_1 : K_1 \in \bar{\mathcal{D}}\} \cup H_0$ for some subset $\bar{\mathcal{D}}$ of $\mathcal{D}(\mathcal{H})$, then $\bar{\mathcal{D}}$ is an upper set of $(\mathcal{D}(\mathcal{H}), \sqsupseteq)$.*

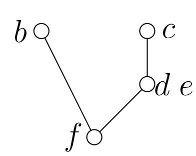
Proof. By Lemma 3.28, there is at most one subset of $\mathcal{D}(\mathcal{H})$ whose union of the elements together with H_0 gives H. On the other hand, Theorem 3.29 implies that there exists one such subset which is also an upper set of $(\mathcal{D}(\mathcal{H}), \sqsupseteq)$. The claim follows thereafter. ■

We elucidate in Example 3.32 the definitions and facts above.

Example 3.32. Consider the Hasse diagram of the ring of sets given in Figure 3.3a with base set $B = \{a, b, c, d, e, f\}$ and $\mathcal{H} = \{H_1, \cdots, H_7\}$. The irreducible elements are $\mathcal{I}(\mathcal{H}) = \{H_1, H_2, H_3, H_5, H_7\}$. The center(s) of each irreducible element is underlined, and $\mathcal{D}(\mathcal{H}) = \{\{b\}, \{c\}, \{d, e\}, \{f\}\}$. The poset of $\mathcal{D}(\mathcal{H})$ is represented in Figure 3.3b. The upper sets of poset $\mathcal{D}(\mathcal{H})$ corresponding to $H_1, \cdots, H_7$ in the exact order are: $\emptyset$; $\{\{b\}\}$; $\{\{c\}\}$; $\{\{b\}, \{c\}\}$; $\{\{c\}, \{d, e\}\}$; $\{\{b\}, \{c\}, \{d, e\}\}$; and $\{\{b\}, \{c\}, \{d, e\}, \{f\}\}$. Affine function is $g(u) = Au + x^0$ with $(x^0)^\intercal = (1, 0, 0, 0, 0, 0)$ and matrix A given below in Figure 3.3c. Note that columns of A correspond to $\{b\}$, $\{c\}$, $\{d, e\}$, $\{f\}$ in this order. △



(a) $(\mathcal{H}, \subseteq)$



(b) $(\mathcal{D}(\mathcal{H}), \sqsupseteq)$

$$A = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

(c) Matrix A

Figure 3.3: Hasse diagrams of a ring of sets and its representation poset, as well as the matrix A for affine representability for Example 3.32.

Alternatively, one can view $\mathcal{D}(\mathcal{H})$ as the set of *minimal differences* between elements of $\mathcal{H}$.

The following lemma is established directly from Lemma 2.4.3 and Corollary 2.4.1 of Gusfield and Irving (1989).

Lemma 3.33. $\mathcal{D}(\mathcal{H}) = \{H \setminus H' : H'$ *is an immediate predecessor of* H *in* $(\mathcal{H}, \subseteq)\}$.

A direct consequence of Lemma 3.28 and Lemma 3.33 is the following.

Lemma 3.34. *Let* $H', H \in \mathcal{H}$. *If* $H' \subseteq H$ *and* $H \setminus H' \in \mathcal{D}(\mathcal{H})$, *then* H' *is an immediate predecessor of* H *in* $(\mathcal{H}, \subseteq)$.

Proof. Let $K_1 := H \setminus H'$. Assume by contradiction that there exists $\bar{H} \in \mathcal{H}$ with $H' \subsetneq \bar{H} \subsetneq H$. Then, because of Lemma 3.33, there exists a center $K_2 \in \mathcal{D}(\mathcal{H})$ such that $\emptyset \neq K_2 \subsetneq K_1$. However, this contradicts Lemma 3.28. ■

3.3.3 Representation of $(\mathcal{S}, \succeq)$ via the poset of rotations

As discussed in Section 3.3.2, the poset $(\mathcal{D}(\mathcal{P}), \sqsupseteq)$ associated with $(\mathcal{P}, \subseteq)$ provides a compact representation of $(\mathcal{P}, \subseteq)$ and can be used to *reconstruct* $\mathcal{P}$ via Theorem 3.30. In this section, we show how to associate with $(\mathcal{S}, \succeq)$ a poset that is isomorphic to $(\mathcal{D}(\mathcal{P}), \sqsupseteq)$, which can be used to reconstruct $\mathcal{S}$. The precise statement is given in Theorem 3.37 below.

For $\mu, \mu' \in \mathcal{S}$, with μ' being an immediate predecessor of μ in the stable matching lattice, let

$$\rho^+(\mu', \mu) = \{(f, w) : f \in F, w \in \mu(f) \setminus \mu'(f)\}$$

and

$$\rho^-(\mu', \mu) = \{(f, w) : f \in F, w \in \mu'(f) \setminus \mu(f)\}.$$

Note that by definition,

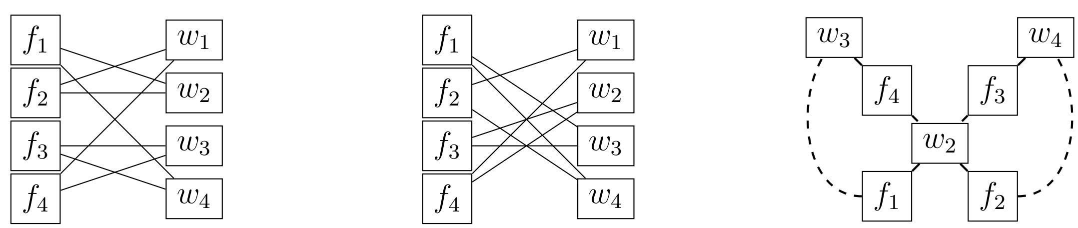
$$\mu = \mu' \triangle \rho^-(\mu', \mu) \triangle \rho^+(\mu', \mu) = \mu' \setminus \rho^-(\mu', \mu) \cup \rho^+(\mu', \mu).$$

We call $\rho(\mu', \mu) := (\rho^+(\mu', \mu), \rho^-(\mu', \mu))$ a *rotation* of $(\mathcal{S}, \succeq)$. Let $\Pi(\mathcal{S})$ denote the set of rotations of $(\mathcal{S}, \succeq)$. That is,

$$\Pi(\mathcal{S}) \coloneqq \{\rho(\mu', \mu) : \mu' \text{ is an immediate predecessor of } \mu \text{ in } (\mathcal{S}, \succeq)\}.$$

Remark 3.35. It is interesting to compare rotations in the current model (Alkan, 2002) with the analogous concept in the MM-MODEL. While in the latter case, rotations are simple cycles in the associated bipartite graph of agents (Baïou and Balinski, 2000), this may not be the case for our model, as Example 3.36 shows.

Example 3.36. Consider the two stable matchings μ' and $\bar{\mu}$ shown in Example 3.63, where μ' is an immediate predecessor of $\bar{\mu}$. As shown in Figure 3.4, their symmetric difference is not a simple cycle. In Figure 3.4c, solid lines are edges from μ' and dashed lines are those from $\bar{\mu}$. $\triangle$



(a) stable matching μ' (b) stable matching $\bar{\mu}$ (c) symmetric difference $\bar{\mu} \triangle \mu'$

Figure 3.4: Two stable matchings neighboring in $(\mathcal{S}, \succeq)$ and their symmetric difference.

In the following, we focus on proving a bijection between $\mathcal{D}(\mathcal{P})$ and $\Pi(\mathcal{S})$, and we often abbreviate $\Pi \coloneqq \Pi(\mathcal{S})$ and $\mathcal{D} \coloneqq \mathcal{D}(\mathcal{P})$. In particular, we show the following.

Theorem 3.37. *Assume $\mathcal{C}_F$ and $\mathcal{C}_W$ are path-independent and cardinal monotone. Then,*

(1) the mapping $Q : \Pi \to \mathcal{D}$, with $Q(\rho) = \rho^+$, is a bijection;

(2) $(\mathcal{D}, \sqsupseteq)$ is isomorphic to the rotation poset *$(\Pi, \succeq^\star)$ where for two rotations $\rho_1, \rho_2 \in \Pi$, $\rho_1 \succeq^\star \rho_2$ if $Q(\rho_1) \sqsupseteq Q(\rho_2)$;*

(3) $(\Pi, \succeq^\star)$ is a representation poset for $(\mathcal{S}, \succeq)$ with representation function $\psi_\mathcal{S}$ such that for any upper set $\bar{\Pi}$ of $(\Pi, \succeq^\star)$, $P(\psi_\mathcal{S}^{-1}(\bar{\Pi})) = \psi_\mathcal{P}^{-1}(\{Q(\rho) : \rho \in \bar{\Pi}\})$ where $\psi_\mathcal{P}$ is the representation

function of $(\mathcal{P}, \subseteq)$ *per Theorem 3.29; and* $\psi_{\mathcal{S}}^{-1}(\bar{\Pi}) = \left(\triangle_{\rho \in \bar{\Pi}} (\rho^- \triangle \rho^+)\right) \triangle \mu_F$*, where* $\triangle$ *is the symmetric difference operator. Moreover, equivalently, we have* $\psi_{\mathcal{S}}^{-1}(\bar{\Pi}) = \mu_F \cup (\bigcup_{\rho \in \bar{\Pi}} \rho^+) \setminus (\bigcup_{\rho \in \bar{\Pi}} \rho^-)$.

Lemma 3.38. *Let* $\mu, \mu' \in \mathcal{S}$ *such that* $\mu' \succ \mu$*. If* $w \in \mu(f) \setminus \mu'(f)$ *for some* f*, then* $w \notin P_f(\mu')$.

Proof. Since $\mu' \succ \mu$, we have $\mathcal{C}_f(\mu'(f) \cup \mu(f)) = \mu'(f)$. By path-independence and consistency, we have

$$w \notin \mu'(f) = \mathcal{C}_f(\mu'(f) \cup \mu(f)) = \mathcal{C}_f(\mathcal{C}_f(\mu'(f) \cup \mu(f) \setminus \{w\}) \cup \{w\}) = \mathcal{C}_f(\mu'(f) \cup \{w\}).$$

Therefore, $w \notin P_f(\mu')$, concluding the proof. ■

Lemma 3.39. *Let* $\mu, \mu' \in \mathcal{S}$ *such that* μ' *is an immediate predecessor of* μ *in the stable matching lattice. Then,* $\mu(f) \setminus \mu'(f) = P_f(\mu) \setminus P_f(\mu')$ *for all* $f \in F$*. In particular,* $P(\mu) \setminus P(\mu') = \rho^+(\mu', \mu)$.

Proof. Fix a firm f. $\mu(f) \setminus \mu'(f) \subseteq P_f(\mu) \setminus P_f(\mu')$ follows by definition and from Lemma 3.38. For the reverse direction, assume by contradiction that there exists $w \in P_f(\mu) \setminus P_f(\mu')$ but $w \notin \mu(f) \setminus \mu'(f)$. Since $w \notin P_f(\mu')$ implies that $w \notin \mu'(f)$ by definition of $P_f(\cdot)$, we also have $w \notin \mu(f)$. By Lemma 3.23, there exists a stable matching $\bar{\mu}$ such that $\mu' \succeq \bar{\mu} \succeq \mu$ and $w \in \bar{\mu}(f)$. However, since μ' is an immediate predecessor of μ in the stable matching lattice, we either have $\bar{\mu} = \mu$ or $\bar{\mu} = \mu'$. However, both are impossible since we deduced $w \notin \mu(f) \cup \mu'(f)$. ■

Lemma 3.40. *Let* $\mu_1, \mu_2, \mu_3 \in \mathcal{S}$ *such that* $\mu_1 \succ \mu_2 \succ \mu_3$*. If* $w \in \mu_1(f) \setminus \mu_2(f)$ *for some firm* f*, then* $w \notin \mu_3(f)$.

Proof. First, note that $\mu_1 \succ \mu_2$ implies $w \in \mu_1(f) = \mathcal{C}_f(\mu_1(f) \cup \mu_2(f))$. Thus, by substitutability, we have $w \in \mathcal{C}_f(\mu_2(f) \cup \{w\})$. Assume by contradiction that $w \in \mu_3(f)$. Then, applying Lemma 3.38 on μ_2 and μ_3, we have that $w \notin P_f(\mu_2)$, which is a contradiction. ■

Lemma 3.41. *Let* $\mu_1, \mu_1', \mu_2, \mu_2' \in \mathcal{S}$ *and assume that* μ_1', μ_2' *are immediate predecessors of* μ_1, μ_2 *in the stable matching lattice, respectively. In addition, assume that* $\mu_1 \succ \mu_2$*. If* $P(\mu_1) \setminus P(\mu_1') = P(\mu_2) \setminus P(\mu_2')$*, then* $\mu_1'(f) \setminus \mu_1(f) = \mu_2'(f) \setminus \mu_2(f)$ *for all firms* $f \in F$.

Proof. Fix a firm f. Due to Lemma 3.39, we know $\mu_1(f) \setminus \mu_1'(f) = \mu_2(f) \setminus \mu_2'(f)$. By the equal-quota property, we have $|\mu_1(f)| = |\mu_1'(f)|$ and $|\mu_2(f)| = |\mu_2'(f)|$. Thus, $|\mu_1'(f) \setminus \mu_1(f)| = |\mu_2'(f) \setminus \mu_2(f)|$ ($\natural$). If $\mu_1'(f) \setminus \mu_1(f) = \emptyset$, the claim follows immediately, and thus, in the following, we assume $\mu_1'(f) \setminus \mu_1(f) \neq \emptyset$. Assume by contradiction that there exists $w \in \mu_1'(f) \setminus \mu_1(f)$ but $w \notin \mu_2'(f) \setminus \mu_2(f)$. Since $\mu_1 \succ \mu_2$ and $\mu_i' \succ \mu_i$ for $i \in \{1, 2\}$, by Theorem 3.19, we have $P(\mu_1) \subsetneq P(\mu_2)$ and $P(\mu_i') \subsetneq P(\mu_i)$ for $i \in \{1, 2\}$. Therefore, $P(\mu_1') \subsetneq P(\mu_2')$ due to the assumption that $P(\mu_1) \setminus P(\mu_1') = P(\mu_2) \setminus P(\mu_2')$. Again by Theorem 3.19, we have $\mu_1' \succ \mu_2'$. Hence, $\mu_1 \vee \mu_2' = \mu_1'$ and we must have $w \in \mu_2'(f)$ and thus, $w \in \mu_2(f)$. However, since $\mu_1' \succ \mu_1 \succ \mu_2$ and $w \in \mu_1'(f) \setminus \mu_1(f)$, we can apply Lemma 3.40 and conclude that $w \notin \mu_2(f)$, which is a contradiction. This shows $\mu_1'(f) \setminus \mu_1(f) \subseteq \mu_2'(f) \setminus \mu_2(f)$. Together with ($\natural$), we have $\mu_1'(f) \setminus \mu_1(f) = \mu_2'(f) \setminus \mu_2(f)$. ∎

Lemma 3.42. *Let A, B, A', B' be sets such that $A \subseteq A'$ and $B \subseteq B'$. In addition, assume that $A' \setminus A = B' \setminus B$. Then, $(A' \cap B') \setminus (A \cap B) = A' \setminus A$.*

Proof. Let $X := A' \setminus A = B' \setminus B$. Notice that $A' = A \sqcup X$ and $B' = B \sqcup X$, where $\sqcup$ is the disjoint union operator. Therefore, we have $A \cap B = (A' \setminus X) \cap (B' \setminus X) = (A' \cap B') \setminus X$ and the claim follows. ∎

Lemma 3.43. *Let $\mu_1, \mu_1', \mu_2, \mu_2' \in \mathcal{S}$ and assume that μ_1', μ_2' are immediate predecessors of μ_1, μ_2 in the stable matching lattice, respectively. If $P(\mu_1) \setminus P(\mu_1') = P(\mu_2) \setminus P(\mu_2')$, then $\mu_1'(f) \setminus \mu_1(f) = \mu_2'(f) \setminus \mu_2(f)$ for every firm f. In particular, $\rho^-(\mu_1', \mu_1) = \rho^-(\mu_2', \mu_2)$.*

Proof. We first consider the case where $\mu_1 = \mu_2$. By Lemma 3.21, we have $P(\mu_i') \subseteq P(\mu_i)$ for $i \in \{1, 2\}$. Therefore,

$$P(\mu_1') = P(\mu_1) \setminus (P(\mu_1) \setminus P(\mu_1')) = P(\mu_2) \setminus (P(\mu_2) \setminus P(\mu_2')) = P(\mu_2'),$$

where the second equality is due to our assumptions that $\mu_1 = \mu_2$ and $P(\mu_1) \setminus P(\mu_1') = P(\mu_2) \setminus P(\mu_2')$. Thus, $\mu_1' = \mu_2'$ because of Theorem 3.19, and the thesis then follows. Since the cases when

$\mu_1 \succ \mu_2$ or $\mu_2 \succ \mu_1$ have already been considered in Lemma 3.41, for the following, we assume that μ_1 and μ_2 are not comparable. Let $\mu_3 \coloneqq \mu_1 \vee \mu_2$ and $\mu_3' \coloneqq \mu_1' \vee \mu_2'$. Note that $\mu_3' \succeq \mu_3$. Then, applying Lemma 3.24 and Lemma 3.42, we have

$$P(\mu_3) \setminus P(\mu_3') = (P(\mu_1) \cap P(\mu_2)) \setminus (P(\mu_1') \cap P(\mu_2')) = P(\mu_1) \setminus P(\mu_1').$$

By Theorem 3.19, Lemma 3.33 and Lemma 3.34, we also have that μ_3' is an immediate predecessor of μ_3 in the stable matching lattice. Note that by construction, we have $\mu_3 \succ \mu_1$ and $\mu_3 \succ \mu_2$ since μ_1 and μ_2 are incomparable. Applying Lemma 3.41 on μ_1 and μ_3 as well as on μ_2 and μ_3, we have $\mu_1'(f) \setminus \mu_1(f) = \mu_3'(f) \setminus \mu_3(f) = \mu_2'(f) \setminus \mu_2(f)$ for all firms $f \in F$, as desired. ■

Theorem 3.44. *Let $\mu_1, \mu_1', \mu_2, \mu_2' \in \mathcal{S}$ and assume that μ_1', μ_2' are immediate predecessors of μ_1, μ_2 in the stable matching lattice, respectively. Then, $P(\mu_1) \setminus P(\mu_1') = P(\mu_2) \setminus P(\mu_2')$ if and only if $\rho(\mu_1', \mu_1) = \rho(\mu_2', \mu_2)$.*

Proof. For the "only if" direction, assume $P(\mu_1) \setminus P(\mu_1') = P(\mu_2) \setminus P(\mu_2')$. Then, $\rho^+(\mu_1', \mu_1) = \rho^+(\mu_2', \mu_2)$ by Lemma 3.39 and $\rho^-(\mu_1', \mu_1) = \rho^-(\mu_2', \mu_2)$ by Lemma 3.43. Thus, $\rho(\mu_1', \mu_1) = \rho(\mu_2', \mu_2)$. For the "if" direction, assume $\rho(\mu_1', \mu_1) = \rho(\mu_2', \mu_2)$. Then, immediately from Lemma 3.39, we have that $P(\mu_1) \setminus P(\mu_1') = \rho^+(\mu_1', \mu_1) = \rho^+(\mu_2', \mu_2) = P(\mu_2) \setminus P(\mu_2')$. ■

Remark 3.45. In the SM-MODEL with P-sets defined as by Gusfield and Irving (1989) stated in Remark 3.20, Theorem 3.44 immediately follows from the definition of P-set. In fact, one can explicitly and uniquely construct $\rho(\mu', \mu)$ from $P(\mu) \setminus P(\mu')$. In particular, $\rho^+(\mu', \mu)$ is the set of edges (f, w) such that $P_f(\mu) \neq P_f(\mu')$ and w is the least preferred partner of f among $P_f(\mu) \setminus P_f(\mu')$, and $\rho^-(\mu', \mu)$ is the set of edges (f, w) such that $P_f(\mu) \neq P_f(\mu')$ and w is the partner that, in the preference list $\geq_f$, is immediately before the most preferred partner of f among $P_f(\mu) \setminus P_f(\mu')$.

Proof of Theorem 3.37. Because of Theorem 3.19 and Lemma 3.39, for every $K_1 \in \mathcal{D}$, there exist stable matchings μ' and μ with μ' being an immediate predecessor of μ such that $K_1 =$

$P(\mu) \setminus P(\mu') = \rho^+(\mu', \mu)$. Thus, the mapping Q is onto. Theorem 3.44 further implies that Q is injective. Hence, the mapping Q is a bijection. This bijection and the definition of $\succeq^\star$ immediately imply that $(\mathcal{D}, \sqsupseteq)$ is isomorphic to $(\Pi, \succeq^\star)$. Together with the isomorphism between $(\mathcal{S}, \succeq)$ and $(\mathcal{P}, \subseteq)$, and the fact that $(\mathcal{D}, \sqsupseteq)$ is a representation poset of $(\mathcal{P}, \subseteq)$, we deduce a bijection between elements of $(\mathcal{S}, \succeq)$ and upper sets of $(\Pi, \succeq^\star)$. That is, $(\Pi, \succeq^\star)$ is a representation poset of $(\mathcal{S}, \succeq)$ and its representation function $\psi_{\mathcal{S}}$ satisfies that for every $\mu \in \mathcal{S}$, $\{Q(\rho) : \rho \in \psi_{\mathcal{S}}(\mu)\} = \psi_{\mathcal{P}}(P(\mu))$. It remains to show that the formula for the inverse of $\psi_{\mathcal{S}}$ given in the statement of the theorem is correct. Let $\mu \in \mathcal{S}$ and let $\mu_0, \mu_1, \cdots, \mu_k$ be a sequence of stable matchings such that μ_{i-1} is an immediate predecessor of μ_i in $(\mathcal{S}, \succeq)$ for all $i \in [k]$, $\mu_0 = \mu_F$ and $\mu_k = \mu$. In addition, let $\rho_i = \rho(\mu_{i-1}, \mu_i)$ for all $i \in [k]$. Note that $\mu = \mu_F \triangle (\rho_1^- \triangle \rho_1^+) \triangle (\rho_2^- \triangle \rho_2^+) \triangle \cdots \triangle (\rho_k^- \triangle \rho_k^+)$ ($\natural$). By Theorem 3.19, $P(\mu_0) \subseteq P(\mu_1) \subseteq \cdots \subseteq P(\mu_k)$, and thus,

$$P(\mu) = P(\mu_0) \cup \big(P(\mu_1) \setminus P(\mu_0)\big) \cup \big(P(\mu_2) \setminus P(\mu_1)\big) \cup \cdots \cup \big(P(\mu_k) \setminus P(\mu_{k-1})\big).$$

Therefore, by Lemma 3.39, $P(\mu) = P(\mu_F) \cup Q(\rho_1) \cup \cdots \cup Q(\rho_k)$. By Lemma 3.31, we know that $\{Q(\rho_i) : i \in [k]\}$ is an upper set of $\mathcal{D}$ and thus, $\psi_{\mathcal{P}}(P(\mu)) = \{Q(\rho_i) : i \in [k]\}$ due to Theorem 3.29. Hence, $\psi_{\mathcal{S}}(\mu) = \{\rho_i : i \in [k]\}$. The inverse of $\psi_{\mathcal{S}}$ must be as in the first definition in the thesis so that ($\natural$) holds.

Let (f, w) be a firm-worker pair. If $(f, w) \in \rho_i^-$ for some $i \in [k]$, then $(f, w) \notin \mu$ due to Lemma 3.40. In addition, because of Lemma 3.28 and the bijection Q, μ_F, ρ_1^+, ρ_2^+, $\cdots$, ρ_i^+ are disjoint. Hence, if $(f, w) \in \mu_F \cup (\bigcup\{\rho_i^+ : i \in [k]\})$ but $(f, w) \notin \bigcup\{\rho_i^- : i \in [k]\}$, then $(f, w) \in \mu$. The second definition of $\psi_{\mathcal{S}}$ from the thesis follows immediately from these facts and the previous definition. ■

Example 3.46. Consider the following instance where each agent has a quota of 2.

$$
\begin{array}{llll}
f_1: & \geq_{f_1,1}: & w_4\ w_2\ w_1\ w_3 \\
 & \geq_{f_1,2}: & w_3\ w_1\ w_2\ w_4 \\
f_2: & \geq_{f_2,1}: & w_2\ w_3\ w_4\ w_1 \\
 & \geq_{f_2,2}: & w_1\ w_4\ w_3\ w_2 \\
f_3: & \geq_{f_3,1}: & w_1\ w_2\ w_4\ w_3 \\
 & \geq_{f_3,2}: & w_3\ w_4\ w_2\ w_1 \\
f_4: & \geq_{f_4,1}: & w_4\ w_3\ w_1\ w_2 \\
 & \geq_{f_4,2}: & w_2\ w_1\ w_3\ w_4
\end{array}
\qquad
\begin{array}{llll}
w_1: & \geq_{w_1,1}: & f_2\ f_1\ f_3\ f_4 \\
 & \geq_{w_1,2}: & f_4\ f_1\ f_3\ f_2 \\
 & \geq_{w_1,3}: & f_2\ f_4\ f_3\ f_1 \\
w_2: & \geq_{w_2,1}: & f_1\ f_2\ f_4\ f_3 \\
 & \geq_{w_2,2}: & f_3\ f_2\ f_4\ f_1 \\
 & \geq_{w_2,3}: & f_1\ f_3\ f_4\ f_2 \\
w_3: & \geq_{w_3,1}: & f_4\ f_3\ f_1\ f_2 \\
 & \geq_{w_3,2}: & f_2\ f_1\ f_3\ f_4 \\
w_4: & \geq_{w_4,1}: & f_2\ f_3\ f_4\ f_1 \\
 & \geq_{w_4,2}: & f_1\ f_4\ f_3\ f_2
\end{array}
$$

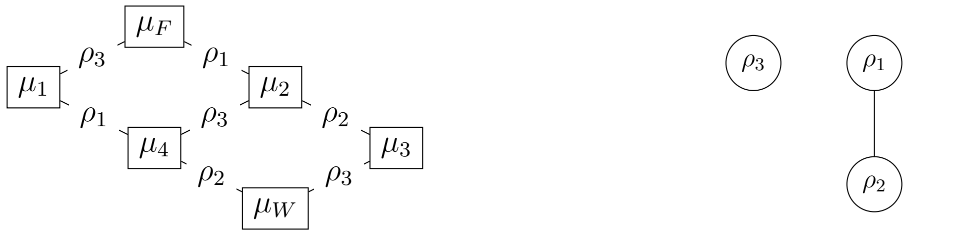
The stable matchings of this instance and their corresponding P-sets are listed below. To be concise, for matching μ, we list the assigned partners of firms f_1, f_2, f_3, f_4 in the exact order. Similarly, for P-set $P(\mu)$, we list in the order of $P_{f_1}(\mu)$, $P_{f_2}(\mu)$, $P_{f_3}(\mu)$, $P_{f_4}(\mu)$ and replace $\{w_1, w_2, w_3, w_4\}$ with W.

$$
\begin{array}{ll}
\mu_F = (\{w_3, w_4\}, \{w_1, w_2\}, \{w_1, w_3\}, \{w_2, w_4\}) & P(\mu_F) = (\{w_3, w_4\}, \{w_1, w_2\}, \{w_1, w_3\}, \{w_2, w_4\}) \\
\mu_1 = (\{w_3, w_4\}, \{w_1, w_2\}, \{w_1, w_4\}, \{w_2, w_3\}) & P(\mu_1) = (\{w_3, w_4\}, \{w_1, w_2\}, \{w_1, w_3, w_4\}, \{w_2, w_3, w_4\}) \\
\mu_2 = (\{w_3, w_4\}, \{w_1, w_2\}, \{w_2, w_3\}, \{w_1, w_4\}) & P(\mu_2) = (\{w_3, w_4\}, \{w_1, w_2\}, \{w_1, w_2, w_3\}, \{w_1, w_2, w_4\}) \\
\mu_3 = (\{w_2, w_4\}, \{w_1, w_3\}, \{w_2, w_3\}, \{w_1, w_4\}) & P(\mu_3) = (\{w_2, w_3, w_4\}, \{w_1, w_2, w_3\}, \{w_1, w_2, w_3\}, \{w_1, w_2, w_4\}) \\
\mu_4 = (\{w_3, w_4\}, \{w_1, w_2\}, \{w_2, w_4\}, \{w_1, w_3\}) & P(\mu_4) = (\{w_3, w_4\}, \{w_1, w_2\}, W, W) \\
\mu_W = (\{w_2, w_4\}, \{w_1, w_3\}, \{w_2, w_4\}, \{w_1, w_3\}) & P(\mu_W) = (\{w_2, w_3, w_4\}, \{w_1, w_2, w_3\}, W, W)
\end{array}
$$

The stable matching lattice $(\mathcal{S}, \succeq)$ and the rotation poset $(\Pi, \succeq^\star)$ are shown in Figure 3.5.

Due to Theorem 3.19 and Theorem 3.37, one can also view Figure 3.5a and Figure 3.5b respectively as the ring of sets $(\mathcal{P}, \subseteq)$ and the poset of minimal differences $(\mathcal{D}, \sqsupseteq)$.

Below, we list the rotations in Π and their corresponding minimal differences in $\mathcal{D}$. In addition,



(a) Stable matching lattice $(\mathcal{S}, \succeq)$

(b) Rotation poset $(\Pi, \succeq^\star)$

Figure 3.5: The stable matching lattice and its rotation poset of Example 3.46

we label in Figure 3.5a the edges of the Hasse Diagram by these rotations.

$$\rho_1 : \ Q(\rho_1) = \rho_1^+ = \{\{f_3, w_2\}, \{f_4, w_1\}\}; \ \rho_1^- = \{\{f_3, w_1\}, \{f_4, w_2\}\}$$

$$\rho_2 : \ Q(\rho_2) = \rho_2^+ = \{\{f_1, w_2\}, \{f_2, w_3\}\}; \ \rho_2^- = \{\{f_1, w_3\}, \{f_2, w_2\}\}$$

$$\rho_3 : \ Q(\rho_3) = \rho_3^+ = \{\{f_3, w_4\}, \{f_4, w_3\}\}; \ \rho_3^- = \{\{f_3, w_3\}, \{f_4, w_4\}\}$$

$\triangle$

3.3.4 Concluding the proof for the first part of Theorem 3.4

Because of Theorem 3.37, part (3), we know that poset $(\Pi, \succeq^\star)$ represents lattice $(\mathcal{S}, \succeq)$. Let $\psi_{\mathcal{S}}$ be the representation function as defined in Theorem 3.37. We denote by $E \subseteq F \times W$ the set of acceptable firm-worker pairs. Hence, E is the base set of lattice $(\mathcal{S}, \succeq)$. We deduce the following, proving the structural statement from Theorem 3.4.

Lemma 3.47. *Let $\bar{\Pi}_1, \bar{\Pi}_2$ be two upper sets of $(\Pi, \succeq^\star)$ and let $\mu_i = \psi_{\mathcal{S}}^{-1}(\bar{\Pi}_i)$ for $i \in \{1, 2\}$. If $\bar{\Pi}_1 \subseteq \bar{\Pi}_2$, then $\mu_1 \succeq \mu_2$.*

Proof. Let $\bar{\mathcal{D}}_i := \{Q(\rho) : \rho \in \bar{\Pi}_i\}$ and let $P_i := \psi_{\mathcal{P}}^{-1}(\bar{\mathcal{D}}_i)$ for $i \in \{1, 2\}$. Since $\bar{\Pi}_1 \subseteq \bar{\Pi}_2$, we have $\bar{\mathcal{D}}_1 \subseteq \bar{\mathcal{D}}_2$ and thus subsequently $P_1 \subseteq P_2$. Since $\psi_{\mathcal{P}}^{-1}(\bar{\mathcal{D}}_i) = P(\psi_{\mathcal{S}}^{-1}(\bar{\Pi}_i))$ by Theorem 3.37, $P_i = P(\mu_i)$ for both $i = 1, 2$. Therefore, by Theorem 3.19, $\mu_1 \succeq \mu_2$. ■

Lemma 3.48. *Let $\rho_1, \rho_2 \in \Pi$. If $\rho_1^+ \cap \rho_2^- \neq \emptyset$, then $\rho_1 \succ^\star \rho_2$.*

Proof. Assume by contradiction that $\rho_1 \not\succ^\star \rho_2$, that is, either $\rho_2 \succ^\star \rho_1$ or that they are not comparable. Let $\bar{\Pi}_1 \coloneqq \{\rho \in \Pi : \rho \succeq \rho_2\}$ be the inclusion-wise smallest upper set of Π that contains ρ_2, let $\bar{\Pi}_0 \coloneqq \bar{\Pi}_1 \setminus \{\rho_2\}$, and let $\bar{\Pi}_2 \coloneqq \{\rho \in \Pi : \rho \succeq \rho_1 \text{ or } \rho \succeq \rho_2\}$ be the inclusion-wise smallest upper set of Π that contains both ρ_1 and ρ_2. Note that $\bar{\Pi}_0 \subsetneq \bar{\Pi}_1 \subsetneq \bar{\Pi}_2$, where the second strict containment is due to our assumption that $\rho_1 \not\succ^\star \rho_2$ and thus $\rho_1 \notin \bar{\Pi}_1$. For $i \in \{0, 1, 2\}$, let $\mu_i \coloneqq (\triangle_{\rho \in \bar{\Pi}_i}(\rho^- \triangle \rho^+)) \triangle \mu_F$. Since $\bar{\Pi}_i$ is an upper set of $(\Pi, \succeq^\star)$, μ_i is a stable matching by Theorem 3.37. Moreover, $\mu_0 \succ \mu_1 \succ \mu_2$ by Lemma 3.47. Let $(f, w) \in \rho_1^+ \cap \rho_2^-$. Since $\rho(\mu_0, \mu_1) = \rho_2$, we have $(f, w) \in \mu_0 \setminus \mu_1$. Since ρ_1 is a $\succeq$-minimal element in $\bar{\Pi}_2$, $\bar{\Pi}_2 \setminus \{\rho_1\}$ is also an upper set of Π. Then, $\mu_2' \coloneqq (\triangle_{\rho \in \bar{\Pi}_2 \setminus \{\rho_1\}}(\rho^- \triangle \rho^+)) \triangle \mu_F$ is a stable matching by Theorem 3.37, and $\mu_2 = \mu_2' \setminus \rho_1^- \cup \rho_1^+$. Thus, we have $(f, w) \in \mu_2$. Together, we have $w \in (\mu_0(f) \cap \mu_2(f)) \setminus \mu_1(f)$. However, this contradicts Lemma 3.40. ■

Theorem 3.49. *The rotation poset $(\Pi, \succeq^\star)$ affinely represents the stable matching lattice $(\mathcal{S}, \succeq)$ with affine function $g(u) = Au + \chi^{\mu_F}$, where $A \in \{0, \pm 1\}^{E \times \Pi}$ is matrix with columns $\chi^{\rho^+} - \chi^{\rho^-}$ for each $\rho \in \Pi$. Moreover, $|\Pi| = O(|F||W|)$ and matrix A has full column rank.*

Proof. The first claim follows immediately because by Theorem 3.37, part (3), $\chi^\mu = A\chi^{\psi_S(\mu)} + \chi^{\mu_F}$, for any stable matching μ. Because of Theorem 3.37, $|\Pi| = |\mathcal{D}|$. In addition, by Lemma 3.28, we have $|\mathcal{D}| = |E| = O(|F||W|)$. Thus, $|\Pi| = O(|F||W|)$. Finally, we show that matrix A has full column rank. Assume by contradiction that there is a non-zero vector $\lambda \in \mathbb{R}^\Pi$ such that $\sum_{\rho \in \Pi} \lambda_\rho(\chi^{\rho^+} - \chi^{\rho^-}) = \mathbf{0}$. Let $\tilde{\Pi} \coloneqq \{\rho \in \Pi : \lambda_\rho \neq 0\}$ denote the set of rotations whose corresponding coefficients in λ are non-zero. Let ρ_1 be a minimal rotation (w.r.t. $\succeq^\star$) in $\tilde{\Pi}$ and let (f, w) be a firm-worker pair in ρ_1^+. Because of Lemma 3.28 and the bijection Q, there is no rotation $\rho \neq \rho_1$ such that $(f, w) \in \rho^+$. Therefore, there must exist a rotation $\rho_2 \in \tilde{\Pi}$ with $(f, w) \in \rho_2^-$. Note that we must have $\rho_1 \succ^\star \rho_2$ due to Lemma 3.48. However, this contradicts the choice of ρ_1. ■

3.4 Algorithms

Because of Theorem 3.49, in order to conclude the proof of Theorem 3.4, we are left to explicitly construct $(\Pi, \succeq^\star)$. That is, we need to find elements of Π, and how they relate to each other via $\succeq^\star$. We fix an instance $(F, W, \mathcal{C}_F, \mathcal{C}_W)$ and abbreviate $\mathcal{S} := \mathcal{S}(\mathcal{C}_F, \mathcal{C}_W)$.

In this section, we further assume workers' choice functions to be quota-filling. Under this additional assumption, for each worker $w \in W$, the family of sets of partners w is assigned to under all stable matchings (denoted as Φ_w) satisfies an additional property, which we call the *full-quota*[3] property (see Lemma 3.50). Recall that q_w denote the quota of worker w and $\bar{q}_w$ is the number of firms matched to w under every stable matching, which is constant due to the equal-quota property (i.e., $|S| = \bar{q}_w$ for all $S \in \Phi_w$).

Lemma 3.50. *For every worker $w \in W$, if $\bar{q}_w < q_w$, then w is matched to the same set of firms in all stable matchings. That is,*

$$\bar{q}_w < q_w \implies |\Phi_w| = 1. \tag{full-quota}$$

Proof. Assume by contradiction that $\bar{q}_w < q_w$ but $|\Phi_w| > 1$. Let S_1, S_2 be two distinct elements from Φ_w and let μ_i be the matching such that $\mu_i(w) = S_i$ for $i = 1, 2$. Note that due to the equal-quota property, we have $|S_1| = |S_2| = \bar{q}_w$. Consider the stable matching $\mu := \mu_1 \wedge \mu_2$. Then,

$$|\mu(w)| = |\mathcal{C}_w(\mu_1(w) \cup \mu_2(w))| = |\mathcal{C}_w(S_1 \cup S_2)| = \min(|S_1 \cup S_2|, q_w|) > \bar{q}_w,$$

where the first equality is by Theorem 3.16 and the last two relations are by quota-filling. However, this contradicts the equal-quota property since μ is a stable matching. ■

Our approach to construct $(\Pi, \succeq^\star)$ is as follows. First, we recall Roth's adaptation of the Deferred Acceptance algorithm to find a firm- or worker-optimal stable matching (Section 3.4.1). Second, we feed the output of Roth's algorithm to an algorithm that produces a maximal chain

[3]Note that the full-quota property is analogous to the *Rural Hospital Theorem* (Roth, 1986) in the SA-MODEL where agents have preferences over individual partners instead of over sets of partners.

$C_1, C_2, \ldots, C_k$ of $(\mathcal{S}, \succeq)$ and the set Π (Section 3.4.2). In Section 3.4.3, we give an algorithm that, given a maximal chain of a ring of sets, constructs the partial order of the poset of minimal differences. This and previous facts are then exploited in Section 3.4.4 to construct the partial order $\succeq^\star$ on elements of Π. We sum up our algorithm in Section 3.4.5, where we show that the overall running time is $O(|F|^3|W|^3\texttt{oracle-call})$.

We start with a definition and properties which will be used in later algorithms. For a matching μ, let

$$\bar{X}_f(\mu) \coloneqq \{w \in W(f) : \mathcal{C}_f(\mu(f) \cup \{w\}) = \mu(f)\},$$

and define the *closure* of μ, denoted by $\bar{X}(\mu)$, as the collection of sets $\{\bar{X}_f(\mu) : f \in F\}$. Note that $\mu(f) \subseteq \bar{X}_f(\mu)$ for every firm f and individually rational matching μ.

Lemma 3.51. *Let μ be an individually rational matching. Then, for every firm f, we have $\mathcal{C}_f(\bar{X}_f(\mu)) = \mu(f)$.*

Proof. Fix a firm f. Since μ is individually rational, $\mathcal{C}_f(\mu(f)) = \mu(f)$. The claim then follows from a direct application of Lemma 3.13 with $A_1 = \mu(f)$ and $A_2 = \bar{X}_f(\mu)$. ■

Lemma 3.52. *Let $\mu_1, \mu_2 \in \mathcal{S}(\mathcal{C}_F, \mathcal{C}_W)$ such that $\mu_1 \succeq \mu_2$. Then, for every firm f, $\mu_2(f) \subseteq \bar{X}_f(\mu_1)$.*

Proof. Since $\mu_1 \succeq \mu_2$, we have $\mathcal{C}_f(\mu_1(f) \cup \mu_2(f)) = \mu_1(f)$ for every firm f. Thus, by the consistency property of $\mathcal{C}_f$, for every $w \in \mu_2(f)$, we have $\mathcal{C}_f(\mu_1(f) \cup \{w\}) = \mu_1(f)$. The claim follows. ■

Lemma 3.53. *The following three operations can be performed in polynomial times:*

(1). given a matching μ, computing its closure $\bar{X}(\mu)$ can be performed in time $O(|F||W|\texttt{oracle-call})$;

(2). given a matching μ, deciding whether it is stable can be performed in time $O(|F||W|\texttt{oracle-call})$;

(3). given stable matchings $\mu, \mu' \in \mathcal{S}$, deciding whether $\mu' \succeq \mu$ can be performed in time $O(|F|\texttt{oracle-call})$.

Proof. (1). For any firm f, computing $\bar{X}_f(\mu)$ requires $O(|W|)$ `oracle-calls` by definition and thus, computing the closure of μ takes $O(|F||W|)$ `oracle-calls`. (2). To check if a matching μ is stable, we need to check first if it is individually rational, which takes $O(|F|+|W|)$ `oracle-calls`, and then to check if it admits any blocking pair, which takes $O(|F||W|)$ `oracle-calls`. (3). To decide if $\mu' \succeq \mu$, one need to check if for every firm $f \in F$, $\mathcal{C}_f(\mu'(f) \cup \mu(f)) = \mu'(f)$, and this takes $O(|F|)$ `oracle-calls`. ■

3.4.1 Deferred acceptance algorithm

The deferred acceptance algorithm introduced in Roth (1984a)[4] can be seen as a generalization of the algorithm proposed in Gale and Shapley (1962). For the following, we assume that firms are the proposing side. Initially, for each firm f, let $X_f \coloneqq W(f)$, i.e., the set of acceptable workers of f. At every step, every firm f *proposes* to workers in $\mathcal{C}_f(X_f)$. Then, every worker w considers the set of firms X_w who made a proposal to w, *temporarily accepts* $Y_w \coloneqq \mathcal{C}_w(X_w)$, and *rejects* the rest. Afterwards, each firm f removes from X_f all workers that rejected f. The *firm-proposing* algorithm iterates until there is no rejection. Hence, throughout the algorithm, X_f denotes the set of acceptable workers of f that have not rejected f. A formal description is given in Algorithm 3.1.

Note that for every step s other than the final step, there exists a firm $f \in F$ such that $X_f^{(s)} \subsetneq X_f^{(s-1)}$. Therefore, the algorithm terminates, since there is a finite number of firms and workers. Moreover, the output has interesting properties.

Theorem 3.54 (Theorem 2, Roth, 1984a)**.** *Let $\bar{\mu}$ be the output of Algorithm 3.1 over a matching market $(F, W, \mathcal{C}_F, \mathcal{C}_W)$ assuming $\mathcal{C}_F, \mathcal{C}_W$ are path-independent. Then, $\bar{\mu} = \mu_F$.*

Due to the symmetry between firms and workers in a market where the only assumption on choice functions is path-independence, swapping the role of firms and workers in Algorithm 3.1, we have the *worker-proposing* deferred acceptance algorithm, which outputs μ_W.

[4]The model considered in Roth (1984a) is more general than our setting here, where choice functions are only assumed to be substitutable and consistent, not necessarily quota-filling.

Algorithm 3.1 Firm-proposing DA algorithm for an instance $(F, W, \mathcal{C}_F, \mathcal{C}_W)$.

initialize the step count $s \leftarrow 0$
for each firm f **do** initialize $X_f^{(s)} \leftarrow W(f)$ **end for**
repeat
    for each worker w **do**
        $X_w^{(s)} \leftarrow \{f \in F : w \in \mathcal{C}_f(X_f^{(s)})\}$
        $Y_w^{(s)} \leftarrow \mathcal{C}_w(X_w^{(s)})$
    end for
    for each firm f **do**
        update $X_f^{(s+1)} \leftarrow X_f^{(s)} \setminus \{w \in W : f \in X_w^{(s)} \setminus Y_w^{(s)}\}$
    end for
    update the step count $s \leftarrow s + 1$
until $X_f^{(s)} = X_f^{(s-1)}$ for every firm f
Output: matching $\bar{\mu}$ with $\bar{\mu}(w) = Y_w^{(s-1)}$ for every worker w

3.4.2 Constructing Π via a maximal chain of $(\mathcal{S}, \succeq)$

Let $(\mathcal{H}, \subseteq)$ be a ring of sets. A *chain* $C_0, \cdots, C_k$ in $(\mathcal{H}, \subseteq)$ is an ordered subset of $\mathcal{H}$ such that C_{i-1} is a predecessor of C_i in $(\mathcal{H}, \subseteq)$ for all $i \in [k]$. The chain is *complete* if moreover C_{i-1} is an immediate predecessor of C_i for all $i \in [k]$; it is *maximal* if it is complete, $C_0 = H_0$ and $C_k = H_z$. Consider $K \in \mathcal{D}(\mathcal{H})$. If $K = C_i \setminus C_{i-1}$ for some $i \in [k]$, then we say that the chain *contains* the minimal difference K. We start with the theorem below, where it is shown that the set $\mathcal{D}(\mathcal{H})$ can be obtained by following any maximal chain of $(\mathcal{H}, \subseteq)$.

Theorem 3.55 (Theorem 2.4.2, Gusfield and Irving, 1989)**.** *Let $H', H \in \mathcal{H}$ such that $H' \subseteq H$. Then, there exists a complete chain from H' to H in $(\mathcal{H}, \subseteq)$, and every such chain contains exactly the same set of minimal differences. In particular, for any maximal chain $(C_0, \cdots, C_k)$ in $(\mathcal{H}, \subseteq)$, we have $\{C_i \setminus C_{i-1} : i \in [k]\} = \mathcal{D}(\mathcal{H})$ and $k = |\mathcal{D}(\mathcal{H})|$.*

In this section, we present Algorithm 3.3 that, on inputs μ', outputs a stable matching μ that is an immediate descendant of μ' in $(\mathcal{S}, \succeq)$. Then, using Algorithm 3.3 as a subroutine, Algorithm 3.4 gives a maximal chain of $(\mathcal{S}, \succeq)$.

We start by extending to our setting the *break-marriage* idea proposed by McVitie and Wilson

(1971) for finding the full set of stable matchings in the one-to-one stable marriage model. Given a stable matching μ' and a firm-worker pair $(f', w') \in \mu' \setminus \mu_W$, the break-marriage procedure, denoted as `break-marriage(`μ', f', w'`)`, works as follows. We first initialize X_f to be $\bar{X}_f(\mu')$ for every firm $f \neq f'$, while we set $X_{f'} = \bar{X}_{f'}(\mu') \setminus \{w'\}$. We then restart the deferred acceptance process. The algorithm continues in iterations as in the **repeat** loop of Algorithm 3.1, with the exception that worker w' temporarily accepts $Y_{w'} := \mathcal{C}_{w'}(X_{w'} \cup \{f'\}) \setminus \{f'\}$. As an intuitive explanation, this acceptance rule of w' ensures that for the output matching $\bar{\mu}$, we have $\mathcal{C}_{w'}(\bar{\mu}(w') \cup \mu'(w')) = \bar{\mu}(w')$, as we show in Lemma 3.58. The formal break-marriage procedure is summarized in Algorithm 3.2. See Example 3.63 for a demonstration. Note that by choice of the pair (f', w'), we have $|\mu'(w')| = q_{w'}$.

Algorithm 3.2 `break-marriage(`μ', f', w'`)`, with $(f', w') \in \mu' \setminus \mu_W$ and $\mu' \in \mathcal{S}$

for each firm $f \neq f'$ **do** initialize $X_f^{(0)} \leftarrow \bar{X}_f(\mu')$ **end for**
initialize $X_{f'}^{(0)} \leftarrow \bar{X}_{f'}(\mu') \setminus \{w'\}$
initialize the step count $s \leftarrow 0$
repeat
    for each worker w **do**
        $X_w^{(s)} \leftarrow \{f \in F : w \in \mathcal{C}_f(X_f^{(s)})\}$
        if $w \neq w'$ **then** $Y_w^{(s)} \leftarrow \mathcal{C}_w(X_w^{(s)})$ **else** $Y_w^{(s)} \leftarrow \mathcal{C}_w(X_w^{(s)} \cup \{f'\}) \setminus \{f'\}$
    end for
    for each firm f **do**
        update $X_f^{(s+1)} \leftarrow X_f^{(s)} \setminus \{w \in W : f \in X_w^{(s)} \setminus Y_w^{(s)}\}$
    end for
    update the step count $s \leftarrow s + 1$
until $X_f^{(s-1)} = X_f^{(s)}$ for every firm f

Output: matching $\bar{\mu}$ with $\bar{\mu}(w) = Y_w^{(s-1)}$ for every worker w

With the same reasoning as for the DA algorithm, the `break-marriage(`μ', f', w'`)` procedure is guaranteed to terminate. Let $s^\star$ be the value of step count s at the end of the algorithm. Note that, for every firm $f \in F$, we have

$$\bar{X}_f(\mu') \supseteq X_f^{(0)} \supseteq X_f^{(1)} \supseteq \cdots \supseteq X_f^{(s^\star)}, \tag{3.1}$$

where the first containment is an equality unless $f = f'$. In particular, (3.1) implies that $f' \notin X_{w'}^{(s)}$ for all $s \in \{0, 1, \cdots, s^\star\}$. Also note that the termination condition implies

$$\bar{\mu}(f) = \mathcal{C}_f(X_f^{(s^\star)}) = \mathcal{C}_f(X_f^{(s^\star - 1)}) \tag{3.2}$$

for every firm f, while for every worker $w \neq w'$ it implies that

$$\bar{\mu}(w) = Y_w^{(s^\star - 1)} = \mathcal{C}_w(X_w^{(s^\star - 1)}) = X_w^{(s^\star - 1)}. \tag{3.3}$$

Let $(f, w) \in F \times W$, we say f is *rejected by w at step s* if $f \in X_w^{(s)} \setminus Y_w^{(s)}$, and we say f is *rejected by w* if f is rejected by w at some step during the break-marriage procedure. Note that a firm f is rejected by all and only the workers in $X_f^{(0)} \setminus X_f^{(s^\star)}$.

In the following, we prove Theorem 3.56.

Theorem 3.56. *Let $\mu', \mu \in \mathcal{S}(\mathcal{C}_F, \mathcal{C}_W)$ be two stable matchings and assume μ' is an immediate predecessor of μ in the stable matching lattice. Pick $(f', w') \in \mu' \setminus \mu$ and let $\bar{\mu}$ be the output matching of* `break-marriage(`μ', f', w'`)`*. Then, $\bar{\mu} = \mu$.*

We start by outlining the proof steps of Theorem 3.56. We first show in Lemma 3.57 that the output matching $\bar{\mu}$ of `break-marriage(`μ', f', w'`)` is individually rational. We then show in Lemma 3.62 that under a certain condition (i.e., the break-marriage operation being *successful*), $\bar{\mu}$ is a stable matching and $\mu' \succ \bar{\mu}$. Lastly, we show that under the assumptions in the statement of Theorem 3.56, the above-mentioned condition is satisfied and $\bar{\mu} \succeq \mu$.

Lemma 3.57. *Let $\mu' \in \mathcal{S}$ be a stable matching that is not the worker-optimal stable matching μ_W and let $(f', w') \in \mu' \setminus \mu_W$. Consider the* `break-marriage(`$\mu', f', w'$`)` *procedure with output $\bar{\mu}$. Then, $\bar{\mu}$ is individually rational.*

Proof. By (3.2) and (3.3) above, for every agent $a \in F \cup W \setminus \{w'\}$, $\bar{\mu}(a) = \mathcal{C}_a(X_a^{(s^\star - 1)})$ and thus, $\mathcal{C}_a(\bar{\mu}(a)) = \mathcal{C}_a(\mathcal{C}_a(X_a^{(s^\star - 1)})) = \mathcal{C}_a(X_a^{(s^\star - 1)}) = \bar{\mu}(a)$, where the second equality is due to path-independence. For worker w', note that $X_{w'}^{(s^\star - 1)} = Y_{w'}^{(s^\star - 1)} = \bar{\mu}(w') = \mathcal{C}_{w'}(X_{w'}^{(s^\star - 1)} \cup \{f'\}) \setminus \{f'\}$, where the first equality is due to the termination criterion. Then, by the substitutability property, with $T = X_{w'}^{(s^\star-1)}$ and $S = X_{w'}^{(s^\star-1)} \cup \{f'\}$, we have that for every firm $f \in \bar{\mu}(w')$, $f \in \mathcal{C}_{w'}(X_{w'}^{(s^\star-1)})$ holds. Thus, $\bar{\mu}(w') \subseteq \mathcal{C}_{w'}(\bar{\mu}(w'))$. Since $\mathcal{C}_{w'}(X) \subseteq X$ for any X in the domain of $\mathcal{C}_{w'}$, we have $\bar{\mu}(w') = \mathcal{C}_{w'}(\bar{\mu}(w'))$. Therefore, $\bar{\mu}$ is individually rational. ■

Lemma 3.58. *Consider the* `break-marriage(`μ', f', w'`)` *procedure with output matching $\bar{\mu}$. Then, for every firm f, $\mathcal{C}_f(\bar{\mu}(f) \cup \mu'(f)) = \mu'(f)$.*

Proof. For a firm f, we have

$$
\begin{aligned}
C_f(\bar{\mu}(f) \cup \mu'(f)) &= \mathcal{C}_f(\mathcal{C}_f(X_f^{(s^\star)}) \cup \mathcal{C}_f(\bar{X}_f(\mu')) \\
&= \mathcal{C}_f(X_f^{(s^\star)} \cup \bar{X}_f(\mu')) = \mathcal{C}_f(\bar{X}_f(\mu')) = \mu'(f),
\end{aligned}
$$

where the first and last equality holds since $\mu'(f) = C_f(\bar{X}_f(\mu'))$ by Lemma 3.51 and $\bar{\mu}(f) = C_f(X_f^{(s^\star)})$ by (3.2), the second equality is by path-independence, and the third equality is due to $X_f^{(s^\star)} \subseteq X_f^{(0)} \subseteq \bar{X}_f(\mu')$ by (3.1). ■

The following two properties of the break-marriage procedure are direct consequences of the path-independence assumption imposed on choice functions. These properties are also true for the deferred acceptance algorithm, as shown in Roth (1984a). Let $f \in F$ and $w \in W$ be an arbitrary firm and worker. Lemma 3.59 states that once f proposes to w in some step of the algorithm, it will keep proposing to w in future steps until w rejects f. Lemma 3.60 states that once w rejects f, w would never accept f in later steps even if the proposal is offered again.

Lemma 3.59. *For all $s \in [s^\star - 1]$ and $w \in W$, we have $Y_w^{(s-1)} \subseteq X_w^{(s)}$.*

Proof. Let $f \in Y_w^{(s-1)}$. By construction, we have $w \in \mathcal{C}_f(X_f^{(s-1)}) \cap X_f^{(s)}$. Since $X_f^{(s)} \subseteq X_f^{(s-1)}$ by (3.1), we deduce that $w \in \mathcal{C}_f(X_f^{(s)})$ by the substitutability property. Hence, $f \in X_w^{(s)}$ by definition. ■

Lemma 3.60. *Let $s \in [s^\star - 1]$, $f \in F$, and $w \in W$. Assume $f \in X_w^{(s-1)} \setminus Y_w^{(s-1)}$, i.e., f is rejected by w at step $s-1$. If $w \neq w'$, then for every step $s' \geq s$, $f \notin \mathcal{C}_w(X_w^{(s')} \cup \{f\})$; and if $w = w'$, then for every step $s' \geq s$, $f \notin \mathcal{C}_w(X_w^{(s')} \cup \{f'\} \cup \{f\})$.*

Proof. By construction, $w \notin X_f^{(s)}$. Hence, $f \notin X_w^{(s')}$ for all $s' \geq s$ because of (3.1) and the definition of $X_w^{(s')}$. Fix a value of $s' \geq s$. First consider the case when $w \neq w'$. By repeated application of the path-independence property of $\mathcal{C}_w$ and Lemma 3.59, we have

$$\begin{aligned}
\mathcal{C}_w(X_w^{(s')} \cup \{f\}) &= \mathcal{C}_w(X_w^{(s')} \cup Y_w^{(s'-1)} \cup \{f\}) = \mathcal{C}_w(X_w^{(s')} \cup \mathcal{C}_w(Y_w^{(s'-1)} \cup \{f\})) \\
&= \mathcal{C}_w(X_w^{(s')} \cup \mathcal{C}_w(\mathcal{C}_w(X_w^{(s'-1)}) \cup \{f\})) \\
&= \mathcal{C}_w(X_w^{(s')} \cup \mathcal{C}_w(X_w^{(s'-1)} \cup \{f\})) \\
&= \cdots \\
&= \mathcal{C}_w(\underbrace{X_w^{(s')}}_{\not\ni f} \cup \underbrace{X_w^{(s'-1)}}_{\not\ni f} \cup \cdots \cup \underbrace{\mathcal{C}_w(X_w^{(s-1)} \cup \{f\})}_{=\mathcal{C}_w(X_w^{(s-1)}) = Y_w^{(s-1)} \not\ni f}).
\end{aligned}$$

Therefore, $f \notin \mathcal{C}_w(X_w^{(s')} \cup \{f\})$ as desired. We next consider the case where $w = w'$. Since $w \notin X_{f'}^{(0)}$ by construction, we have $w \notin X_{f'}^{(s-1)}$ by (3.1), which then implies $f' \notin X_w^{(s-1)}$ by definition. Thus, we have $f \neq f'$. Again, by repeated application of the path-independence property of $\mathcal{C}_w$ and Lemma 3.59, we have

$$\begin{aligned}
\mathcal{C}_w(X_w^{(s')} \cup \{f'\} \cup \{f\}) &= \mathcal{C}_w(X_w^{(s')} \cup Y_w^{(s'-1)} \cup \{f'\} \cup \{f\}) \\
&= \mathcal{C}_w(X_w^{(s')} \cup \{f'\} \cup \mathcal{C}_w(Y_w^{(s'-1)} \cup \{f'\} \cup \{f\})) \\
&= \mathcal{C}_w(X_w^{(s')} \cup \{f'\} \cup \mathcal{C}_w((\mathcal{C}_w(X_w^{(s'-1)} \cup \{f'\}) \setminus \{f'\}) \cup \{f'\} \cup \{f\})) \\
&= \mathcal{C}_w(X_w^{(s')} \cup \{f'\} \cup \mathcal{C}_w(X_w^{(s'-1)} \cup \{f'\} \cup \{f\})) \\
&= \cdots \\
&= \mathcal{C}_w(\underbrace{X_w^{(s')}}_{\not\ni f} \cup \underbrace{X_w^{(s'-1)}}_{\not\ni f} \cup \cdots \cup \{f'\} \cup \underbrace{\mathcal{C}_w(X_w^{(s-1)} \cup \{f'\} \cup \{f\})}_{=\mathcal{C}_w(X_w^{(s-1)} \cup \{f'\}) \not\ni f}).
\end{aligned}$$

Therefore, $f \notin \mathcal{C}_w(X_w^{(s')} \cup \{f'\} \cup \{f\})$ as desired in this case as well. ■

We say the procedure `break-marriage`(μ', f', w') is *successful* if $f' \notin \mathcal{C}_{w'}(X_{w'}^{(s^\star-1)} \cup \{f'\})$. We next show that when the procedure is successful, the output matching is stable.

Remark 3.61. For the SM-MODEL, McVitie and Wilson (1971) defines the break-marriage procedure `break-marriage`(μ', f', w') to be successful if w' receives a proposal from a firm that w' prefers to f'. To translate this condition the CM-QF-MODEL, we interpret it as the follows: if w' were to choose between this proposal and f', w' would not choose f'.

Lemma 3.62. *If* `break-marriage`(μ', f', w') *is successful, then the output matching $\bar{\mu}$ is stable. Moreover, $\mu' \succ \bar{\mu}$.*

Proof. Since `break-marriage`(μ', f', w') is successful, applying the consistency property with $T = X_{w'}^{(s^\star-1)}$ and $S = T \cup \{f'\}$, we have $\mathcal{C}_{w'}(X_{w'}^{(s^\star-1)} \cup \{f'\}) = \mathcal{C}_{w'}(X_{w'}^{(s^\star-1)})$ and thus, $Y_{w'}^{(s^\star-1)} = \mathcal{C}_{w'}(X_{w'}^{(s^\star-1)})$. In addition, by the termination condition, $Y_{w'}^{(s^\star-1)} = X_{w'}^{(s^\star-1)}$. Therefore, we have the following identity

$$\bar{\mu}(w') = Y_{w'}^{(s^\star-1)} = X_{w'}^{(s^\star-1)} = \mathcal{C}_{w'}(X_{w'}^{(s^\star-1)}) = \mathcal{C}_{w'}(X_{w'}^{(s^\star-1)} \cup \{f'\}), \tag{3.4}$$

which is similar to (3.3) for other workers.

Claim 3.62.1. *Let $(f, w) \in F \times W$. If f is rejected by w during the break-marriage procedure, then $f \notin \mathcal{C}_w(\bar{\mu}(w) \cup \{f\})$.*

Proof. If $w \neq w'$, then by Lemma 3.60, $f \notin \mathcal{C}_w(X_w^{(s^\star-1)} \cup \{f\}) = \mathcal{C}_w(\bar{\mu}(w) \cup \{f\})$ where the equality is due to (3.3). This is also true if $w = w'$ because again by Lemma 3.60,

$$f \notin \mathcal{C}_{w'}(X_{w'}^{(s^\star-1)} \cup \{f'\} \cup \{f\}) = \mathcal{C}_{w'}(\mathcal{C}_{w'}(X_{w'}^{(s^\star-1)} \cup \{f'\}) \cup \{f\}) = \mathcal{C}_{w'}(\bar{\mu}(w') \cup \{f\}),$$

where the first equality is by path-independence, and the second equality by (3.4). ■

Claim 3.62.2. $\mathcal{C}_w(\mu'(w) \cup \bar{\mu}(w)) = \bar{\mu}(w)$ *for all $w \in W$.*

Proof. Let $f \in \mu'(w) \setminus \bar{\mu}(w)$, and suppose first $(f, w) \neq (f', w')$. Because of Lemma 3.51 and Lemma 3.59, f must be rejected by w during the break-marriage procedure since otherwise $f \in X_w^{(s)}$ for all $s \in [s^\star] \cup \{0\}$, which in particular implies $w \in \bar{\mu}(f)$ due to (3.3). Then, by Claim 3.62.1, $f \notin \mathcal{C}_w(\bar{\mu}(w) \cup \{f\})$. Next assume $(f, w) = (f', w')$. By (3.4), we know that $X_w^{(s^\star-1)} = \bar{\mu}(w)$. Since `break-marriage(`$\mu', f', w'$`)` is successful, we have $f' \notin \mathcal{C}_w(X_w^{(s^\star-1)} \cup \{f'\}) = \mathcal{C}_w(\bar{\mu}(w) \cup \{f'\})$. We conclude that in both cases, $\mathcal{C}_w(\bar{\mu}(w) \cup \{f\}) = \mathcal{C}_w(\bar{\mu}(w))$ by consistency. Thus, we can apply Lemma 3.13 with $A_1 = \bar{\mu}(w)$ and $A_2 = \mu'(w)$ and conclude that $\mathcal{C}_w(\mu'(w) \cup \bar{\mu}(w)) = \mathcal{C}_w(\bar{\mu}(w))$. The claim then follows from Lemma 3.57. ■

Fix an acceptable firm-worker pair $(f, w) \notin \bar{\mu}$. We show that (f, w) does not block $\bar{\mu}$. Assume by contradiction that $f \in \mathcal{C}_w(\bar{\mu}(w) \cup \{f\})$ (†) and $w \in \mathcal{C}_f(\bar{\mu}(f) \cup \{w\})$ (‡). We claim that $(f, w) \notin \mu'$. If this is not the case, the consistency property of $\mathcal{C}_w$, with $S = \mu'(w) \cup \bar{\mu}(w)$ and $T = \bar{\mu}(w) \cup \{f\}$, implies $\mathcal{C}_w(\bar{\mu}(w) \cup \{f\}) = \mathcal{C}_w(\mu'(w) \cup \bar{\mu}(w)) = \bar{\mu}(w)$, where the last equality is by Claim 3.62.2. Thus, $f \notin \mathcal{C}_w(\bar{\mu}(w) \cup \{f\})$, which contradicts our assumption (†). Thus, $(f, w) \notin \mu'$. Note that in particular, $(f, w) \neq (f', w')$. By Lemma 3.15 and Claim 3.62.2, (†) implies $f \in \mathcal{C}_w(\mu'(w) \cup \{f\})$. Hence, we must have $w \notin \mathcal{C}_f(\mu'(f) \cup \{w\})$ since μ' is stable, i.e., not blocked by (f, w). This implies $\mathcal{C}_f(\mu'(f) \cup \{w\}) = \mathcal{C}_f(\mu'(f)) = \mu'(f)$ due to the consistency property or $\mathcal{C}_f$ and the fact that μ' is individually rational. Thus, $w \in \bar{X}_f(\mu') = X_f^{(0)}$ since $f \neq f'$.

Suppose first $w \notin X_f^{(s^\star)}$. Then, worker w rejected firm f during the break-marriage procedure. This implies $f \notin \mathcal{C}_w(\bar{\mu}(w) \cup \{f\})$ by Claim 3.62.1, contradicting assumption (†). Suppose next $w \in X_f^{(s^\star)}$. Since $(f, w) \notin \bar{\mu}$, we have $w \notin \bar{\mu}(f) = \mathcal{C}_f(X_f^{(s^\star)})$, where the equality is due to (3.2). Then by the consistency property, with $S = X_f^{(s^\star)}$ and $T = \bar{\mu}(f) \cup \{w\}$, we have that $w \notin \mathcal{C}_f(\bar{\mu}(f) \cup \{w\})$. However, this contradicts (‡). Therefore, $\bar{\mu}$ must be stable.

By Lemma 3.58, $\mu' \succeq \bar{\mu}$. Moreover, we have $\mu' \neq \bar{\mu}$ since $f' \in \mu'(w') \setminus \bar{\mu}(w')$. Hence, $\mu' \succ \mu$ as desired. ■

We now give the proof of Theorem 3.56.

Proof of Theorem 3.56. Note that by Lemma 3.52, $\mu(f) \subseteq \bar{X}_f(\mu')$ for every $f \in F$. We start by showing that during the break-marriage procedure, for every firm f, no worker in $\mu(f)$ rejects f. Assume by contradiction that this is not true. Let s' be the first step where such a rejection happens, with firm f_1 being rejected by worker $w_1 \in \mu(f_1)$. Hence, $f_1 \in X_{w_1}^{(s')} \setminus Y_{w_1}^{(s')}$.

Claim 3.62.3. *There exists a firm $f_2 \in Y_{w_1}^{(s')} \setminus \mu(w_1)$ such that $f_2 \in \mathcal{C}_{w_1}(\mu(w_1) \cup \{f_2\})$.*

Proof. Assume by contradiction that such a firm f_2 does not exist. We first consider the case when $w_1 \neq w'$. By Corollary 3.14 with $A_1 = \mu(w_1)$ and $A_2 = Y_{w_1}^{(s')}$, we have $\mathcal{C}_{w_1}(\mu(w_1) \cup Y_{w_1}^{(s')}) = \mathcal{C}_{w_1}(\mu(w_1)) = \mu(w_1)$, where the last equality is because μ is individually rational. Hence, $f_1 \in \mathcal{C}_{w_1}(\mu(w_1) \cup Y_{w_1}^{(s')})$, and using substitutability, we deduce $f_1 \in \mathcal{C}_{w_1}(Y_{w_1}^{(s')} \cup \{f_1\})$. However, using consistency, with $T = Y_{w_1}^{(s')} \cup \{f_1\}$ and $S = X_{w_1}^{(s')}$, we conclude $\mathcal{C}_{w_1}(Y_{w_1}^{(s')} \cup \{f_1\}) = \mathcal{C}_{w_1}(X_{w_1}^{(s')}) = Y_{w_1}^{(s')} \not\ni f_1$, a contradiction.

We next consider the case when $w_1 = w'$. Note that $f_1 \neq f'$, because $(f', w') \notin \mu$ by choice of (f', w'). Since $\mu' \succeq \mu$, by Theorem 3.16, $\mathcal{C}_{w'}(\mu'(w') \cup \mu(w')) = \mu(w')$. Thus, by the consistency property, with $S = \mu'(w') \cup \mu(w')$ and $T = \mu(w') \cup \{f'\}$, we have $\mathcal{C}_{w'}(\mu(w') \cup \{f'\}) = \mu(w') \not\ni f'$. As in the case $w_1 \neq w'$, by Corollary 3.14 with $A_1 = \mu(w')$ and $A_2 = Y_{w_1}^{(s')} \cup \{f'\}$ and the fact that μ is individually rational, $\mu(w') = \mathcal{C}_{w'}(\mu(w')) = \mathcal{C}_{w'}(\mu(w') \cup \{f'\} \cup Y_{w'}^{(s')})$. Then, since $f_1 \in \mu(w') \cap X_{w'}^{(s')}$, by substitutability and path independence, we have:

$$
\begin{aligned}
f_1 \in \mathcal{C}_{w'}(Y_{w'}^{(s')} \cup \{f'\} \cup \{f_1\}) &= \mathcal{C}_{w'}(\mathcal{C}_{w'}(X_{w'}^{(s')} \cup \{f'\}) \setminus \{f'\} \cup \{f'\} \cup \{f_1\}) \\
&= \mathcal{C}_{w'}(X_{w'}^{(s')} \cup \{f'\}).
\end{aligned}
$$

However, since $f_1 \notin Y_{w'}^{(s')}$ by our choice and $f_1 \neq f'$, we should have $f_1 \notin \mathcal{C}_{w'}(X_{w'}^{(s')} \cup \{f'\})$, which is again a contradiction. ∎

Now let f_2 be the firm whose existence is guaranteed by Claim 3.62.3. In particular, $f_2 \in Y_{w_1}^{(s')}$ implies $w_1 \in \mathcal{C}_{f_2}(X_{f_2}^{(s')}) \subseteq X_{f_2}^{(s')}$. Note that by our choice of f_1, $\mu(f_2) \subseteq X_{f_2}^{(s')}$. Therefore, by substitutability and $w_1 \in \mathcal{C}_{f_2}(X_{f_2}^{(s')})$, we have $w_1 \in \mathcal{C}_{f_2}(\mu(f_2) \cup \{w_1\})$. However, this means that (f_2, w_1) is a blocking pair of μ, which contradicts stability of μ. Thus, for every firm $f \in F$, no worker in $\mu(f)$ rejects f during the break-marriage procedure as we claimed, which, together with the fact that $\mu(f) \subseteq \bar{X}_f(\mu')$, implies $\mu(f) \subseteq X_f^{(s^\star)}$. By path-independence and (3.2), we have that for every firm f:

$$\begin{aligned} \mathcal{C}_f(\bar{\mu}(f) \cup \mu(f)) &= \mathcal{C}_f(\mathcal{C}_f(X_f^{(s^\star)}) \cup \mu(f)) = \mathcal{C}_f(X_f^{(s^\star)} \cup \mu(f)) \\ &= \mathcal{C}_f(X_f^{(s^\star)}) = \bar{\mu}(f). \end{aligned} \tag{3.5}$$

Moreover,

$$|\mu(f)| = |\mu'(f)| = |\bar{\mu}(f)|, \; \forall f \in F \tag{3.6}$$

because

$$\begin{aligned} |\mu(f)| = |\mu'(f)| = |\mathcal{C}_f(\bar{\mu}(f) \cup \mu'(f))| \geq |\mathcal{C}_f(\bar{\mu}(f))| = |\bar{\mu}(f)| \\ = |\mathcal{C}_f(\bar{\mu}(f) \cup \mu(f))| \geq |\mathcal{C}_f(\mu(f))| = |\mu(f)|, \end{aligned}$$

where the first equality is due to the equal-quota property, the second and the fourth equalities are by Lemma 3.58 and (3.5) respectively, the remaining two equalities are due to the fact that $\bar{\mu}$ and μ are individually rational, and the two inequalities hold because of cardinal monotonicity.

We next show that the break-marriage procedure is successful. Consider the following two cases for a worker $w \neq w'$. The first is when $|\mu'(w)| < q_w$. By the full-quota property, w has the same set of partners in all stable matchings. In particular, $\mu'(w) = \mu(w)$. We claim that only firms from $\mu(w)$ propose to w during the break-marriage procedure. Assume by contradiction that a firm $f \notin \mu(w)$ proposes to w at step s (i.e., $w \in \mathcal{C}_f(X_f^{(s)})$). Then, since $\bar{\mu}(f) = \mathcal{C}_f(X_f^{(s^\star)}) \subseteq X_f^{(s^\star)} \subseteq X_f^{(s)}$ due to (3.1) and (3.2), by substitutability, we have $w \in \mathcal{C}_f(\bar{\mu}(f) \cup \{w\})$ and thus, $w \in C_f(\mu(f) \cup \{w\})$ because of (3.5) and Lemma 3.15. Since $|\mu(w)| < q_w$, we also have that $f \in C_w(\mu(w) \cup \{f\})$ by the quota-filling property of $\mathcal{C}_w$. However, this contradicts stability of μ. Therefore, $Y_w^{(s)} = X_w^{(s)} = \mu'(w)$ for all $s \in \{0, 1, \cdots, s^\star\}$ by Lemma 3.51 and Lemma 3.59. Hence, $\bar{\mu}(w) = \mu'(w)$ by (3.3).

We next consider the second case for worker $w \neq w'$, which is when $|\mu'(w)| = q_w$, and we claim that $|Y_w^{(s)}| = q_w$ for all $s \in \{0\} \cup [s^\star]$. We will show this by induction. For the base case with $s = 0$, we want to show that $X_w^{(0)} \supseteq \mu'(w)$ because then we have $|X_w^{(0)}| \geq q_w$ and thus $|Y_w^{(0)}| = q_w$ by quota-filling. Let $f \in \mu'(w)$. If $f \neq f'$, then by Lemma 3.51, we have $w \in \mathcal{C}_f(X_f^{(0)})$; and if $f = f'$, by substitutability of $\mathcal{C}_{f'}$, we also have $w \in \mathcal{C}_f(X_f^{(0)})$ since $w \neq w'$. Hence, $f \in X_w^{(0)}$ by definition of $X_w^{(0)}$. For the inductive step, assume that $|Y_w^{(s-1)}| = q_w$ and we want to show that $|Y_w^{(s)}| = q_w$. Because of Lemma 3.59, $X_w^{(s)} \supseteq Y_w^{(s-1)}$. Hence, similar to the base case, we have $|X_w^{(s)}| \geq q_w$ and subsequently $|Y_w^{(s)}| = q_w$ by quota-filling. Therefore, $|\bar{\mu}(w)| = |\mu'(w)|$ by (3.3).

Combining both cases, we have $|\bar{\mu}(w)| = |\mu'(w)|$ for every worker $w \neq w'$. Together with (3.6), we have:

$$\begin{aligned} \sum_{w \in W \setminus \{w'\}} |\bar{\mu}(w)| + |\bar{\mu}(w')| &= \sum_{w \in W} |\bar{\mu}(w)| = \sum_{f \in F} |\overline{\mu}(f)| = \sum_{f \in F} |\mu(f)| \\ &= \sum_{w \in W} |\mu(w)| = \sum_{w \in W \setminus \{w'\}} |\mu(w)| + |\mu(w')|. \end{aligned}$$

Hence, we must also have $|\bar{\mu}(w')| = |\mu'(w')| = q_{w'}$. Therefore, $f' \notin \mathcal{C}_{w'}(X_{w'}^{(s^\star-1)} \cup \{f'\})$ because otherwise $|\bar{\mu}(w')| = |\mathcal{C}_{w'}(X_{w'}^{(s^\star-1)} \cup \{f'\}) \setminus \{f'\}| \leq q_{w'} - 1$, where the inequality is by quota-filling. Hence, the break-marriage procedure is successful.

Finally, by Lemma 3.62, we have $\bar{\mu} \in \mathcal{S}$ and $\mu' \succ \bar{\mu}$. We also have $\bar{\mu} \succeq \mu$ by (3.5). Therefore, it must be that $\bar{\mu} = \mu$ since μ is an immediate descendant of μ' in $\mathcal{S}$. ■

Example 3.63. Consider the following instance adapted from the one given in Martínez et al.

(2004). One can check that every choice function is quota-filling with quota 2.

$$
\begin{array}{lll}
f_1: & \geq_{f_1,1}: & w_1\ \underline{w_4}\ w_3\ w_2 \\
 & \geq_{f_1,2}: & \underline{w_2}\ w_3\ w_4\ w_1 \\
 & & \\
f_2: & \geq_{f_2,1}: & \underline{w_1}\ w_3\ w_4\ w_2 \\
 & \geq_{f_2,2}: & \underline{w_2}\ w_4\ w_3\ w_1 \\
f_3: & \geq_{f_3,1}: & \underline{w_3}\ w_1\ w_2\ w_4 \\
 & \geq_{f_3,2}: & \underline{w_4}\ w_2\ w_1\ w_3 \\
 & & \\
f_4: & \geq_{f_4,1}: & \underline{w_3}\ w_2\ w_1\ w_4 \\
 & \geq_{f_4,2}: & w_4\ \underline{w_1}\ w_2\ w_3
\end{array}
\qquad
\begin{array}{lll}
w_1: & \geq_{w_1,1}: & f_3\ \underline{f_2}\ f_1\ f_4 \\
 & \geq_{w_1,2}: & \underline{f_4}\ f_2\ f_1\ f_3 \\
 & \geq_{w_1,3}: & f_3\ \underline{f_4}\ f_1\ f_2 \\
w_2: & \geq_{w_2,1}: & f_3\ \underline{f_1}\ f_2\ f_4 \\
 & \geq_{w_2,2}: & f_4\ \underline{f_2}\ f_1\ f_3 \\
w_3: & \geq_{w_3,1}: & f_1\ \underline{f_3}\ f_4\ f_2 \\
 & \geq_{w_3,2}: & f_2\ \underline{f_3}\ f_4\ f_1 \\
 & \geq_{w_3,3}: & f_1\ f_2\ \underline{f_4}\ f_3 \\
w_4: & \geq_{w_4,1}: & \underline{f_1}\ f_3\ f_4\ f_2 \\
 & \geq_{w_4,2}: & f_2\ \underline{f_3}\ f_4\ f_1 \\
 & \geq_{w_4,3}: & \underline{f_1}\ f_2\ f_4\ f_3
\end{array}
$$

Consider the stable matching $\mu' = (\{w_2, w_4\}, \{w_1, w_2\}, \{w_3, w_4\}, \{w_1, w_3\})$, where, to be concise, we list the assigned partners of firms f_1, f_2, f_3, f_4 in the exact order. Matched pairs are underlined above. The closure of μ' is

$$\bar{X}(\mu') = \{\{w_2, w_3, w_4\}, \{w_1, w_2, w_3, w_4\}, \{w_1, w_2, w_3, w_4\}, \{w_1, w_2, w_3\}\}.$$

In the following, we describe the iterations of the `break-marriage`(μ', f_1, w_2) procedure. The rejected firms are bolded.

	$s=0$	$s=1$	$s=2$	$s=3$
$X_{f_1}^{(s)}$	$\{w_3, w_4\}$	$\{w_3, w_4\}$	$\{w_3, w_4\}$	$\{w_3, w_4\}$
$X_{f_2}^{(s)}$	$\{w_1, w_2, w_3, w_4\}$	$\{w_1, w_2, w_3, w_4\}$	$\{w_1, w_3, w_4\}$	$\{w_1, w_3, w_4\}$
$X_{f_3}^{(s)}$	$\{w_1, w_2, w_3, w_4\}$	$\{w_1, w_2, w_3, w_4\}$	$\{w_1, w_2, w_3, w_4\}$	$\{w_1, w_2, w_3\}$
$X_{f_4}^{(s)}$	$\{w_1, w_2, w_3\}$	$\{w_1, w_2\}$	$\{w_1, w_2\}$	$\{w_1, w_2\}$
$X_{w_1}^{(s)}$	$\{f_2, f_4\}$	$\{f_2, f_4\}$	$\{f_2, f_4\}$	$\{f_2, f_4\}$
$X_{w_2}^{(s)}$	$\{f_2\}$	$\{\boldsymbol{f_2}, f_4\}$	$\{f_4\}$	$\{f_3, f_4\}$
$X_{w_3}^{(s)}$	$\{f_1, f_3, \boldsymbol{f_4}\}$	$\{f_1, f_3\}$	$\{f_1, f_3\}$	$\{f_1, f_3\}$
$X_{w_4}^{(s)}$	$\{f_1, f_3\}$	$\{f_1, f_3\}$	$\{f_1, f_2, \boldsymbol{f_3}\}$	$\{f_1, f_2\}$
$Y_{w_1}^{(s)}$	$\{f_2, f_4\}$	$\{f_2, f_4\}$	$\{f_2, f_4\}$	$\{f_2, f_4\}$
$Y_{w_2}^{(s)}$	$\{f_2\}$	$\boxed{\{f_4\}}$	$\{f_4\}$	$\{f_3, f_4\}$
$Y_{w_3}^{(s)}$	$\{f_1, f_3\}$	$\{f_1, f_3\}$	$\{f_1, f_3\}$	$\{f_1, f_3\}$
$Y_{w_4}^{(s)}$	$\{f_1, f_3\}$	$\{f_1, f_3\}$	$\{f_1, f_2\}$	$\{f_1, f_2\}$

The output matching is $\bar{\mu} = (\{w_3, w_4\}, \{w_1, w_4\}, \{w_2, w_3\}, \{w_1, w_2\})$, which, one can check, is stable. Note the step highlighted in box above where $Y_{w_2}^{(1)} = \mathcal{C}_{w_2}(\{f_2, f_4\} \cup \{f_1\}) \setminus \{f_1\} = \{f_1, f_4\} \setminus \{f_1\} = \{f_4\}$. If instead, w_2 used the original (i.e., same as in the DA algorithm) acceptance rule and accepted both f_2 and f_4, the algorithm would prematurely stopped after $s = 1$, without leading to a stable matching. $\triangle$

We are now ready to present the algorithm that finds an immediate descendant for any given stable matching, using the break-marriage procedure. The details of the algorithm are presented in Algorithm 3.3.

Theorem 3.64. *The output μ^* of Algorithm 3.3 is an immediate descendant of μ' in the stable matching lattice $(\mathcal{S}, \succeq)$.*

Algorithm 3.3 Immediate descendant of stable matching $\mu' \neq \mu_W$

Input: μ', μ_W

initialize $\mathcal{T} \leftarrow \emptyset$
for each $(f', w') \in \mu' \setminus \mu_W$ **do**
  run the `break-marriage`(μ', f', w') procedure
  if the procedure is successful **then** add the output matching $\bar{\mu}$ to $\mathcal{T}$
end for
let μ^* be a matching in $\mathcal{T}$
for each $\mu \in \mathcal{T} \setminus \{\mu^*\}$ **do**
  if $\mu \succeq \mu^*$ **then** update $\mu^* \leftarrow \mu$
end for ▷ μ^* is a maximal matching from $\mathcal{T}$

Output: μ^*

Proof. First note that due to Lemma 3.62, all matchings in the set $\mathcal{T}$ constructed by Algorithm 3.3 are stable matchings and $\mu' \succeq \mu$ for all $\mu \in \mathcal{T}$. Moreover, we claim that $\mathcal{T} \neq \emptyset$. Let $\mu_1 \in \mathcal{S}$ such that μ' is an immediate predecessor of μ_1 in $(\mathcal{S}, \succeq^\star)$. Such a stable matching μ_1 exists because $\mu' \neq \mu_W$. Because of Lemma 3.40, we have $\mu' \setminus \mu_1 \subseteq \mu' \setminus \mu_W$ and thus by Theorem 3.56, we have $\mu_1 \in \mathcal{T}$. Hence, $\mathcal{T} \neq \emptyset$ as desired. Now, to prove the theorem, assume by contradiction that the output matching μ^* is not an immediate descendant of μ' in $(\mathcal{S}, \succeq)$. Then, there exists a stable matching μ such that $\mu' \succ \mu \succ \mu^*$. By Lemma 3.40, for every firm-worker pair $(f', w') \in \mu' \setminus \mu$, we also have $(f', w') \notin \mu_W$. Thus, $\mu \in \mathcal{T}$ due to Theorem 3.56. However, this means that μ^* is not a maximal matching from $\mathcal{T}$, which is a contradiction. ■

Finally, putting everything together, Algorithm 3.4 finds a maximal chain of the stable matching lattice, as well as the set of rotations. Its correctness follows from Theorem 3.64, Theorem 3.55, and Theorem 3.37.

3.4.3 Finding irreducible elements via maximal chains

The goal of this section is to prove the following. Note that the result below holds for any ring of sets.

Algorithm 3.4 A maximal chain of $(\mathcal{S}, \succeq)$ and the set of rotations Π

Input: μ_F and μ_W

initialize counter $k \leftarrow 0$ and $C_k \leftarrow \mu_F$
while $C_k \neq \mu_W$ **do**
    run Algorithm 3.3 with input C_k and μ_W, and let μ^* be its output
    update counter $k \leftarrow k+1$ and $C_k \leftarrow \mu^*$
end while

Output: maximal chain $C_0, C_1, \cdots, C_k$; and $\Pi = \{\rho_i \coloneqq \rho(C_{i-1}, C_i) : i \in [k]\}$.

Theorem 3.65. *Consider a ring of sets $(\mathcal{H}, \subseteq)$ with base set B. Let $C_0, C_1, \cdots, C_k$ be a maximal chain of $(\mathcal{H}, \subseteq)$ and let $K_i \coloneqq C_i \setminus C_{i-1}$ for all $i \in [k]$. For $H \subseteq B$, let* `ros-membership` *denote the running time of an algorithm that decides if $H \in \mathcal{H}$. There exists an algorithm with running time $O(k^2\texttt{ros-membership})$ that takes C_0, C_1, $\cdots$, C_k as input and outputs, for each minimal difference K_i, a set of indices $\Lambda(K_i)$ such that $I(K_i) = \bigcup\{K_j : j \in \Lambda(K_i)\} \cup C_0$. In particular, this algorithm can be used to obtain the partial order $\sqsupseteq$ over $\mathcal{D}(\mathcal{H})$.*

We start with the theorem below, which gives an alternative definition of the partial order $\sqsupseteq$.

Theorem 3.66 (Theorem 2.4.4, Gusfield and Irving, 1989)**.** *Let $K_1, K_2 \in \mathcal{D}(\mathcal{H})$. Then, $K_1 \sqsupseteq K_2$ if and only if K_1 appears before K_2 on every maximal chain in $(\mathcal{H}, \subseteq)$.*

We now present the algorithm stated in Theorem 3.65 in Algorithm 3.5. The idea is as follows. In order to find $I(K_i)$ (i.e., the minimal element in $\mathcal{H}$ that contains K_i), the algorithm tries to remove from the set C_i as many items as possible, while keeping $C_i \in \mathcal{H}$. That is, the algorithm removes from C_i all minimal differences $K \in \{K_1, K_2, \cdots, K_i\}$ such that $K \not\sqsupseteq K_i$. As we show in the proof of Theorem 3.65, the resulting set is $I(K_i)$. A demonstration of this algorithm is given in Example 3.67 on the ring of sets from Example 3.32.

Example 3.67. Consider the ring of sets given in Example 3.32, and assume Algorithm 3.5 takes in the maximal chain $C_0 = H_1$, $C_1 = H_2$, $C_2 = H_4$, $C_3 = H_6$, $C_4 = H_7$. Then, $K_1 = \{b\}$, $K_2 = \{c\}$, $K_3 = \{d, e\}$ and $K_4 = \{f\}$. Now, image we would like to obtain $\Lambda(K_3)$. From Figure 3.3a, it is clear that $I(K_3) = H_5$ and thus, $\Lambda(K_3) = \{2, 3\}$. During Algorithm 3.5, at the

Algorithm 3.5

Input: A maximal chain $C_0, C_1, \cdots, C_k$ of $(\mathcal{H}, \subseteq)$.

```
1: for i = 1, 2, ..., k do
2:     define K_i ← C_i \ C_{i-1}
3:     initialize H ← C_i and Λ(K_i) ← {1, 2, ..., i}
4:     for j = i-1, i-2, ..., 1 do
5:         if H \ K_j ∈ 𝓗 then
6:             update H ← H \ K_j and Λ(K_i) ← Λ(K_i) \ {j}
7:         end if
8:     end for
9: end for
```

Output: $\Lambda(K_i)$ for all $i \in [k]$

outer **for** loop with $i = 3$, H is initialized to be $C_3 = H_6$. In the first iteration of the inner **for** loop, since $H_6 \setminus K_2 = \{a, b, d, e\} \notin \mathcal{H}$, $\Lambda(K_3)$ remains $\{1, 2, 3\}$. Next, H is updated to be $H_6 \setminus K_1 = H_5$ and $\Lambda(K_3)$ is updated to be $\{2, 3\}$. The output is as expected. $\triangle$

We now give the proof of Theorem 3.65.

Proof of Theorem 3.65. It is clear that the running time of Algorithm 3.5 is $O(k^2 \texttt{ros-membership})$. Suppose first the output of Algorithm 3.5 is correct, that is, $I(K_i) = \bigcup\{K_j : j \in \Lambda(K_i)\} \cup C_0$. Then, for two minimal differences $K_{i_1}, K_{i_2} \in \mathcal{D}(\mathcal{H})$, $K_{i_1} \sqsupseteq K_{i_2}$ if and only if $\Lambda(K_{i_1}) \subseteq \Lambda(K_{i_2})$ by definition of $\sqsupseteq$. Hence, the partial order $\sqsupseteq$ can be obtained in time $O(k^2)$ from the output of Algorithm 3.5. It remains to show the correctness of Algorithm 3.5. Fix a value of $i \in [k]$ and for the following, consider the i^{th} iteration of the outer **for** loop of the algorithm. Let $\{j_1, j_2, \cdots, j_M\}$ be an enumeration of $\Lambda(K_i)$ at the end of the iteration such that $j_1 < j_2 < \cdots < j_M$. Note that $j_M = i$. We start by showing the following claim.

Claim 3.67.1. *For all $m \in [M-1]$, $K_{j_m} \sqsupseteq K_i$.*

Proof. We prove this by induction on m, where the base case is $m = M - 1$. We start with the base case. Note that j_m is the first index for which the **if** statement at Line 5 is evaluated to be false. That is, $(\bigcup_{\ell=1}^{j_m} K_\ell) \cup K_i \cup C_0 \in \mathcal{H}$ but $(\bigcup_{\ell=1}^{j_m - 1} K_\ell) \cup K_i \cup C_0 \notin \mathcal{H}$. By

Lemma 3.31, we have $\{K_1, K_2, \cdots, K_{j_m}, K_i\}$ is an upper set of $(\mathcal{D}(\mathcal{H}), \sqsupseteq)$, and by Theorem 3.29, we have $\{K_1, K_2, \cdots, K_{j_m-1}, K_i\}$ is not an upper set of $(\mathcal{D}(\mathcal{H}), \sqsupseteq)$. Since for all $j' < j_m$, $K_{j_m} \not\sqsupseteq K_{j'}$ because of Theorem 3.66, the reason why $\{K_1, K_2, \cdots, K_{j_m-1}, K_i\}$ is not an upper set of $(\mathcal{D}(\mathcal{H}), \sqsupseteq)$ must be that $K_{j_m} \sqsupseteq K_i$. For the inductive step, assume the claim is true for all $m' > m$ and we want to show that $K_{j_m} \sqsupseteq K_i$. Note that again by Theorem 3.29, we have that the set $\{K_1, K_2, \cdots, K_{j_m}, K_{j_{m+1}}, K_{j_{m+2}}, \cdots, K_i\}$ is an upper set of $(\mathcal{D}(\mathcal{H}), \sqsupseteq)$, but the set $\{K_1, K_2, \cdots, K_{j_m-1}, K_{j_{m+1}}, K_{j_{m+2}}, \cdots, K_i\}$ is not an upper set of $(\mathcal{D}(\mathcal{H}), \sqsupseteq)$. With the same argument as in the base case, since for all $j' < j_m$, $K_{j_m} \not\sqsupseteq K_{j'}$ by Theorem 3.66, it must be that $K_{j_m} \sqsupseteq K_{j_{m'}}$ for some $m' > m$. Therefore, applying the inductive hypothesis, we have $K_{j_m} \sqsupseteq K_i$ as desired. ■

Let H^* be set H at the end of i^{th} iteration of the outer **for** loop. Note that $H^* = \bigcup\{K_j : j \in \Lambda(K_i)\} \cup C_0$ by construction. Since $K_i \subseteq H^*$, we have $I(K_i) \subseteq C_i \subseteq H^*$ by definition. Also note that by definition, $I(K_i) \in \mathcal{H}$. Assume by contradiction that $H^* \neq I(K_i)$ (i.e., $H^* \not\subseteq I(K_i)$). Consider a complete chain from the minimal element H_0 of $(\mathcal{H}, \subseteq)$ to $I(K_i)$ in $(\mathcal{H}, \subseteq)$, whose existence is guaranteed by Theorem 3.55. Then, at least one minimal difference from $\{K_j : j \in \Lambda(K_i) \setminus \{i\}\}$, call it K', is not contained in this complete chain. However, this means $K' \not\sqsupseteq K_i$ due to Theorem 3.55, which contradicts Claim 3.67.1. Therefore, we must have $I(K_i) = H^*$. ■

3.4.4 Partial order $\succeq^\star$ over Π

In this section, we show how to obtain the partial order $\succeq^\star$ over the rotation poset Π. Recall that as stated in Theorem 3.65 of the previous section, there exists an algorithm that finds the partial order $\sqsupseteq$ over $\mathcal{D} := \mathcal{D}(\mathcal{P})$ when given as input a maximal chain of $\mathcal{P}$. Employing the isomorphism between $\mathcal{S}$ and $\mathcal{P}$ shown in Theorem 3.19 and that between $\mathcal{D}$ and Π shown in Theorem 3.37, we adapt the algorithm so that from a maximal chain of $\mathcal{S}$, we obtain the partial order $\succeq^\star$ over Π.

Theorem 3.68. *Let $\Lambda(\rho)$ and $\Lambda(\rho')$ be the outputs of Algorithm 3.6 for rotations $\rho, \rho' \in \Pi$, respectively. Then, $\rho \succeq^\star \rho'$ if and only if $\Lambda(\rho) \subseteq \Lambda(\rho')$.*

Algorithm 3.6

Input: outputs of Algorithm 3.4 – maximal chain $C_0, \cdots, C_k$ of $(\mathcal{S}, \succeq)$ and the set of rotations $\Pi = \{\rho_i \coloneqq \rho(C_{i-1}, C_i) : i \in [k]\}$

for $i = 1, 2, \cdots, k$ **do**
  initialize $\mu \leftarrow C_i$ and $\Lambda(\rho_i) \leftarrow \{1, 2, \cdots, i\}$
  for $j = i-1, i-2, \cdots, 1$ **do**
    if $\mu \triangle \rho_j^- \triangle \rho_j^+ \in \mathcal{S}$ **then**
      update $\mu \leftarrow \mu \triangle \rho_j^- \triangle \rho_j^+$ and $\Lambda(\rho_i) \leftarrow \Lambda(\rho_i) \setminus \{j\}$
    end if
  end for
end for

Output: $\Lambda(\rho_i)$ for all $i \in [k]$

Proof. To distinguish between the inputs of Algorithm 3.6 and Algorithm 3.5, we let $\mu_0, \mu_1, \cdots, \mu_k$ denote the maximal chain in the input of Algorithm 3.6. Consider the outputs of Algorithm 3.5 with inputs $C_i = P(\mu_i)$ for all $i \in [k] \cup \{0\}$. Then, because of the isomorphism between $(\mathcal{S}, \succeq)$ and $(\mathcal{P}, \subseteq)$ and the isomorphism between $(\Pi, \succeq^\star)$ and $(\mathcal{D}, \sqsupseteq)$ as respectively stated in Theorem 3.19 and Theorem 3.37, $K_i = Q(\rho_i)$ and $\Lambda(\rho_i) = \Lambda(K_i)$ for all $i \in [k]$, where $K_i = C_i \setminus C_{i-1}$ as defined in Algorithm 3.5. Thus, together with Theorem 3.65,

$$\rho \succeq^\star \rho' \Leftrightarrow Q(\rho) \sqsupseteq Q(\rho') \Leftrightarrow \Lambda(Q(\rho)) \subseteq \Lambda(Q(\rho')) \Leftrightarrow \Lambda(\rho) \subseteq \Lambda(\rho'),$$

concluding the proof. ■

Example 3.69. Consider the instance given in Example 3.46 and assume the maximal chain we obtained is $C_0 = \mu_F, C_1 = \mu_2, C_2 = \mu_3, C_3 = \mu_W$ so that we exactly have $\rho_i = \rho(C_{i-1}, C_i)$ for all $i \in [3]$ as denoted in Example 3.46. Imagine we want to compare ρ_1 and ρ_2. As shown in Figure 3.5b, $\rho_1 \succeq^\star \rho_2$. First, consider the outer **for** loop of Algorithm 3.6 with $i = 1$. Then the body of the inner **for** loop is not executed and immediately we have $\Lambda(\rho_1) = \{1\}$. Next, consider the outer **for** loop of Algorithm 3.6 with $i = 2$. Then, μ is initialized to be μ_3 and $\Lambda(\rho_2)$ is initialized to be $\{1, 2\}$. During the first and only iteration of the inner **for** loop, since $\mu_3 \triangle \rho_1^- \triangle \rho_1^+$

is not a stable matching, $\Lambda(\rho_2)$ is not updated and remains $\{1, 2\}$. Finally, $\Lambda(\rho_1) \subseteq \Lambda(\rho_2)$ and thus, $\rho_1 \succeq^\star \rho_2$ as expected. $\triangle$

3.4.5 Summary and time complexity analysis

The complete procedure to build the rotation poset is summarized in Algorithm 3.7.

Algorithm 3.7 Construction of the rotation poset $(\Pi, \succeq^\star)$

1: Run Algorithm 3.1's, firm-proposing and worker-proposing, to obtain μ_F and μ_W.
2: Run Algorithm 3.4 to obtain a maximal chain $C_0, C_1, \cdots, C_k$ of the stable matching lattice $(\mathcal{S}, \succeq)$, and the set of rotations $\Pi \equiv \{\rho_1, \rho_2, \cdots, \rho_k\}$.
3: Run Algorithm 3.6 to obtain the sets $\Lambda(\rho_i)$ for each rotation $\rho_i \in \Pi$.
4: Define the partial order relation $\succeq^\star$: for $\rho_i, \rho_j \in \Pi$, $\rho_i \succeq^\star \rho_j \Leftrightarrow \Lambda(\rho_i) \subseteq \Lambda(\rho_j)$.

The rest of the section focuses on time complexity analysis.

Theorem 3.70. *Algorithm 3.7 runs in time* $|W|^3|F|^3$`oracle-call`.

DA algorithm (Algorithm 3.1). Because of Lemma 3.59 and Lemma 3.60, Algorithm 3.1 can be implemented as in Algorithm 3.8 to reduce the number of `oracle-calls`. In particular, during each **repeat** loop, only firms that are rejected in the previous step (i.e., in $\bar{F}$) and only workers who receive new proposals (i.e., in $\bar{W}$) need to invoke their choice functions. Therefore, the **for** loop at Line 5 is entered at most $|F||W|$ times, and similarly, the **for** loop at Line 13 is entered at most $|F||W|$ times. That is, the total number of `oracle-calls` is $O(|F||W|)$. Moreover, and for each firm-worker pair (f, w), w is removed from X_f at most once and f is added to X_w at most once. That is, Line 8 (resp. Line 16) is repeated at most $|F||W|$ times. Therefore, the running time of the DA algorithm is $O(|F||W|$`oracle-call`$)$.

Break-marriage procedure (Algorithm 3.2). Since the core steps (i.e., the loops) of the break-marriage procedure is the same as that of the DA algorithm, the running time of the break-marriage procedure is $O(|F||W|$`oracle-call`$)$, with the same arguments as above.

Algorithm 3.8 Efficient implementation of Algorithm 3.1

set $\bar{F} \leftarrow F$ and $\bar{W} \leftarrow \emptyset$
for each firm f **do** initialize $X_f \leftarrow W(f)$ and $Y_f^{\text{prev}} \leftarrow \emptyset$ **end for**
for each worker w **do** initialize $X_w \leftarrow \emptyset$ and $Y_w^{\text{prev}} \leftarrow \emptyset$ **end for**
repeat
  for each firm $f \in \bar{F}$ **do**
    $A_f \leftarrow \mathcal{C}_f(X_f)$
    for each worker $w \in A_f \setminus Y_f^{\text{prev}}$ **do**
      update $X_w \leftarrow X_w \cup \{f\}$ and $\bar{W} \leftarrow \bar{W} \cup \{w\}$
    end for
    update $Y_f^{\text{prev}} \leftarrow A_f$
  end for
  re-set $\bar{F} \leftarrow \emptyset$
  for each worker $w \in \bar{W}$ **do**
    $X_w \leftarrow \mathcal{C}(X_w)$
    for each firm $f \in Y_w^{\text{prev}} \setminus X_w$ **do**
      update $X_f \leftarrow X_f \setminus \{w\}$ and $\bar{F} \leftarrow \bar{F} \cup \{f\}$
    end for
    update $Y_w^{\text{prev}} \leftarrow X_w$
  end for
  re-set $\bar{W} \leftarrow \emptyset$
until $\bar{F} = \emptyset$

Output: matching $\bar{\mu}$ with $\bar{\mu}(w) = Y_w^{\text{prev}}$ for every worker w; closure $\tilde{X}(\bar{\mu})$ with $\tilde{X}_f(\bar{\mu}) = X_f$ for every firm f

Immediate descendant (Algorithm 3.3). Recall that $\bar{q}_f$ denotes the number of workers matched to firm f under any stable matching (see the equal-quota property). Let $\Upsilon := \sum_{f \in F} \bar{q}_f$ denote the number of worker-firm pairs in any stable matching. Then, Algorithm 3.2 is run for at most Υ times. In addition, finding one maximal element μ^* from $\mathcal{T}$ requires at most Υ comparisons of pairs of stable matchings, each of which requires $|F|$ `oracle-calls` by Part (3) of Lemma 3.53. All together, since $\Upsilon \leq |F||W|$, the running time of Algorithm 3.3 is $O(|F|^2|W|^2\texttt{oracle-call})$.

Maximal chain (Algorithm 3.4). Since the length of a maximal chain of $\mathcal{P}$, and equivalently of $\mathcal{S}$ due to Theorem 3.19, is at most the size of its base set due to Lemma 3.28 and Theorem 3.55, Algorithm 3.3 is repeated for at most $|F||W|$ times. Thus, the running time of Algorithm 3.4 is $O(|F|^3|W|^3\texttt{oracle-call})$.

Partial order $\succeq^\star$ (Algorithm 3.6). Recall that checking if a matching is stable requires $O(|F||W|)$ `oracle-calls` by Part (2) of Lemma 3.53. Thus, `ros-membership` is $O(|F||W|\texttt{oracle-call})$. Since k is at most $|F||W|$, the running time of Algorithm 3.6 is $O(|F|^3|W|^3\texttt{oracle-call})$.

Rotation poset $(\Pi, \succeq^\star)$ (Algorithm 3.7). Summing up the time of running Algorithm 3.1 twice, then the time of running Algorithm 3.4 and Algorithm 3.6, the time complexity for building $(\Pi, \succeq^\star)$ is $O(|F|^3|W|^3\texttt{oracle-call})$.

3.5 The convex hull of lattice elements

Consider a poset $(Y, \succeq^\star)$. Its associated *order polytope* is defined as

$$\mathcal{O}(Y, \succeq^\star) := \{y \in [0,1]^Y : y_i \geq y_j,\ \forall i, j \in Y \text{ s.t. } i \succeq^\star j\}.$$

A characterization of vertices and facets of $\mathcal{O}(X, \succeq^\star)$ is given in Stanley (1986).

Theorem 3.71 (Stanley, 1986). *The vertices of $\mathcal{O}(Y, \succeq^\star)$ are the characteristic vectors of upper sets of Y. The facets of $\mathcal{O}(Y, \succeq^\star)$ are all and only the following: $y_i \geq 0$ if i is a minimal element of the poset; $y_i \leq 1$ if i is a maximal element of the poset; $y_i \geq y_j$ if i is an immediate predecessor of j.*

Proof of Theorem 3.6. Let $(Y, \succeq^\star)$ affinely represent $(X, \succeq)$ via functions ψ and $g(u) = Au + x^0$. We claim that

$$\begin{aligned} \text{conv}(\mathcal{X}) := \text{conv}(\{\chi^\mu : \mu \in X\}) &= \{x^0\} \oplus A \cdot \mathcal{O}(Y, \succeq^\star) \\ &= \{x \in \mathbb{R}^X : x = x^0 + Ay, y \in \mathcal{O}(Y, \succeq^\star)\}, \end{aligned} \tag{3.7}$$

where $\oplus$ denotes the Minkowski sum operator. Indeed, by definition of affine representation and the fact that both polytopes, conv$(\mathcal{X})$ and $\mathcal{O}(Y, \succeq^\star)$, have $0/1$ vertices, g defines a bijection between vertices of these two polytopes. Convexity then implies (3.7). As $\mathcal{O}(Y, \succeq^\star)$ has $O(|Y|^2)$ facets shown in Theorem 3.71, we conclude the first statement from Theorem 3.6.

Now suppose that A has full column rank. This implies that conv($\mathcal{X}$) is affinely isomorphic to $\mathcal{O}(Y, \succeq^\star)$. Hence, there is a one-to-one correspondence between facets of $\mathcal{O}(Y, \succeq^\star)$ and facets of conv($\mathcal{X}$), concluding the proof. ■

Following the proof of Theorem 3.6, when a poset $\mathcal{B} = (Y, \succeq^\star)$ affinely represent a lattice $\mathcal{L} = (\mathcal{X}, \succeq^*)$ via a function $g(u) = Au + x^0$, with A having full column rank, many properties of conv($\mathcal{X}$) can be derived from the analogous properties of $\mathcal{O}(Y, \succeq^*)$. For instance, the following immediately follows from the fact that $\mathcal{O}(Y, \succeq^*)$ is full-dimensional.

Corollary 3.72. *Let $\mathcal{B} = (Y, \succeq^\star)$ affinely represent the lattice $\mathcal{L} = (\mathcal{X}, \succeq)$ via functions ψ and $g(u) = Au + x^0$, with A having full column rank. Then the dimension of* conv($\mathcal{X}$) *is equal to the number of elements in $\mathcal{B}$.*

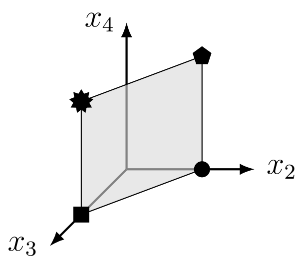
Example 3.73 shows that statements above need not hold when A does not have full column-rank.

Example 3.73. Consider the lattice $(\mathcal{X}, \succeq)$ and its representation poset $(Y, \succeq^\star)$ from Example 3.8. Note that

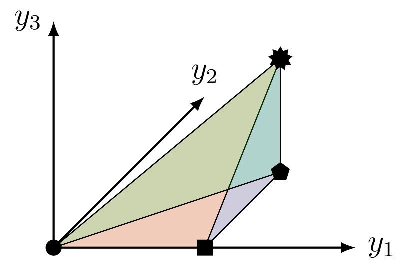
$$\text{conv}(\mathcal{X}) = \{x \in [0,1]^4 : x_1 = 1,\ x_2 + x_3 = 1\}.$$

Thus, conv($\mathcal{X}$) has dimension 2. On the other hand, $\mathcal{O}(Y, \succeq^\star)$ has dimension 3. So the two polytopes are not affinely isomorphic. Polytopes conv($\mathcal{X}$) and $\mathcal{O}(Y, \succeq^\star)$ are shown in Figure 3.6.

More generally, one can easily construct a "trivial" distributive lattice $(\mathcal{X}, \succeq)$ such that the number of facets of $\mathcal{O}(Y, \succeq^\star)$ gives no useful information on the number of facets of conv($\mathcal{X}$), where $(Y, \succeq^\star)$ is a poset that affinely represents $(\mathcal{X}, \succeq)$. In fact, the vertices of any $0/1$ polytope can be arbitrarily arranged in a chain to form a distributive lattice $(\mathcal{X}, \succeq)$. A poset $\mathcal{O}(Y, \succeq^\star)$ that affinely represents $(\mathcal{X}, \succeq)$ is given by a chain with $|Y| = |\mathcal{X}| - 1$. It is easy to see that $\mathcal{O}(Y, \succeq^\star)$ is a simplex and has therefore $|Y| + 1 = |\mathcal{X}|$ facets. However, conv($\mathcal{X}$) could have much more (or much less) facets than the number of its vertices. △



(a) $\text{conv}(\mathcal{X})$ in the space of $x_1 = 1$



(b) $\mathcal{O}(Y, \succeq^\star)$

Figure 3.6: Polytopes for Example 3.73.

When A has full column rank, we can also deduce a *minimal* (i.e., complete and irredundant) description of $\text{conv}(\mathcal{X})$ from a minimal description of the order polytope, using basic linear algebra facts.

Theorem 3.74. *Let $P = \{y \in \mathbb{R}^n : c_i^\intercal y \geq \delta_i,\ \forall i = 1, 2, \cdots, I\} \subseteq \mathbb{R}^n$ be a full-dimensional polytope. Let $A \in \mathbb{R}^{m \times n}$ be a matrix with full column rank and let $a_1, a_2, \cdots, a_{m-n} \in \mathbb{R}^m$ be $m-n$ linearly independent vectors that span the left null space of matrix A. Moreover, let $d_i \in \mathbb{R}^m$ be a vector such that $d_i^\intercal A = c_i^\intercal$ for all $i = 1, 2, \cdots, I$. Then, for any $x^0 \in \mathbb{R}^m$, $Q \coloneqq \{x^0\} \oplus A \cdot P$ can be described as*

$$\begin{aligned} \{x \in \mathbb{R}^m :\ \ & d_i^\intercal (x - x^0) \geq \delta_i \quad \forall i = 1, 2, \cdots, I \\ & a_i^\intercal (x - x^0) = 0 \quad \forall i = 1, 2, \cdots, m-n\}. \end{aligned} \tag{3.8}$$

Moreover, if the description of P is irredundant, then the description of Q in (3.8) *is also irredundant.*

Proof. We first show that (in)equalities in the description (3.8) are valid for Q. Let $\bar{x} \in Q$ and assume that $\bar{x} = A\bar{y} + x^0$ for some $\bar{y} \in P$. Then, for all $i = 1, 2, \cdots, I$,

$$d_i^\intercal (\bar{x} - x^0) = d_i^\intercal (A\bar{y} + x^0 - x^0) = d_i^\intercal A\bar{y} = c_i^\intercal \bar{y} \geq \delta;$$

and for all $i = 1, 2, \cdots, n-m$,

$$a_i^\intercal(\bar{x} - x^0) = a_i^\intercal(A\bar{y} + x^0 - x^0) = a_i^\intercal A\bar{y} = \mathbf{0}^\intercal \bar{y} = 0.$$

We next show the reverse. Assume $\bar{x} \in \mathbb{R}^m$ is valid for (3.8) and we want to show that $\bar{x} \in Q$. The fact that $\bar{x}$ satisfies all the equalities in (3.8) implies that there exists $\bar{y} \in \mathbb{R}^n$ such that $\bar{x} = A\bar{y} + x^0$. It then suffices to show that $\bar{y} \in P$. Assume not, then for some $i \in [I]$, we have $c_i^\intercal \bar{y} < \delta_i$. This then implies that $d_i^\intercal(\bar{x} - x^0) = c_i^\intercal \bar{y} < \delta$, contradicting the assumption that $\bar{x}$ is valid for (3.8).

The last statement of the theorem for irredudance follows from the assumption that matrix A has full column rank. ■

Note that the vectors a_i's and d_i's in the statement of Theorem 3.74 can be computed using known algorithms such as *Gaussian elimination*. We show next how to we deduce a (known) minimal description of the stable matching polytope in the SM-MODEL.

3.5.1 Minimal description of the stable matching polytope in the SM-MODEL

In the following, we assume that the choice function $\mathcal{C}_a$ of every agent $a \in F \cup W$ comes from an underlying strict preference list, denoted as $\geq_a$. That is, for every subset of acceptable partners S, we have $\mathcal{C}_a(S) = \max(S, \geq_a)$.

Let E_s denote the set of stable pairs, and let $\bar{E}_s \subset E_s$ be any subset of stable pairs such that exactly one firm-worker pair from ρ^- for each $\rho \in \Pi$ is included. Note that $\bar{E}_s$ is not unique and $|\bar{E}_s| = |\Pi|$, since for any two rotations $\rho_1, \rho_2 \in \Pi$, $\rho_1^- \cap \rho_2^- = \emptyset$.

Let $\Pi_0 \subseteq \Pi$ (resp. $\Pi_z \subseteq \Pi$) be all the rotations that have no predecessor (resp. successor) in $(\Pi, \succeq^\star)$. Let E_0 (resp. E_z) be a family of firm-worker pairs such that it contains exactly one firm-worker pair from ρ^- (resp. ρ^+) for each $\rho \in \Pi_0$ (resp. $\rho \in \Pi_z$), and no other firm-worker pairs.

Let Γ be a collection of pairs of rotations (ρ_i, ρ_j) such that ρ_i is an immediate predecessor of ρ_j in $(\Pi, \succeq^\star)$. The following claim (Lemma 3.75) regarding the precedence relations in Γ was shown in e.g., Gusfield and Irving (1989) and Eirinakis et al. (2014).

Lemma 3.75. *Consider $(\rho_i, \rho_j) \in \Gamma$. Then, one of the following two situations must be true.*

(i) There is a firm-worker pair $(f, w) \in E_s$ such that $(f, w) \in \rho_i^+ \cap \rho_j^-$

(ii) There is a firm-worker pair $(f, w) \notin E_s$ such that the following two conditions hold: (i) for some firm f' and f'' with $f' >_w f >_w f''$, $(f', w) \in \rho_i^+$ and $(f'', w) \in \rho_i^-$; (ii) for some worker w' and w'' with $w' >_f w >_f w''$, $(f, w') \in \rho_j^-$ and $(f, w'') \in \rho_j^+$.

Thus, we can partition Γ into two groups, Γ_1 and Γ_2, so that for each pair $(\rho_i, \rho_j) \in \Gamma_1$, there is a stable pair $e \in E_s$ such that $e \in \rho_i^+ \cap \rho_j^-$, but for each $(\rho_i, \rho_j) \in \Gamma_2$, such a stable pair does not exist. That is, Γ_1 is a collection of immediate precedence relations that can be defined by stable pairs, and Γ_2 is a collection of immediate precedence relations that can only be defined by non-stable pairs.

For every $(\rho_i, \rho_j) \in \Gamma_1$, let e_{ρ_i,ρ_j} be one of the stable firm-worker pairs in $\rho_i^+ \cap \rho_j^-$. For every $(\rho_i, \rho_j) \in \Gamma_2$, let e_{ρ_i,ρ_j} be the non-stable firm-worker pair which satisfies the condition in Lemma 3.75 (ii). In particular, e_{ρ_i,ρ_j} represents the firm-worker pair that results in the precedence relation $\rho_i \succeq^\star \rho_j$.

Eirinakis et al. (2014) gives a minimal description of the stable matching polytope in the SM-MODEL.

Theorem 3.76 (Eirinakis et al., 2014)**.** *The stable matching polytope, $\{x^0\} \oplus A \cdot \mathcal{O}(\Pi, \succeq^\star)$, in the*

SM-MODEL *can be minimally described as follows:*

$$x_{f,w} + \sum_{(f',w)\in E_s:f'>_w f} x_{f',w} + \sum_{(f,w')\in E_s:w'>_f w} x_{f,w'} = 1 \qquad \forall (f,w)\in E_s\setminus \bar{E}_s \tag{3.9}$$

$$x_{f,w} = 0 \qquad \forall (f,w)\in E\setminus E_s \tag{3.10}$$

$$x_{f,w} \geq 0 \qquad \forall (f,w)\in E_0 \tag{3.11}$$

$$x_{f,w} \geq 0 \qquad \forall (f,w)\in E_z \tag{3.12}$$

$$x_{f,w} \geq 0 \qquad \forall (\rho_i,\rho_j)\in \Gamma_1, e_{\rho_i,\rho_j}=(f,w) \tag{3.13}$$

$$\sum_{(f',w)\in E_s:f'>_w f} x_{f',w} + \sum_{(f,w')\in E_s:w'>_f w} x_{f,w'} \geq 1 \qquad \forall (\rho_i,\rho_j)\in \Gamma_2, e_{\rho_i,\rho_j}=(f,w) \tag{3.14}$$

In the following, we give an alternative proof that (3.9) – (3.14) minimally describe the stable matching polytope by connecting the stable matching polytope with the order polytope associated with its representation poset. In particular, we show that the minimal description given by Eirinakis et al. (2014) can precisely be viewed as an application of Theorem 3.74 where each inequality is provided with a combinatorial interpretation special for the stable matching model.

We start with some known results. Theorem 3.77 can be viewed as a consequence of Theorem 3.76. However, since the goal of this section to give an alternative proof of Theorem 3.76, we present Theorem 3.77 as a known result on its own. Theorem 3.78 is immediate from Theorem 3.71.

Theorem 3.77 (Eirinakis et al., 2014)**.** *The system of equations* (3.9) *and* (3.10) *is linearly independent.*

Theorem 3.78 (Stanley, 1986)**.** *The order polytope* $\mathcal{O}(\Pi,\succeq^\star)$ *is full-dimensional and can be minimal described by the following facet-defining inequalities.*

$$y_{\rho_i} \leq 1 \qquad \forall i \text{ s.t. } \rho_i\in\Pi_0 \tag{3.15}$$

$$y_{\rho_i} \geq 0 \qquad \forall i \text{ s.t. } \rho_i\in\Pi_z \tag{3.16}$$

$$y_{\rho_i} \geq y_{\rho_j} \qquad \forall (\rho_i,\rho_j)\in I \tag{3.17}$$

We next summarize some facts known for matrix A which can be deduced from our results in Section 3.3. These properties are also known for the SM-MODEL (see, e.g., Gusfield and Irving, 1989).

Lemma 3.79. *Matrix A has the following properties:*

(i) Matrix A has full column rank.

(ii) Every row of A has at most one $+1$ and at most one -1.

(iii) Consider a row of A corresponding to a firm-worker pair (f, w) in the firm-optimal stable matching. There is at most one column, say ρ, with $A_{(f,w),\rho} = -1$ and all other columns have entry zero.

(iv) Consider a row of A corresponding to a firm-worker pair (f, w) in the worker-optimal stable matching. There is at most one column, say ρ, with $A_{(f,w),\rho} = +1$ and all other columns have entry zero.

In Propositions 3.80, 3.81 and 3.82, we show that (3.11) – (3.14) correspond to (3.15) – (3.17) in a sense as how c_i's relate to d_i's in the statement of Theorem 3.74. Then, in Proposition 3.83, we show that (3.9) and (3.10) correspond to the a_i vectors in the statement of Theorem 3.74.

Proposition 3.80. *Inequalities* (3.11) *are facet-defining for* $\{x^0\} \oplus A \cdot \mathcal{O}(\Pi, \succeq^\star)$*, and they are in bijection with* (3.15).

Proof. Consider a firm-worker pair $(f, w) \in E_0$ and assume $(f, w) \in \rho^-$ for some $\rho \in \Pi_0$. Note that (f, w) is in the firm-optimal stable matching by definition of Π_0 and thus $x^0_{f,w} = 1$. Therefore, by Lemma 3.79 (iii), $A_{(f,w),\rho} = -1$ and $A_{(f,w),\rho'} = 0$ for all other $\rho' \in \Pi$ such that $\rho' \neq \rho$ (♮). In the framework of Theorem 3.74, $y_\rho \leq 1$ corresponds to $c_i^\intercal y \geq \delta_i$ where c_i is a vector indexed by Π with all entries being 0, except for the one corresponding to ρ which has value -1, and $\delta_i = -1$. Consider the vector d_i indexed by E with all entries being 0 except for the one corresponding to (f, w) which has value 1. Due to (♮), it is not hard to see that $d_i^\intercal A = c_i^\intercal$. Moreover, $d_i^\intercal(x - x^0) \geq \delta_i$

reduces to $1 \cdot (x_{f,w} - 1) \geq -1 \Leftrightarrow x_{f,w} \geq 0$. We conclude the proof by Lemma 3.79 (i) and Theorem 3.78. ■

Proposition 3.81. *Inequalities* (3.12) *are facet-defining for* $\{x^0\} \oplus A \cdot \mathcal{O}(\Pi, \succeq^\star)$ *and they are in bijection with* (3.16).

Proof. Consider a firm-worker pair $(f, w) \in E_z$ and assume $(f, w) \in \rho^+$ for some $\rho \in \Pi_z$. Note that (f, w) is not in the firm-optimal stable matching and thus $x^0_{f,w} = 0$. Therefore, by Lemma 3.79 (iv), $A_{(f,w),\rho} = 1$ and $A_{(f,w),\rho'} = 0$ for all other $\rho' \in \Pi$ such that $\rho' \neq \rho$ ($\natural$). In the framework of Theorem 3.74, $y_\rho \geq 0$ corresponds to $c_i^\intercal y \geq \delta_i$ where c_i is a vector indexed by Π with all entries being 0, except for the one corresponding to ρ which has value 1, and $\delta_i = 0$. Consider the vector d_i indexed by E with all entries being 0 except for the one corresponding to (f, w) which has value 1. Due to ($\natural$), it is not hard to see that $d_i^\intercal A = c_i^\intercal$. Moreover, $d_i^\intercal (x - x^0) \geq \delta_i$ reduces to $1 \cdot (x_{f,w} - 0) \geq 0 \Leftrightarrow x_{f,w} \geq 0$. We conclude the proof again by Lemma 3.79 (i) and Theorem 3.78. ■

Proposition 3.82. *Inequalities* (3.13) *and* (3.14) *are facet-defining for* $\{x^0\} \oplus A \cdot \mathcal{O}(\Pi, \succeq^\star)$ *and they are in a bijection with* (3.17).

Proof. Consider a firm-worker pair (f, w) such that $(f, w) = e_{\rho_i,\rho_j}$ for some $(\rho_i, \rho_j) \in I$. Note that $x^0_{f,w} = 0$. We consider first the case when $(\rho_i, \rho_j) \in \Gamma_1$. In this case, we have $A_{(f,w),\rho_i} = 1$, $A_{(f,w),\rho_j} = -1$ and $A_{(f,w),\rho} = 0$ for all other $\rho \in \Pi \backslash \{\rho_i, \rho_j\}$ ($\natural$). In the framework of Theorem 3.74, $y_{\rho_i} \geq y_{\rho_j}$ corresponds to $c_i^\intercal y \geq \delta_i$ where c_i is a vector indexed by Π with all entries being 0, except for the one corresponding to ρ_i and ρ_j which have value 1 and -1 respectively, and $\delta_i = 0$. Consider the vector d_i indexed by E with all entries being 0 except for the one corresponding to (f, w) which has value 1. Due to ($\natural$), it is not hard to see that $d_i^\intercal A = c_i^\intercal$. Moreover, $d_i^\intercal (x - x^0) \geq \delta_i$ reduces to $1 \cdot (x_{f,w} - 0) \geq 0 \Leftrightarrow x_{f,w} \geq 0$.

We next consider the case when $(i, j) \in \Gamma_2$. Let $\bar{\rho}_1 \succ \bar{\rho}_2 \succ \cdots \succ \bar{\rho}_p \equiv \rho_j$ with $\bar{\rho}_1 \in \Pi_0$ be a series of rotations which sequentially match firm f to different workers. In particular, there is a collection of workers $\bar{w}_1, \bar{w}_2, \cdots, \bar{w}_p$ with $\bar{w}_1 >_f \bar{w}_2 >_f \cdots >_f \bar{w}_p >_f w$ such that $(f, \bar{w}_1) \in \bar{\rho}_1^-$

	$\bar{\rho}_1$	$\bar{\rho}_2$	$\bar{\rho}_3$	$\cdots$	$\bar{\rho}_{p-1}$	$\bar{\rho}_p$	$\bar{\rho}_{p+1}$	$\bar{\rho}_{p+2}$	$\bar{\rho}_{p+3}$	$\cdots$	$\bar{\rho}_{p+q-1}$	$\bar{\rho}_{p+q}$
$(f, \bar{w}_1)$	-1											
$(f, \bar{w}_2)$	$+1$	-1										
$(f, \bar{w}_3)$		$+1$	-1									
$\vdots$			$\vdots$		$\vdots$							
$(f, \bar{w}_p)$					$+1$	-1						
$(\bar{f}_1, w)$							$+1$	-1				
$(\bar{f}_2, w)$								$+1$	-1			
$\vdots$									$\vdots$		$\vdots$	
$(\bar{f}_{q-1}, w)$											$+1$	-1
$(\bar{f}_q, w)$												$+1$

Table 3.1: Matrix entries for certain firm-worker pairs (Proof of Proposition 3.82)

and $(f, \bar{w}_i) \in \bar{\rho}^+_{k-1} \cap \bar{\rho}^-_k$ for all $k = 2, 3, \cdots, p$. Moreover, define

$$\bar{E}_f \coloneqq \{(f, \bar{w}_k) : k \in [p]\} = \{(f, w') \in E_s : w' >_f w\}.$$

Similarly, let $\rho_i \equiv \bar{\rho}_{p+1} \succ \bar{\rho}_{p+2} \succ \cdots \succ \bar{\rho}_{p+q} \in \Pi_z$ be a series of rotations which sequentially match worker w to different firms. That is, there is a collection of firms $\bar{f}_1, \bar{f}_2, \cdots, \bar{f}_q$ with $\bar{f}_q >_w \bar{f}_{q-1} >_w \cdots >_w \bar{f}_1 >_w f$ such that $(\bar{f}_k, w) \in \bar{\rho}^+_{p+k} \cap \rho^-_{p+k+1}$ for all $k = 1, 2, \cdots, q-1$ and $(\bar{f}_q, w) \in \bar{\rho}^+_{p+q}$. Moreover, define

$$\bar{E}_w \coloneqq \{(\bar{f}_k, w) : k \in [q]\} = \{(f', w) \in E_s : f' >_w f\}.$$

Rows of matrix A corresponding to these firm-worker pairs in $\bar{E}_f \cup \bar{E}_w$ are given in Table 3.1 with zero entries omitted. Note that columns corresponding to any other rotation in $\Pi \setminus \{\bar{\rho}_1, \cdots, \bar{\rho}_{p+q}\}$ are also omitted because the entries are all zero. In addition, $(f, \bar{w}_1)$ is the only firm-work pair in $\bar{E}_f \cup \bar{E}_w$ that is in the firm-optimal stable matching.

In the framework of Theorem 3.74, $y_{\rho_i} \geq y_{\rho_j}$ corresponds to $c_i^\intercal y \geq \delta_i$ where c_i and δ_i are defined as above (in the first paragraph of the proof). Consider the vector d_i indexed by E with all entries being 0 except for the ones corresponding to edges in $\bar{E}_f \cup \bar{E}_w$ which have value 1. It is not hard to see that $d_i^\intercal A = c_i^\intercal$. Moreover, $d_i^\intercal(x - x^0) \geq \delta_i$ reduces to

$$\sum_{(f',w')\in \bar{E}_f\cup\bar{E}_w} \left(x_{(f',w')} - x^0_{(f',w')}\right) = \sum_{(f',w')\in \bar{E}_f\cup\bar{E}_w} x_{(f',w')} - x^0_{(f,\bar{w}_1)} \geq 0,$$

which is then equivalent to

$$\sum_{(f',w)\in E_s : f' >_w f} x_{f',w} + \sum_{(f,w')\in E_s : w' >_f w} x_{f,w'} \geq 1.$$

Again, we conclude the proof by Lemma 3.79 (i) and Theorem 3.78. ■

Proposition 3.83. *Equalities* (3.9) *and* (3.10) *are valid for* $\{x^0\} \oplus A \cdot \mathcal{O}(\Pi, \succeq^\star)$*, and they correspond to the equalities in the description given in* (3.8)*.*

Proof. We start with equalities in (3.9). Consider the equality corresponding to $(f, w) \in E_s \setminus \bar{E}_s$. Let $\bar{w}_1, \bar{w}_2, \cdots, \bar{w}_p$ be an enumeration of the workers in $\{w' \in W : (f, w') \in E_s, w' >_f w\}$ and let $\bar{f}_1$, $\bar{f}_2$, $\cdots$, $\bar{f}_q$ be an enumeration of the firms $\{f' \in F : (f', w) \in E_s, f' >_w f\}$. We assume without loss of generality that $\bar{w}_1 >_f \bar{w}_2 >_f \cdots >_f \bar{w}_p$, and $\bar{f}_q >_w \bar{f}_{q-1} >_w \cdots >_w \bar{f}_1$. Let $\bar{\rho}_1, \bar{\rho}_2, \cdots, \bar{\rho}_{p+q}$ be a sequence of rotations such that (i) $(f, \bar{w}_1) \in \bar{\rho}_1^-$; (ii) $(f, \bar{w}_i) \in \bar{\rho}_{i-1}^+ \cap \bar{\rho}_i^-$ for all $i = 2, 3, \cdots, p$; (iii) $(f, w) \in \bar{\rho}_p^+ \cap \bar{\rho}_{p+1}^-$; (iv) $(\bar{f}_i, w) \in \rho_{p+i}^+ \cap \rho_{p+i+1}^-$ for all $i = 1, 2, \cdots, q-1$; and (v) $(\bar{f}_q, w) \in \rho_{p+q}^+$. Let $\bar{E}_{(f,w)}$ denote the firm-worker pairs analyzed in (i)-(v). Rows of matrix A corresponding to $\bar{E}_{(f,w)}$ are given in Table 3.2 with zero entries omitted. Note that columns corresponding to any other rotation in $\Pi \setminus \{\bar{\rho}_1, \cdots, \bar{\rho}_{p+q}\}$ are also omitted from Table 3.2 because the entries are all zero. In addition, $(f, \bar{w}_1)$ is the only firm-worker pair in $\bar{E}_{(f,w)}$ that is in the firm-optimal stable matching.

In the framework of Theorem 3.74, consider vector a_i indexed by E with all entries being 0 except for the ones corresponding to edges in $\bar{E}_{(f,w)}$ which have value 1. It is not hard to see that

	$\bar{\rho}_1$	$\bar{\rho}_2$	$\bar{\rho}_3$	$\cdots$	$\bar{\rho}_{p-1}$	$\bar{\rho}_p$	$\bar{\rho}_{p+1}$	$\bar{\rho}_{p+2}$	$\cdots$	$\bar{\rho}_{p+q-1}$	$\bar{\rho}_{p+q}$
$(f, \bar{w}_1)$	-1										
$(f, \bar{w}_2)$	$+1$	-1									
$(f, \bar{w}_3)$		$+1$	-1								
$\vdots$			$\vdots$		$\vdots$						
$(f, \bar{w}_p)$					$+1$	-1					
(f, w)						$+1$	-1				
$(\bar{f}_1, w)$							$+1$	-1			
$\vdots$								$\vdots$		$\vdots$	
$(\bar{f}_{q-1}, w)$										$+1$	-1
$(\bar{f}_q, w)$											$+1$

Table 3.2: Matrix entries for certain firm-worker pairs (Proof of Proposition 3.83)

$a_i^\intercal A = \mathbf{0}^\intercal$ and $a_i^\intercal (x - x^0) = 0$ reduces to

$$\sum_{(f',w') \in \bar{E}_{(f,w)}} \left(x_{(f',w')} - x^0_{(f',w')} \right) = \sum_{(f',w') \in \bar{E}_{(f,w)}} x_{(f',w')} - x^0_{(f,\bar{w}_1)} = 0,$$

which is then equivalent to

$$x_{f,w} + \sum_{(f',w) \in E_s : f' >_w f} x_{f',w} + \sum_{(f,w') \in E_s : w' >_f w} x_{f,w'} = 1.$$

Therefore, inequality (3.9) corresponding to (f, w) is valid for $\{x^0\} \oplus A \cdot \mathcal{O}(\Pi, \succeq^\star)$.

Next, for equalities in (3.10), consider the one corresponding to $(f, w) \in E \setminus E_s$. Since (f, w) is not a stable pair, the row of matrix A corresponding to (f, w) are all zeros. In the framework of Theorem 3.74, consider vector a_i indexed by E with all entries being 0 except for the one corresponding to edge (f, w) which has value 1. Then, clearly, $a_i^\intercal A = \mathbf{0}^\intercal$. Moreover, $a_i^\intercal (x - x^0) = 0$ simply reduces to $x_{(f,w)} = 0$.

Lastly, since these equalities are linearly independent due to Theorem 3.77, they correspond to the equalities in the description given in (3.8). ■

We now give the proof for Theorem 3.76.

Proof of Theorem 3.76. It follows from Theorem 3.74 and 3.78; and Propositions 3.80, 3.81, 3.82, and 3.83. ■

3.6 Representations of choice functions and algorithms

Recall our previous observation that a choice function may be defined on all the (exponentially many) subsets of agents from the opposite side. The oracle model bypasses the computational concerns of representing choice functions explicitly. However, one drawback of this model is that it requires multiple rounds of communication between the "central planner" and each agent in the market. This, from an application point of view, is time-consuming: one of the major improvements brought about by the implementations of the Deferred Acceptance algorithm when applied, e.g., to the New York City school system, lies in the fact that it does not require multiple rounds of communication between the agents and the central planner (Abdulkadiroğlu, Pathak, and Roth, 2005a).

This observation leads to the following practically relevant and theoretically intriguing questions: is there a way to represent choice functions "compactly", and do our algorithms perform efficiently in such a model? A natural starting point is the MC-representation defined in Section 3.2.3. We show in Section 3.6.1 that the time complexity of our algorithms in the model where choice functions are given through their MC-representation is polynomial in the input size (where now the input includes the MC-representations). However, the MC-representation of a choice function may need a number of preference relations that are exponential in the number of agents (see Remark 3.84).

It is therefore interesting to investigate whether there are other ways to represent choice functions that is of size polynomial in the number of agents. Via a counting argument, we give a negative answer to this question in Section 3.6.2 for choice functions that are substitutable, consistent, and cardinal monotone (see Theorem 3.87 and Remark 3.88). We remark that our argument leaves it open whether a similar result holds if we replace cardinal monotonicity with quota-filling.

3.6.1 Algorithms with MC-representation

In this section, we show how to modify the algorithms and analyze their time complexities when agents' choice functions are explicitly given via the MC-representations.

In Algorithm 3.1 and Algorithm 3.2, instead of relying on an oracle model, we need to compute the outcomes of choice functions $\mathcal{C}_a(S)$ for agent $a \in F \cup W$ and subset of acceptable partners S. Using results in Section 3.2.3, $\mathcal{C}_a(S)$ can be obtained as a set of maximizers as the following: $\{\max(S, \geq_{a,i}) : i \in [p(\mathcal{C}_a)]\}$. Since each $\max(S, \geq_{a,i})$ requires $O(\max(|F|, |W|)$ time to compute, the time-complexity for obtaining $\mathcal{C}_a(S)$ is $O(\max(|F|, |W|)p(\mathcal{C}_a))$. Thus, for all previous results in terms of time complexity, one can simply replace $O(\texttt{oracle-calls})$ with $O(\max(|F|, |W|) \max_{a \in F \cup W} p(\mathcal{C}_a))$. Note that this time complexity bound is polynomial in the input size, but could be exponential in the number of agents, since $\max_{a \in F \cup W} p(\mathcal{C}_a)$ maybe exponential in the number of the agents as discussed in Remark 3.84.

Remark 3.84. Doğan, Doğan, and Yıldız (2021) constructed strict preference lists with quotas (i.e., choice functions for the MM-MODEL) whose MC-representation needs exponentially many preference relations. Since such choice functions are a special case of the quota-filling choice functions, in general the MC-representation of quota-filling choice functions is not polynomial in the number of agents.

3.6.2 On the number of substitutable, consistent, and cardinal monotone choice functions

In this section, the domain of all choice functions is the family of subsets of X, with $|X| = n$. The simplest choice functions $\mathcal{C}$ appears in the SM-MODEL, where there is a single underlying strict preference list. The number of such choice functions is

$$\sum_{i=0}^{n} \binom{n}{i} i! = \sum_{i=0}^{n} \frac{n!}{(n-i)!} = \sum_{i=0}^{n} \frac{n!}{(n-i)!} = n! \sum_{i=0}^{n} \frac{1}{i!} \leq en!,$$

hence, singly exponential in n. On the other extreme, the number of all choice functions is doubly-exponential in n (see, e.g., Echenique, 2007). We give the proof of this fact for completeness.

Theorem 3.85. *The number of choice functions on subsets of X with $|X| = n$ is $2^{n2^{n-1}}$.*

Proof. Since for each set of partners $S \subseteq X$ with $|S| = i$, $\mathcal{C}(S)$ can take 2^i possible values and there are $\binom{n}{i}$ subsets of X with size i, the number of possible choice function is $\prod_{i=1}^{n} (2^i)^{\binom{n}{i}}$. Taking the logarithm with base 2, we have

$$\log_2 \left(\prod_{i=1}^{n} (2^i)^{\binom{n}{i}} \right) = \sum_{i=1}^{n} \binom{n}{i} i = \sum_{i=1}^{n} \frac{n!}{(n-i)!(i-1)!} = n \sum_{i'=0}^{n-1} \frac{(n-1)!}{(n-1-i')!(i')!} = n2^{n-1}.$$

■

It has also been shown by Echenique (2007) that when choice functions are assumed to be substitutable and consistent (i.e., path-independent), the number of choice functions remains doubly exponential in n.

Theorem 3.86 (Echenique, 2007)). *The number of substitutable and consistent choice functions on subsets of X with $|X| = n$ is $2^{\Omega\left(\frac{2^{n-1}}{\sqrt{n-1}}\right)}$.*

In the rest of the section, we show that the number of choice functions that additionally satisfies cardinal monotonicity remains doubly exponentially in n. The proof idea follows from that given in Echenique (2007).

Theorem 3.87. *The number of substitutable, consistent, and cardinal monotone choice functions on subsets of X with $|X| = n$ is $2^{\Omega\left(\frac{2^{n-1}}{\sqrt{n-1}}\right)}$.*

Remark 3.88. Because of Theorem 3.87, in order to encode all substitutable, consistent, and cardinal monotone choice function in binary strings, we need a number of strings that is superpolynomial in n, i.e., the number of agents in the market.

A family of subsets $\mathcal{A} \subseteq 2^X$ is an *antichain* of $(2^X, \subseteq)$ if for any subsets $A, B \in \mathcal{A}$, they are not comparable, i.e., $A \setminus B \neq \emptyset$ and $B \setminus A \neq \emptyset$. A family of subsets $\mathcal{F} \subseteq 2^X$ is a *filter* (i.e., *lower set*) if for all $F \in \mathcal{F}$, $F' \supseteq F$ implies $F' \in \mathcal{F}$. Moreover, we say filter $\mathcal{F}$ is a *filter at* x if for all $F \in \mathcal{F}$, we have $x \in F$. Note that $\emptyset$ is a filter at x.

Theorem 3.89 (Echenique, 2007)**.** *There is an injective function mapping collections of antichains $\boldsymbol{\mathcal{A}} = \{\mathcal{A}_x : x \in X\}$ where each $\mathcal{A}_x$ is an antichain of the poset $(2^{X\setminus\{x\}}, \subseteq)$ to substitutable choice functions. The image of $\boldsymbol{\mathcal{A}}$ is defined as follows: for all $S \subseteq X$,*

$$\mathcal{C}(S) \coloneqq \{x \in S : S \notin \mathcal{T}_x\},$$

where

$$\mathcal{T}_x \coloneqq \{B \subseteq X : A \cup \{x\} \subseteq B \text{ for some } A \in \mathcal{A}_x\}.$$

Moreover, $\mathcal{T}_x$ is a filter at x for all $x \in X$.

Because of Theorem 3.89, let $\mathcal{C}[\boldsymbol{\mathcal{A}}]$ denote the substitutable choice function corresponding to the collection of antichains $\boldsymbol{\mathcal{A}}$ constructed by the statement of the theorem.

Lemma 3.90. *Let (Y, W) be a partition of X with $W = \{w\}$. Let $\boldsymbol{\mathcal{A}} = \{\mathcal{A}_x : x \in X\}$ be a collection of antichains such that (i) for all $x \in Y$, $\mathcal{A}_x = \emptyset$ and (ii) $\mathcal{A}_w$ is an antichain of $(2^Y, \subseteq)$. Then $\mathcal{C}[\boldsymbol{\mathcal{A}}]$ is consistent and cardinal monotone.*

Proof. We abbreviate $\mathcal{C} := \mathcal{C}[\boldsymbol{\mathcal{A}}]$. Let $\mathcal{T}_x$ be as defined in the statement of Theorem 3.89. That is, $\mathcal{T}_x = \emptyset$ for all $x \in Y$ and $\mathcal{T}_w$ is a filter at w. Hence, note that $S \cap Y \subseteq \mathcal{C}(S)$ for all $S \subseteq X$ ($\sharp$).

Let $T \subseteq X$. We consider first the case when $w \notin T$. Then, $\mathcal{C}(T) = T$ because of ($\sharp$). Let $S \subseteq X$ be such that $\mathcal{C}(T) \subseteq S \subseteq T$. Then it must be that $S = T$ and it follows immediately that $\mathcal{C}(T) = \mathcal{C}(S)$. In addition, for all $S \subseteq T$, we also have $S \subseteq Y$ and thus, using ($\sharp$) again, $|\mathcal{C}(S)| = |S| \leq |T| = |\mathcal{C}(T)|$.

We next consider the case when $w \in T$. Then, either $\mathcal{C}(T) = T$ or $\mathcal{C}(T) = T \setminus \{w\}$, again because of ($\sharp$). We start with the consistency property. Assume we are in the former case, and let $S \subseteq X$ be such that $\mathcal{C}(T) \subseteq S \subseteq T$. Since $T = \mathcal{C}(T)$, we have $S = T$ and thus $\mathcal{C}(T) = \mathcal{C}(S)$. Now assume we are in the latter case: $\mathcal{C}(T) = T \setminus \{w\}$. If $S \subseteq X$ satisfies $\mathcal{C}(T) \subseteq S \subseteq T$, we either have $S = T$ or $S = T \setminus \{w\}$. Regardless, we have $\mathcal{C}(S) = \mathcal{C}(T)$. Lastly, we show the cardinality monotonicity property, and we consider both cases at once. For all $S \subsetneq T$, we either have $\mathcal{C}(S) = S$ or $\mathcal{C}(S) = S \setminus \{w\}$ due to ($\sharp$). Either way, $|\mathcal{C}(S)| \leq |S| \leq |T| - 1 \leq |\mathcal{C}(T)|$. Hence, $\mathcal{C}$ is both consistent and cardinal monotone, concluding the proof. ■

Thus, a lower bound to the number of substitutable, consistent, and cardinal monotone choice functions can be obtained by counting the number of antichains. The problem of counting the number of antichains of a poset is called the *Dedekind's problem*. Let $\mathcal{N}(k)$ denote the collection of antichains of poset $(2^{[k]}, \subseteq)$. The following result is well-known and we include the proof for completeness.

Lemma 3.91. $|\mathcal{N}(k)| \geq 2^{\binom{k}{\lfloor k/2 \rfloor}} = 2^{\Theta(2^k/\sqrt{k})}$.

Proof. Consider any two distinct subsets $A, B \subseteq X$ with $|A| = |B|$, then it must be that $A \setminus B \neq \emptyset$ and $B \setminus A \neq \emptyset$. Thus, a collection of subsets, each with the same size, is an antichain of $(2^{[k]}, \subseteq)$. Therefore, the number of antichains of $(2^{[k]}, \subseteq)$ is at least the number of subsets of $\{A \subseteq X : |A| = \lfloor k/2 \rfloor\}$, which is exactly $2^{\binom{k}{\lfloor k/2 \rfloor}}$ since there are $\binom{k}{\lfloor k/2 \rfloor}$ subsets of X with size $\lfloor k/2 \rfloor$. The last equality follows from Stirling's approximation. ■

We now present the proof for Theorem 3.87.

Proof of Theorem 3.87. Let (Y, W) be a partition of X with $|Y| = n - 1$ and $|W| = 1$, as in the statement of Lemma 3.90. By Lemma 3.91, the possible choices of antichains $\mathcal{A}_x$ for $x \in W$ is at least $\mathcal{N}(n - 1)$. Hence, the number of $\boldsymbol{\mathcal{A}}$ (i.e., collection of antichains) in the statement of Lemma 3.90 is also at least $\mathcal{N}(n - 1)$. Finally, together with Theorem 3.89, we have that the number of substitutable, consistent, and cardinal monotone choice functions is again at least $\mathcal{N}(n - 1)$. ■

Chapter 4: Discovering Opportunities in New York City's Discovery Program: an Analysis of Affirmative Action Mechanisms

4.1 Introduction

There is a pervasive problem in the way students are evaluated and given access to higher education (Ashkenas, Park, and Pearce, 2017; Boschma and Brownstein, 2016; Capers IV et al., 2017). Promising students are often unable to get admission at the top schools because the path to getting admitted to these schools requires extensive training at various levels, starting as early as when students are 3 years old (Shapiro, 2019b). It is no surprise then that underrepresented minorities, especially those with lower household income and lower family education, are systematically screened-out of the education pipeline. In fact, in many cities, schools remain highly segregated (Shapiro, 2021; Shapiro, 2019a). Disparate opportunities in accessing high-quality education is one of the main causes of income imbalance and social immobility in the United States (Orfield and Lee, 2005). It is expected that this disparity will only become more acute due to COVID-induced loss of jobs and strain on low-income families. Now more than ever, affirmative action policies, such as quota-based mechanisms and training programs, are critical and offer practical remedies for increasing representation of under-represented minorities and disadvantaged groups in public schools in the U.S. (Hafalir, Yenmez, and Yildirim, 2013).

In this work, we study theoretically and empirically the characteristics of the *Discovery Program*[1], which is an affirmative action program used by the New York City Department of Education (NYC DOE) in an effort to increase the number of disadvantaged students at specialized high schools (SHS) (NYCDOE, 2019). SHSs span the five boroughs of NYC (Table 4.3), and

[1] `https://www.schools.nyc.gov/enrollment/enrollment-help/meeting-student-needs/diversity-in-admissions`

are among the most competitive ones in NYC. For admission, these high schools consider *only* students' score on the Specialized High School Admissions Test (SHSAT). Around 5000 students are admitted every year to SHSs. The discovery program reserves some seats for disadvantaged students that are assigned at the end of the regular admission process, after student's participation in a 3-week enrichment program during the summer.

The discovery program has been instrumental in creating opportunities for disadvantaged students (classified with respect to socio-economic factors), increasing the number of admitted students to these extremely competitive public high schools in NYC. In 2020, for example, Mayor Bill de Blasio called for an expansion of discovery program, with 20% seats at SHSs reserved for the program. This expansion resulted in $1,350$ more disadvantaged students being admitted to these specialized schools (NYCDOE, 2019; Veiga, 2020).

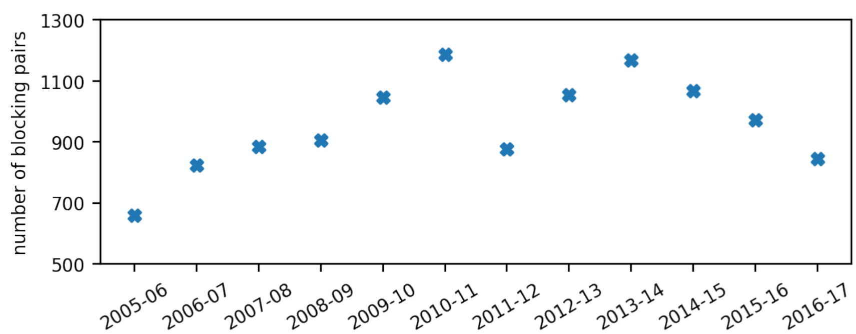


Figure 4.1: Number of blocking pairs amongst disadvantaged students under the discovery program mechanism across the last 12 years, which impacted around 650 students each year.

In this work, we dive deep into the student-school matching produced by the discovery program. Our empirical analysis shows that under a reasonable assumption on students' preferences over schools which we term *school-over-seat*[2], the matchings from 12 recent academic years (2005-06 to 2016-17) created about 950 in-group blocking pairs each year amongst disadvantaged students, impacting about 650 disadvantaged students every year (see Figure 4.1). A blocking pair is a pair of student s_1 and school c_1 that prefer each other to their matches, thus violating the priority of student s_1 at school c_1 and creating dissatisfaction among students and schools.

[2]This hypothesis assumes that students' preference lists over schools are not affected by whether they are required to participate in the three-week summer enrichment program. See Section 4.1.1 for details.

Moreover, we find that this program benefits *lower-performing* disadvantaged students more than *top-performing* disadvantaged students (in terms of their rankings of their assigned schools), thus unintentionally creating an incentive to under-perform. See Figure 4.2 for a depiction of our empirical analysis, where top-performing students (with ranks $0 \sim 500$) attend less preferred schools under the discovery program, unlike low-performing students (rank 500-1000) who get matched to better ranked schools (lower numeric rank is better). These drawbacks of the discovery program are not simply an artifact of the data from NYC DOE, but are, as we show theoretically, properties about the current implementation of the discovery program. Therefore, our goal in this chapter is to explore other affirmative action mechanisms, so that we can propose practical modifications to how the discovery program is implemented, while alleviating the above-mentioned drawbacks.

In particular, we compare the discovery program (`DISC`) together with two other affirmative action mechanism: *minority reserve* (`MR`) and *joint seat allocation* (`JSA`). These latter mechanisms are also quota-based, where schools reserve a certain proportion of their seats for disadvantaged students. Minority reserve, in contrast to the discovery program, allocates the reserved seats to disadvantaged students before the general admission. This mechanism has been well studied in the literature (see, e.g., Hafalir, Yenmez, and Yildirim, 2013). The joint seat allocation, on the other hand, allocates reserved and general (i.e., non-reserved) seats at the same time, while allowing disadvantaged students to take general seats (if they are able to compete) and otherwise revert to reserved seats. This mechanism is inspired by the joint seat allocation process for admission to Indian Institutes of Technology[3] (JoSAA, 2020) and this is the first work to study this to the best of our knowledge. We compare these three affirmative action policies with respect to the baseline stable matching mechanism, `noAA`, which does not incorporate affirmative action policies (Gale and Shapley, 1962). We discuss our key contributions next.

[3]The actual mechanism used by the Joint Seat Allocation Authority is more complicated than the version we study here in the chapter. In particular, in our setting, we assume that there are two disjoint types of students: disadvantaged and non-disadvantaged. However, in the actual implementation (see, e.g., Baswana et al., 2019), students are categorized through multiple dimensions (e.g., caste, gender).

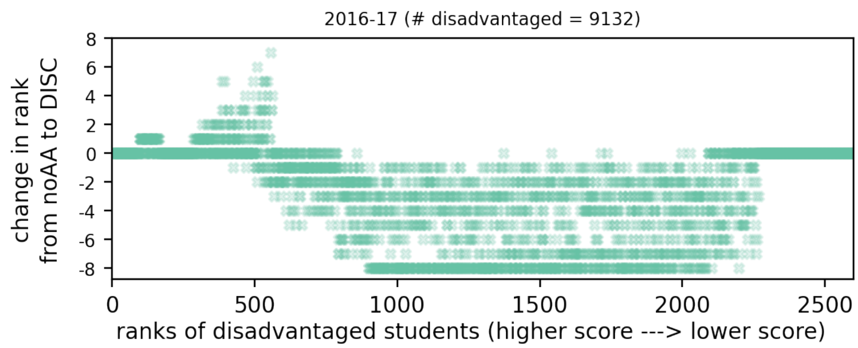


Figure 4.2: Change in rank (where a negative change means getting to a more preferred school) of assigned schools for all disadvantaged students from `noAA` to `DISC` (we plot `DISC` - `noAA`), ranked by the quality of students. Top students (ranked 0-500) are matched to worse schools under DISC, whereas the lower performing disadvantaged students are matched to better schools.

4.1.1 Main results

We first show properties of affirmative action mechanisms under the school-over-seat hypothesis, i.e., students' preferences over schools are not influenced by whether they are admitted via general seats or reserved seats (in the case of NYC SHSs, reserved seats additionally require a 3-week summer program). We next discuss weak dominance amongst the three affirmative action mechanisms, showing that `JSA` outperforms `MR` under a condition that we term *high competitiveness* of markets. Finally, we empirically validate our theoretical results using data from NYC DOE, and make a policy recommendation for the discovery program.

Properties of Affirmative Action Mechanisms.

Question 1. *Which affirmative action mechanisms considered in the chapter satisfy reasonable notions of fairness such as absence of in-group blocking pairs and strategy-proofness? What is the impact of these affirmative action policies on the disadvantaged group of students?*

We explore four useful properties for affirmative action mechanisms for each of `noAA` (the mechanism that does not reserve seats for disadvantaged students), `DISC`, `JSA` and `MR` mechanisms and briefly explain these properties here (see Sections 4.2 and 4.3 for formal definitions): (i) *strategy-proofness*: this property means that the best strategy of students is to honestly report their preferences; (ii) *absence of in-group blocking pairs*: this is a fairness condition which ensures there is no priority violation for students; (iii) the third property asks for the mechanism not to worsen (with respect to the mechanism with no affirmative action) the assignment of *at least* one disadvantaged student[4]; and (iv) the fourth property asks *all* disadvantaged students not to be worse-off in a restricted scenario called *smart-reserve*. Reservation quotas are a smart reserve[5] if the number of seats reserved for disadvantaged students is no less than the number of disadvantaged students admitted without affirmation actions.

We summarize our results in Table 4.1. As one can immediately see from the table, the current implementation of the discovery program does not satisfy any of the attractive features we investigate, yet the other two affirmative action mechanisms, `MR` and `JSA`, satisfy all these properties. This is even true when *all the schools rank students in the same order*, as in the NYC SHS admission market where students are ranked based on their SHSAT scores. We additionally demonstrate these findings empirically by computational experiments using the admission data on NYC SHSs (the details can be found in Section 4.5). These results suggest that the discovery program could benefit by replacing the current implementation with either minority reserve or joint seat allocation. This result calls for a direct comparison of those mechanisms.

	`noAA`	`DISC`	`MR`	`JSA`
weakly group strategy-proof	✓ [DF]	✗ (Ex 4.10)	✓ [HYY]	✓ (Prop 4.12)
no in-group blocking pairs	✓ [GS]	✗ (Ex 4.10)	✓ (Prop 4.8)	✓ (Prop 4.16)
at least one disadvantaged student not worse off	`NA`	✗ (Ex 4.9)	✓ [HYY]	✓ (Thm 4.13)
no disadvantaged student worse off if smart reserve	`NA`	✗ (Ex 4.10)	✓ [HYY]	✓ (Thm 4.14)

Table 4.1: Summary of properties of affirmative action mechanisms under the school-over-seat assumption. `NA` means not applicable. Previously known results and their corresponding citations are given in square brackets, with: [DF] Dubins and Freedman (1981); [HYY] Hafalir, Yenmez, and Yildirim (2013); and [GS] Gale and Shapley (1962); other results are accompanied by the labels of examples, propositions, or theorems used to answer the questions.

[4]Intuitively, one might expect property (iii) to be so weak that it is trivially satisfied. However, the discovery program does not satisfy it in general.

[5]This requirement was first proposed and studied by Hafalir, Yenmez, and Yildirim (2013), and they showed that such a condition is achievable either in an ad-hoc fashion or by using historical data on school admissions.

Dominance across Affirmative Action Mechanisms.

Question 2. *Considering a fixed reservation quota, does one of the affirmative action mechanisms (*`DISC`*,* `JSA` *or* `MR`*)* (weakly) dominate *another one for disadvantaged students, i.e., do all disadvantaged students weakly prefer the schools they are matched to under one mechanism compared to the other?*

We say that a mechanism A *(weakly) dominates* another mechanism B for disadvantaged students if A places all disadvantaged students in schools they like at least as much as the schools they are placed in by B. Our results from Table 4.1 seem to suggest that the discovery program mechanism could be dominated by either minority reserve or joint seat allocation. However, this is *not* the case, as shown by the results we summarize in Table 4.2. All three mechanisms are incomparable, even under some pretty restrictive hypothesis: (1) schools rank students in the same order; and/or (2) reservation quotas being a smart reserve. The first hypothesis is common in markets where students' ranking is based on an entrance exam, such as the one for NYC SHSs. The only exception to the incomparability results is that the mechanism `noAA` without affirmative action, under the second hypothesis, is dominated by minority reserve and joint seat allocation[6].

	`noAA`	`MR`	`DISC`	`JSA`
`noAA`		(✗) (✗) (✗ Ex 4.17)	(✗) (✗) (✗ Ex 4.17)	(✗) (✗) (✗ Ex 4.17)
`MR`	(✗ [HYY]) (✓ [HYY]) (✓)		(✗) (✗) (✗ Ex 4.18)	(✗) (✗) (✗ Ex 4.19)
`DISC`	(✗) (✗) (✗ Ex 4.10)	(✗) (✗) (✗ Ex 4.18)		(✗) (✗) (✗ Ex 4.18)
`JSA`	(✗ 4.15) (✓ Thm 4.14) (✓)	(✗) (✗) (✗ Ex 4.19)	(✗) (✗) (✗ Ex 4.18)	

Table 4.2: The table answer the following question under the school-over-seat assumption: does the "row" mechanism dominates the "column" mechanism for disadvantaged students? We answer the question for three restricted domains: (1) schools share a common ranking of the students, (2) the reservation quotas is a smart reserve, and (3) both. The answers are given in the exact order. All answers are accompanied by the citations with [HYY] Hafalir, Yenmez, and Yildirim (2013) or the labels of the examples or theorems used to answer the questions, except for cases when the answer for one domain can be inferred from that of another domain.

To be able to identify crucial interventions for the discovery program, we study the behavior of the `JSA` and `MR` mechanisms in markets that satisfy a condition which we call *high competitiveness*.

[6]We would like to point out that this exception is simply another way of expressing the same results related to the third property in Table 4.1.

This is a novel ex-post condition which guarantees that JSA weakly dominates MR for disadvantaged students. This condition is verified by our data from NYC DOE, where in fact JSA outperforms MR for disadvantaged students. We also show reasonable conditions on the primitives of the market that imply high competitiveness. Roughly speaking, the high competitiveness condition is satisfied when the demand for seats (i.e., number of students) is much larger than the supply, and when disadvantaged students are performing systematically worse than advantaged students[7]. See Theorem 4.20 for the formal statement. We discuss next how our experiments validate our theoretical result and provide a practical policy recommendation for changes to the discovery program.

Case Study based on Data from New York City's Department of Education.

We validate our theoretical results with extensive computational experiments using data we obtained from NYC DOE for the 2005-2006 to 2016-2017 academic years, where we label students as advantaged or disadvantaged based on the criteria given by the discovery program. First, we show that, in practice as well, the discovery program suffers from all the theoretical drawbacks we presented in Table 4.1 (see Section 4.5.1 for details), except the third property (as it requires the construction of an extreme case). We find that for reservation quotas set to 20%, on average there are 950 blocking pairs for disadvantaged students which impact around 650 students each year. Considering the changes in rank to matched schools, DISC mechanisms is the only one under which disadvantaged students can be worse-off (i.e., which hurts some disadvantaged students). In particular, this hurts the top-performing disadvantaged students much more, and helps the low-performing disadvantaged students (see Figure 4.2). The discovery program is also not strategy proof: some of the aforementioned top-performing students may truncate their preference lists (i.e., remove some less preferred schools from their honestly submitted preference lists), so that they skip the competition for general seats at these less preferred schools and aim directly for reserved seats at more preferred schools.

In addition, by observing the distribution of SHSAT scores for both the advantaged and disad-

[7] We hereafter refer to non-disadvantaged students as advantaged students for the sake of nomenclature, and it only implies that advantaged students do not suffer from the disadvantages that affect disadvantaged students.

vantaged groups of students, we notice that disadvantaged students are performing systematically worse than advantaged students (see Figure 4.3b), which would undoubtedly lead to underrepresentation of disadvantaged students at these SHSs without affirmative actions. Because of this observation and of the very limited number of seats when compared to the students applying to SHSs, we expect the market to the highly competitive and thus all disadvantaged students would weakly prefer their assignment under `JSA` than under `MR`. We indeed observe these characteristics for the NYC SHS admission market across all academic years we have data for (see Figure 4.3a and Figure 4.4b). This leads to the policy recommendation we present in this work.

Policy Recommendation.

Overall, our work paves the way to make the discovery program fairer for disadvantaged students. In particular, we provide an answer to how the existing practice of the discovery program can be changed minimally to improve the outcome for the disadvantaged group of students, so that the program aligns with the incentives to perform better.

Our Proposal: We propose that the program takes into account the preferences of students in terms of the schools v/s seats. Is attending a particular school more important than the type of seat they are assigned to or vice versa? We believe that most students should be willing to take a one-time 3-week summer program to attend a school they prefer, rather than not taking the program and attending, for 4 years, a school they prefer less (e.g., we find that this hypothesis is supported by the fact that preferences appear often to be strongly polarized for certain schools due to, e.g., geographical considerations, details are reported in the Appendix, Section B.2). Although this seems reasonable, unfortunately such preferences are currently not collected in the data provided by the NYC DOE.

Under the school-over-seat assumption, we find that the many drawbacks of the current implementation of the discovery program can be corrected by following the *joint seat allocation* mechanism. For the NYC Specialized High School market – and, more generally, for highly competitive markets – joint seat allocation gives a matching that is weakly better for disadvantaged students, when compared to matching output by the other replacement mechanism studied in this chapter, both in theory and in practice.

Although powerful, the modification we propose requires *minimal* modification: there is essentially no change in terms of what students and schools should report to the DOE (preference lists for both and admission capacity for schools), and there is no change in terms of the algorithm (the deferred acceptance algorithm (Gale and Shapley, 1962), which is currently in implementation). Given this information, to implement the `JSA` mechanism, one only needs to compute an equivalent instance where students' preference lists are expanded to be over reserved and general seats at schools, so that the matching we desire to obtain can be easily recovered from the matching obtained under the classical stable matching model on this equivalent instance. See Section 4.4.3 for details.

Before we delve deeper into our model and results, we would like to highlight a trade-off that any constrained resource allocation problem faces. Diverting some resources to the disadvantaged groups implies taking some resources that are currently assigned to the advantaged groups. In this work as well, we find from our empirical analysis, that advantaged students always weakly prefer their assignment under `MR` compared to `JSA`. For all the academic years we analyze, we find that about 3% of the advantaged students are worse off under `JSA` than under `MR` (i.e., about 97% of them are matched to the same school under the two mechanisms); and among the 3%, most of them experience a drop in the rank of assigned schools that is at most two. See Figure 4.4b for details of one academic year. We consider this impact to be minimal compared to the ill-treatment faced by the disadvantaged students.

4.1.2 The techniques

The affirmative action mechanisms introduced in this chapter seem to entail different algorithms applied to the same preferences lists of students and schools. However, it turns out that an equivalent, yet mathematically more convenient way is to view their assignment outputs as obtained from the same algorithm applied, however, to different input instances. There are two approaches by which we can obtain such a reformulation.

This first approach is to employ *choice functions*, which are a general and powerful way to model the preference lists of agents in matching markets. In particular, all choice functions needed to model the mechanisms in this chapter satisfy the *substitutability*, *consistency*, and q_c*-acceptance* properties (see Section 4.2.2). Under such properties, stable matchings are known to exist and satisfy strong structural and algorithmic properties (see, e.g., Alkan (2002), Faenza and Zhang (2021), and Roth (1984a)). This reformulation[8] allows us to analyze the assignments under different mechanisms as the outputs of one or more rounds of Roth's generalization (Roth, 1984a) of the classical deferred acceptance algorithm by Gale and Shapley (1962). As a result, to show properties of the assignment obtained from an affirmative action mechanism, we can directly use properties of its choice functions, of stable matchings, as well as the properties of the generalized deferred acceptance algorithm.

The second approach is to expand students' original preferences over schools to preferences over reserved and general seats at schools. Under this reformulation, assignments under different affirmative action mechanisms can be obtained simply by applying the classical deferred acceptance algorithm over the equivalent instances. This allows us to deduce interesting properties of the mechanisms (e.g., strategy-proofness), by leveraging on classical results on stable matchings.

4.1.3 Related literature

The problem of assigning students to schools (without affirmative action) was first studied by Gale and Shapley in their seminal work (Gale and Shapley, 1962). Abdulkadiroğlu and Sönmez (2003) then analyzed the algorithm in the context of school choice and recommended school districts to replace their current mechanisms with either this algorithm or another algorithm, called the *top trading cycle algorithm*. Since then, these mechanisms have been widely adopted by many

[8] We note in passing, that, this reformulation allows a central planner to access many stable matchings, using recent results by Faenza and Zhang (2021), which provide alternatives to the matchings output by the mechanisms considered in this chapter.

cities in the United States, such as New York City and Boston.

The first attempt of incorporating affirmative action with the stable mechanism occurred in this pioneering work (Abdulkadiroğlu and Sönmez, 2003), where they extended their analysis to a simple affirmative action policy, using majority quotas. However, Kojima (2012) then analyze the effects of these proposed affirmative action policies, as well as priority-based policies, and showed that in some cases, the mechanisms might hurt disadvantaged students, the very group these policies are trying to help. Hafalir, Yenmez, and Yildirim (2013) further analyze the effect empirically through simulated data and suggested that this phenomenon might be quite common, and does not just happen in theory due to special edge cases. In addition, to overcome the efficiency loss, they propose the minority reserve mechanism.

Since then, there has been an abundance of literature, studying and proposing solutions for the efficiency loss due to affirmative action, such as Afacan and Salman (2016), Doğan (2016), Echenique and Yenmez (2015), Ehlers et al. (2014), Fragiadakis and Troyan (2017), Jiao and Shen (2021), Kominers and Sönmez (2016), and Nguyen and Vohra (2019).

Another popular form of affirmative action is what is called *priority-based* (see, e.g., Hafalir, Yenmez, and Yildirim, 2013; Jiao and Shen, 2021; Kojima, 2012), which creates a higher priority for disadvantaged students by, e.g., boosting their scores. Though this mechanism satisfies important properties such as strategy-proofness and absence of in-group blocking pairs, its practical use is being largely debated. For example, in 2019, the college board proposed adding an adversity score to SAT scores to account for socio-economic differences, however, this was met with severe pushback (Jaschik, 2019). In another lawsuit at the University of Michigan challenging a priority-based mechanism that assigned 20 points extra to disadvantaged students, the system was declared unconstitutional by the Supreme Court (Gratz_vs_Bollinger, 2003). Faenza, Gupta, and Zhang, 2022b investigates the effects of policies where scores for minority students are boosted before the admission process by extra training, additional resources, etc. Since the goal of this work is to focus on operational suggestions to the discovery program, we do not explore priority-based mechanisms.

4.1.4 Outline

The rest of the chapter is organized as follows. In Section 4.2, we introduce the basic model and related concepts for stable matchings and stable matching mechanisms. In Section 4.3, we formally introduce the affirmative action mechanisms considered in this chapter and investigate their properties (i.e., answering Question 1). We then compare these mechanisms in Section 4.4 and provide the answer to Question 2. In Section 4.4.3, we show that the three affirmative action mechanisms considered in the chapter differ in terms of how students' preference over reserved seats and general seats are interpreted. Lastly, in Section 4.5, we dive into the data on NYC SHS admission, demonstrate our theoretical findings empirically.

4.2 Model and Notations

For this chapter, the two sides of the markets are students and schools, denoted by S and C, respectively. In the following, we recall and re-introduce some concepts already introduced in Chapter 2 and Chapter 3 to align with notations of this chapter, as well as introducing some new concepts.

4.2.1 Matchings and mechanisms

Let $G = (S \cup C, E)$ be a bipartite graph, where the edge set E represents the schools which students find *acceptable* (i.e., would like to attend). Every student $s \in S$ has a strict preference relation over the schools they find acceptable and the option of being unassigned (denoted by $\emptyset$), which we call the *preference list* of student s, and we denote it by $>_s$. On the other hand, every school c has a quota $q_c \in \mathbb{N} \cup \{0\}$ and a strict *priority order* $>_c$ over the students. In addition, we assume that there are two types of students, *advantaged* (or *majority*) and *disadvantaged* (or *minority*), denote by S^M and S^m respectively. That is, $S = S^M \dot{\cup} S^m$ where $\dot{\cup}$ is the disjoint union operator.

Let $>_S \equiv \{>_s : s \in S\}$, $>_C \equiv \{>_c : c \in C\}$, and $\mathbf{q} \equiv \{q_c : c \in C\}$ denote the collection of students' preference lists, schools' priority orders, and schools' quotas, respectively. Moreover, we write $>\equiv \{>_S, >_C\}$. An *instance (or market)* is thus denoted by $(G, >_S, >_C, \mathbf{q})$ or $(G, >, \mathbf{q})$.

A *matching* μ (of an instance) is a collection of student-school pairs such that every student is incident to at most one edge in μ and every school c is incident to at most q_c edges in μ. For student $s \in S$ and school $c \in C$, we denote by $\mu(s)$ the school student s is matched (or assigned) to, and by $\mu(c)$ the set of students school c is matched (or assigned) to, under matching μ.

For every school $c \in C$, let $q_c^R \in \{0, 1, \cdots, q_c\}$ denote the number of seats reserved to disadvantaged students at school c, and let $q_c^G := q_c - q_c^R$ denote the number of general seats at school c. We call $\mathbf{q}^R := \{q_c^R : c \in C\}$ the *reservation quotas*.

A *(quota-based matching) mechanism* is a function that maps every instance, together with reservation quotas, to a *matching*. Given an instance $I = (G, >, \mathbf{q})$, a mechanism ϕ, and reservation quotas $\mathbf{q}^R$, let $\phi(I, \mathbf{q}^R)$ denote the matching obtained under the mechanism ϕ with reservation quotas $\mathbf{q}^R$. Sometimes, when the reservation quotas are clear from context, we simply denote the matching as $\phi(I)$.

Let μ_1, μ_2 be two matchings. We say μ_1 *(weakly) dominates* μ_2 *for disadvantaged students* if $\mu_1(s) \geq_s \mu_2(s)$ for all disadvantaged students $s \in S^m$. If moreover $\mu_1 \neq \mu_2$ (i.e., there is at least one disadvantaged student $s \in S^m$ such that $\mu_1(s) >_s \mu_2(s)$), then we say μ_1 *Pareto dominates* μ_2 *for disadvantaged students*. Consider a student-school pair $(s, c) \in E$, it is a blocking pair of matching μ *for disadvantaged students* if $s \in S^m$, $c >_s \mu(s)$, and there exists a disadvantaged student $s' \in \mu(c) \cap S^m$ such that $s >_c s'$; and it is a blocking pair of matching μ *for advantaged students* if $s \in S^M$, $c >_s \mu(s)$, and there exists an advantaged student $s' \in \mu(c) \cap S^M$ such that $s >_c s'$. Collectively, a blocking pair is called an *in-group* blocking pair if it is a blocking pair for either disadvantaged or advantaged students.

Fix reservation quotas $\mathbf{q}^R$. A mechanism ϕ is *strategy-proof* if for any instance I and for any student $s \in S$, there is no preference list $\tilde{>}_s$ such that $\phi(\tilde{I}, \mathbf{q}^R)(s) >_s \phi(I, \mathbf{q}^R)(s)$, where $\tilde{I}$ is obtained from I by replacing $>_s$ with $\tilde{>}_s$. In other words, a mechanism is strategy-proof if no student has the incentive to misreport their preference list. As a stronger concept, a mechanism is *weakly group strategy-proof* if for any instance I and for any *group* of students $S_1 \subseteq S$, there are no preference lists $\{\tilde{>}_s : s \in S_1\}$ such that for every student $s \in S_1$, $\phi(\tilde{I}, \mathbf{q}^R)(s) >_s \phi(I, \mathbf{q}^R)(s)$, where $\tilde{I}$ is obtained from I by replacing $>_s$ with $\tilde{>}_s$ for every $s \in S_1$. That is, a mechanism is weakly group strategy-proof if no group of students can jointly misreport their preference lists so that everyone in the group is strictly better off. Note that if a mechanism is weakly group strategy-proof, it is strategy-proof.

Consider two mechanisms ϕ_1 and ϕ_2. If $\phi_1(I, \mathbf{q}^R)$ (weakly) dominates $\phi_2(I, \mathbf{q}^R)$ for disadvantaged students for all instances I, we say that mechanism ϕ_1 *(weakly) dominates* mechanism ϕ_2 for disadvantaged students. If neither ϕ_1 nor ϕ_2 dominates the other mechanism, we say they are *not comparable* or *incomparable*.

4.2.2 Choice functions

To unify the treatment of different affirmative action mechanisms, we use the concept of choice functions. Under each mechanism, every school $c \in C$ is endowed with a *choice function* $\mathcal{C}_c : 2^S \to 2^S$. Recall that for every subset of students $S_1 \subseteq S$, $\mathcal{C}_c(S_1)$ represents the students whom school c would like to admit among those in S_1. In particular, for every $S_1 \subseteq S$, we have $\mathcal{C}_c(S_1) \subseteq S_1$ and $|\mathcal{C}_c(S_1)| \leq q_c$. Choice function $\mathcal{C}_c$ is a function of the priority order $>_c$ and quotas q_c^R and q_c^G, and its exact definition depends on the specific mechanism (see Section 4.4). Students' preferences are still described by a strict order over a subset of schools.

For all the affirmative action mechanisms studied in this chapter, every school c's choice function $\mathcal{C}_c$ satisfies the following properties: *substitutability*, *consistency*, and q_c*-acceptance*[9]. Thus, for the rest of the chapter, unless otherwise specified, these properties are always assumed. See Chapter 3, Section 3.2.2 for their definitions. For some mechanisms, $\mathcal{C}_c$ is additionally q_c-*responsive*, which we define in the following. For any nonnegative integer q, a priority order over the students $>$, and a subset of students $S_1 \subseteq S$, let $\max(S_1, >, q)$ denote the $\min(q, |S_1|)$ highest ranked students (i.e., students with the highest priorities) of S_1 according to the priority order $>$.

[9] q_c-acceptance is also referred to as *quota-filling*. However, we use q_c-acceptance in this chapter to highlight the quota.

Definition 4.1 (q_c-responsive)**.** Choice function $\mathcal{C}_c$ is q_c-responsive if there exists a priority order $>$ over the students such that for any set of students S_1, $\mathcal{C}_c(S_1) = \max(S_1, >, q_c)$. In such case, we say $\mathcal{C}_c$ is *induced* by priority order $>$ (and quota q_c).

We further note that q-responsiveness implies substitutability, consistency, and q-acceptance. Indeed, q-responsive choice functions are the "simplest" choice functions and are mostly studied in the matching literature, including the seminal work by Gale and Shapley (1962) and in practical school choice (Abdulkadiroğlu, Pathak, and Roth, 2005b; Abdulkadiroğlu et al., 2005).

4.2.3 Stable matchings

Consider an arbitrary collection of schools' choice functions $\mathcal{C} := \{\mathcal{C}_c : c \in C\}$. Note that the q_c-acceptant property implies that for every school c, we must have $\mathcal{C}_c(\mu(c)) = \mu(c)$ by any matching μ by the definition of matchings. A matching μ is *stable* (in instance I under choice functions $\mathcal{C}$) if there is no student-school pair $(s, c) \in E$ such that $c >_s \mu(s)$ and $s \in \mathcal{C}_c(\mu(c) \cup \{s\})$. When such a student-school pair exists, we call it a *blocking pair* of μ, or we say that the edge (or pair) *blocks* μ. Note that the definition of matchings only depends on the instance, not on the choice functions; whereas the definition of stability depends on both.

When the choice function is q_c-responsive (i.e., induced by a priority order and a quota), the definition of stability with respect to choice functions is equivalent to the standard definition in the classical model without choice functions. In particular, the condition $s \in \mathcal{C}_c(\mu(c) \cup \{s\})$ can then be stated as: either school c's seats are not fully assigned (i.e., $|\mu(c)| < q_c$) or s has a higher priority over some students that are assigned to c (i.e., $\exists s' \in \mu(c)$ such that $s >_c s'$).

Among all stable matchings of a given instance and choice functions, there is one that *dominates* every stable matching, where matching μ_1 is said to *dominate* matching μ_2 if $\mu_1(s) \geq_s \mu_2(s)$ for all students $s \in S$. This stable matching is called the *student-optimal* stable matching, and it can be obtained by the *student-proposing deferred acceptance algorithm* (Gale and Shapley, 1962; Roth, 1984a), which we describe next. The algorithm runs in *rounds*. At each round k, every student applies to their most preferred school that has not rejected them; and every school c, with $S_c^{(k)}$

denoting the set of students who applied to it in the current round, *temporarily* accepts students in $\mathcal{C}_c(S_c^{(k)})$ and rejects the rest. The algorithm terminates when there is no rejection. For any instance I and choice functions $\mathcal{C}$, we denote by $\mathtt{SDA}(I, \mathcal{C})$ the matching output by the student-proposing deferred acceptance algorithm.

4.3 Affirmative Action Mechanisms

For the rest of the section, we fix an instance $I = (G, >, \mathbf{q})$ and reservation quotas $\mathbf{q}^R$. The choice functions of schools depend on the mechanisms, and we introduce them in details in each subsection. We also discuss the features of the mechanisms in their corresponding subsections.

4.3.1 No affirmative action

The simplest mechanism is the one without affirmative action. That is, schools do not distinguish students of different types. The choice function of school c under the no affirmative action mechanism is q_c-responsive, simply induced from its priority order: for all subset of students $S_1 \subseteq S$,

$$\mathcal{C}_c^{\mathtt{noAA}}(S_1) := \max(S_1, >_c, q_c).$$

We denote by $\mu^{\mathtt{noAA}} := \mathtt{SDA}(I, \mathcal{C}^{\mathtt{noAA}})$ the matching under the no affirmative action mechanism. Although this matching can be obtained from the original and simpler deferred acceptance algorithm proposed by Gale and Shapley (1962), we present the mechanism from a choice function point of view so that it is consistent with later sections. The no affirmative action mechanism has the following two properties. Theorem 4.2 was shown by Dubins and Freedman (1981), and Proposition 4.3 is immediate from the fact that $\mu^{\mathtt{noAA}}$ admits no blocking pairs under $\mathcal{C}^{\mathtt{noAA}}$ and the definition of choice functions $\mathcal{C}^{\mathtt{noAA}}$.

Theorem 4.2 (Dubins and Freedman, 1981)**.** *The no affirmative action mechanism is weakly group strategy-proof.*

Proposition 4.3. $\mu^{\mathtt{noAA}}$ *does not admit in-group blocking pairs.*

4.3.2 Minority reserve

Under minority reserve, the choice function of every school $c \in C$, denoted by $\mathcal{C}_c^{\texttt{MR}}$, is defined as follows (Hafalir, Yenmez, and Yildirim, 2013): for every subset of students $S_1 \subseteq S$,

$$\mathcal{C}_c^{\texttt{MR}}(S_1) = \underbrace{\max(S_1 \cap S^m, >_c, q_c^R)}_{=:S_1^R;\text{ reserved seats}} \,\dot{\cup}\, \underbrace{\max\left(S_1 \setminus S_1^R, >_c, q_c - |S_1^R|\right)}_{\text{remaining seats}}.$$

That is, every school first accepts disadvantaged students from its pool of candidates up to its reservation quota, and then fills up the remaining seats from the remaining candidates. Note that if there is a shortage of disadvantage students (i.e., $|S_1 \cap C^m| < q_c^R$), then the remaining reserved seats become open to advantaged students.

Proposition 4.4. *Choice function $\mathcal{C}_c^{\texttt{MR}}$ is substitutable, consistent, and q_c-acceptant.*

Proof of Proposition 4.4. The substitutability property was shown in Hafalir, Yenmez, and Yildirim, 2013, but we include the proof here for completeness. Let $S_1 \subseteq S$ be a subset of students, $s \in \mathcal{C}_c^{\texttt{MR}}(S_1)$ be a student selected by the choice function, and S_2 be a subset of students such that $s \in S_2 \subseteq S_1$. We want to show that $s \in \mathcal{C}_c^{\texttt{MR}}(S_2)$. Consider the following two cases. The first case is when $s \in S_1^R$. Here, it is immediate that $s \in S_2^R \coloneqq \max(S_2 \cap S^m, >_c, q_c^R)$ since $S_2 \cap S^m \subseteq S_1 \cap S^m$ and thus, $s \in \mathcal{C}_c^{\texttt{MR}}(S_2)$. The other case is when $s \in \mathcal{C}_c^{\texttt{MR}}(S_1) \setminus S_1^R$. Our argument for the first case implies that $S_1^R \cap S_2 \subseteq S_2^R$ and thus, we have $S_2 \setminus S_2^R \subseteq S_2 \setminus S_1^R \subseteq S_1 \setminus S_1^R$. Hence, we also have $s \in \mathcal{C}_c^{\texttt{MR}}(S_2)$.

Next, for consistency, let S_2 be a subset of students with $\mathcal{C}_c^{\texttt{MR}}(S_1) \subseteq S_2 \subseteq S_1$, and we want to show that $\mathcal{C}_c^{\texttt{MR}}(S_1) = \mathcal{C}_c^{\texttt{MR}}(S_2)$. By the definition of the choice function, it is clear that $S_1^R = S_2^R$ since $S_1^R \subseteq S_2$. With the same reasoning, we additionally have $\max(S_1 \setminus S_1^R, >_c, q_c - |S_1^R|)) = \max(S_2 \setminus S_1^R, >_c, q_c - |S_1^R|)) = \max(S_2 \setminus S_2^R, >_c, q_c - |S_2^R|))$. Therefore, the claim follows.

Lastly, for q_c-acceptance, we first have that $|\mathcal{C}_c^{\texttt{MR}}(S_1)| \leq |S_1^R| + q_c - |S_1^R| = q_c$, where the inequality follows directly from the definition. It remains to show that when $|S_1| < q_c$, we have $\mathcal{C}_c^{\texttt{MR}}(S_1) = S_1$. This is immediate from the definition of the choice function. ■

Since substitutability and consistency guarantee the existence of stable matchings (Aygün and Sönmez, 2013; Hatfield and Milgrom, 2005; Roth, 1984a), stable matchings exist under choice functions $\mathcal{C}^{\texttt{MR}}$ and we denote by $\mu^{\texttt{MR}} := \texttt{SDA}(I, \mathcal{C}^{\texttt{MR}})$ the matching under minority reserve with reservation quotas $\mathbf{q}^R$. Minority reserve has several desirable properties, which we formally state below. Theorem 4.6 states that at least one disadvantaged student is not worse off when compared to the no affirmative action mechanism, which is not necessarily true for other affirmative action mechanisms (see, e.g., Kojima, 2012); and Theorem 4.7 states that when the reservation quotas are "carefully" chosen, no disadvantaged student is worse off. Reservation quotas $\mathbf{q}^R$ are said to be a *smart reserve* if $q_c^R \geq |\mu^{\texttt{noAA}}(c)|$ for all $c \in C$.

Proposition 4.5 (Hafalir, Yenmez, and Yildirim, 2013)**.** *Minority reserve is weakly group strategy-proof.*

Theorem 4.6 (Hafalir, Yenmez, and Yildirim, 2013)**.** *For any reservation quota $\mathbf{q}^R$, there exists a disadvantaged student $s \in S^m$ such that $\mu^{\texttt{MR}}(s) \geq_s \mu^{\texttt{noAA}}(s)$.*

Theorem 4.7 (Hafalir, Yenmez, and Yildirim, 2013)**.** *If the reservation quotas $\mathbf{q}^R$ is a smart reserve, then $\mu^{\texttt{MR}}$ dominates $\mu^{\texttt{noAA}}$ for disadvantaged students.*

The following claim follows directly from the fact that $\mu^{\texttt{MR}}$ is stable under choice functions $\mathcal{C}^{\texttt{MR}}$ and the definition of $\mathcal{C}^{\texttt{MR}}$.

Proposition 4.8. *$\mu^{\texttt{MR}}$ does not admit in-group blocking pairs.*

Proof of Propsoition 4.8. Assume by contradiction that (s, c) is an in-group blocking pair for $\mu^{\texttt{MR}}$. Let s' be the student in the same group as s such that $s' \in \mu^{\texttt{MR}}(c)$ and $s >_c s'$. Then, by definition of $\mathcal{C}_c^{\texttt{MR}}$, we have $s \in \mathcal{C}_c^{\texttt{MR}}(\mu^{\texttt{MR}}(c) \cup \{s\})$, which means (s, c) is a blocking pair for $\mu^{\texttt{MR}}$. However, this contradicts stability of $\mu^{\texttt{MR}}$. ■

4.3.3 Discovery program

This mechanism is adapted from the mechanism used by NYC DOE for increasing the number of disadvantaged students at the city's eight specialized schools, which are considered to be the best public schools. Instead of distributing reserved seats to disadvantaged students at the beginning as the minority reserve (i.e., to top ranked disadvantaged students), the discovery program mechanism[10] distributes reserved seats to disadvantaged student at the end of seat-assignment procedure. One of the reasons for allocating reserved seats to lower ranked disadvantages students is that disadvantaged students who are admitted via reserved seats are required to participate in a 3-weeks *summer enrichment program* as a preparation for the specialized high schools.

However, for the sake of comparison (with other mechanisms), we assume that students' preference for schools are not affected by whether they are required to participate in the summer program – that is, students are indifferent between general and reserved seats at each school. We assume this *school-over-seat* hypothesis for the rest of the chapter, and we discuss its validity in the Appendix, Section B.2.

When there is a shortage of disadvantaged students, reserved seats could go unassigned under the discovery program mechanism. Although this is usually not of concern in real-world applications, since there are usually more students than available seats, we nevertheless present the discovery program mechanism in a more general case where vacant reserved seats are de-reserved.

The algorithm for the discovery program mechanism has three stages. Schools' choice functions at all stages are the simple q-responsive choice function $\mathcal{C}^{\texttt{noAA}}$. The mechanism starts by running the deferred acceptance algorithm on instance $(G, >, \mathbf{q}^G)$ to obtain matching $\mu_1^{\texttt{DISC}}$ for the general seats; it then runs the deferred acceptance algorithm on the instance restricted to the disadvantaged students that are not yet assigned $(G[C \cup \{s \in S^m : \mu_1^{\texttt{DISC}}(s) = \emptyset\}], >, \mathbf{q}^R)$ to obtain matching $\mu_2^{\texttt{DISC}}$ for reserved seats; and it lastly runs the deferred acceptance algorithm on the instance restricted to the advantaged students that are not yet assigned $(G[C \cup \{s \in S^M : \mu_1^{\texttt{DISC}}(s) = \emptyset\}, >, \mathbf{q}^E)$ with $q_c^E = q_c^R - |\mu_2^{\texttt{DISC}}(c)|$ $\forall c \in C$ to obtain matching $\mu_3^{\texttt{DISC}}$ for vacant reserved seats. The final matching combines the matchings obtained at these three stages: $\mu^{\texttt{DISC}} := \mu_1^{\texttt{DISC}} \dot\cup \mu_2^{\texttt{DISC}} \dot\cup \mu_3^{\texttt{DISC}}$.

Although the mechanism intends to help disadvantaged students, it could actually hurt them.

[10] `https://www.schools.nyc.gov/enrollment/enrollment-help/meeting-student-needs/diversity-in-admissions`

As we show through Example 4.9, under the discovery program mechanism, it is possible that all disadvantaged students are worse off.

Example 4.9. Consider the instance with students $S^M = \{s_1^M, s_2^M\}$, $S^m = \{s_1^m\}$ and schools $C = \{c_1, c_2\}$. The quotas of schools are $q_{c_1} = 2$ and $q_{c_2} = 1$, and both schools have priority order $s_1^M > s_2^M > s_1^m$. Both advantaged students prefer c_1 to c_2, whereas the disadvantaged student prefers c_2 to c_1. It is easy to see that under the no affirmative action mechanism,

$$\mu^{\texttt{noAA}} = \{(s_1^M, c_1), (s_2^M, c_1), (s_1^m, c_2)\}.$$

Now consider the discovery program mechanism with reservation quotas $q_{c_1}^R = 1$ and $q_{c_2}^R = 0$. Then,

$$\mu^{\texttt{DISC}} = \{(s_1^M, c_1), (s_2^M, c_2), (s_1^m, c_1)\}.$$

Under the discovery program mechanism, the disadvantaged student s_1^m is not only assigned to a school less preferred less, but is also now required to participate in the summer program. $\triangle$

Moreover, the discovery program mechanism could create blocking pairs for disadvantaged students, incentivize disadvantaged students to misrepresent their preference lists, and might hurt disadvantaged students even when the reservation quotas are a smart reserve. See the example below.

Example 4.10. Consider the instance with students $S^M = \{s_1^M, s_2^M, s_3^M\}$, $S^m = \{s_1^m, s_2^m, s_3^m\}$ and schools $C = \{c_1, c_2\}$. The quotas of schools are $q_{c_1} = 3$ and $q_{c_2} = 2$, and both schools have priority order $s_1^M > s_2^M > s_1^m > s_3^M > s_2^m > s_3^m$. All students prefer c_1 to c_2. Without affirmative action, we have

$$\mu^{\texttt{noAA}}(c_1) = \{s_1^M, s_2^M, s_1^m\}, \quad \mu^{\texttt{noAA}}(c_2) = \{s_3^M, s_2^m\}.$$

Now assume that the reservation quotas are $q_{c_1}^R = q_{c_2}^R = 1$, which in particular is a smart reserve.

Under the discovery program mechanism with these reservation quotas, we have

$$\mu^{\texttt{DISC}}(c_1) = \{s_1^M, s_2^M, s_2^m\}, \quad \mu^{\texttt{DISC}}(c_2) = \{s_1^m, s_3^m\}.$$

Disadvantaged student s_1^m is worse off under $\mu^{\texttt{DISC}}$ than under $\mu^{\texttt{noAA}}$. In addition, $\mu^{\texttt{DISC}}$ admits a blocking pair (s_1^m, c_1) for disadvantages students as s_1^m prefers c_1 to c_2 and s_1^m has a higher priority than s_2^m at c_1. Moreover, s_1^m has the incentive to misreport the preference list: if s_1^m were to report the preference list as $c_1 > \emptyset$, the matching under the discovery program mechanism would have been the same as $\mu^{\texttt{noAA}}$. $\triangle$

4.3.4 Joint seat allocation

The mechanism of joint seat allocation we introduce here is inspired by the mechanism used for admission to Indian Institutes of Technology (JoSAA, 2020). It allocates the general and reserved seats at the same time, while only allowing disadvantaged students to take the reserved seats when they cannot get admitted via the general seats. Under this mechanism, the choice function of every school $c \in C$, denoted by $\mathcal{C}_c^{\texttt{JSA}}$, is defined as follows. For every subset of students $S_1 \subseteq S$,

$$\mathcal{C}_c^{\texttt{JSA}}(S_1) = \underbrace{\max(S_1, >_c, q_c^G)}_{=:S_1^G;\text{ general seats}} \ \dot{\cup}\ \underbrace{\max\left(S_1 \cap S^m \setminus S_1^G, >_c, q_c^R\right)}_{=:S_1^R;\text{ reserved seats}} \ \dot{\cup}\ \underbrace{\max(S_1 \setminus (S_1^G \cup S_1^R), >_c, q_c - |S_1^G \cup S_1^R|)}_{\text{remainning seats}}.$$

A prominent distinction between joint seat allocation and minority reserve is that in the former, "highly ranked" disadvantaged students are admitted via general seats and do not take up the quotas for reserved seats. Intuitively, this opens up more opportunities for disadvantaged students and one would expect all disadvantaged students to be weakly better off under joint seat allocation than under minority reserve. This is true for instances where the competition for seats is high, but is not true for general instances. See Section 4.4 and Theorem 4.20 for more discussions on the comparison between these two mechanisms.

Proposition 4.11. *Choice function $\mathcal{C}_c^{\texttt{JSA}}$ is substitutable, consistent, and q_c-acceptant.*

Proof of Proposition 4.11. The proof steps are similar to that of Proposition 4.4 for minority reserve. Let $S_1 \subseteq S$ be a subset of students. First, for substitutability, let $s \in \mathcal{C}_c^{\texttt{JSA}}(S_1)$ and let S_2 be a subset of students such that $s \in S_2 \subseteq S_1$. We want to show that $s \in \mathcal{C}_c^{\texttt{JSA}}(S_2)$ and we consider the following three cases. The first case is when $s \in S_1^G$. In this case, it is immediate that $s \in S_2^G := \max(S_2, >_c, q_c^G)$ since $S_2 \subseteq S_1$. This first case in particular implies that $S_1^G \cap S_2 \subseteq S_2^G$ and thus, $S_2 \setminus S_2^G \subseteq S_2 \setminus S_1^G \subseteq S_1 \setminus S_1^G$. Hence, in the second case where $s \in S_1^R$, we similarly have $s \in S_2^R := \max(S_2 \cap S^m \setminus S_2^G, >_c, q_c^R)$. Note that this argument for the second case also implies that $S_2 \setminus (S_2^G \cup S_2^R) \subseteq S_1 \setminus (S_1^G \cup S_1^R)$. Hence, for the last case where $s \in \max(S_1 \setminus (S_1^G \cup S_1^R), >_c, q_c - |S_1^G \cup S_1^R|)$, we also have $s \in \max(S_2 \setminus (S_2^G \cup S_2^R), >_c, q_c - |S_2^G \cup S_2^R|)$. Therefore, in all these three cases, we have $s \in \mathcal{C}_c^{\texttt{JSA}}(S_2)$ and thus $\mathcal{C}_c^{\texttt{JSA}}$ is substitutable.

Next, for consistency, let S_2 be a subset of students with $\mathcal{C}_c^{\texttt{JSA}}(S_1) \subseteq S_2 \subseteq S_1$, and we want to show that $\mathcal{C}_c^{\texttt{JSA}}(S_1) = \mathcal{C}_c^{\texttt{JSA}}(S_2)$. By the definition of the choice function, it is clear that $S_1^G = S_2^G$ since $S_1^G \subseteq S_2$. Moreover, we have $S_1^R = S_2^R$ since $S_1^R \subseteq S_2 \cap S^m \setminus S_2^G$. With the same reasoning, we additionally have that $\max(S_1 \setminus (S_1^G \cup S_1^R), >_c, q_c - |S_1^G \cup S_1^R|) = \max(S_2 \setminus (S_2^G \cup S_2^R), >_c, q_c - |S_2^G \cup S_2^R|)$. Therefore, the choice function is consistent.

Lastly, for q_c-acceptant, we first have that $|\mathcal{C}_c^{\texttt{JSA}}(S_1)| \leq |S_1^G| + |S_1^R| + q_c - |S_1^G| - |S_1^R| = q_c$, where the inequality follows directly from the definition. It remains to show that when $|S_1| < q_c$, we have $\mathcal{C}_c^{\texttt{JSA}}(S_1) = S_1$. This is immediate from the definition of the choice function. ■

Proposition 4.11 implies that stable matchings exist under joint seat allocation, and we denote the student-optimal stable matching by $\mu^{\texttt{JSA}} := \texttt{SDA}(I, \mathcal{C}^{\texttt{JSA}})$.

All positive results of minority reserve extend to joint seat allocation. We formalize the statements below. The proof of Proposition 4.12 and Theorem 4.13 follow by constructing an equivalent instance where, in particular, students have preference lists over general and reserved seats at different schools. This idea is similar to that given in Hafalir, Yenmez, and Yildirim (2013), but the equivalent instances are different under minority reserve and joint seat allocation (see Section 4.4.3 for details, where we additionally construct similar equivalent instances for the discovery program mechanism). The main reason for establishing such equivalent instances is that it allows us to directly use the strategy-proofness result for the classical stable matching model (i.e., no affirmative action).

Proposition 4.12. *Joint seat allocation is weakly group strategy-proof.*

Proof of Proposition 4.12. Assume by contradiction that there exists a group of students $S_1 \subseteq S$ who can jointly misreport their preference lists so that every one in S_1 is strictly better off. Now consider the auxiliary instance introduced in Section 4.4.3, where the relative ranking of schools by each student remains the same as that of the original instance. As a result, this strategic behavior by S_1 can be translated to a strategic behavior in the auxiliary instance due to Proposition 4.24. That is, $S_1 \subseteq S$ can accordingly misreport their preferences lists in the auxiliary instance so that every one in S_1 is better off. However, this contradicts Theorem 4.2, which states that strategic behaviors are not possible in the auxiliary instance. This concludes the proof. ■

Theorem 4.13. *For any reservation quota $\mathbf{q}^R$, there exists a disadvantaged student $s \in S^m$ such that $\mu^{\texttt{JSA}}(s) \geq_s \mu^{\texttt{noAA}}(s)$.*

Proof of Theorem 4.13. Assume by contradiction that there is reservation quotas $\mathbf{q}^R$ such that $\mu^{\texttt{noAA}}(s) >_s \mu^{\texttt{JSA}}(s)$ for every disadvantaged student $s \in S^m$. Then, consider an alternative instance where every disadvantaged student s misreports his or her preference list where $\mu^{\texttt{noAA}}(s)$ is the only acceptable school. Let $\tilde{G}$ and $\tilde{>}_S$ be the resulting graph and preference lists of the students. In the following, we consider the alternative instance $\tilde{I} = (\tilde{G}, \tilde{>}_S, >_C, \mathbf{q})$ and we claim that $\mu^{\texttt{noAA}}$ is stable in instance $\tilde{I}$ under choice functions $\mathcal{C}^{\texttt{JSA}}$. Assume by contradiction that $\mu^{\texttt{noAA}}$ admits a blocking pair (s, c). Since all disadvantaged students are matched to their first choice, it must be that $s \in S^M$. Then, $s \in \mathcal{C}_c^{\texttt{JSA}}(\mu^{\texttt{noAA}}(c) \cup \{s\})$ implies that there is a student $s' \in \mu^{\texttt{noAA}}(c)$ such that $s >_c s'$. However, this means $s \in \mathcal{C}^{\texttt{noAA}}(\mu^{\texttt{noAA}}(c) \cup \{s\})$, which contradicts stability of $\mu^{\texttt{noAA}}$ in the original instance I under choice functions $\mathcal{C}^{\texttt{noAA}}$. Hence, $\mu^{\texttt{noAA}}$ is stable in instance $\tilde{I}$ with choice functions $\mathcal{C}^{\texttt{JSA}}$. Since $\texttt{SDA}(\tilde{I}, \mathcal{C}^{\texttt{JSA}})$ is the student-optimal stable matching, it dominates $\mu^{\texttt{noAA}}$ and thus, every disadvantaged student is strictly better off under $\texttt{SDA}(\tilde{I}, \mathcal{C}^{\texttt{JSA}})$ as compared to $\mu^{\texttt{JSA}}$.

However, this contradicts Proposition 4.12 which states that the joint seat allocation mechanism is weakly group strategy-proof. ■

For the following theorem, we give a novel proof that directly follow the procedure of the deferred acceptance algorithm and use the properties of choice functions $\mathcal{C}^{\texttt{JSA}}$. Our approach is different from the one given in Hafalir, Yenmez, and Yildirim, 2013 for the similar property of minority reserve.

Theorem 4.14. *If the reservation quotas are a smart reserve, then $\mu^{\texttt{JSA}}$ dominates $\mu^{\texttt{noAA}}$ for disadvantaged students.*

Proof of Theorem 4.14. Assume by contradiction that there exists disadvantaged students s with $\mu^{\texttt{noAA}}(s) >_s \mu^{\texttt{JSA}}(s)$. Let s_1 be the *first* disadvantaged student that is rejected by $c_1 := \mu^{\texttt{noAA}}(s_1)$ during the deferred acceptance algorithm on instance I with choice functions $\mathcal{C}^{\texttt{JSA}}$. Assume this rejection happens at round k. Let $S_k^{\texttt{JSA}}$ denote the set of students who apply to school c_1 during round k. In addition, let $S^{\texttt{noAA}}$ denote the set of students who have ever applied to c_1 throughout the deferred acceptance on instance I with choice functions $\mathcal{C}^{\texttt{noAA}}$. It has been shown in Roth, 1984a that $\mathcal{C}_{c_1}^{\texttt{noAA}}(S^{\texttt{noAA}}) = \mu^{\texttt{noAA}}(c_1)$. Thus, $s_1 \in \max(S^{\texttt{noAA}} \cap S^m, >_{c_1}, q_{c_1}^R)$ by definition of choice function $\mathcal{C}_{c_1}^{\texttt{noAA}}$ and the assumption that the reservation quotas are a smart reserve (i.e., $q_{c_1}^R \geq |\mu^{\texttt{noAA}}(c_1)|$). Moreover, by our choice of s_1, we have $S_k^{\texttt{JSA}} \cap S^m \subseteq S^{\texttt{noAA}} \cap S^m$. Therefore, $s_1 \in \max(S_k^{\texttt{JSA}} \cap S^m, >_{c_1}, q_{c_1}^R)$, which then implies $s_1 \in \mathcal{C}_{c_1}^{\texttt{JSA}}(S_k^{\texttt{JSA}})$ by definition of choice function $\mathcal{C}_{c_1}^{\texttt{JSA}}$. However, this contradicts our assumption that s_1 is rejected by c_1 at round k, concluding the proof. ■

When the reservation quota is not a smart reserve, it is possible that $\mu^{\texttt{noAA}}$ Pareto dominates $\mu^{\texttt{JSA}}$ for disadvantaged students, which can be readily seen from the same example for minority reserve presented in Hafalir, Yenmez, and Yildirim (2013). See Example 4.15 below.

Example 4.15. Consider the instance with students $S^M = \{s_1^M\}$, $S^m = \{s_1^m, s_2^m\}$ and schools $C = \{c_1, c_2, c_3\}$, each with a quota of 1. All schools have priority order $s_1^M > s_1^m > s_2^m$. Students' preference lists are given below:

s_1^M	s_1^m	s_2^m
c_1	c_3	c_1
c_3	c_1	c_2

Without affirmative action, the resulting matching is

$$\mu^{\texttt{noAA}} = \{(s_1^M, c_1), (s_2^m, c_2), (s_1^m, c_3)\}.$$

Consider the reservation quotas $q_{c_1}^R = 1$ and $q_{c_2}^R = q_{c_3}^R = 0$. Then,

$$\mu^{\texttt{MR}} = \mu^{\texttt{JSA}} = \{(s_1^m, c_1), (s_2^m, c_2), (s_1^M, c_3)\}.$$

Disadvantaged student s_2^m is indifferent between the two matchings, but disadvantaged student s_1^m strictly prefers $\mu^{\texttt{noAA}}$ to $\mu^{\texttt{JSA}}$. That is, $\mu^{\texttt{noAA}}$ Pareto dominates $\mu^{\texttt{JSA}}$ for disadvantaged students. $\triangle$

As Proposition 4.8, the following claim follows directly from the fact that $\mu^{\texttt{JSA}}$ is stable under choice functions $\mathcal{C}^{\texttt{JSA}}$ and the definition of $\mathcal{C}^{\texttt{JSA}}$.

Proposition 4.16. *$\mu^{\texttt{JSA}}$ does not admit in-group blocking pairs.*

Proof of Proposition 4.16. Assume by contradiction that (s, c) is an in-group blocking pair. Let s' be the student in the same group as s such that $s' \in \mu^{\texttt{JSA}}(c)$ and $s >_c s'$. Then, by definition of $\mathcal{C}_c^{\texttt{JSA}}$, we have $s \in \mathcal{C}_c^{\texttt{JSA}}(\mu^{\texttt{JSA}}(c) \cup \{s\})$, which means (s, c) is a blocking pair of $\mu^{\texttt{JSA}}$. However, this contradicts stability of $\mu^{\texttt{JSA}}$. ■

4.4 Comparison of Affirmative Action Mechanisms

In this section, we investigate how different mechanisms introduced in the previous section compare with each other.

4.4.1 Is there a winning mechanism for disadvantaged students?

To begin with, we would like to answer the following question regarding any two mechanisms: does one mechanism dominate the other mechanism for disadvantaged students? We consider three domains which impose restrictions on the instance or the reservation quotas. They are: (1) the reservation quotas are a smart reserve, (2) schools share a common priority order over the students (i.e., *universal* priority order), and (3) both smart reserve and universal priority order. We summarized the results in Table 4.2. Note that for a pair of mechanisms, a positive answer for (1) or (2) implies a positive answer for (3) and a negative answer for (3) implies negative answers for both (1) and (2). These allow us to simplify the presentations given in Table 4.2.

From Table 4.2, we can see that no two mechanisms are comparable in the general domain (i.e., all instances included). In addition, even in the restricted domains, most of the mechanisms are not comparable, with the exception that minority reserve and joint seat allocation dominate the no affirmative action mechanism when the reservation quotas are a smart reserve.

These results are show as follows. We first observe that `noAA` does not dominate the other mechanisms, through a rather trivial example below.

Example 4.17. Consider the instance with students $S^M = \{s_1^M\}$, $S^m = \{s_1^m, s_2^m\}$ and schools $C = \{c_1, c_2\}$. Both schools have a quota of 1, and a reservation quota of 1. All students prefer school c_1 to c_2. Both schools have priority order $s_1^M > s_1^m > s_2^m$. Then,

$$\mu^{\texttt{noAA}} = \{s_1^M, c_1\}, \{s_1^m, c_2\}, \text{ and } \mu^{\texttt{MR}} = \mu^{\texttt{DISC}} = \mu^{\texttt{JSA}} = \{s_1^m, c_1\}, \{s_2^m c_2\}.$$

That is, the matching under any of the mechanisms with affirmative action Pareto dominates the matching obtained without affirmative action for disadvantaged students. △

We then show, through Example 4.18 and Example 4.19, that the three affirmative action mechanisms are not comparable.

Example 4.18. Consider the instance with students $S^M = \{s_1^M, s_2^M\}$, $S^m = \{s_1^m, s_2^m\}$ and schools

$C = \{c_1, c_2\}$. Both schools have a quota of 2 and a reservation quota of 1. All students prefer school c_1 to c_2, and all schools have priority order $s_1^M > s_1^m > s_2^M > s_2^m$. Then,

$$\mu^{\texttt{noAA}} = \mu^{\texttt{MR}} = \mu^{\texttt{JSA}} = \{s_1^M, c_1\}, \{s_1^m, c_1\}, \{s_2^M, c_2\}, \{s_2^m, c_2\},$$

and

$$\mu^{\texttt{DISC}} = \{s_1^M, c_1\}, \{s_2^m, c_1\}, \{s_1^m, c_2\}, \{s_2^M, c_2\}.$$

Note that the reservation quotas is a smart reserve. Disadvantaged student s_2^m strictly prefers $\mu^{\texttt{DISC}}$ to the other matching, while s_1^m strictly prefers the other matching to $\mu^{\texttt{DISC}}$. $\triangle$

Example 4.19. Consider the instance with students $S^M = \{s_1^M, s_2^M, s_3^M\}$, $S^m = \{s_1^m, s_2^m, s_3^m, s_4^m\}$ and schools $C = \{c_1, c_2, c_3, c_4\}$. The quotas and reservation quotas of schools, and the preference lists of students are given below.

c	c_1	c_2	c_3	c_4
q_c	1	1	1	2
q_c^R	1	1	0	1

s_1^M	s_2^M	s_3^M	s_1^m	s_2^m	s_3^m	s_4^m
c_2	c_1	c_4	c_2	c_4	c_3	c_4
	c_3	c_3	c_1			

All schools have priority order $s_1^M > s_1^m > s_2^M > s_2^m > s_3^M > s_3^m > s_4^m$. To see that the reservation quotas is a smart reserve, the matching under the no affirmative action mechanism is

$$\mu^{\texttt{noAA}} = \{s_1^m, c_1\}, \{s_1^M, c_2\}, \{s_2^M, c_3\}, \{s_2^m, c_4\}, \{s_3^M, c_4\}.$$

The matchings under minority reserve and joint seat allocation are:

$$\mu^{\texttt{MR}} = \{s_2^M, c_1\}, \{s_1^m, c_2\}, \{s_3^m, c_3\}, \{s_2^m, c_4\}, \{s_3^M, c_4\};$$
$$\mu^{\texttt{JSA}} = \{s_2^M, c_1\}, \{s_1^m, c_2\}, \{s_3^M, c_3\}, \{s_2^m, c_4\}, \{s_4^m, c_4\}.$$

Disadvantaged student s_1^m and s_2^m are indifferent between $\mu^{\texttt{MR}}$ and $\mu^{\texttt{JSA}}$, s_3^m strictly prefers $\mu^{\texttt{MR}}$ to

$\mu^{\texttt{JSA}}$, but s_4^m strictly prefers $\mu^{\texttt{JSA}}$ to $\mu^{\texttt{MR}}$. $\triangle$

4.4.2 Joint seat allocation vs minority reserve: the high competitiveness hypothesis

To further compare minority reserve and joint seat allocation, we consider a special condition on the market, that we term *high competitiveness of the market:*

$$|\mu^{\texttt{MR}}(c) \cap S^m| \leq q_c^R \text{ for every school } c \in C.$$

Note that this is an ex-post condition that is based on the outcome $\mu^{\texttt{MR}}$ of a specific mechanism – namely, minority reserve. In particular, this condition asks that minority students not occupy general seats in the matching $\mu^{\texttt{MR}}$.

Under the high competitiveness hypothesis, joint seat allocation dominates minority reserve for disadvantaged students. We formalize the statement in Theorem 4.20.

Theorem 4.20. *If $\mu^{\texttt{MR}}$ satisfies that for every school $c \in C$, $|\mu^{\texttt{MR}}(c) \cap S^m| \leq q_c^R$ (*high competitiveness hypothesis*), then $\mu^{\texttt{JSA}}$ dominates $\mu^{\texttt{MR}}$ for disadvantaged students.*

Proof of Theorem 4.20. Assume by contradiction there exists disadvantaged students s such that $\mu^{\texttt{MR}}(s) >_s \mu^{\texttt{JSA}}(s)$. Consider the execution of the deferred acceptance algorithm with choice functions $\mathcal{C}^{\texttt{JSA}}$, and let s_1 be the first disadvantaged student who is rejected by $\mu^{\texttt{MR}}(s_1) := c_1$. Assume this rejection happens at round k of the deferred acceptance algorithm. Let $S_k^{\texttt{JSA}}$ denote the set of students who apply to school c_1 during round k. In addition, let $S^{\texttt{MR}}$ denote the set of students who have ever applied to school c_1 during the execution of the deferred acceptance algorithm with choice functions $\mathcal{C}^{\texttt{MR}}$. It has been shown in Roth, 1984a that $\mathcal{C}_{c_1}^{\texttt{MR}}(S^{\texttt{MR}}) = \mu^{\texttt{MR}}(c_1)$, which then implies that $s_1 \in \max(S^{\texttt{MR}} \cap S^m, >_{c_1}, q_{c_1}^R)$ by definition of choice function $\mathcal{C}_{c_1}^{\texttt{MR}}$ and our assumption that $|\mu^{\texttt{MR}}(c_1)| \leq q_{c_1}^R$. Moreover, our choice of student s_1 implies that $S_k^{\texttt{JSA}} \cap S^m \subseteq S^{\texttt{MR}} \cap S^m$ and thus, we also have $s_1 \in \max(S_k^{\texttt{JSA}} \cap S^m, >_{c_1}, q_{c_1}^R)$. Therefore, $s_1 \in \mathcal{C}_{c_1}^{\texttt{JSA}}(S_k^{\texttt{JSA}})$ by definition of choice function $\mathcal{C}_{c_1}^{\texttt{JSA}}$. However, this contradicts our assumption that s_1 is rejected by c_1 at round k, concluding the proof. $\blacksquare$

High competitiveness can be connected to primitives of the market. Intuitively, it is satisfied when disadvantaged students are systematically performing worse than advantaged students and when there is a shortage of seats at all schools. In other words, this condition is satisfied if after the initial allocation of reserve seats to top ranked disadvantaged students, the remaining disadvantaged students are not able to compete with the advantaged students for general seats[11]. This condition is not uncommon in markets with limited resources.

In Section 4.5 we show empirically that, in particular, the market of NYC SHS is highly competitive using their admission data. Below we state a rigorous statement connecting primitives of the market and high competitiveness. We provide *ex-ante* conditions on *random* matching markets that guarantee the high competitive condition with high probability. The main proof idea is to connect the assignment problem of students to schools as the classic *balls into bins* problem, with schools as bins and students as balls.

Theorem 4.21. *Consider a family of markets with an increasing number of students and schools, where the preference lists of students are i.i.d. such that the probabilities of any two schools ranking first in a student's preference list coincide. Assume that schools have the same (reservation) quota and they share the same ranking of students, and that* $q - 1 > q^R > n \log n$*, where* n *is the number of schools. If, for some* $\epsilon \in (0, 1)$ *the* $(n \log n + (q - q^R)n \log \log n)$*-ranked advantaged student exists and is ranked above the* $(1-\epsilon)q^R n$*-ranked disadvantaged student (where rankings of students are within their respective groups), then the market is highly competitive with probability* $1 - o(1)$.

Proof of Theorem 4.21. Recall that, under MR, a student applies to her favorite school's reserved seats, and, if rejected, to the same school's non-reserved seat (see Section 4). We want to estimate the ranking, among disadvantaged students, of the *bottleneck* student – that is, the first disadvantaged student that is not admitted through a reserved seat at her most preferred school (hence, the student may either be admitted to her most preferred school via a general seat, or be admitted to

[11]High competitiveness is also satisfied in the trivial case when there are so many reserved seats, that all disadvantaged students get one, but this is rarely seen in the real world – and does not happen in our data from NYC SHSs.

another school, or not be admitted to any school).

We reformulate this problem in the classical balls in bins setting: given n bins and a series of balls, each inserted in exactly one bin chosen uniformly at random, which is the first ball k that is inserted in a bin with already q^R balls? Classical bounds (see, e.g., Raab and Steger, 1998) imply that, in the $q^R > n \log n$ regimen, $k \geq (1-\epsilon)q^R n$ with probability $1-o(1)$ for any $\epsilon \in (0,1)$ – in particular, for the ϵ from the hypothesis of the theorem. Interpreting schools as bins, disadvantaged students as balls, and assigning students to their most preferred schools as inserting balls to bins, we obtain that, with probability $1-o(1)$, the bottleneck student is ranked at least $(1-\epsilon)q^R n$ among disadvantaged students.

The market is highly competitive if and only if any disadvantaged student ranked at par or worse than the bottleneck student does not get a general seat in any school. For this to happen, the bottleneck student must be ranked worse than an advantaged student that we call *lucky applicant*. This is the worst-ranked advantaged student that would get a non-reserved seat in the market obtained from the original market with the number of seats being $q - q^R$, no reservation quota, and no disadvantaged student (call such a market *restricted*). So we want to compute the ranking, among advantaged students, of the lucky applicant. We can use again the balls and bins analogy from above. Denote by $b(q-q^R, n)$ the random variable denoting the smallest p such that, when ball p is extracted, all bins already have at least $(q-q^R)$ balls inserted. From Erdös and Rényi (1961), we know that for any real x, we have

$$\lim_{n\to\infty} \mathbb{P}(b(q-q^R,n) - 1 < n\log n + n(q-q^R-1)\log\log n + nx) = e^{-\frac{e^{-x}}{(q-q^R-1)!}}.$$

Taking $x = \log\log\log n$, we have

$$\begin{aligned}
&\lim_{n\to\infty} \mathbb{P}(b(q-q^R,n) - 1 < n\log n + n(q-q^R-1)\log\log n + n\log\log\log n) \\
&= \lim_{n\to\infty} e^{-\frac{e^{-\log\log\log n}}{(q-q^R-1)!}} \geq \lim_{n\to\infty} e^{-e^{-\log\log\log n}} = 1.
\end{aligned}$$

Hence, with probability $1-o(1)$, each school is ranked first at least $(q-q^R-1)$ times when we look at the preference lists of the best $n \log n + (q - q^R) n \log\log n$ advantaged students. Thus, with high probability, all the advantaged students that are admitted to a seat in the restricted market – in particular, the lucky applicant – are contained in the $(n \log n + (q - q^R) n \log\log n)$-best ranked advantaged students. It suffices therefore that the worst of them is ranked above the bottleneck student – as it is required by the hypothesis – to conclude that the market is highly competitive. ■

Let us discuss the hypothesis from Theorem 4.21. We restricted to markets where schools share a unique preference list of students. This condition applies, for instance, to the way universities rank incoming student across China and in Indian IITs, as well as in NYC SHSs. The condition on the the equal probability of each school appearing first in preference lists apply, for instance, in classical random markets, such as Knuth, Motwani, and Pittel (1990), Pittel (1989), and Pittel (1992). $q^R > n \log n$ applies when there are few schools compared to the number of seats, while the condition on the relative rankings of students applies when disadvantaged students perform systematically worse than advantaged students. For a comparison, in the data from NYC DOE, we have that the average reservation quota is $q^r = 208 > n = 8$, the average number of seats at each school is $q = 635$, $n + n(q - q^R) = 3424$, and $q^R n = 1664$. Omitting from the comparison the terms logarithmic and sublogarithmic in n ($n = 8$, hence they would only help the hypothesis of Theorem 4.21 to be satisfied), we see that the 1664-th ranked disadvantaged student performs at par with the 6848-th advantaged student, hence well within the hypothesis of the theorem.

4.4.3 Equivalent Interpretation

In this subsection, we take a different approach and instead of comparing the outputs. We compare how mechanisms interpret the inputs, and particularly how students' original preferences over schools are translated to their preferences over reserved and general seats at all schools.

We present alternative representations of the inputs under three mechanisms. That is, for each of the three matchings – $\mu^{\texttt{MR}}$, $\mu^{\texttt{DISC}}$, and $\mu^{\texttt{JSA}}$ – we show how to construct an auxiliary instance such that the matching corresponds to the student-optimal stable matching of the auxiliary instance without affirmative action.

The reason for developing these auxiliary instances is three-fold. First, it allows us to prove many of the properties (e.g., weakly group strategy-proofness) of the joint seat allocation mechanism, since we can now apply results developed for the classical stable matching model. Second, it completely removes the cost of implementing a new mechanism for the DOE. That is, the DOE does not need to develop a new algorithm incorporating choice functions, and can use the same algorithm as in their current system. Lastly, these auxiliary instances elucidate a simple difference of the three mechanisms: they differ in how students' preferences over general and reserved seats at all schools are extracted from their original preferences over schools.

We start by describing the common components of these auxiliary instances, which are the set of schools, their quotas, and their priority orders over the students. Every school $c \in C$ is divided into two schools c' and c'', where c' represents the part with general seats and has quota $q^{\mathsf{aux}}_{c'} := q_c - q^R_c$, and c'' is the part with reserved seats and has quota $q^{\mathsf{aux}}_{c''} := q^R_c$. Let $C^{\mathsf{aux}} = \{c' : c \in C\} \cup \{c'' : c \in C\}$ be the new set of schools after the division, and for every $c \in C^{\mathsf{aux}}$, let $\omega(c)$ denote its corresponding school in the original instance. Then, graph G^{aux} has vertices and edges:

$$V(G^{\mathsf{aux}}) = C^{\mathsf{aux}} \cup S, \text{ and } E(G^{\mathsf{aux}}) = \{(s, c) : s \in S, c \in C^{\mathsf{aux}}, (s, \omega(c)) \in E\}.$$

The priority order over the students by school c' is the same as that of school c (i.e., $>^{\mathsf{aux}}_{c'} = >_c$); and that by school c'' is defined as follows: for two students $s_1, s_2 \in S$,

$$s_1 >^{\mathsf{aux}}_{c''} s_2 \iff \begin{cases} s_1 \in S^m \text{ and } s_2 \in S^M; \text{ or} \\ s_1, s_2 \in S^m \text{ and } s_1 >_c s_2; \text{ or} \\ s_1, s_2 \in S^M \text{ and } s_1 >_c s_2. \end{cases}$$

The choice function $\mathcal{C}^{\mathsf{aux}}_c$ of every school $c \in C^{\mathsf{aux}}$ is q^{aux}_c-responsive and is simply induced from priority order $>^{\mathsf{aux}}_c$. We state the choice functions here to be consistent with our approach in previous sections. However, they are not necessary to obtain the student-optimal stable matching as the classical deferred acceptance algorithm suffice.

The only component remaining is the preference lists of students, which depends on the specific affirmative action mechanism, and we describe those next.

Minority reserve. The original preference list $c_1 >_s c_2 >_s \cdots >_s c_k$ of student s is modified as:

$$c_1'' >_s^{\texttt{MR-a}} c_1' >_s^{\texttt{MR-a}} c_2'' >_s^{\texttt{MR-a}} c_2' >_s^{\texttt{MR-a}} \cdots >_s^{\texttt{MR-a}} c_k'' >_s^{\texttt{MR-a}} c_k'.$$

Although the relative ranking of the schools remains the same, students prefer reserved seats to general seats. Let $I^{\texttt{MR-a}} := (G^{\text{aux}}, >_S^{\texttt{MR-a}}, >_C^{\text{aux}}, \mathbf{q}^{\text{aux}})$ denote the auxiliary instance, and let $\mu^{\texttt{MR-a}} := \texttt{SDA}(I^{\texttt{MR-a}}, \mathcal{C}^{\text{aux}})$ denote the student-optimal stable matching of the auxiliary instance.

Proposition 4.22 (Hafalir, Yenmez, and Yildirim, 2013)**.** *For every student $s \in S$, $\mu^{\texttt{MR}}(s) = \omega(\mu^{\texttt{MR-a}}(s))$.*

Discovery program. The original preference list $c_1 >_s c_2 >_s \cdots >_s c_k$ of student s becomes:

$$c_1' >_s^{\texttt{DISC-a}} c_2' >_s^{\texttt{DISC-a}} \cdots >_s^{\texttt{DISC-a}} c_k' >_s^{\texttt{DISC-a}} c_1'' >_s^{\texttt{DISC-a}} \cdots >_s^{\texttt{DISC-a}} c_k''.$$

Students prefer general seats over reserved seats; and within each type of seats, the ranking of the schools is the same as that of the original instance. Similarly, we denote the auxiliary instance by $I^{\texttt{DISC-a}} := (G^{\text{aux}}, >_S^{\texttt{DISC-a}}, >_C^{\text{aux}}, \mathbf{q}^{\text{aux}})$, and let $\mu^{\texttt{DISC-a}} := \texttt{SDA}(I^{\texttt{DISC-a}}, \mathcal{C}^{\text{aux}})$ denote the student-optimal stable matching of the auxiliary instance.

Proposition 4.23. *For every student $s \in S$, $\mu^{\texttt{DISC}}(s) = \omega(\mu^{\texttt{DISC-a}}(s))$.*

Proof of Proposition 4.23. To prove the proposition, instead of carrying out the deferred acceptance algorithm as we introduced in Section 4.2 based on Roth, 1984a for choice function models, we consider an equivalent execution of the algorithm when choice functions $\mathcal{C}$ are responsive. This algorithm was introduced by McVitie and Wilson (1971) and it similarly runs in rounds. The algorithm starts with all students unmatched. In every round, one student s who is not (temporarily) matched applies to his or her most preferred school c that has not yet rejected him or her. Let S_c denote the set of students c has temporarily accepted at the end of the previous round. School c

temporarily accepts $\mathcal{C}_c(S_c \cup \{s\})$ and rejects the rest. Note that during the algorithm, at every round, the student s can be arbitrarily selected. Hence, we now consider a particular execution of the algorithm on the auxiliary instance (i.e., the order in which students are selected). The execution has three stages, and they match exactly to the three stages of the discovery program mechanism. In the first stage, the algorithm can only select students who would apply to schools of type c'. Since after this stage, students will only apply to schools of type c'', the students who are temporarily matched in the first stage would not be rejected in later stages. That is, the temporary assignment at the end of the first stage becomes permanent, and it is matching $\mu_1^{\texttt{DISC}}$. For the second stage, the algorithm can only select disadvantaged students. Since schools of type c'' prefers disadvantaged students to advantaged students, the temporary assignment at the end of the second stage is also permanent and it corresponds to $\mu_2^{\texttt{DISC}}$. In the last stage, the algorithm continues without restriction until it terminates. Since there are only advantaged students applying to schools of type c'' at this final stage, the matching finalized at this stage is $\mu_3^{\texttt{DISC}}$. ■

Joint seat allocation. The original preference list $c_1 >_s c_2 >_s \cdots >_s c_k$ of student s becomes:

$$c_1' >_s^{\texttt{JSA-a}} c_1'' >_s^{\texttt{JSA-a}} c_2' >_s^{\texttt{JSA-a}} c_2'' >_s^{\texttt{JSA-a}} \cdots >_s^{\texttt{JSA-a}} c_k' >_s^{\texttt{JSA-a}} c_k''.$$

Similar to minority reserve, the relative ranking of the schools remains the same as that of the original instance; but different from minority reserve, students prefer general seats to reserved seats. Again, we let $I^{\texttt{JSA-a}} := (G^{\mathrm{aux}}, >_S^{\texttt{JSA-a}}, >_C^{\mathrm{aux}}, \mathbf{q}^{\mathrm{aux}})$ denote the auxiliary instance, and let $\mu^{\texttt{JSA-a}} := \texttt{SDA}(I^{\texttt{JSA-a}}, \mathcal{C}^{\mathrm{aux}})$ denote the student-optimal stable matching of the auxiliary instance.

Proposition 4.24. *For every student $s \in S$, $\mu^{\texttt{JSA}}(s) = \omega(\mu^{\texttt{JSA-a}}(s))$.*

Proof of Proposition 4.24. We first show that matchings in the original instance $I_1 := (G, >, \mathbf{q})$ and matchings in the auxiliary instance $I_2 := (G^{\mathrm{aux}}, >_S^{\texttt{JSA-a}}, >_C^{\mathrm{aux}}, \mathbf{q})$ can be transformed from each other. One direction is straightforward. Given a matching μ_2 in instance I_2, its corresponding matching μ_1 in instance I_1 has $\mu_1(s) = \omega(\mu_2(s))$ for all students $s \in S$. For the other direction, let μ_1 be a matching in instance I_1, we can construct its corresponding matching μ_2 in instance I_2

as follows. For every school c, $\mu_2(c') = \max(\mu_1(c), >_c, q_c^G)$ and $\mu_2(c'') = \mu_1(c) \setminus \mu_2(c')$. Let ψ denote the above mapping from matchings in I_2 to matchings in I_1, and let ψ^{-1} denote the above mapping for the reverse direction. By construction, a matching μ of I_1 is stable in I_1 if and only if $\psi^{-1}(\mu)$ is stable in I_2. Therefore, the student-optimal stable matching in I_1 can be obtained from the student-optimal stable matching in I_2 via mapping ψ^{-1}, and the claim follows. ■

4.5 Data on NYC Specialized High Schools

In this section, we analyze and compare the mechanisms on real-world datasets[12]. There is a total of 12 anonymized datasets, each for one of the 12 consecutive academic years from 2005-06 to 2016-17. Entries of each dataset include (1) students' IDs, (2) their scores for the Specialized High School Admissions Test (see Table 4.3 for a list of specialized high schools), (3) their (possibly, non-complete) preference lists of these eight schools, (4) their middle schools, (5) which school they are admitted to (which could be empty), and other information that are not relevant for our analysis.

B	Bronx High School of Science
T	Brooklyn Technical High School
R	Staten Island Technical High School
L	Brooklyn Latin
Q	Queens High School for the Sciences at York
M	High School of Mathematics, Science and Engineering at City College
S	Stuyvesant High School
A	High School of American Studies at Lehman College

Table 4.3: School code and school name of NYC specialized high schools.

Immediately from the dataset, we can extract the number of students applying for these specialized high schools and the capacities of each schools (i.e., the number of students admitted). On average, about $27,000$ students take the SHSAT exam every year, and among them, about $8,000$ (which is about 30%) are disadvantaged students. In terms of admission, about $5,100$ students receive an offer, out of whom about 820 (which is about 16%) are disadvantaged students.

[12]The dataset is under a non-disclosure agreement with NYC DOE, *Data request #1046*.

To label each student as advantaged or disadvantaged, we follow the definition currently used by NYC DOE for the discovery program:

To be eligible for the Discovery program, a Specialized High Schools applicant must

1. *Be one or more of the following: a student from a low-income household, a student in temporary housing, or an English Language Learner who moved to NYC within the past four years; and*
2. *Have scored within a certain range below the cutoff score on the SHSAT; and*
3. *Attend a high-poverty school. A school is defined as high-poverty if it has an Economic Need Index (ENI) of at least 60%.*

The second condition is related to eligibility, and not specifically to whether a student is disadvantaged, so we do not incorporate that when labeling the students. For the first set of conditions, we use an accompanying dataset which contains students' demographic information. However, since the information given in the dataset are not exactly the same as those specified in the definition, we slightly modify the first condition: "be one or more of the following: (1) eligible for free or reduced price lunch or has been identified by the Human Resources Administration (HRA) as receiving certain types of public assistance; or (2) an English Language Learner". For the last condition, we obtain the ENIs of NYC middle schools from a school quality report of academic year 2017-2018, which can be downloaded from the NYC Open Data website[13].

To obtain schools' universal priority order $>_C$ over the students, we assign to every student a unique *lottery* number, denoted as ℓ_s, for tie-breaking. For any two students $s_1, s_2 \in S$, s_1 has a higher priority than s_2 (i.e., $s_1 >_C s_2$) only when s_1 has a higher score than s_2 or when they have the same score but $\ell_{s_1} < \ell_{s_2}$. This idea of using lottery numbers for tie breaking has been used in practice (see, e.g., Abdulkadiroğlu, Pathak, and Roth (2009)).

[13] `https://data.cityofnewyork.us/Education/2017-2018-School-Quality-Reports-Elem-Middle-K-8/g6v2-wcvk`

Combining all components, the final dataset for analysis contains the following information for each student: unique identification number, test score, preference list, indicator for whether they are disadvantaged students, and lottery number.

First in Section 4.5.1, we analyze the outcome of the discovery program mechanism under the current guideline, and we provide some additional observations besides the theoretical results in Section 4.3.3. We then compare, in Section 4.5.2, the outcomes from all three mechanisms. For most of the experiments, we only include results of the latest academic year, since they are qualitatively similar for all academic years. Full results of all academic years can be found in Appendix B.1.

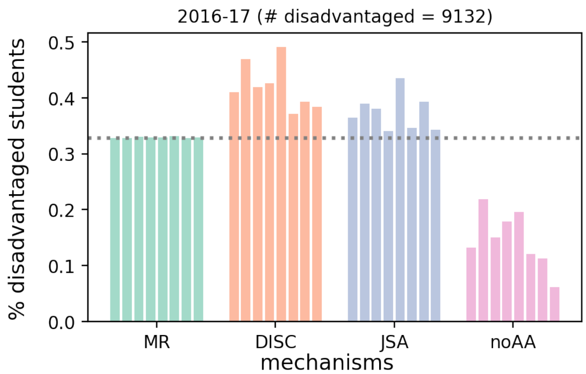
We also investigate and discuss the school-over-seat hypothesis by analyzing the patterns of students' preference lists, which can be found in Appendix B.2.

4.5.1 Results: the discovery program

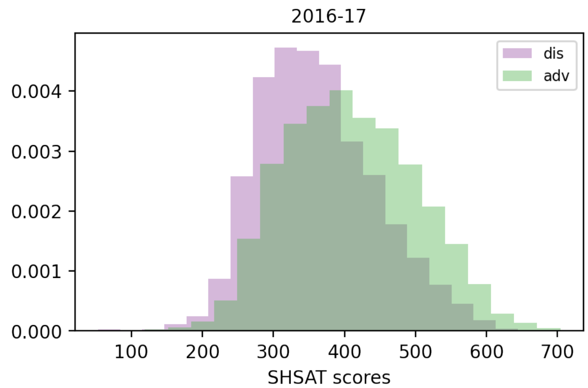
We start by analyzing the performance of the discovery program mechanism, where the reservation quota of every school c is set to be $q_c^R := \lceil q_c \times 20\% \rceil$, since 20% is the number recommended in a proposal by NYCDOE (2019). We show two negative results of the discovery program mechanism, one of which has been discussed theoretically in Section 4.3.

Recall that the discovery program is the only mechanism that admits in-group blocking pairs (see the summary in Table 4.1). In Figure 4.1, we show the number of blocking pairs for disadvantaged students across all academic years. On average, there are about 950 blocking pairs for disadvantaged students every academic year involving about 650 disadvantaged students.

We also conducted a simple experiment to show that the discovery program is not strategy-proof. In this experiment, we first identify the top ranked disadvantaged student s who is not admitted to his most preferred school, and we then modify the preference list of s so that this most preferred school is the only school on the preference list (i.e., removing all other schools and considering them as unacceptable). We notice that with the modified preference list, disadvantaged student s is then able to go to the most preferred school. Hence, under the discovery program,



(a) Proportions of disadvantaged students admitted, with bars from left to right corresponding to schools: B, T, R, L, Q, M, S, A. The dotted line represents the proportion of disadvantaged students among all applicants.



(b) The distribution of the SHSAT scores of advantaged students (labeled "adv") and disadvantaged students (labeled "dis").

Figure 4.3: Affirmative action increases the number of disadvantaged students admitted.

students could lie about their preferences in order to go to more preferred schools.

4.5.2 Results: comparison of three mechanisms

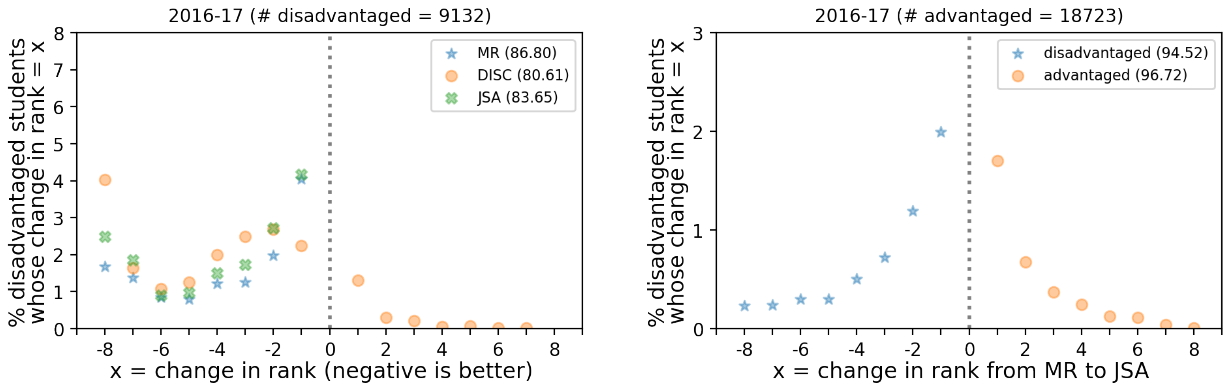
For experiments in this section, we choose the reservation quotas so that they are consistent with the proportion of disadvantaged students in the market: $q_c^R = \lceil q_c \times \frac{|S^m|}{|S^M|} \rceil, \forall c \in C$. We choose these reservation quotas simply because they are a reasonable choice and are a smart reserve, and we would like to point out that one could slightly increase or decrease these numbers without affecting the findings in this section qualitatively.

Proportion of disadvantaged students admitted. In Figure 4.3a, we show that all mechanisms with affirmative action can increase the proportion of disadvantaged students admitted to these schools. More specifically, under joint seat allocation and the discovery program mechanism, the numbers of disadvantaged students admitted exceeds the reservation quotas. This is because disadvantaged students with high scores can take up general seats under these two mechanisms. On the other hand, for minority reserve, the numbers of disadvantaged students admitted match exactly the reservation quota. This is because after disadvantaged students take up the reserved seats, the remaining disadvantaged students cannot compete against advantaged students for the general seats and are thus not admitted. The phenomenon is exactly the high competitiveness condition we discussed in Section 4.4.2 and is particularly true for our dataset since the number of students are much higher than the number of available seats, and disadvantaged students are performing systematically worse than advantaged students, as one can see in Figure 4.3b.

The figure seems to suggest that, for a fixed quota, the discovery program mechanism is better for disadvantaged students, as the number of disadvantaged students admitted to any school is the largest. However, this is not true when we examine the matching more closely.

Effects of affirmative actions to individual students. As opposed to Figure 4.3a which shows the effects of affirmative action mechanisms on disadvantaged students as a whole group, we show in Figure 4.4a these effects on individual levels. In particular, we examine the change in rank of the schools assigned to students under these mechanisms as compared to under the no affirmative action mechanism. For instance, if a student is matched to their third choice (i.e., rank of assigned school is 3) under the no affirmative action mechanism, but is matched to their first choice (i.e., rank of assigned school is 1) under minority reserve, then their change in rank of assigned school is -2 under minority reserve.

The main takeaway of Figure 4.4a is that when the reservation quotas are a smart reserve, the discovery program mechanism is the only one under which disadvantaged students can be worse off, as it is the only mechanism with markers on the positive axis. This is consistent with our discussion in Section 4.3 (see Table 4.1). We further investigate who are the disadvantaged students that are worse off under the discovery program, and we show the results in Figure 4.2. Interestingly, the disadvantaged students who are performing relatively well are the ones who are being admitted to schools they prefer less (dots on the upper left side of Figure 4.2). These are essentially the disadvantaged students who are assigned to general seats during the first stage of the discovery program mechanism. Because there are fewer seats during the first stage of the discovery program mechanism (as compared to the no affirmative action mechanism), the competition is fiercer and thus, these disadvantaged students got assigned to worse schools. Not only does this phenomenon imply that the discovery program mechanism is unfair to these well-performing disadvantaged



(a) Change from `noAA` to an affirmative action mechanism, for disadvantaged students

(b) Change from `MR` to `JSA`, for both advantaged and disadvantaged students.

Figure 4.4: Percentage of (dis)advantaged students (w.r.t. the total number of (dis)advantaged students) whose change in rank of assigned schools is a certain value. The number in each legend label is for when $x = 0$.

students, but it also hints at a situation where students have the incentive to under-perform in the admission exams. This certainly is in sharp contrast to the purpose of education and should not be a consequence of any applicable mechanism.

Joint seat allocation dominates minority reserve. In Figure 4.4a, we see that for each negative change in rank of assigned schools, the markers of joint seat allocation are in general higher than those of minority reserve. It seems to suggest that matching $\mu^{\texttt{JSA}}$ dominates matching $\mu^{\texttt{MR}}$ for disadvantaged students. To understand if this is true, we directly compare these two matchings and confirm the hypothesis (see Figure 4.4b). In fact, we observe the same dominance relation for all academic years. This prompts us to investigate the reason behind it, especially given that this dominance relation is not true in general as we discussed in Section 4.4. This dominance is a consequence of the data satisfying the high competitiveness hypothesis defined in Section 4.4.2 (see Figure 4.3a): the number of disadvantaged students admitted under minority reserve should not exceed the reservation quotas.

4.6 Conclusion and Discussion

In this chapter, we study three quota-based affirmative action mechanisms, and compare their outcomes for disadvantaged students under the school-over-seat hypothesis. We show that although the discovery program is instrumental in providing opportunities for disadvantaged students, the current implementation suffers from some drawbacks both theoretically and empirically. In addition, we show that to improve the discovery program, although there is no clear winner between joint seat allocation and minority reserve in general settings, the former is better for the NYC specialized high school market.

One caveat of our results is that they are based on the school-over-seat hypothesis, for which current data do not offer a definitive validation. Our experiments on the polarization of the preference data (see Appendix B.2) and the fact that the length of the summer program (3 weeks) is minimal when compared to the length of a high-school cycle (4 years) seem to suggest that this hypothesis is reasonable. However, other factors may come into play, such as the social stigma attached to being admitted via reserved seats[14].

This leads to two interesting directions for future work. As a first step, we believe it would be beneficial to explicitly collect students' expanded preference. Not only will these data confirm or invalidate the school-over-seat hypothesis, but they will also provide insights on the similarity or heterogeneity of the structure of students' expanded preference lists. In the case where the school-over-seat hypothesis fails, then a valuable next step would be to design a matching mechanism that account for individual students' expanded preference lists, while maintaining a number of desirable features such as strategy-proof and absence of in-group blocking pairs.

[14]We are not aware of this stigma being present in NYC SHSs, but it is definitely present in other markets employing some form of affirmative action mechanisms (Aygun and Turhan, 2020).

Conclusion

In my thesis, I developed theories and algorithms, as well as empirical results, for models that extend beyond the traditional stable matching problem by Gale and Shapley (1962). Some extensions focus on the *output* of the model, such as legal assignments and EADAM in Chapter 2; some extensions focus on the *input* of the model such as the model with choice functions in Chapter 3; and some focus on both, such as affirmative action mechanisms in Chapter 4. I believe that results in my thesis serve as an important contribution to the rich literature on stable matchings.

My theoretical results further add a touch of elegance to the theories surrounding stable matchings, and provide the foundation for provably better and practically implementable suggestions to policymakers; my algorithmic results drastically increase the applicability of these extensions for real-world markets; and my empirical results not only validate these theoretical and algorithmic results, but also allow me to identify key characteristics of real-world markets, which then lead to further theoretical findings.

The works in this thesis have deepened my understanding of the stable matching problem as well as its extensions, which in turn brings about many interesting research directions, which I will describe next.

1. For models with substitutable and consistent choice functions, Ehlers and Morrill (2020) showed that the concept of legal assignments is well-defined and the set of legal assignments has a lattice structure. Although Ehlers and Morrill (2020) gave an algorithm that finds the student-optimal legal assignment, it is not known if one can optimize efficiently over the set of stable matchings. To begin with, due to our result that legal assignments are stable assignments in disguise

(see Appendix A.1), the lattice structure formed by legal assignments might not be distributive (Blair, 1988). Hence, in order to develop fast algorithms using the properties of distributive lattice due to Birkhoff (1937), additional assumptions need to be imposed upon choice functions, such as quota-filling and cardinal monotonicity. Even then, the techniques we developed in Chapter 2 and Chapter 3 do not apply directly since the certificate for illegal edges relies on the rotation digraph, which is not known for the choice function model. Therefore, it is interesting to study efficient algorithms for optimization problems over the set of stable matching under the choice function model, which I believe could potentially lead to many new theories for the choice function model.

2. A natural extension of the previous direction is to develop faster algorithms of EADAM for the choice functions model. Ideally, we would like to design an algorithm whose time complexity again matches that of the deferred acceptance algorithm.

3. Eirinakis et al. (2014) presented a minimal description of the stable matching polytope for the traditional model. In Section 3.5.1 of Chapter 3, we gave an alternative proof for this description by connecting the stable matching polytope with the order polytope associated with the rotation poset. It would be interesting to investigate whether one can devise a minimal description of the stable matching polytope for the choice function model, using the same idea.

4. In Chapter 3, our structural results in Section 3.3 assumed choice functions to be substitutable and cardinal monotone. However, our algorithmic results in Section 3.4 further assumed that the choice functions of one side of the market are quota-filling. That is because a critical counting argument in the proof of Theorem 3.56 relies on the quota-filling assumption. However, this could be an artifact of our proof technique, and we would like to investigate further whether the same algorithmic results hold without the quota-filling assumption.

5. In Section 3.6.2 of Chapter 3, we show that the number of substitutable and cardinal monotonicity choice functions is doubly exponentially in the number of potential partners. It remains open whether a similar result holds for substitutable and quota-filling choice functions. It seems

the counting techniques developed by Echenique (2007) does not apply for quota-filling choice functions.

6. There has been some recent work showing how feasible regions of certain problems in combinatorial optimization can be seen as a distributive lattice (Garg, 2020). This fact, combined with our approach in Chapter 3, may lead to (known or new) efficient algorithms for optimizing linear functions over the associated polytopes.

7. For the traditional stable matching problem, it is well-known, and it is not hard to see that when one side of the market has the same preference list, there is a unique stable matching (Gusfield and Irving, 1989). In addition, Clark (2006) gave a more general condition, which he called the *No Crossing Condition (NCC)*, which is sufficient for the uniqueness of stable matchings. Thus, it is interesting to investigate the same question for models with choice functions. In particular, if one side of the market has the same choice function, does it guarantee that there is a unique stable matching? Moreover, can we find a condition that is similar to NCC for the choice function model?

8. The idea of EADAM extends to the choice function model (see, e.g. Ehlers and Morrill, 2020; Doğan, 2016), although the time complexity of the naive execution does not match that of the deferred acceptance algorithm. As we discussed previously, developing efficient execution of EADAM for general choice functions might be challenging. However, for the affirmative action mechanisms we discussed in Chapter 4, efficient execution might be possible due to the alternative interpretations given in Section 4.4.3. A faster algorithm for EADAM here has a crucial application impact because Doğan (2016) showed that the mechanism based on EADAM, together with choice functions as those in the minority reserve mechanism, is minimally responsive. We also believe that a similar result holds for choice functions as those in the joint seat allocation mechanism.

References

Abdulkadiroğlu, Atila, Yeon-Koo Che, and Yosuke Yasuda (2015). “Expanding “choice" in school choice”. In: *American Economic Journal: Microeconomics* 7.1, pp. 1–42.

Abdulkadiroğlu, Atila, Parag A Pathak, and Alvin E Roth (2005a). “The New York City High School Match”. In: *American Economic Review* 95.2, pp. 364–367.

Abdulkadiroğlu, Atila, Parag A Pathak, and Alvin E Roth (2005b). “The New York City high school match”. In: *American Economic Review* 95.2, pp. 364–367.

Abdulkadiroğlu, Atila, Parag A Pathak, and Alvin E Roth (2009). “Strategy-proofness versus efficiency in matching with indifferences: Redesigning the NYC high school match”. In: *American Economic Review* 99.5, pp. 1954–78.

Abdulkadiroğlu, Atila, Parag A Pathak, Alvin E Roth, and Tayfun Sönmez (2005). “The Boston public school match”. In: *American Economic Review* 95.2, pp. 368–371.

Abdulkadiroğlu, Atila and Tayfun Sönmez (2003). “School choice: A mechanism design approach”. In: *American economic review* 93.3, pp. 729–747.

Afacan, Mustafa Oğuz, Zeynel Harun Alioğulları, and Mehmet Barlo (2017). “Sticky matching in school choice”. In: *Economic Theory* 64.3, pp. 509–538.

Afacan, Mustafa Oğuz and Umutcan Salman (2016). “Affirmative actions: The Boston mechanism case”. In: *Economics Letters* 141, pp. 95–97.

Aizerman, Mark and Andrew Malishevski (1981). “General theory of best variants choice: Some aspects”. In: *IEEE Transactions on Automatic Control* 26.5, pp. 1030–1040.

Alkan, Ahmet (2001). “On preferences over subsets and the lattice structure of stable matchings”. In: *Review of Economic Design* 6.1, pp. 99–111.

Alkan, Ahmet (2002). “A class of multipartner matching markets with a strong lattice structure”. In: *Economic Theory* 19.4, pp. 737–746.

Aprile, Manuel, Alfonso Cevallos, and Yuri Faenza (2018). “On 2-level polytopes arising in combinatorial settings”. In: *SIAM Journal on Discrete Mathematics* 32.3, pp. 1857–1886.

Ashkenas, Jeremy, Haeyoun Park, and Adam Pearce (2017). “Even with affirmative action, Blacks and Hispanics are more underrepresented at top colleges than 35 years ago”. In: *New York Times*, pp. 1–18.

Ashlagi, Itai and Afshin Nikzad (2016). “What matters in school choice tie-breakings? How competition guides design”. In: *Proceedings of the 2016 ACM Conference on Economics and Computation*, pp. 767–768.

Aygün, Orhan and Tayfun Sönmez (2013). “Matching with contracts: Comment”. In: *American Economic Review* 103.5, pp. 2050–51.

Aygün, Orhan and Bertan Turhan (2016). “Dynamic reserves in matching markets: Theory and applications”. In: *Available at SSRN 2743000*.

Aygun, Orhan and Bertan Turhan (2020). “Designing Direct Matching Mechanisms for India with Comprehensive Affirmative Action”. In: *Available at SSRN 3582669*.

Baïou, Mourad and Michel Balinski (2000). “Many-to-many matching: stable polyandrous polygamy (or polygamous polyandry)”. In: *Discrete Applied Mathematics* 101.1-3, pp. 1–12.

Baïou, Mourad and Michel Balinski (2000). “The stable admissions polytope”. In: *Mathematical programming* 87.3, pp. 427–439.

Baïou, Mourad and Michel Balinski (2004). “Student admissions and faculty recruitment”. In: *Theoretical Computer Science* 322.2, pp. 245–265.

Bando, Keisuke (2014). “On the existence of a strictly strong Nash equilibrium under the student-optimal deferred acceptance algorithm”. In: *Games and Economic Behavior* 87, pp. 269–287.

Bansal, Vipul, Aseem Agrawal, and Varun S Malhotra (2007a). “Polynomial time algorithm for an optimal stable assignment with multiple partners”. In: *Theoretical Computer Science* 379.3, pp. 317–328.

Bansal, Vipul, Aseem Agrawal, and Varun S Malhotra (2007b). “Polynomial time algorithm for an optimal stable assignment with multiple partners”. In: *Theoretical Computer Science* 379.3, pp. 317–328.

Baswana, Surender, Partha Pratim Chakrabarti, Sharat Chandran, Yashodhan Kanoria, and Utkarsh Patange (2019). “Centralized admissions for engineering colleges in India”. In: *INFORMS Journal on Applied Analytics* 49.5, pp. 338–354.

Benjamin, Arthur T, Cherlyn Converse, and Henry A Krieger (1995). “How do I marry thee? Let me count the ways”. In: *Discrete Applied Mathematics* 59.3, pp. 285–292.

Birkhoff, Garrett (1937). “Rings of sets”. In: *Duke Mathematical Journal* 3.3, pp. 443–454.

Blair, Charles (1988). “The lattice structure of the set of stable matchings with multiple partners”. In: *Mathematics of operations research* 13.4, pp. 619–628.

Boschma, Janie and Ronald Brownstein (2016). “The concentration of poverty in American schools”. In: *The Atlantic* 29.

Capers IV, Quinn, Daniel Clinchot, Leon McDougle, and Anthony G Greenwald (2017). “Implicit racial bias in medical school admissions”. In: *Academic Medicine* 92.3, pp. 365–369.

Chambers, Christopher P and M Bumin Yenmez (2017). “Choice and matching”. In: *American Economic Journal: Microeconomics* 9.3, pp. 126–47.

Cheriyan, Joseph, Torben Hagerup, and Kurt Mehlhorn (1996). “An $o(n^3)$-Time Maximum-Flow Algorithm”. In: *SIAM Journal on Computing* 25.6, pp. 1144–1170.

Clark, Simon (2006). “The uniqueness of stable matchings”. In: *Contributions in Theoretical Economics* 6.1.

Doğan, Battal (2016). “Responsive affirmative action in school choice”. In: *Journal of Economic Theory* 165, pp. 69–105.

Doğan, Battal, Serhat Doğan, and Kemal Yıldız (2021). “On capacity-filling and substitutable choice rules”. In: *Mathematics of Operations Research*.

Dubins, Lester E and David A Freedman (1981). “Machiavelli and the Gale-Shapley algorithm”. In: *The American Mathematical Monthly* 88.7, pp. 485–494.

Dur, Umut, A Arda Gitmez, and O Yılmaz (2015). *School choice under partial fairness*. Tech. rep. Technical report. Working paper. North Carolina State University, 2015.[19].

Echenique, Federico (2007). “Counting combinatorial choice rules”. In: *Games and Economic Behavior* 58.2, pp. 231–245.

Echenique, Federico and M Bumin Yenmez (2015). “How to control controlled school choice”. In: *American Economic Review* 105.8, pp. 2679–94.

Ehlers, Lars (2007). “von Neumann–Morgenstern stable sets in matching problems”. In: *Journal of Economic Theory* 134.1, pp. 537–547.

Ehlers, Lars, Isa E Hafalir, M Bumin Yenmez, and Muhammed A Yildirim (2014). “School choice with controlled choice constraints: Hard bounds versus soft bounds”. In: *Journal of Economic theory* 153, pp. 648–683.

Ehlers, Lars and Thayer Morrill (2020). “(Il)legal assignments in school choice”. In: *The Review of Economic Studies* 87.4, pp. 1837–1875.

Eirinakis, Pavlos, Dimitrios Magos, Ioannis Mourtos, and Panayiotis Miliotis (2014). “Polyhedral aspects of stable marriage”. In: *Mathematics of Operations Research* 39.3, pp. 656–671.

Erdil, Aytek and Haluk Ergin (2008). “What’s the matter with tie-breaking? Improving efficiency in school choice”. In: *American Economic Review* 98.3, pp. 669–89.

Erdös, Paul and Alfréd Rényi (1961). “On a classical problem of probability theory”. In: *Magyar Tud. Akad. Mat. Kutató Int. Közl* 6.1-2, pp. 215–220.

Ergin, Haluk I (2002). “Efficient resource allocation on the basis of priorities”. In: *Econometrica* 70.6, pp. 2489–2497.

Faenza, Yuri, Swati Gupta, and Xuan Zhang (2022a). “Discovering Opportunities in New York City’s Discovery Program: an Analysis of Affirmative Action Mechanisms”. In: *Manuscript in preparation*.

Faenza, Yuri, Swati Gupta, and Xuan Zhang (2022b). “Reducing the Feeder Effect in Public School Admissions: A Bias-aware Analysis for Targeted Interventions”. In: *arXiv preprint arXiv:2004.10846*.

Faenza, Yuri and Xuan Zhang (2021). “Affinely Representable Lattices, Stable Matchings, and Choice Functions”. In: *Integer Programming and Combinatorial Optimization*. Ed. by Mohit Singh and David P. Williamson. Cham: Springer International Publishing, pp. 89–103. ISBN: 978-3-030-73879-2.

Faenza, Yuri and Xuan Zhang (2022). “Legal Assignments and Fast EADAM with Consent via Classic Theory of Stable Matchings”. In: *Operations Research*.

Fleiner, Tamás (2003). “On the stable b-matching polytope”. In: *Mathematical Social Sciences* 46.2, pp. 149–158.

Fragiadakis, Daniel and Peter Troyan (2017). “Improving matching under hard distributional constraints”. In: *Theoretical Economics* 12.2, pp. 863–908.

Gale, David and Lloyd S Shapley (1962). “College admissions and the stability of marriage”. In: *The American Mathematical Monthly* 69.1, pp. 9–15.

Garg, Vijay K (2020). “Predicate detection to solve combinatorial optimization problems”. In: *Proceedings of the 32nd ACM Symposium on Parallelism in Algorithms and Architectures*, pp. 235–245.

Gratz_vs_Bollinger (2003). *Gratz v. Bollinger*. URL: `www.oyez.org/cases/2002/02-516`.

Gusfield, Dan (1987). “Three fast algorithms for four problems in stable marriage”. In: *SIAM Journal on Computing* 16.1, pp. 111–128.

Gusfield, Dan and Robert W Irving (1989). *The stable marriage problem: structure and algorithms*. MIT press.

Hafalir, Isa E, M Bumin Yenmez, and Muhammed A Yildirim (2013). “Effective affirmative action in school choice”. In: *Theoretical Economics* 8.2, pp. 325–363.

Harris, Elizabeth A. and Ford Fessenden (2017). “The Broken Promises of Choice in New York City Schools”. In: *The New York Times*.

Hatfield, John William and Paul R Milgrom (2005). “Matching with contracts”. In: *American Economic Review* 95.4, pp. 913–935.

Hitsch, Gunter J, Ali Hortaçsu, and Dan Ariely (2010). “Matching and sorting in online dating”. In: *American Economic Review* 100.1, pp. 130–63.

Irving, Robert W and Paul Leather (1986). “The complexity of counting stable marriages”. In: *SIAM Journal on Computing* 15.3, pp. 655–667.

Irving, Robert W, Paul Leather, and Dan Gusfield (1987). “An efficient algorithm for the “optimal” stable marriage”. In: *Journal of the ACM (JACM)* 34.3, pp. 532–543.

Jaschik, Scott (2019). *New SAT Score: Adversity*. URL: https://www.insidehighered.com/admissions/article/2019/05/20/college-board-will-add-adversity-score-everyone-taking-sat.

Jiao, Zhenhua and Ziyang Shen (2021). “School choice with priority-based affirmative action: A responsive solution”. In: *Journal of Mathematical Economics* 92, pp. 1–9.

JoSAA, Joint Seat Allocation Authority (2020). *Business rules for joint seat allocation for the academic programs offered by the IITs, NITs, IIEST, IIITs and other-GFTIs for the academic year 2020-21*. URL: https://josaa.nic.in/webinfo/File/GetFile?FileId=6&LangId=P.

Kamada, Yuichiro and Fuhito Kojima (2015). “Efficient matching under distributional constraints: Theory and applications”. In: *American Economic Review* 105.1, pp. 67–99.

Karlin, Anna R, Shayan Oveis Gharan, and Robbie Weber (2018). “A simply exponential upper bound on the maximum number of stable matchings”. In: *Proceedings of the 50th Annual ACM SIGACT Symposium on Theory of Computing*. ACM, pp. 920–925.

Kelso Jr, Alexander S and Vincent P Crawford (1982). “Job matching, coalition formation, and gross substitutes”. In: *Econometrica: Journal of the Econometric Society*, pp. 1483–1504.

Kesten, Onur (2010). “School choice with consent”. In: *The Quarterly Journal of Economics* 125.3, pp. 1297–1348.

Knuth, Donald E, Rajeev Motwani, and Boris Pittel (1990). “Stable husbands”. In: *Random Structures & Algorithms* 1.1, pp. 1–14.

Knuth, Donald Ervin (1976). “Marriages stables”. In: *Technical report*.

Kojima, Fuhito (2012). “School choice: Impossibilities for affirmative action”. In: *Games and Economic Behavior* 75.2, pp. 685–693.

Kominers, Scott Duke and Tayfun Sönmez (2016). “Matching with slot-specific priorities: Theory”. In: *Theoretical Economics* 11.2, pp. 683–710.

Kunysz, Adam, Katarzyna Paluch, and Pratik Ghosal (2016). “Characterisation of strongly stable matchings”. In: *Proceedings of the twenty-seventh annual ACM-SIAM symposium on Discrete algorithms*. Society for Industrial and Applied Mathematics, pp. 107–119.

Manlove, David F (2002). “The structure of stable marriage with indifference”. In: *Discrete Applied Mathematics* 122.1-3, pp. 167–181.

Manlove, David F (2013). *Algorithmics of matching under preferences*. Vol. 2. World Scientific.

Martínez, Ruth, Jordi Massó, Alejandro Neme, and Jorge Oviedo (2004). “An algorithm to compute the full set of many-to-many stable matchings”. In: *Mathematical Social Sciences* 47.2, pp. 187–210.

McVitie, David G and Leslie B Wilson (1971). “The stable marriage problem”. In: *Communications of the ACM* 14.7, pp. 486–490.

Morrill, Thayer (2016). *Which School Assignments Are Legal?* Tech. rep.

Narita, Yusuke (2016). *Match or mismatch: learning and inertia in school choice*. Tech. rep. Working paper.

Nguyen, Thành, Hai Nguyen, and Alexander Teytelboym (2019). “Stability in Matching Markets with Complex Constraints”. In: *Proceedings of the 2019 ACM Conference on Economics and Computation*. ACM, pp. 61–61.

Nguyen, Thành and Rakesh Vohra (2019). “Stable matching with proportionality constraints”. In: *Operations Research*.

NYCDOE, New York City Department of Education (2019). *Specialized High Schools Proposal: Making Admissions to the Specialized High Schools More Equitable for All Students*. URL: https://infohub.nyced.org/docs/default-source/default-document-library/specialized-high-schools-proposal_english.pdf.

Orfield, Gary and Chungmei Lee (2005). “Why segregation matters: Poverty and educational inequality.” In: *Civil Rights Project at Harvard University (The)*.

Picard, Jean-Claude (1976). "Maximal closure of a graph and applications to combinatorial problems". In: *Management science* 22.11, pp. 1268–1272.

Pittel, Boris (1989). "The average number of stable matchings". In: *SIAM Journal on Discrete Mathematics* 2.4, pp. 530–549.

Pittel, Boris (1992). "On likely solutions of a stable marriage problem". In: *The Annals of Applied Probability* 2.2, pp. 358–401.

Quinn Capers, IV, Daniel Clinchot, Leon McDougle, and Anthony G Greenwald (2017). "Implicit racial bias in medical school admissions". In: *Academic Medicine* 92.3, pp. 365–369.

Raab, Martin and Angelika Steger (1998). ""Balls into bins"—A simple and tight analysis". In: *International Workshop on Randomization and Approximation Techniques in Computer Science*. Springer, pp. 159–170.

Roth, Alvin E (1984a). "Stability and polarization of interests in job matching". In: *Econometrica: Journal of the Econometric Society*, pp. 47–57.

Roth, Alvin E (1984b). "The evolution of the labor market for medical interns and residents: a case study in game theory". In: *Journal of political Economy* 92.6, pp. 991–1016.

Roth, Alvin E (1986). "On the allocation of residents to rural hospitals: a general property of two-sided matching markets". In: *Econometrica: Journal of the Econometric Society*, pp. 425–427.

Roth, Alvin E and Elliott Peranson (1999). "The redesign of the matching market for American physicians: Some engineering aspects of economic design". In: *American economic review* 89.4, pp. 748–780.

Roth, Alvin E and Uriel G Rothblum (1999). "Truncation strategies in matching markets—in search of advice for participants". In: *Econometrica* 67.1, pp. 21–43.

Roth, Alvin E, Uriel G Rothblum, and John H Vande Vate (1993). "Stable matchings, optimal assignments, and linear programming". In: *Mathematics of operations research* 18.4, pp. 803–828.

Roth, Alvin E and Marilda Sotomayor (1990). *Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis.* Cambridge University Press.

Roth, Alvin E and Marilda Sotomayor (1992). "Two-sided matching". In: *Handbook of game theory with economic applications* 1, pp. 485–541.

Rothblum, Uriel G (1992). "Characterization of stable matchings as extreme points of a polytope". In: *Mathematical Programming* 54.1-3, pp. 57–67.

Schrijver, Alexander (2003). *Combinatorial optimization: polyhedra and efficiency*. Vol. 24. Springer Science & Business Media.

Shapiro, Eliza (2019a). “Segregation Has Been the Story of New York City’s Schools for 50 Years”. In: *The New York Times Magazine*.

Shapiro, Eliza (2019b). “Should a Single Test Decide a 4-Year-Old’s Educational Future?” In: *New York Times*.

Shapiro, Eliza (2021). “Only 8 Black Students Are Admitted to Stuyvesant High School”. In: *The New York Times Magazine*.

Stanley, Richard P (1986). “Two poset polytopes”. In: *Discrete & Computational Geometry* 1.1, pp. 9–23.

Tang, Qianfeng and Jingsheng Yu (2014). “A new perspective on Kesten’s school choice with consent idea”. In: *Journal of Economic Theory* 154, pp. 543–561.

Tang, Qianfeng and Yongchao Zhang (2017). “Weak Stability and Pareto Efficiency in School Choice”. In.

Tomoeda, Kentaro (2018). “Finding a stable matching under type-specific minimum quotas”. In: *Journal of Economic Theory* 176, pp. 81–117.

Troyan, Peter, David Delacrétaz, and Andrew Kloosterman (2020). “Essentially stable matchings”. In: *Games and Economic Behavior* 120, pp. 370–390.

Vate, John H Vande (1989). “Linear programming brings marital bliss”. In: *Operations Research Letters* 8.3, pp. 147–153.

Veiga, Christina (2020). *Program aiming to integrate NYC’s specialized high schools continues to enroll few black and Hispanic students*. URL: `https://ny.chalkbeat.org/2020/5/1/21244612/discovery-few-black-and-hispanic-students/`.

Von Neumann, John and Oskar Morgenstern (1953). *Theory of Games and Economic Behavior: 3d Ed*. Princeton University Press.

Wako, Jun (2010). “A polynomial-time algorithm to find von Neumann-Morgenstern stable matchings in marriage games”. In: *Algorithmica* 58.1, pp. 188–220.

Wang, Xing, Niels Agatz, and Alan Erera (2018). “Stable matching for dynamic ride-sharing systems”. In: *Transportation Science* 52.4, pp. 850–867.

Appendix A: Additional Details for Chapter 2

A.1 Extension of Theorem 2.3 to the Setting of Ehlers and Morrill (2020)

In Ehlers and Morrill (2020), each school b, instead of having a strict preference ordering of the students and having a quota, has a choice function $C_b : 2^{A_b} \to 2^{A_b}$, where $A_b = \{a : ab \in E\}$ and for each $X \subseteq A_b$, $C_b(X) \subseteq X$, where E is a subset of $A \times B$. $C_b(X)$ can be interpreted as the students school b chooses from X. Ehlers and Morrill (2020) assume that, for every $b \in B$, the choice function C_b is substitutable and consistent.

As usual, each student $a \in A$ has a strict ordering over schools $\{b \in B : ab \in E\}$. We denote by $(G(A \cup B, E), <, \mathcal{C})$ an instance of the stable assignment problem in the setting of Ehlers and Morrill (2020), where $<= \{<_v\}_{v \in A}$ and $\mathcal{C} = \{C_b\}_{b \in B}$. An assignment[1] M of this instance is a collection of edges of G such that for each $a \in A$, at most one edge of M is incident to a, and for each $b \in B$, $C_b(\{a : ab \in M\}) = \{a : ab \in M\}$. For an assignment M and $x \in A \cup B$, we again write $M(x) := \{y : xy \in M\}$.

An edge $ab \in E$ is said to *block* assignment M if $b >_a M(a)$ and $a \in C_b(M(b) \cup \{a\})$. Similarly to the one-to-many setting, an assignment is stable if and only if there is no edge blocking it, and an assignment M' is said to block an assignment M if there is an edge $ab \in M'$ such that ab blocks M. The set of legal assignments is defined the same way as before. Moreover, it is shown in Ehlers and Morrill (2020) that Theorem 2.5 holds in this setting.

For a subgraph $G' \subseteq G$, we denote by $(G', <, \mathcal{C})$ the instance where students' preferences are those induced by $<$ on schools in G', and schools' choice functions are defined as the restriction of $\mathcal{C}$ to students in G'. In particular, for every school $b \in B$, the restriction of function C_b also satisfies the substitutability and LAD properties, and thus $(G', <, \mathcal{C})$ is also an instance of the stable

[1]This is called an "individually rational" assignment in Ehlers and Morrill (2020).

assignment problem in the setting of Ehlers and Morrill (2020).

Because of the above-mentioned properties, the three lemmas in Section 2.3 – Lemma 2.6, Lemma 2.7, and Lemma 2.8 – extend to the setting of Ehlers and Morrill (2020) with exactly the same arguments, and thus so does Theorem 2.3.

A.2 Details of Implementations and Examples

As all implementation of our algorithms run in time $O(|E|)$, we can preprocess the input in time $O(|E|)$ and assume that: for each agent, we have its preference list given as an ordered list; given a student $a \in A$ and a school $b \in B$ and an assignment M, in constant time we can access $M(a)$ and the least preferred student in $M(b)$; given $x \in A \cup B$ and two neighbors y_1 and y_2 of x, we can decide in constant time if $y_1 >_x y_2$.

Before going through the details and proof of each implementation, we give an example for an execution of the algorithm.

A.2.1 Example of Algorithm 2.1 Execution

Example A.1. Consider the following instance with 5 students and 5 schools, where each school has quota 1. The student-optimal stable assignment is $\{a_1b_4, a_2b_3, a_3b_2, a_4b_1, a_5b_5\}$, denoted succinctly by $(4, 3, 2, 1, 5)$ (ordered list of school to which each student is matched).

$$
\begin{array}{llll}
a_1: & b_1 > b_2 > b_3 > b_4 > b_5 & b_1: & a_4 > a_5 > a_3 > a_2 > a_1 \\
a_2: & b_2 > b_1 > b_4 > b_3 > b_5 & b_2: & a_3 > a_5 > a_4 > a_1 > a_2 \\
a_3: & b_3 > b_4 > b_1 > b_2 > b_5 & b_3: & a_2 > a_5 > a_1 > a_4 > a_3 \\
a_4: & b_4 > b_3 > b_2 > b_1 > b_5 & b_4: & a_1 > a_5 > a_2 > a_3 > a_4 \\
a_5: & b_4 > b_3 > b_2 > b_1 > b_5 & b_5: & a_5 > a_1 > a_2 > a_3 > a_4
\end{array}
$$

For the fast implementation of `school-rotate-remove`, at each iteration i, together with M^i, we will additionally keep the following items:

(i) a directed path P^i of the school-rotation digraph D^i stored as a doubly-linked list which we will constructed step-by-step until a sink is reached or a cycle is closed;

(ii) for each $b \in B$, a position p_b for which the algorithm maintains the following invariant: with $b(p_b)$ denoting the student at position p_b on b's preference list, if b is not in the path P^i, then for all $a \geq_b b(p_b)$, we have $M^i(a) \geq_a b$, and if b is on the path P^i, then for all $a >_b b(p_b)$, we have $M^i(a) \geq_a b$;

(iii) a Boolean array W^i of dimension $|B|$, recording whether each school is in P^i;

(iv) a subset T^i of sinks[2] of D^i, stored as a Boolean array of dimension $|B|$;

(v) an index f such that $b_1, \cdots, b_{f-1}$ are all in T^i but b_f is not in T^i.

In Table A.1, we outline the updates that occurred at all steps (denoted by j) of all iterations (denoted by i) using the fast implementation of `school-rotate-remove`. The steps of iteration i illustrate the steps in building the directed path P^i. A cell is left blank if no update happens. W^i can be easily deduced from P^i and is therefore not included in the table.

The main idea of the construction is that, in order to find a directed cycle or a sink in a digraph, it suffices to follow a path. (i) allows us to carry over information on such paths from one iteration of the algorithm to the next. (ii) and (v) allow us to extend such a path quickly, without going through the full preference lists of agents again. (iii) and (iv) allow a quick detection when a sink or a cycle has been found while following a path.

When extending the directed path P^i, if $P^i = []$, as in (0.0) and (10.0), we add the first school not in T^i to the directed path, which is achieved by repeatedly checking if $b_f \in T^i$ and while so, updating $f := f + 1$. Assume P^i is non-empty and has b at the tail. If b is not the node corresponding to $\emptyset$ nor does p_b exceed the length of b's preference list (i.e., $p_b > 5$), then we rely on p_b to find $s_{M^i}(b)$. That is, we repeatedly update $p_b := p_b + 1$ until either $p_b > 5$ or $a := b(p_b)$ satisfies $b >_a M^i(a)$. Note that here, p_b is incremented before the conditions are checked and thus, p_b strictly increases every time an extension happens with b at the tail. The only time that p_b will

[2]Note that T^i can be easily deduced from p_b, given that a school b is in T^i if and only if $p_b(b) = \deg_G(b) + 1$. However, we keep T^i in our illustration to elucidate the steps.

$(i.j)$	P^i	$\{p_b\}_{b\in B}$	M^i	T^i	f
(0.0)	$[] \to b_1$	$[1,1,1,1,1]$	$(4,3,2,1,5)$	$\emptyset$	1
(0.1)	b_1, a_5, b_5	$[2,1,1,1,1]$			
(1.0)	b_1	$[2,1,1,1,6]$		b_5	
(1.1)	b_1, a_3, b_2	$[3,1,1,1,6]$			
(1.2)	b_1, a_3, b_2, a_5, b_5	$[3,2,1,1,6]$			
(2.0)	b_1, a_3, b_2				
(2.1)	$b_1, a_3, b_2, a_4, (b_1)$	$[3,3,1,1,6]$			
(3.0)	$[] \to b_1$		$(4,3,1,2,5)$		
(3.1-2)	b_1, a_2, b_3, a_5, b_5	$[4,3,2,1,6]$			
(4.0)	b_1, a_2, b_3				
(4.1-2)	$b_1, a_2, b_3, a_1, b_4, a_5, b_5$	$[4,3,3,2,6]$			
(5.0)	b_1, a_2, b_3, a_1, b_4				
(5.1)	$b_1, a_2, b_3, a_1, b_4, (a_2)$	$[4,3,3,3,6]$			
(6.0)	b_1	$[3,3,3,3,6]$	$(3,4,1,2,5)$		
(6.1-2)	$b_1, a_2, b_4, a_3, (b_1)$	$[4,3,3,4,6]$			
(7.0)	$[] \to b_1$		$(3,1,4,2,5)$		
(7.1-3)	$b_1, a_1, b_3, a_4, b_2, (a_1)$	$[5,4,4,4,6]$			
(8.0)	b_1	$[4,4,4,4,6]$	$(2,1,4,3,5)$		
(8.1-2)	$b_1, a_1, b_2, a_2, (b_1)$	$[5,5,4,4,6]$			
(9.0)	$[] \to b_1$		$(1,2,4,3,5)$		
(10.0)	$[] \to b_2$	$[6,5,4,4,6]$		b_1, b_5	2
(11.0)	$[] \to b_3$	$[6,6,4,4,6]$		b_1, b_2, b_5	3
(11.1-2)	$b_3, a_3, b_4, a_4, (b_3)$	$[6,6,5,5,6]$			
(12.0)	$[] \to b_3$		$(1,2,3,4,5)$		
(13.0)	$[] \to b_4$	$[6,6,6,5,6]$		b_1, b_2, b_3, b_5	4
(14.0)	$[]$	$[6,6,6,6,6]$		b_1, b_2, b_3, b_4, b_5	∞

Table A.1: Iterations of `school-rotate-remove` of Example A.1.

decrease is when b points to a directed cycle, as the school b_1 in (5.1) and (7.3). In such case, p_b is decremented by 1 after the rotation elimination, as seen in (6.0) and (8.0). This is because it is possible to have $s_{M^{i+1}}(b) = s_{M^i}(b)$ unchanged, and if that is the case, p_b should remain the same in the next iteration when extending the directed path. There are two scenarios, corresponding to Step 6 and Step 8 in Algorithm 2.1, where we stop extending the directed path P^i: one is when the tail b is a sink, implied by having $p_b > 5$ or having $b = \emptyset$; the other is when the additional node is already in the directed path, which can be checked against W_i. In the latter case, such nodes are written as (node) in Table A.1. $\diamondsuit$ $\triangle$

A.2.2 Proof of Theorem 2.25

Before going into the proof, we first state the following results. Lemma A.2 states the time complexity of Gale-Shapley's algorithm for the stable assignment problems and its proof can be found in, e.g., Manlove (2013) and Gusfield and Irving (1989). Lemma A.3 is shown by Bansal, Agrawal, and Malhotra (2007a) and it implies that the number of student-rotations an instance can have is at most linear in the number of edges.

Lemma A.2. *Gale-Shapley's algorithm with students or schools proposing can be implemented to run in time* $O(|E|)$.

Lemma A.3. *For any* $xy \in E$*, there is at most one* X*-rotation* $y_0, x_0, y_1, \ldots, y_{r-1}, x_{r-1}$ *exposed in some stable assignment of* $(G, <, \mathbf{q})$ *such that* $x = x_i$ *and* $y = y_i$ *for some* $i \in \{0, 1, \cdots, r-1\}$.

Proof of Theorem 2.25. We show details for Algorithm 2.1 with X taken as the set of schools (i.e., `school-rotate-remove`), as those with X taken as the set of students follow in a similar fashion. For simplicity, we call "school-rotations" simply "rotations" throughout the proof.

Algorithm 2.1 first finds the student-optimal stable assignment. This takes time $O(|E|)$ by Lemma A.2. Then the algorithm enters the **while** loop. A key fact we will resort to multiple times in our arguments is the following: (‡) for any pair of iterations i_1, i_2 of the while loop such that $i_1 > i_2$, we have $M^{i_1} \succeq M^{i_2}$. Given an assignment M, we say that we *scan* an edge ab when we check $b >_a M(a)$. From what is assumed above, scanning ab requires constant time. We denote by $b(i)$ the student at the i^{th} position on the preference list of b. Assume schools are sorted as $B = \{\bar{b}_1, \bar{b}_2, \cdots, \bar{b}_{|B|}\}$. Recall that, for all i, all sinks of D^i are either school nodes or the node corresponding to $\emptyset$.

At each iteration i, we keep the following:

- the assignment M^i as an $|A|$-dimensional array with the k^{th} position recording the school the k^{th} student is assigned to;
- a subset T^i of sinks of D^i stored as a Boolean array of dimension $|B|$;

- an index f such that $b_1, \cdots, b_{f-1}$ are all in T^i but b_f is not in T^i;
- a directed path P^i of D^i, stored as a doubly-linked list;
- a collection W^i of schools that are in P^i, stored as a $|B|$-dimensional Boolean array; and
- for each $b \in B$, a position p_b such that, in determining $s_{M^i}(b)$, we do not need to scan ab for all a such that $a \geq_b b(p_b)$.

We initialize $M^0 = M_0$, $T^0 := \emptyset$, $f := 1$, $P^0 := []$, $W^0 := \emptyset$, and p_b to be the position of the least preferred student in $M^0(b)$ on b's preference list for every $b \in B$. Note that the choices of p_b are correct due to stability of M^0. Clearly, the initialization takes $O(|E|)$ time.

We start by showing, for each iteration i, how to update the aforementioned pieces of information through two series of operations: those underlined in the text, which require constant time, and those wave underlined. Second, we show the correctness of these updates. Lastly, we bound the running time of the algorithm by investigating the number of times we repeat each of the underlined operations and the total time needed to perform wave underlined operations.

For each iteration of the **while** loop, we perform the following updates.

- If P^i is empty, we select the first school that is not in T^i and add it to P^i. This school can be obtained by checking if $\overline{b}_f \in T^i$ and, while $\overline{b}_f \in T^i$, updating $f := f + 1$. So we may assume P^i is non-empty, and represented as $P^i = b_0, a_1, b_1, \cdots, a_k, b_k$.
- Within the iteration, we extend P^i and simultaneously maintain W^i, by finding $a_{k+1} = s_{M^i}(b_k)$, $b_{k+1} = next_{M^i}(b_k)$, $\cdots$ until we reach a node b_j such that either (1) b_j is a sink (step 6); or (2) $next_{M^i}(b_j) = b_\ell$ for some $\ell < j$ (step 8). In particular,
  a) Check if $b_k = \emptyset$. If so, we are in case (1). Otherwise, to obtain $s_{M^i}(b_k)$, we will repeatedly update $p_{b_k} := p_{b_k} + 1$ until $p_{b_k} > \deg_G(b_k)$ (i.e., b_k is a sink and we are in case (1) above) or by scanning of $b_k(p_{b_k})b_k$ we deduce $s_{M^i}(b_k) = b_k(p_{b_k})$.
  b) If $s_{M^i}(b_k)$ is found, we check if $b_{k+1} := next_{M^i}(b_k) \in W^i$. If this happens, we are in case (2) above, otherwise we set $k := k + 1$, and go to a).

- In case (1), the removal of $a_j b_{j-1}$ is achieved by setting $P^{i+1} := P^i \setminus \{a_j, b_j\}$. If $b_j \neq \emptyset$, we also set $W^{i+1} := W^i \setminus \{b_j\}$ and update $T^{i+1} := T^i \cup \{b_j\}$.
- In case (2), a school-rotation exposed in M^i – corresponding to the directed cycle $C^i = b_\ell, \cdots, b_j, a_{j+1}$ – is found and eliminated, as to construct M^{i+1} from M^i. In addition, we update $p_{b_{\ell-1}} := p_{b_{\ell-1}} - 1$ if $\ell > 0$, and set $P^{i+1} := P^i \setminus C^i$, $W^{i+1} := W^i \setminus C^i$.

We now argue about the correctness of these updates. In both cases (1) and (2), P^{i+1} is a directed path of D^{i+1} and W^{i+1}, M^{i+1} is correctly computed. Moreover, because of (‡), sinks of D^i are also sinks in D^{i+1}, justifying the update on T^i and f. Lastly, consider any node $b \in B$ whose associated position p_b is updated in this iteration. There are two scenarios. The first scenario is when looking for $s_{M^i}(b)$, where p_b is repeatedly updated until $p_b > \deg_G(b)$ or until $b(p_b)$ is added to the directed path P^i. In either case, because of (‡) and the fact that every time p_b is updated, it is incremented only by 1, the update of p_b is correct. The second scenario is when we are in case (2) and $b = b_{\ell-1}$, where p_b is updated to be $p_b - 1$. In this case, we found a rotation ρ with $b \notin \rho$ and $next_{M^i}(b) \in \rho$. We carry out the decrement because it is possible to have $s_{M^{i+1}}(b) = s_{M^i}(b)$ and thus re-scanning of $s_{M^i}(b)b$ is required. No further decrements on p_b is needed again because of (‡).

Finally, we will argue about the time complexity. First, note that the number of iterations is clearly bounded by the number of edges plus the number of rotations eliminated. Because of Lemma 2.26, all rotations eliminated throughout the algorithm are also rotations in $\mathcal{SR}(G_L, <, \mathbf{q})$. Note that if M^i is obtained from M^{i-1} by eliminating a rotation, $M^i \succ M^{i-1}$. Hence, no rotation is eliminated twice and thus the number of rotations eliminated is $O(|E|)$ due to Lemma A.3. Thus, the number of iterations is $O(|E|)$.

The total number of updates on P^i, W^i and T^i in case (1) is then also $O(|E|)$. Since f only increases, we update $f := f + 1$ at most $O(|V|)$ times. The number of times we check if $\bar{b}_f \in T^i$ is given by the number of positive answers (proportional to the number of updates of f) plus the number of negative answers (proportional to the number of iterations), hence $O(|E|)$. The number of times $\{p_b\}_{b \in B}$, is updated is given by the number of times we update $p_b := p_b + 1$ (proportional to the number of edges) plus twice the number of times we update $p_b := p_b - 1$ (proportional to the number of rotations), hence $O(|E|)$. From the update on p_b, we see that the only time an edge ab is scanned more than once is when a rotation is eliminated and $b = b_{\ell-1}$, $a = a_\ell$. We call this an *exception*. We claim that we scan each edge at most once, with $O(|E|)$ exceptions. Since every rotation corresponds to at most one exception, the number of exceptions does not exceed the number of rotations, which is $O(|E|)$. Note that each time we check if $p_b > \deg_G(b)$, we either find a sink (which happens at most once per iteration), or we scan an edge (which has been shown to happen $O(|E|)$ times). Hence, the number of times we compare p_b and $\deg_G(b)$ is $O(|E|)$. In addition, the number of times we check if $next_{M^i}(b_j) \in W^i$ is upper bounded by the number of edge scans, hence $O(|E|)$. The number of times we check if $b_k \neq \emptyset$ is upper bounded by the number of times we check if $next_{M^i}(b_j) \in W^i$, so again $O(|E|)$.

The number of individual entry updates when constructing M^{i+1} from M^i, P^{i+1} from P^i, or W^{i+1} from W^i in case (2) is at most the number of edges in all rotations from $\mathcal{SR}(G_L, <, \mathbf{q})$, which is $O(|E|)$ from Lemma A.3, concluding the proof. ■

A.2.3 Example of Algorithm 2.3 Execution

Example A.4. Consider the instance in Example A.1. Assume a_5 is not consenting. In Table A.2, we outline the updates, similar to those in Example A.1. When school b points to the nonconsenting student a_5 (whose partner b_5 is a sink) in the rotation digraph, in addition to remove a_5 and b_5 from the directed path P^i, we also remove b from P^i, set $T^i := T^i \cup \{b\}$, and update p_b to a number that is larger than the length of the preference list of b (e.g., $p_b := 6$) in lieu of the edge removals in Step 9. Such updates can be seen in (1.0), (2.0), (3.0), and (4.0). △

A.2.4 Proof of Lemma 2.33

Proof. The implementation follows as in the proof of Theorem 2.25. The only modification regards the update of T^{i+1} in case (1) considered in the proof, which is when extending the directed path P^i, we encounter a node b_j that is a sink. If a_j consents, then the update on T^{i+1} remains unchanged,

$(i.j)$	P^i	$\{p_b\}_{b\in B}$	M^i	T^i	f
(0.0)	$[] \to b_1$	$[1,1,1,1,1]$	$(4,3,2,1,5)$	$\emptyset$	1
(0.1)	b_1, a_5, b_5	$[2,1,1,1,1]$			
(1.0)	$[] \to b_2$	$[6,1,1,1,6]$		b_1, b_5	2
(1.1)	b_2, a_5, b_5	$[6,2,1,1,6]$			
(2.0)	$[] \to b_3$	$[6,6,1,1,6]$		b_1, b_2, b_5	3
(2.1)	b_3, a_5, b_5	$[6,6,2,1,6]$			
(3.0)	$[] \to b_4$	$[6,6,6,1,6]$		b_1, b_2, b_3, b_5	4
(3.1)	b_4, a_5, b_5	$[6,6,6,2,6]$			
(4.0)	$[]$	$[6,6,6,6,6]$		b_1, b_2, b_3, b_4, b_5	∞

Table A.2: Iterations of `school-rotate-remove with consent` of Example A.4

which is to set $T^{i+1} := T^i \cup \{b_j\}$; however, if a_j is nonconsenting, we set $T^{i+1} := T^i \cup \{b_j, b_{j-1}\}$ and update $p_{b_{j-1}} := \deg(b_{j-1}) + 1$. Correctness analysis and the counting arguments used for time complexity analysis in the proof of Theorem 2.25 remain valid. ■

Appendix B: Additional Details for Chapter 4

B.1 Additional Figures for all Academic Years

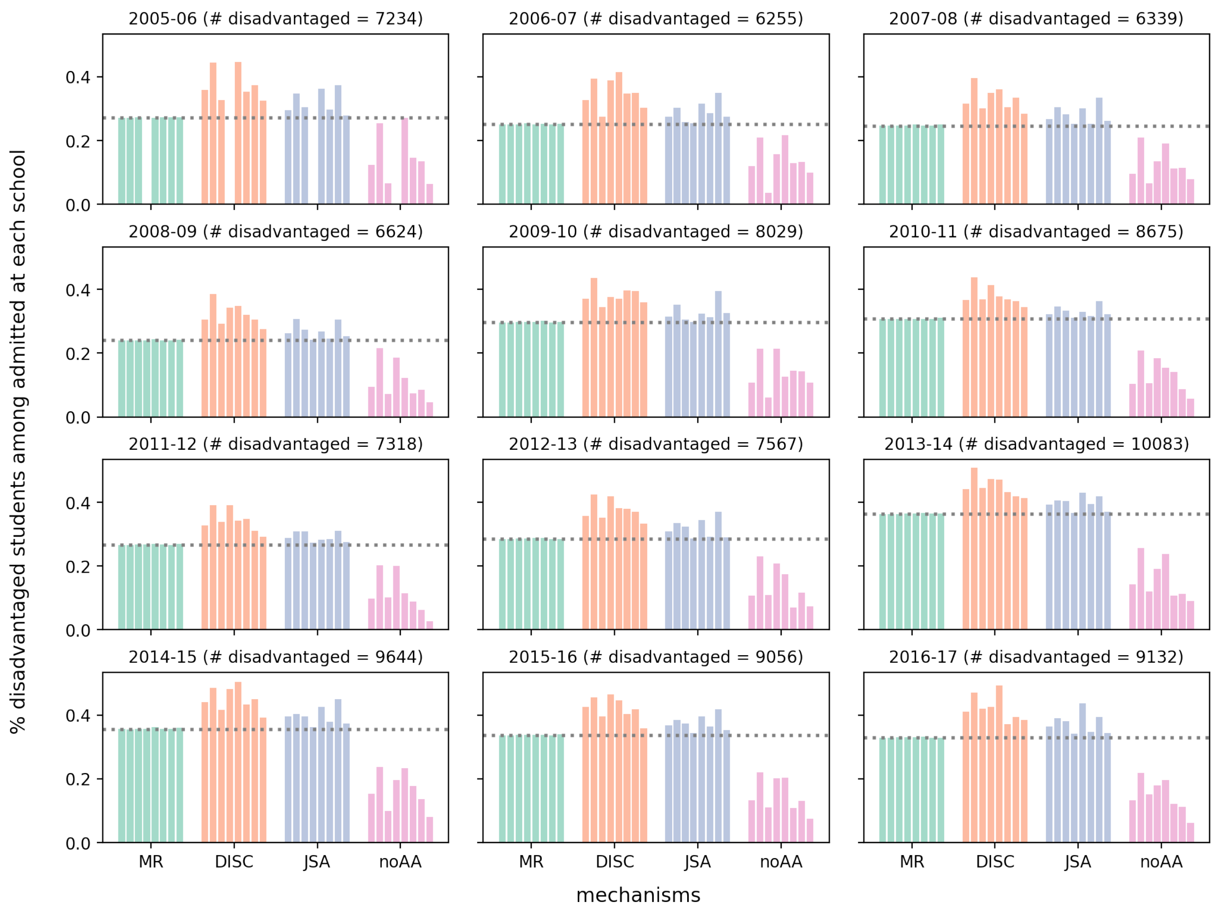


Figure B.1: All academic years of Figure 4.3a.

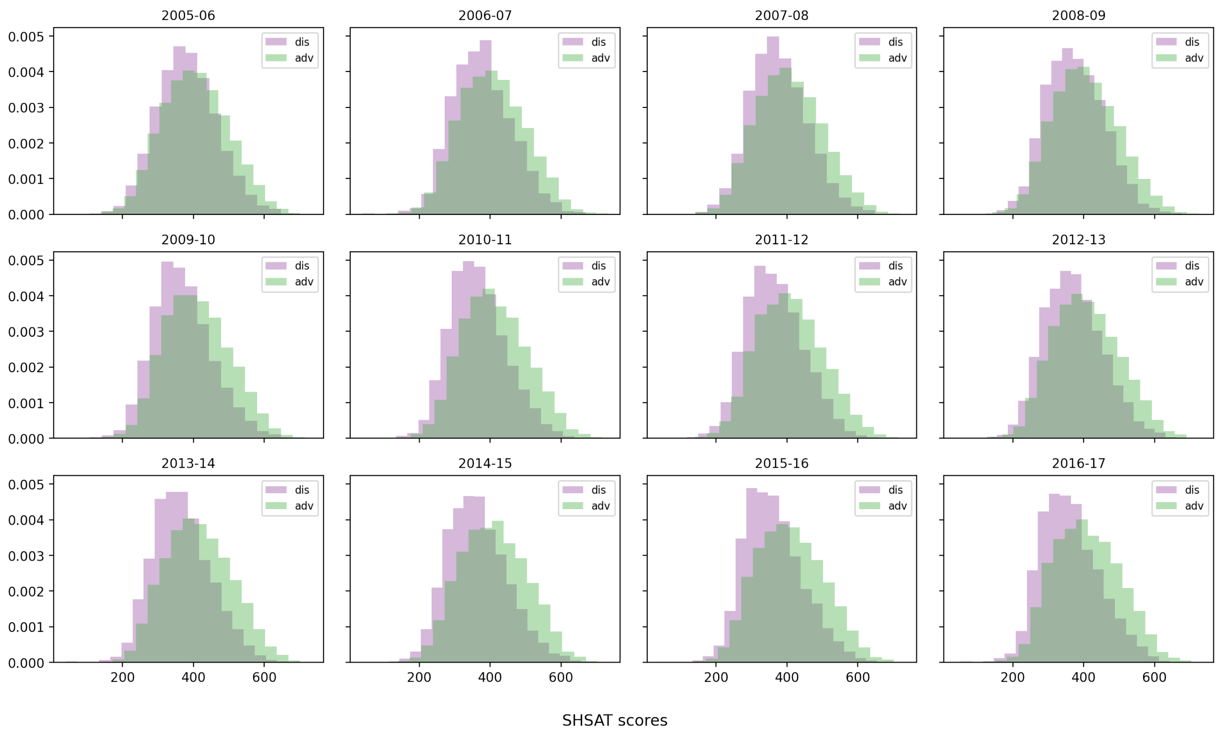


Figure B.2: All academic years of Figure 4.3b.

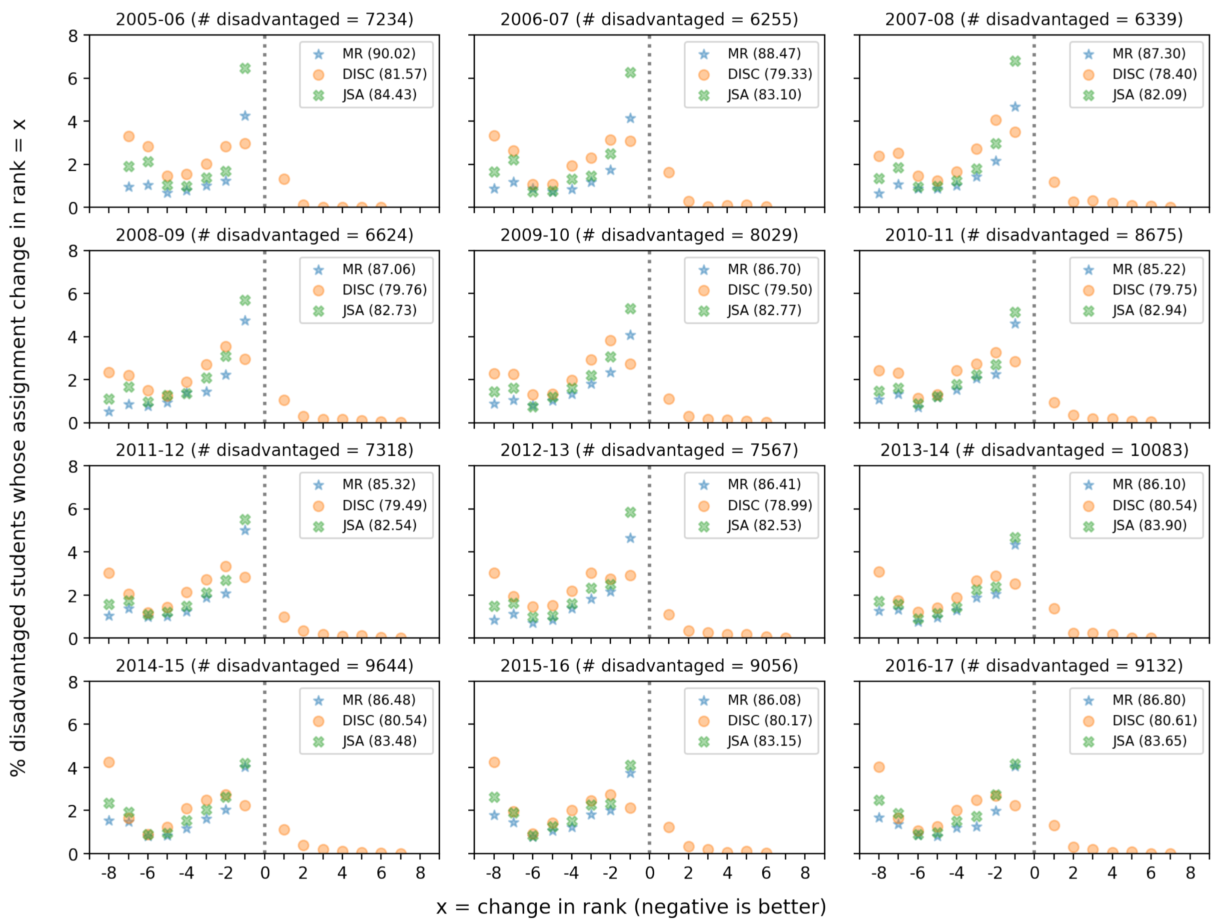


Figure B.3: All academic years of Figure 4.4a.

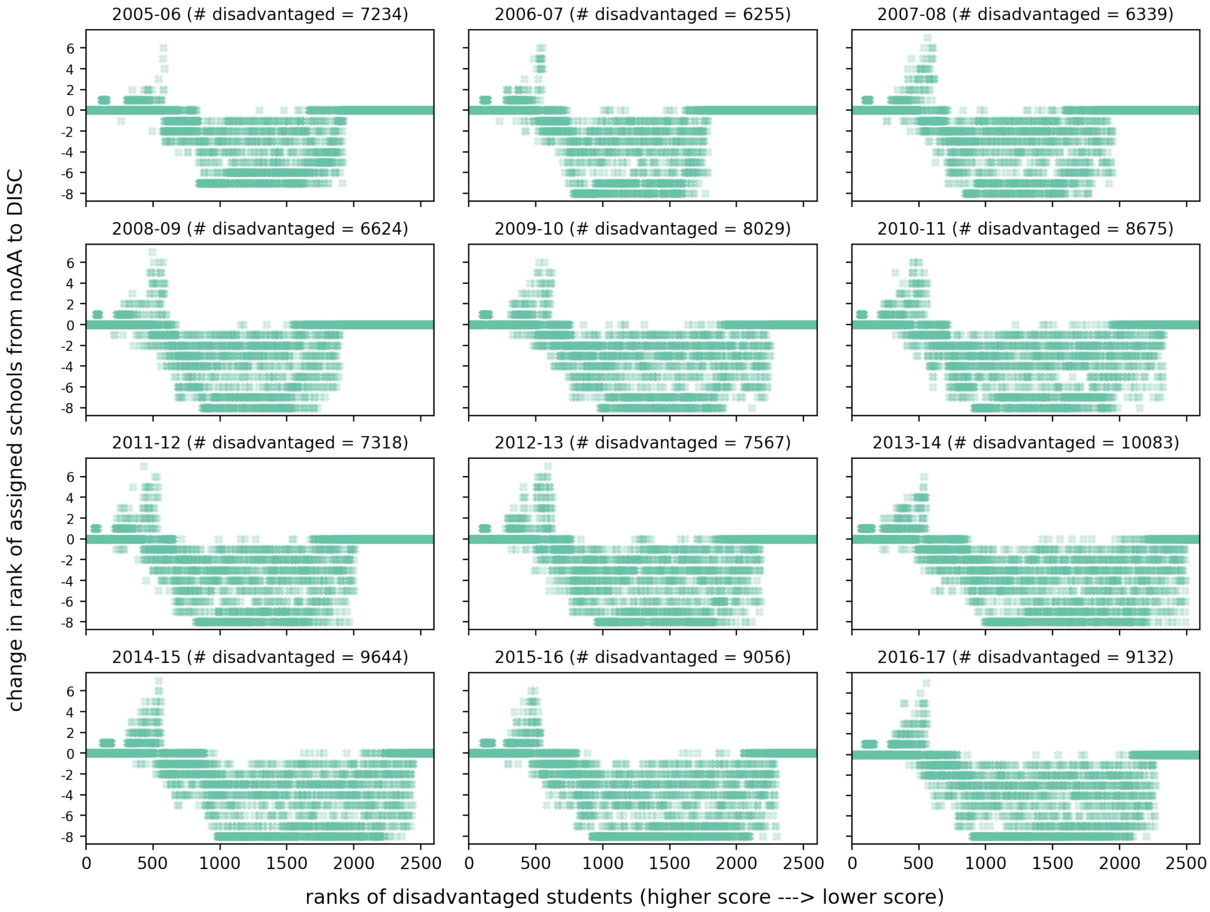


Figure B.4: All academic years of Figure 4.2.

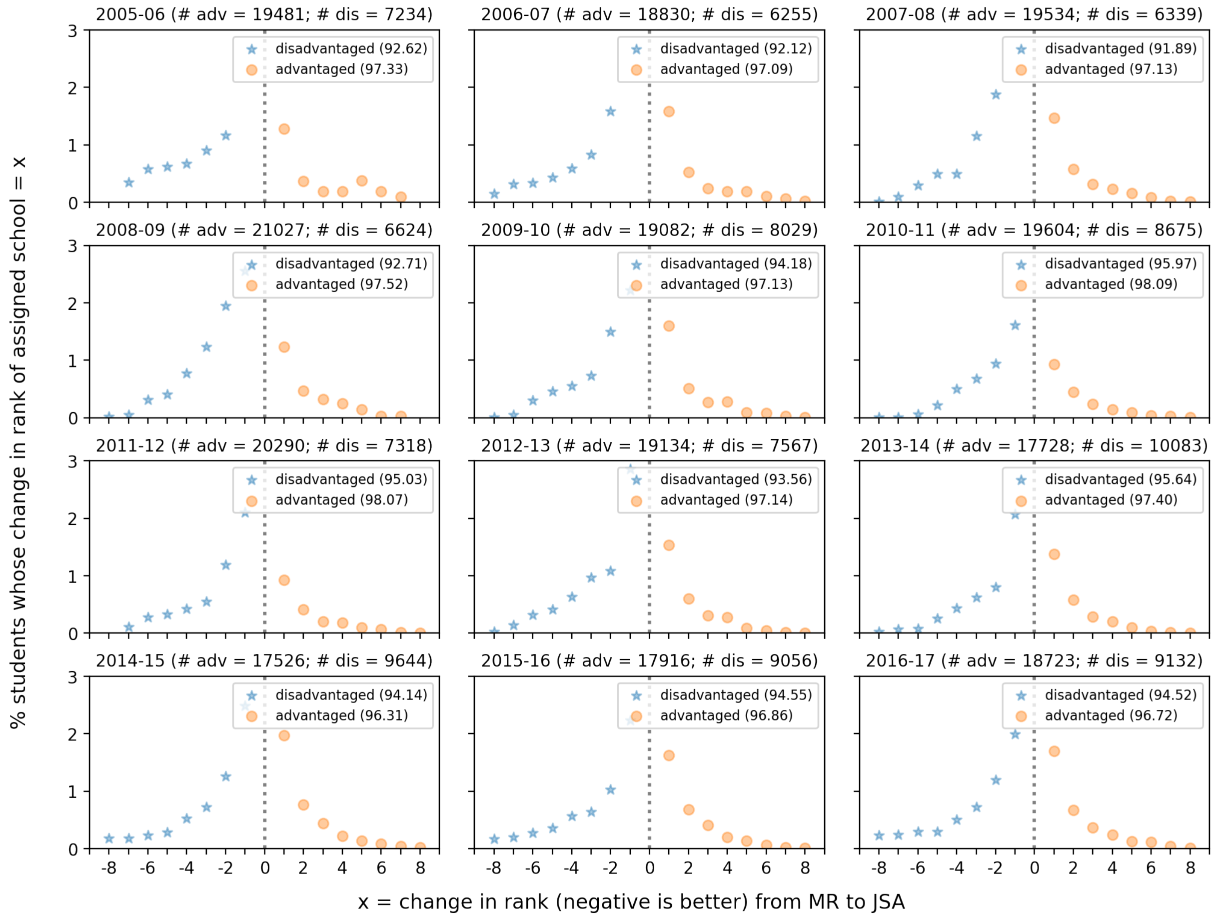


Figure B.5: All academic years of Figure 4.4b.

all students

	S	T	B	Q	R	M	L	A
S	0.0	-0.03	0.24	0.45	0.56	0.32	0.46	0.41
T	0.03	0.0	0.18	0.52	0.65	0.39	0.66	0.47
B	-0.24	-0.18	0.0	0.25	0.39	0.13	0.23	0.28
Q	-0.45	-0.52	-0.25	0.0	0.28	-0.22	-0.01	-0.02
R	-0.56	-0.65	-0.39	-0.28	0.0	-0.41	-0.25	-0.27
M	-0.32	-0.39	-0.13	0.22	0.41	0.0	0.19	0.23
L	-0.46	-0.66	-0.23	0.01	0.25	-0.19	0.0	-0.02
A	-0.41	-0.47	-0.28	0.02	0.27	-0.23	0.02	0.0

students in district 31

	S	T	B	Q	R	M	L	A
S	0.0	0.08	0.48	0.49	-0.87	0.43	0.46	0.48
T	-0.08	0.0	0.48	0.49	-0.94	0.42	0.49	0.47
B	-0.48	-0.48	0.0	0.02	-0.99	-0.13	-0.02	0.02
Q	-0.49	-0.49	-0.02	0.0	-0.99	-0.15	-0.06	-0.01
R	0.87	0.94	0.99	0.99	0.0	0.98	0.98	0.99
M	-0.43	-0.42	0.13	0.15	-0.98	0.0	0.08	0.15
L	-0.46	-0.49	0.02	0.06	-0.98	-0.08	0.0	0.04
A	-0.48	-0.47	-0.02	0.01	-0.99	-0.15	-0.04	0.0

Figure B.6: Each cell in this table represents the extent to which students prefer the row school to the column school. Specifically, the number is calculated as the percentage of students in each district who prefer the row school to the column school minus the percentage of students who prefer the column school to the row school. The cells are color-formatted with numbers in $[-1, 1]$ mapped to a spectrum from red to green.

B.2 Discussion on the school-over-seat hypothesis

In this section, we delve into some empirical observations of students' preference lists and we do so for two reasons. The first one is to investigate the school-over-seat hypothesis. Since students are not asked to report their preferences over different types of seats, we can only make some inferences based on the pattern of the preferences submitted by students. For the second reason, recall that in Section 4.4.3, we show how different mechanisms expand differently students' original preferences over schools to their preferences over reserved and general seats. Hence, our observations aim to shed some light on the validity of these expansions. For the following discussion, we forgo the assumption that participation in the summer enrichment program does not affect students' preference for schools.

The second table in Figure B.6 indicates that geographic proximity could lead to a strong preference for some schools. We observe that students in district 31 strongly prefer Staten Island Tech (S) to any other schools. This is because district 31 is the only school district on Staten Island, and Staten Island Tech is the only specialized high school on Staten Island. Hence, for students residing in Staten Island, since transportation to other boroughs are extremely limited and lengthy, it is reasonable to assume the school-over-seat hypothesis when comparing Staten Island Tech to any other specialized high school. From the same type of tables for other school districts which we include in Appendix B.3, we observe similar patterns: students in district 10 strongly prefers Bronx Science (B) and students in district 29 strongly prefers Queens High School for the Sciences at York (Q). The difference in preferences towards Stuyvesant and Brooklyn Tech seems to be more nuanced. The complete map of school districts in New York City can and the map of specialized high schools can be found in Appendix B.4 and B.5.

Lastly, we would like to point out some concerns that are not directly observable from our data. Aygun and Turhan (2020) noted that for admissions to Indian Institutes of Technology (IIT), there is often social stigma associated with reserved seats and thus, many students prefer to not be admitted via reserved seats. We also note that NYC DOE defines disadvantaged students based on their social economic status instead of a caste system as in the case of IIT admission. Hence, the severity of the social stigma associated with reserved seats might differ between these two markets.

In sum, we believe more study is needed to understand students' preference structure over reserved and general seats for the NYC SHS market. Moreover, as a future direction, it would be interesting to design and study mechanisms which incorporate students' preferences over general and reserved seats at all schools, possibly in orders that are not consistent with those interpreted by the mechanisms.

B.3 Additional Figures for Section B.2

district = 1

	S	T	B	Q	R	M	L	A
S	0.0	0.16	0.47	0.72	0.76	0.51	0.67	0.69
T	-0.16	0.0	0.3	0.81	0.81	0.47	0.76	0.69
B	-0.47	-0.3	0.0	0.49	0.56	0.15	0.35	0.42
Q	-0.72	-0.81	-0.49	0.0	0.28	-0.49	-0.17	-0.18
R	-0.76	-0.81	-0.56	-0.28	0.0	-0.57	-0.31	-0.31
M	-0.51	-0.47	-0.15	0.49	0.57	0.0	0.26	0.4
L	-0.67	-0.76	-0.35	0.17	0.31	-0.26	0.0	0.03
A	-0.69	-0.69	-0.42	0.18	0.31	-0.4	-0.03	0.0

district = 2

	S	T	B	Q	R	M	L	A
S	0.0	0.17	0.36	0.74	0.77	0.44	0.58	0.49
T	-0.17	0.0	0.11	0.86	0.89	0.47	0.81	0.56
B	-0.36	-0.11	0.0	0.65	0.71	0.3	0.49	0.47
Q	-0.74	-0.86	-0.65	0.0	0.23	-0.59	-0.26	-0.38
R	-0.77	-0.89	-0.71	-0.23	0.0	-0.62	-0.38	-0.46
M	-0.44	-0.47	-0.3	0.59	0.62	0.0	0.36	0.23
L	-0.58	-0.81	-0.49	0.26	0.38	-0.36	0.0	-0.14
A	-0.49	-0.56	-0.47	0.38	0.46	-0.23	0.14	0.0

district = 3

	S	T	B	Q	R	M	L	A
S	0.0	0.21	-0.12	0.74	0.8	0.08	0.57	0.11
T	-0.21	0.0	-0.54	0.73	0.78	-0.23	0.61	-0.13
B	0.12	0.54	0.0	0.87	0.88	0.4	0.78	0.4
Q	-0.74	-0.73	-0.87	0.0	0.26	-0.78	-0.3	-0.73
R	-0.8	-0.78	-0.88	-0.26	0.0	-0.8	-0.42	-0.77
M	-0.08	0.23	-0.4	0.78	0.8	0.0	0.64	0.09
L	-0.57	-0.61	-0.78	0.3	0.42	-0.64	0.0	-0.63
A	-0.11	0.13	-0.4	0.73	0.77	-0.09	0.63	0.0

district = 4

	S	T	B	Q	R	M	L	A
S	0.0	0.47	0.27	0.78	0.87	0.31	0.78	0.57
T	-0.47	0.0	-0.35	0.59	0.69	-0.14	0.55	0.21
B	-0.27	0.35	0.0	0.7	0.79	0.17	0.72	0.43
Q	-0.78	-0.59	-0.7	0.0	0.5	-0.64	0.0	-0.35
R	-0.87	-0.69	-0.79	-0.5	0.0	-0.78	-0.39	-0.58
M	-0.31	0.14	-0.17	0.64	0.78	0.0	0.53	0.3
L	-0.78	-0.55	-0.72	0.0	0.39	-0.53	0.0	-0.37
A	-0.57	-0.21	-0.43	0.35	0.58	-0.3	0.37	0.0

district = 5

	S	T	B	Q	R	M	L	A
S	0.0	0.61	0.33	0.75	0.82	0.11	0.73	0.57
T	-0.61	0.0	-0.36	0.5	0.55	-0.44	0.39	0.07
B	-0.33	0.36	0.0	0.74	0.7	-0.09	0.63	0.54
Q	-0.75	-0.5	-0.74	0.0	0.3	-0.71	-0.1	-0.39
R	-0.82	-0.55	-0.7	-0.3	0.0	-0.74	-0.35	-0.42
M	-0.11	0.44	0.09	0.71	0.74	0.0	0.67	0.54
L	-0.73	-0.39	-0.63	0.1	0.35	-0.67	0.0	-0.34
A	-0.57	-0.07	-0.54	0.39	0.42	-0.54	0.34	0.0

district = 6

	S	T	B	Q	R	M	L	A
S	0.0	0.63	0.1	0.82	0.84	-0.06	0.78	0.33
T	-0.63	0.0	-0.58	0.36	0.47	-0.69	0.37	-0.28
B	-0.1	0.58	0.0	0.75	0.77	-0.22	0.76	0.3
Q	-0.82	-0.36	-0.75	0.0	0.29	-0.88	-0.02	-0.57
R	-0.84	-0.47	-0.77	-0.29	0.0	-0.88	-0.23	-0.66
M	0.06	0.69	0.22	0.88	0.88	0.0	0.84	0.52
L	-0.78	-0.37	-0.76	0.02	0.23	-0.84	0.0	-0.56
A	-0.33	0.28	-0.3	0.57	0.66	-0.52	0.56	0.0

district = 7

	S	T	B	Q	R	M	L	A
S	0.0	0.24	-0.67	0.48	0.54	-0.29	0.39	-0.18
T	-0.24	0.0	-0.8	0.37	0.42	-0.53	0.23	-0.41
B	0.67	0.8	0.0	0.9	0.94	0.42	0.88	0.6
Q	-0.48	-0.37	-0.9	0.0	0.3	-0.7	-0.22	-0.61
R	-0.54	-0.42	-0.94	-0.3	0.0	-0.76	-0.29	-0.72
M	0.29	0.53	-0.42	0.7	0.76	0.0	0.65	0.16
L	-0.39	-0.23	-0.88	0.22	0.29	-0.65	0.0	-0.55
A	0.18	0.41	-0.6	0.61	0.72	-0.16	0.55	0.0

district = 8

	S	T	B	Q	R	M	L	A
S	0.0	0.16	-0.7	0.32	0.59	-0.15	0.35	-0.4
T	-0.16	0.0	-0.83	0.26	0.6	-0.24	0.45	-0.39
B	0.7	0.83	0.0	0.97	0.97	0.72	0.93	0.45
Q	-0.32	-0.26	-0.97	0.0	0.47	-0.59	0.09	-0.73
R	-0.59	-0.6	-0.97	-0.47	0.0	-0.72	-0.37	-0.8
M	0.15	0.24	-0.72	0.59	0.72	0.0	0.54	-0.24
L	-0.35	-0.45	-0.93	-0.09	0.37	-0.54	0.0	-0.67
A	0.4	0.39	-0.45	0.73	0.8	0.24	0.67	0.0

district = 9

	S	T	B	Q	R	M	L	A
S	0.0	0.22	-0.72	0.44	0.63	-0.22	0.35	-0.39
T	-0.22	0.0	-0.83	0.27	0.49	-0.45	0.24	-0.53
B	0.72	0.83	0.0	0.91	0.96	0.6	0.86	0.48
Q	-0.44	-0.27	-0.91	0.0	0.38	-0.69	-0.1	-0.73
R	-0.63	-0.49	-0.96	-0.38	0.0	-0.76	-0.33	-0.82
M	0.22	0.45	-0.6	0.69	0.76	0.0	0.59	-0.08
L	-0.35	-0.24	-0.86	0.1	0.33	-0.59	0.0	-0.66
A	0.39	0.53	-0.48	0.73	0.82	0.08	0.66	0.0

district = 10

	S	T	B	Q	R	M	L	A
S	0.0	0.25	-0.77	0.48	0.53	-0.19	0.42	-0.49
T	-0.25	0.0	-0.89	0.29	0.43	-0.4	0.32	-0.66
B	0.77	0.89	0.0	0.95	0.96	0.74	0.94	0.41
Q	-0.48	-0.29	-0.95	0.0	0.27	-0.66	-0.05	-0.84
R	-0.53	-0.43	-0.96	-0.27	0.0	-0.7	-0.28	-0.86
M	0.19	0.4	-0.74	0.66	0.7	0.0	0.61	-0.43
L	-0.42	-0.32	-0.94	0.05	0.28	-0.61	0.0	-0.81
A	0.49	0.66	-0.41	0.84	0.86	0.43	0.81	0.0

district = 11

	S	T	B	Q	R	M	L	A
S	0.0	0.19	-0.76	0.38	0.53	-0.18	0.39	-0.42
T	-0.19	0.0	-0.85	0.31	0.52	-0.32	0.39	-0.44
B	0.76	0.85	0.0	0.94	0.96	0.73	0.9	0.51
Q	-0.38	-0.31	-0.94	0.0	0.38	-0.65	-0.03	-0.71
R	-0.53	-0.52	-0.96	-0.38	0.0	-0.71	-0.35	-0.77
M	0.18	0.32	-0.73	0.65	0.71	0.0	0.55	-0.24
L	-0.39	-0.39	-0.9	0.03	0.35	-0.55	0.0	-0.66
A	0.42	0.44	-0.51	0.71	0.77	0.24	0.66	0.0

district = 12

	S	T	B	Q	R	M	L	A
S	0.0	0.22	-0.68	0.34	0.5	-0.15	0.34	-0.29
T	-0.22	0.0	-0.84	0.28	0.49	-0.49	0.25	-0.46
B	0.68	0.84	0.0	0.91	0.93	0.68	0.9	0.64
Q	-0.34	-0.28	-0.91	0.0	0.35	-0.7	-0.06	-0.6
R	-0.5	-0.49	-0.93	-0.35	0.0	-0.71	-0.34	-0.7
M	0.15	0.49	-0.68	0.7	0.71	0.0	0.56	-0.04
L	-0.34	-0.25	-0.9	0.06	0.34	-0.56	0.0	-0.55
A	0.29	0.46	-0.64	0.6	0.7	0.04	0.55	0.0

district = 13

	S	T	B	Q	R	M	L	A
S	0.0	-0.42	0.61	0.59	0.71	0.38	0.22	0.52
T	0.42	0.0	0.98	0.93	0.97	0.9	0.84	0.95
B	-0.61	-0.98	0.0	0.04	0.22	-0.35	-0.51	-0.11
Q	-0.59	-0.93	-0.04	0.0	0.31	-0.45	-0.51	-0.16
R	-0.71	-0.97	-0.22	-0.31	0.0	-0.55	-0.63	-0.34
M	-0.38	-0.9	0.35	0.45	0.55	0.0	-0.16	0.39
L	-0.22	-0.84	0.51	0.51	0.63	0.16	0.0	0.49
A	-0.52	-0.95	0.11	0.16	0.34	-0.39	-0.49	0.0

district = 14

	S	T	B	Q	R	M	L	A
S	0.0	-0.46	0.65	0.56	0.7	0.48	0.03	0.64
T	0.46	0.0	0.86	0.89	0.92	0.85	0.52	0.91
B	-0.65	-0.86	0.0	-0.1	0.18	-0.19	-0.56	-0.04
Q	-0.56	-0.89	0.1	0.0	0.33	-0.11	-0.6	0.12
R	-0.7	-0.92	-0.18	-0.33	0.0	-0.41	-0.7	-0.28
M	-0.48	-0.85	0.19	0.11	0.41	0.0	-0.5	0.27
L	-0.03	-0.52	0.56	0.6	0.7	0.5	0.0	0.65
A	-0.64	-0.91	0.04	-0.12	0.28	-0.27	-0.65	0.0

district = 15

	S	T	B	Q	R	M	L	A
S	0.0	-0.5	0.63	0.64	0.62	0.47	0.2	0.62
T	0.5	0.0	0.96	0.98	0.94	0.94	0.85	0.96
B	-0.63	-0.96	0.0	0.07	0.0	-0.25	-0.54	-0.0
Q	-0.64	-0.98	-0.07	0.0	-0.05	-0.37	-0.64	-0.13
R	-0.62	-0.94	0.0	0.05	0.0	-0.24	-0.55	-0.02
M	-0.47	-0.94	0.25	0.37	0.24	0.0	-0.38	0.3
L	-0.2	-0.85	0.54	0.64	0.55	0.38	0.0	0.6
A	-0.62	-0.96	0.0	0.13	0.02	-0.3	-0.6	0.0

district = 16

	S	T	B	Q	R	M	L	A
S	0.0	-0.74	0.57	0.52	0.62	0.29	0.22	0.52
T	0.74	0.0	0.94	0.97	0.94	0.97	0.82	0.97
B	-0.57	-0.94	0.0	-0.18	0.22	-0.45	-0.29	-0.14
Q	-0.52	-0.97	0.18	0.0	0.46	-0.45	-0.12	0.05
R	-0.62	-0.94	-0.22	-0.46	0.0	-0.62	-0.45	-0.37
M	-0.29	-0.97	0.45	0.45	0.62	0.0	0.14	0.52
L	-0.22	-0.82	0.29	0.12	0.45	-0.14	0.0	0.22
A	-0.52	-0.97	0.14	-0.05	0.37	-0.52	-0.22	0.0

Figure B.7: These tables are the same as those in Figure B.6, but for districts 1 – 16.

district = 17

	S	T	B	Q	R	M	L	A
S	0.0	-0.66	0.53	0.53	0.61	0.25	0.23	0.55
T	0.66	0.0	0.93	0.91	0.97	0.83	0.8	0.92
B	-0.53	-0.93	0.0	-0.07	0.2	-0.4	-0.33	0.01
Q	-0.53	-0.91	0.07	0.0	0.26	-0.47	-0.35	-0.02
R	-0.61	-0.97	-0.2	-0.26	0.0	-0.55	-0.47	-0.18
M	-0.25	-0.83	0.4	0.47	0.55	0.0	-0.03	0.51
L	-0.23	-0.8	0.33	0.35	0.47	0.03	0.0	0.39
A	-0.55	-0.92	-0.01	0.02	0.18	-0.51	-0.39	0.0

district = 18

	S	T	B	Q	R	M	L	A
S	0.0	-0.59	0.66	0.52	0.67	0.25	0.22	0.63
T	0.59	0.0	0.98	0.92	0.97	0.85	0.86	0.98
B	-0.66	-0.98	0.0	-0.19	0.2	-0.5	-0.5	-0.05
Q	-0.52	-0.92	0.19	0.0	0.34	-0.37	-0.43	0.1
R	-0.67	-0.97	-0.2	-0.34	0.0	-0.61	-0.6	-0.23
M	-0.25	-0.85	0.5	0.37	0.61	0.0	-0.12	0.49
L	-0.22	-0.86	0.5	0.43	0.6	0.12	0.0	0.56
A	-0.63	-0.98	0.05	-0.1	0.23	-0.49	-0.56	0.0

district = 19

	S	T	B	Q	R	M	L	A
S	0.0	-0.6	0.68	0.29	0.72	0.31	0.28	0.62
T	0.6	0.0	0.95	0.78	0.97	0.85	0.85	0.94
B	-0.68	-0.95	0.0	-0.5	0.25	-0.43	-0.39	-0.14
Q	-0.29	-0.78	0.5	0.0	0.67	0.08	0.04	0.43
R	-0.72	-0.97	-0.25	-0.67	0.0	-0.62	-0.56	-0.39
M	-0.31	-0.85	0.43	-0.08	0.62	0.0	0.02	0.39
L	-0.28	-0.85	0.39	-0.04	0.56	-0.02	0.0	0.34
A	-0.62	-0.94	0.14	-0.43	0.39	-0.39	-0.34	0.0

district = 20

	S	T	B	Q	R	M	L	A
S	0.0	-0.34	0.67	0.69	0.23	0.59	0.41	0.67
T	0.34	0.0	0.95	0.97	0.46	0.94	0.94	0.97
B	-0.67	-0.95	0.0	0.06	-0.54	-0.15	-0.33	0.06
Q	-0.69	-0.97	-0.06	0.0	-0.57	-0.3	-0.4	-0.07
R	-0.23	-0.46	0.54	0.57	0.0	0.43	0.31	0.57
M	-0.59	-0.94	0.15	0.3	-0.43	0.0	-0.18	0.29
L	-0.41	-0.94	0.33	0.4	-0.31	0.18	0.0	0.4
A	-0.67	-0.97	-0.06	0.07	-0.57	-0.29	-0.4	0.0

district = 21

	S	T	B	Q	R	M	L	A
S	0.0	-0.38	0.73	0.75	0.41	0.65	0.53	0.73
T	0.38	0.0	0.96	0.98	0.71	0.93	0.94	0.98
B	-0.73	-0.96	0.0	0.09	-0.44	-0.12	-0.23	0.07
Q	-0.75	-0.98	-0.09	0.0	-0.49	-0.28	-0.3	-0.03
R	-0.41	-0.71	0.44	0.49	0.0	0.3	0.21	0.43
M	-0.65	-0.93	0.12	0.28	-0.3	0.0	-0.12	0.26
L	-0.53	-0.94	0.23	0.3	-0.21	0.12	0.0	0.3
A	-0.73	-0.98	-0.07	0.03	-0.43	-0.26	-0.3	0.0

district = 22

	S	T	B	Q	R	M	L	A
S	0.0	-0.55	0.66	0.66	0.62	0.53	0.48	0.67
T	0.55	0.0	0.95	0.96	0.94	0.93	0.9	0.96
B	-0.66	-0.95	0.0	-0.01	-0.08	-0.25	-0.3	-0.0
Q	-0.66	-0.96	0.01	0.0	-0.05	-0.28	-0.3	0.0
R	-0.62	-0.94	0.08	0.05	0.0	-0.17	-0.21	0.07
M	-0.53	-0.93	0.25	0.28	0.17	0.0	-0.09	0.33
L	-0.48	-0.9	0.3	0.3	0.21	0.09	0.0	0.32
A	-0.67	-0.96	0.0	0.0	-0.07	-0.33	-0.32	0.0

district = 23

	S	T	B	Q	R	M	L	A
S	0.0	-0.57	0.74	0.56	0.78	0.41	0.36	0.64
T	0.57	0.0	0.93	0.9	0.97	0.8	0.82	0.91
B	-0.74	-0.93	0.0	-0.28	0.17	-0.47	-0.53	-0.19
Q	-0.56	-0.9	0.28	0.0	0.42	-0.3	-0.34	0.08
R	-0.78	-0.97	-0.17	-0.42	0.0	-0.53	-0.53	-0.29
M	-0.41	-0.8	0.47	0.3	0.53	0.0	-0.12	0.42
L	-0.36	-0.82	0.53	0.34	0.53	0.12	0.0	0.43
A	-0.64	-0.91	0.19	-0.08	0.29	-0.42	-0.43	0.0

district = 24

	S	T	B	Q	R	M	L	A
S	0.0	0.09	0.36	0.28	0.7	0.35	0.48	0.53
T	-0.09	0.0	0.26	0.35	0.83	0.46	0.67	0.67
B	-0.36	-0.26	0.0	0.0	0.62	0.14	0.31	0.38
Q	-0.28	-0.35	-0.0	0.0	0.74	0.14	0.33	0.49
R	-0.7	-0.83	-0.62	-0.74	0.0	-0.66	-0.46	-0.42
M	-0.35	-0.46	-0.14	-0.14	0.66	0.0	0.21	0.42
L	-0.48	-0.67	-0.31	-0.33	0.46	-0.21	0.0	0.09
A	-0.53	-0.67	-0.38	-0.49	0.42	-0.42	-0.09	0.0

district = 25

	S	T	B	Q	R	M	L	A
S	0.0	0.29	0.02	0.23	0.76	0.52	0.66	0.59
T	-0.29	0.0	-0.47	-0.08	0.75	0.39	0.66	0.48
B	-0.02	0.47	0.0	0.4	0.87	0.66	0.77	0.74
Q	-0.23	0.08	-0.4	0.0	0.77	0.42	0.63	0.56
R	-0.76	-0.75	-0.87	-0.77	0.0	-0.55	-0.29	-0.37
M	-0.52	-0.39	-0.66	-0.42	0.55	0.0	0.31	0.25
L	-0.66	-0.66	-0.77	-0.63	0.29	-0.31	0.0	-0.15
A	-0.59	-0.48	-0.74	-0.56	0.37	-0.25	0.15	0.0

district = 26

	S	T	B	Q	R	M	L	A
S	0.0	0.16	-0.13	0.05	0.7	0.44	0.53	0.49
T	-0.16	0.0	-0.55	-0.16	0.75	0.48	0.69	0.46
B	0.13	0.55	0.0	0.45	0.92	0.76	0.8	0.85
Q	-0.05	0.16	-0.45	0.0	0.83	0.61	0.67	0.63
R	-0.7	-0.75	-0.92	-0.83	0.0	-0.5	-0.29	-0.4
M	-0.44	-0.48	-0.76	-0.61	0.5	0.0	0.24	0.16
L	-0.53	-0.69	-0.8	-0.67	0.29	-0.24	0.0	-0.15
A	-0.49	-0.46	-0.85	-0.63	0.4	-0.16	0.15	0.0

district = 27

	S	T	B	Q	R	M	L	A
S	0.0	-0.25	0.5	-0.13	0.69	0.28	0.46	0.59
T	0.25	0.0	0.75	0.21	0.88	0.6	0.82	0.81
B	-0.5	-0.75	0.0	-0.6	0.4	-0.27	-0.07	0.13
Q	0.13	-0.21	0.6	0.0	0.81	0.41	0.56	0.75
R	-0.69	-0.88	-0.4	-0.81	0.0	-0.62	-0.39	-0.28
M	-0.28	-0.6	0.27	-0.41	0.62	0.0	0.19	0.5
L	-0.46	-0.82	0.07	-0.56	0.39	-0.19	0.0	0.19
A	-0.59	-0.81	-0.13	-0.75	0.28	-0.5	-0.19	0.0

district = 28

	S	T	B	Q	R	M	L	A
S	0.0	0.07	0.25	-0.05	0.68	0.35	0.46	0.52
T	-0.07	0.0	0.02	-0.03	0.77	0.41	0.65	0.6
B	-0.25	-0.02	0.0	-0.07	0.65	0.24	0.41	0.46
Q	0.05	0.03	0.07	0.0	0.78	0.44	0.54	0.63
R	-0.68	-0.77	-0.65	-0.78	0.0	-0.58	-0.36	-0.38
M	-0.35	-0.41	-0.24	-0.44	0.58	0.0	0.2	0.36
L	-0.46	-0.65	-0.41	-0.54	0.36	-0.2	0.0	0.06
A	-0.52	-0.6	-0.46	-0.63	0.38	-0.36	-0.06	0.0

district = 29

	S	T	B	Q	R	M	L	A
S	0.0	-0.11	0.34	-0.35	0.64	0.22	0.41	0.49
T	0.11	0.0	0.49	-0.24	0.75	0.34	0.62	0.57
B	-0.34	-0.49	0.0	-0.57	0.43	-0.19	0.1	0.16
Q	0.35	0.24	0.57	0.0	0.86	0.57	0.7	0.77
R	-0.64	-0.75	-0.43	-0.86	0.0	-0.65	-0.36	-0.39
M	-0.22	-0.34	0.19	-0.57	0.65	0.0	0.26	0.43
L	-0.41	-0.62	-0.1	-0.7	0.36	-0.26	0.0	0.01
A	-0.49	-0.57	-0.16	-0.77	0.39	-0.43	-0.01	0.0

district = 30

	S	T	B	Q	R	M	L	A
S	0.0	0.23	0.35	0.37	0.77	0.4	0.58	0.57
T	-0.23	0.0	0.05	0.36	0.82	0.42	0.7	0.58
B	-0.35	-0.05	0.0	0.19	0.7	0.23	0.45	0.45
Q	-0.37	-0.36	-0.19	0.0	0.75	-0.02	0.3	0.3
R	-0.77	-0.82	-0.7	-0.75	0.0	-0.74	-0.48	-0.54
M	-0.4	-0.42	-0.23	0.02	0.74	0.0	0.31	0.36
L	-0.58	-0.7	-0.45	-0.3	0.48	-0.31	0.0	-0.05
A	-0.57	-0.58	-0.45	-0.3	0.54	-0.36	0.05	0.0

district = 31

	S	T	B	Q	R	M	L	A
S	0.0	0.08	0.48	0.49	-0.87	0.43	0.46	0.48
T	-0.08	0.0	0.48	0.49	-0.94	0.42	0.49	0.47
B	-0.48	-0.48	0.0	0.02	-0.99	-0.13	-0.02	0.02
Q	-0.49	-0.49	-0.02	0.0	-0.99	-0.15	-0.06	-0.01
R	0.87	0.94	0.99	0.99	0.0	0.98	0.98	0.99
M	-0.43	-0.42	0.13	0.15	-0.98	0.0	0.08	0.15
L	-0.46	-0.49	0.02	0.06	-0.98	-0.08	0.0	0.04
A	-0.48	-0.47	-0.02	0.01	-0.99	-0.15	-0.04	0.0

district = 32

	S	T	B	Q	R	M	L	A
S	0.0	-0.47	0.67	0.47	0.72	0.4	0.22	0.62
T	0.47	0.0	0.93	0.91	0.96	0.88	0.69	0.94
B	-0.67	-0.93	0.0	-0.39	0.28	-0.5	-0.51	-0.16
Q	-0.47	-0.91	0.39	0.0	0.6	-0.2	-0.25	0.2
R	-0.72	-0.96	-0.28	-0.6	0.0	-0.65	-0.68	-0.43
M	-0.4	-0.88	0.5	0.2	0.65	0.0	-0.2	0.48
L	-0.22	-0.69	0.51	0.25	0.68	0.2	0.0	0.5
A	-0.62	-0.94	0.16	-0.2	0.43	-0.48	-0.5	0.0

Figure B.8: These tables are the same as those in Figure B.6, but for districts 17 – 32.

B.4 Map of NYC School Districts

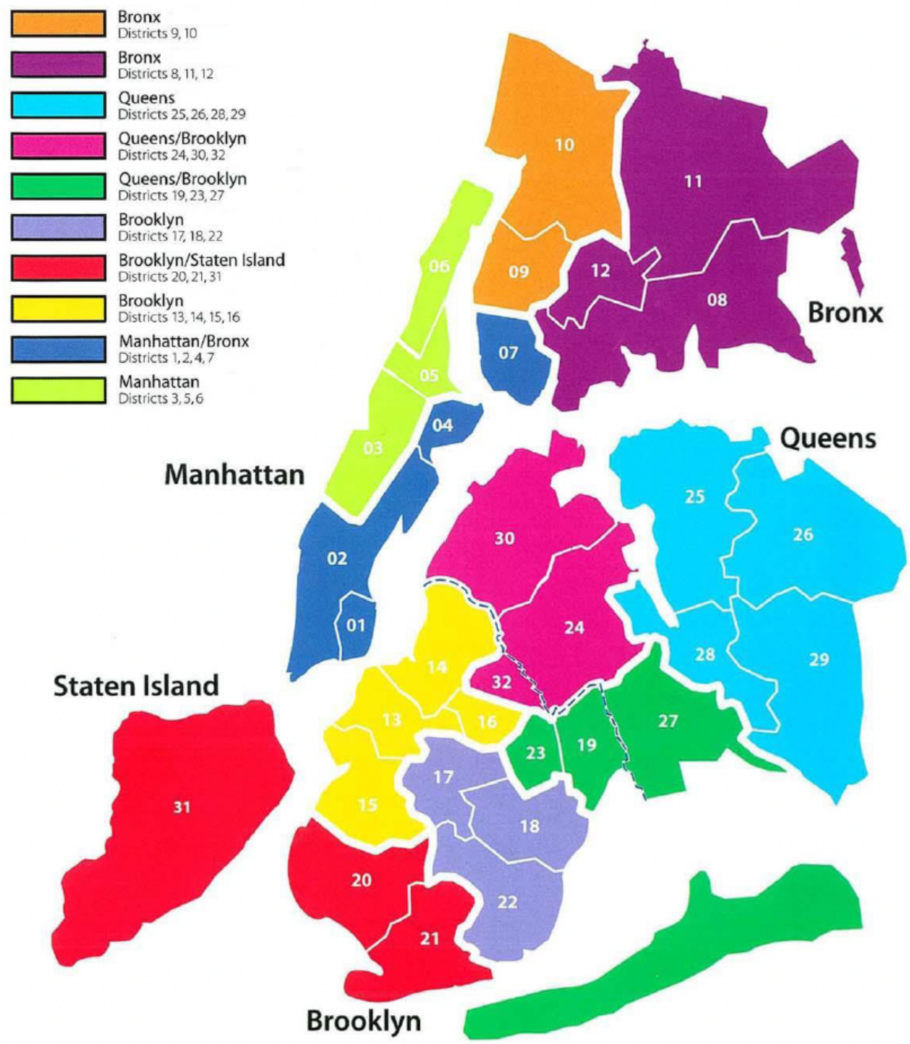


Figure B.9: Map of school districts in New York City, compiled by NYC DOE and available online at `https://video.eschoolsolutions.com/udocs/DistrictMap.pdf`

B.5 Map of NYC Specialized High Schools

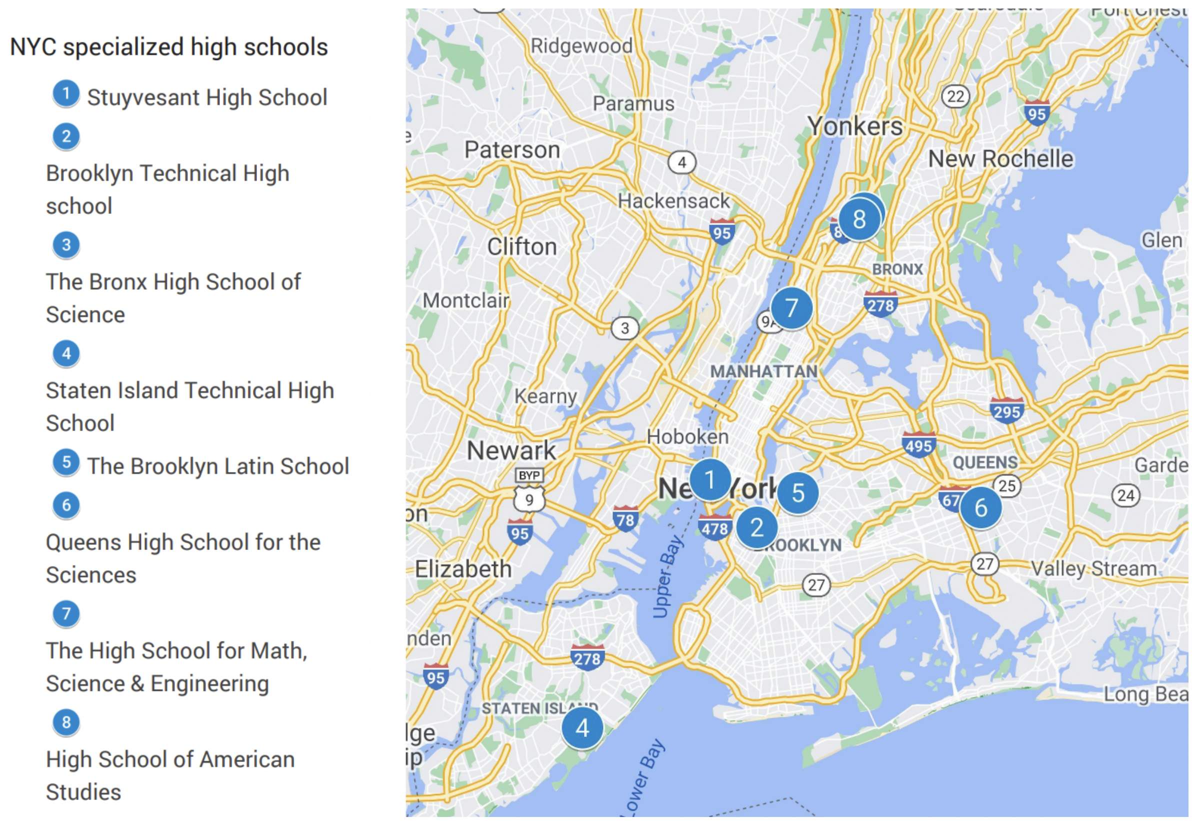


Figure B.10: Map of specialized high schools in New York City. In Bronx, the two schools numbered by 3 and 8 are overlapping on the map. The map is generated by *Google My Maps*.