Matching market design : from theory to practice

Li, Mengling

2015

Li, M. (2015). Matching market design : from theory to practice. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/65410>

<https://doi.org/10.32657/10356/65410>

(On the spine)

MATCHING MARKET DESIGN: FROM THEORY TO PRACTICE

LI MENG LING

2015

**MATCHING MARKET DESIGN:
FROM THEORY TO PRACTICE**

LI MENG LING

SCHOOL OF PHYSICAL AND MATHEMATICAL SCIENCES

2015

MATCHING MARKET DESIGN: FROM THEORY TO PRACTICE

LI MENGLING

Division of Mathematical Sciences
School of Physical and Mathematical Sciences

A thesis submitted to the Nanyang Technological University
in fulfilment of the requirement for the degree of
Doctor of Philosophy

2015

Acknowledgements

First and foremost, I would like to express my heartfelt gratitude to my supervisor, Associate Professor Chen Ning, for his invaluable supervision and encouragement, without which this thesis could not have reached its present form. He has opened my eyes to the possibilities for theories to solve problems in the real-life world. I hope that one day I will finally be able to thank him by having the opportunity to impart a sense of his vision and scholarship on another generation of students.

Furthermore, I am indebted to Associate Professor Chia Wai-Mun from the Division of Economics at Nanyang Technological University for bringing me into the field of economics and guiding me on many other interesting economics projects beyond this thesis. Her diligence, patience, and enthusiasm on research have been steadily influencing me.

I would like to thank my thesis committee members, Associate Professors Wang Huaxiong, Wu Guohua and Nanyang Assistant Professor An Bo, for their time and support. I am also thankful to the three anonymous examiners for reading this thesis. Their comments have improved the presentation of this work.

I am deeply grateful to Xu Shaofeng for providing me instrumental assistance during the job market. I owe immeasurable thanks to him for the insightful advice and constant encouragement. Many thanks to all other mentors and friends who have given me helpful suggestions, including Joseph D. Alba, Chen Xiaoping, Fang Zheng, Feng Qu, Huang Weihong, Yohanes Eko Riyanto, Tang Yang, Laura Wu Guiying and Zheng Huanhuan.

I would like to thank the faculty members and administrative staffs of the Division of Mathematical Sciences and the Division of Economics at Nanyang Technological University, for providing all kinds of assistance during my Ph.D. journey.

I would also like to acknowledge the Nanyang President's Graduate Scholarship for financial support, especially the generous conference funding. I have benefited immensely from each experience. During the conferences, Professors Alvin E. Roth, Tayfun Sönmez, William Thomson and Utku Ünver provided helpful discussions and constructive feedback, which have inspired me a lot to improve my work.

Last but not least, I would like to thank my parents and brother for supporting me in this endeavor. Thank you for loving me and providing me with the unwavering confidence that I would succeed.

Contents

Acknowledgements	i
Summary	vii
1 Introduction	1
1.1 Market Design and Matching	1
1.2 Marriage Problem	3
1.3 Variants of Marriage Problem	7
1.3.1 Multi-unit Capacities	7
1.3.2 Weak Preferences	9
1.4 Roommates Problem	12
1.5 Thesis Outline	14
2 Two-Sided Many-to-Many Matching with Ties	17
2.1 Introduction	17
2.2 Model	20
2.3 Solution Concepts	22
2.3.1 Pairwise Stability	23
2.3.2 Pareto Efficiency	24
2.3.3 Pareto Stability	25
2.4 Computing a Pareto Stable Matching	26
2.4.1 Characterization of Pareto Efficiency	27
2.4.2 A Pareto Stable Matching Algorithm	29
2.4.2.1 Capacity Increment	32
2.4.2.2 Augmenting Path Elimination	33

2.4.3	Analysis of the Algorithm	36
2.4.4	Multiple Preferences	37
2.5	Conclusion	38
2.6	Appendix	39
2.6.1	Responsive Preference with Lattice Structure	39
2.6.2	Proofs	40
3	Two Competing Pareto Stable Matching Mechanisms	49
3.1	Introduction	49
3.2	Model	51
3.3	Solution Concepts	52
3.3.1	One-Sided Pareto Efficiency	53
3.3.2	One-Sided Group Stability	53
3.4	Pareto Stable Matching Mechanisms	56
3.4.1	Pareto-Improving Draft Mechanism	57
3.4.2	Pareto-Improving Dictatorship Mechanism	58
3.4.3	Draft versus Dictatorship	61
3.4.4	Some Remarks on Strategyproofness in Large	62
3.5	Conclusion	63
3.6	Appendix	64
3.6.1	Proofs	64
4	Improving Efficiency in Course Allocation	69
4.1	Introduction	69
4.2	Course Allocation Problem	74
4.3	NTU's Course Allocation Mechanism	77
4.3.1	Inefficiency of the Mechanism	78
4.4	A New Design	80
4.5	Simulations	81
4.5.1	Data and Key Assumptions	81
4.5.2	Simulation Environment I	84
4.5.3	Simulation Environment II	88

4.6	Conclusion	92
4.7	Appendix	93
4.7.1	NTU Curriculum Structure and Course Registration Process .	93
4.7.2	Survey Questions and Results	95
5	Pareto Efficiency in Generalized Roommates Problem	99
5.1	Introduction	99
5.2	Problem Definition and Solution Concept	101
5.3	Pareto Efficient Assignments	104
5.4	Strategic Implications	109
5.5	Conclusion	111
6	Discussion and Future Research	113
6.1	Conclusion and Implications	113
6.2	Future Research	115
	Bibliography	117

Summary

This thesis studies two types of matching markets: the two-sided matching markets where there is a need to match large two-sided populations of agents to one another based on their preferences; and the one-sided matching markets in which everyone can potentially be matched with anyone else. A notable feature in these markets is that price does not do all of the work. The main interest is on the efficiency and fairness of a matching outcome.

We first study a generalized many-to-many matching problem with ties. The complications in such a problem are due to multi-unit capacities and weak preferences, either of which can make a stable matching outcome not necessarily Pareto efficient. A natural solution concept is Pareto stability, which ensures both stability and Pareto efficiency. We show that a Pareto stable matching always exists by developing an efficient algorithm to compute one.

Next, for a more practical matching market design problem where one side of the market has homogeneous preferences, for instance, course allocation, we propose two new competing Pareto stable matching mechanisms known as the Pareto-improving draft and dictatorship mechanisms. These two mechanisms, while both have desirable properties in terms of efficiency and fairness, bring about the tradeoff between strategyproofness and non-callousness.

We then propose a new design to the student-course matching system at Nanyang Technological University (NTU), which is a real-life many-to-many matching market. Using unique course matching data from NTU, our simulations show that the new design with the Pareto stable matching mechanisms can significantly improve the overall efficiency and welfare of the students when compared to the existing mechanism. Furthermore, the Pareto-improving draft mechanism outperforms the Pareto-improving dictatorship mechanism despite it is non-strategyproof for the students.

Finally, we consider the generalized roommates problem with N students to be assigned to M rooms. Students may have weak preferences and rooms can have different capacities. Unlike the two-sided matching problems, multilateral agreement is required to dissolve a *status quo* assignment in the roommates problem. Therefore,

the notion of stability is too restrictive and a more relevant solution concept is Pareto efficiency. We show that a Pareto efficient assignment always exists by introducing efficient algorithms to compute such assignments. Strategic considerations suggest that the proposed Pareto-improving roommate swap algorithm is not strategyproof for the students.

List of Figures

2.1	Augmenting Path and Cycle	28
2.2	Lattice Structure of Responsive Preference.	39
4.1	Survey Results for 1,200 Students at NTU.	72
4.2	Distribution of the lengths of students' PE and UE Preference Lists. .	83
4.3	Summary of Simulation Results: Total Number of Allocations	92
4.4	Summary of Simulation Results: Percentage of Unassigned Students .	93
4.5	Curriculum Structure at NTU.	97

List of Tables

4.1	Course Registration Statistics.	79
4.2	Simulation Results I: Total Number of Allocations	85
4.3	Simulation Results I: Number of Unassigned Students for PE+UE . .	88
4.4	Simulation Results II: Total Number of Allocations	90
4.5	Simulation Results II: Average Rank	91
4.6	Simulation Results II: Number of Unassigned Students for PE+UE .	91

Chapter 1

Introduction

1.1 Market Design and Matching

The decreasingly self-regulating nature of various marketplaces has opened increasingly many opportunities for the deliberate design of markets. Market design examines the reasons why markets or institutions fail and considers the properties of alternative mechanisms in terms of efficiency, fairness, incentives and complexity. It combines insights from economic and game theory together with lessons learned from empirical work and experimental analysis to study how markets could be created and designed to achieve social goals. This field has recently received increasing attention, following the successful implementations of spectrum auctions (McMillan, 1994; Bulow et al., 2009; Cramton, 2013), clearinghouses matching kidney donors and recipients (Roth et al., 2004, 2005, 2007; Ashlagi and Roth, 2012), as well as allocation systems assigning students to schools (Abdulkadiroğlu and Sönmez, 2003; Abdulkadiroğlu et al., 2005a,b, 2006, 2009).¹

A well-designed market mechanism can attract sufficiently many participants and make it strategically simple for them, which leads to a socially desirable outcome where resources are efficiently allocated to the parties that value them the most while there is no incentive to transact outside the marketplace. Market design brings together theoretical, empirical and experimental methods to study policy-relevant

¹See Vulkan et al. (2013) for many successful market design cases, including matching markets as well as other types of markets such as auctions, e-commerce and law design.

tradeoffs with practical consequences on the design of market institutions.

In this thesis, we study two types of matching markets: the two-sided matching markets where there is a need to match large two-sided populations of agents to one another, such as medical residents and hospitals, patients and kidney donors, students and schools; and the one-sided matching markets in which everyone can potentially be matched with anyone else, such as pairing police officers on patrols, pilots on flights (Cechlarova and Ferkova, 2004), students to share double rooms in colleges. In the two-sided matching markets, there are two disjoint sets of agents and each agent has a preference ranking over potential partners on the other side of the market. There is a maximum number of partners that an agent can be matched with, which is known as his/her capacity. Given these data as input, consisting of two sets of preference orderings and capacities of each agent, we want to determine a satisfactory matching between these two sides of the market with some agreed-upon criterion.² The one-sided matching markets are similar to the two-sided matching markets except that agents are no longer distinguished into two disjoint sets.

In contrast to the commodity markets, a notable feature in the matching marketplaces is that price does not do all of the work. Matching is one of the most important functions of various marketplaces — how to arrange the donor-patient kidney exchanges, who gets into which school, and so on, all these help to shape lives, careers and future. The challenge for a market designer is therefore to design matching mechanisms that can achieve socially desirable matching outcome when an equilibrium price is no longer a market clearing solution.

The catalyst for recent success of matching theory has been the seminal work of Gale and Shapley (1962), which introduces the two-sided matching model as a marriage problem and the one-sided matching model as a roommates problem. Since then a growing body of literature began to explore the computational issues, mathematical properties, economic properties and strategic issues in various matching models and their variants. Gale and Shapley (1962) propose the most fundamental solution concept in the matching markets known as stability. A stable matching is a

²Roth and Sotomayor (1990) give a comprehensive game-theoretic analysis on various two-sided matching market models.

set of pairs of acceptable agents such that no two agents could improve their assignments by becoming matched to each other. Empirical evidence has shown that if the arranged matching is not appropriately stable, then the market will unravel and the mechanism is most likely to fail, leading to very inefficient social outcome (Roth, 2002).

Since the discrete marriage game proposed by Gale and Shapley (1962), the theory of stable matching has been growing rapidly, with especially wide, practical and fruitful applications in two-sided contexts such as the U.S. hospital-intern market³ (Roth, 1984b, 1986), the school-student matching (Abdulkadiroğlu and Sönmez, 2003) and the kidney exchange programs in the U.S. and Europe (Roth et al., 2004, 2005, 2007). This trend is continuing further recently with new applications in (often large scale) resource allocation problems of social importance, such as college admissions (Chen and Kesten, 2013), course allocations (Budish and Cantillon, 2012; Budish and Kessler, 2014) and on-campus housing allocations (Guillen and Kesten, 2012).

In this chapter, we first give an overview of the marriage problem in Section 1.2. Then we review various variants of the marriage problem in Section 1.3. The one-sided matching problem, that is, the roommates problem is introduced in Section 1.4. We conclude this chapter with an outline of the thesis in Section 1.5.

1.2 Marriage Problem

The classical two-sided matching model, also known as the **marriage problem**, proposed in Gale and Shapley (1962) involves two disjoint sets of agents, referred to as men $M = \{m_1, m_2, \dots, m_p\}$ and women $W = \{w_1, w_2, \dots, w_q\}$, each of whom ranks a subset of agents from the other set in a strict order, known as their **preferences** and denoted as $\succ_m$ or $\succ_w$. There are no ties in the preferences. If an agent is indifferent between two or more potential partners, he/she is nevertheless required to list them in some order. For example, a man m 's preference over five women: $w_2 \succ_m w_1 \succ_m w_4 \succ_m \emptyset \succ_m w_5 \succ_m w_3$ indicates that he likes w_2 the most, followed

³Also known as the National Resident Matching Program (NRMP).

by w_1 and then w_4 . All other women w_3, w_5 are unacceptable to him, that is, he would rather to stay single, represented by $\emptyset$, than be married to any of these women.⁴ For simplicity, we usually omit the $\emptyset$ and the women followed by it in a preference list. That is, usually only acceptable partners are included in a preference list. Women's preferences are defined similarly. These preferences can be considered as ordinal utility of the agents in marriage. Each agent can be matched to at most one agent from the other side. Formally, each agent has a **capacity**, also known as quota, of one, denoted as $q_m = q_w = 1$. Therefore, the marriage problem is also referred to as **one-to-one matching with strict preferences** and represented by a tuple $(M, W, \succ_m, \succ_w)$.

Given an instance $(M, W, \succ_m, \succ_w)$, a **matching** is a one-to-one function, denoted by $\mu : M \cup W \rightarrow M \cup W$, where $\mu(m) \in W$ is the woman matched to $m \in M$ and $\mu(w) \in M$ is the man matched to $w \in W$, and $\mu(m) = w$ if and only if $\mu(w) = m$. We use $\mu(m) = \emptyset$ to denote that m is not matched to anyone, and similarly for $\mu(w) = \emptyset$.

What is a desirable matching outcome? Gale and Shapley (1962) propose the solution concept of stability for a matching outcome. A **stable** matching has the property that no two agents would both prefer to match with each other as compared to their respective current partners. It ensures fairness and equity in a matching outcome by taking the preferences of both sides into account. Formally, the definition of a stable matching is given as following.

Definition 1.2.1 (Stable Matching). *Given a matching μ for an instance $(M, W, \succ_m, \succ_w)$, a man-woman pair (m, w) is a blocking pair if $w \succ_m \mu(m)$ and $m \succ_w \mu(w)$. A matching μ is called stable if it has no blocking pair.*

For any marriage problem, can we always achieve a stable matching? A stable matching always exists and can be computed with a simple iterative algorithm known as the **Deferred Acceptance Algorithm** which was first proposed by Gale and Shapley (1962). Due to the symmetry of the problem, the algorithm can start from either the men or the women side. The Men-Proposing Deferred Acceptance

⁴Sometimes m is used instead of $\emptyset$ to represent the option of staying single, i.e., being unmatched.

Algorithm is as following.

MEN-PROPOSING DEFERRED ACCEPTANCE ALGORITHM

- Initially all men and women are free
- While there is a man m who is free and hasn't proposed to every woman
 - choose such a man m
 - let w be the highest ranked woman in m 's preference list to whom m hasn't proposed yet
 - m proposes to w
 - if w is free, then (m, w) become engaged
 - else, w is currently engaged to m'
 - * if w prefers m' to m , then m remains free
 - * if w prefers m to m' , then (m, w) become engaged and m' becomes free

In the algorithm, all the men make proposals to the women according to their preferences. For those women with more than one proposal, they reject the least preferred men. They do not accept those men that they have not yet rejected immediately. Instead, they continue to hold these men without any commitment. For those men rejected, they make new proposals to the women who have not rejected them. This results in new rejections until no rejected men wish to continue making further proposals. At this stage, all proposals being held are finally accepted to produce a matching. Similarly, we can reverse the roles of the men and women which gives the Women-Proposing Deferred Acceptance Algorithm. Note that in the implementation of the algorithm, all acceptances are deferred until the end.

Theorem 1.2.2 (Gale and Shapley, 1962). *The Deferred Acceptance Algorithm returns a stable matching.*

Proof. We first show that the algorithm returns a matching in finite steps. In the execution of the algorithm, w remains engaged from the point at which she receives her first proposal. If m is free at some point in the execution of the algorithm, then

there is a woman to whom m hasn't proposed yet. Each man and women is only engaged to at most one partner throughout the algorithm. There is a finite number of men and women. Therefore, the algorithms terminates in finite steps and returns a matching.

Next, we show that the returned matching is stable. Let μ be a matching returned by the algorithm. Assume that (m, w) is a blocking pair, where $(m, w'), (m', w) \in \mu$. That is, $w \succ_m w'$ and $m \succ_w m'$. In the algorithm, m 's last proposal was to w' by definition. Then we consider the following two cases:

- Case 1: m has proposed to w . Since the sequence of partners of w only increases, w will be matched to a man better than m . A contradiction.
- Case 2: m has not proposed to w . By the algorithm, m should propose to w before w' . A contradiction.

□

Apart from the well-established stability solution concept, another important solution concept is called Pareto efficiency (or Pareto optimality) which guarantees the overall efficiency of a matching outcome. In a **Pareto efficient** matching, there is no Pareto improvement in which at least one agent is strictly better off while no other agents are worse off. In other words, it is not possible to benefit anyone without hurting someone. The lack of this property is discouraging from a social planner's perspective. In the marriage model, a stable matching is also Pareto efficient. This result is intuitive: If otherwise such a Pareto improvement exists, then there should be at least a pair of man and woman better off by becoming matched with each other. The existence of such a pair of man and woman contradicts with the stability of the matching.

Theorem 1.2.3. *In the marriage game, any stable matching is Pareto efficient.*

Proof. Consider any stable matching μ . If μ is not Pareto efficient, then there is μ' such that no one gets worse off and at least one gets better off. Assume without loss of generality that m get better off. Denote $w = \mu(m)$ and $w' = \mu'(m)$, then m prefers w' to w . Assume $m' = \mu(w')$. Since no one gets worse off in μ' than μ , w' should prefer m to m' as well, which implies that (m, w') is a blocking pair for μ , a contradiction. □

1.3 Variants of Marriage Problem

The remarkable work of Gale and Shapley on the marriage problem has inspired extensive studies on its various variants, including complications and details that constantly arise in practice. Two most important variants include multi-unit capacities, that is, $q_m \geq 1$ and/or $q_w \geq 1$, and ties/indifferences in preferences (i.e., non-strict/weak preferences)⁵, denoted as $\sim_m$ and $\sim_w$. These generalizations give rise to more general problems which have recently seen many practical applications, such as school choice, online labor market, social lending, and college admission. In this section, we discuss separately how these two types of variants can affect the properties and computations of a stable matching outcome.

1.3.1 Multi-unit Capacities

When agents have multi-unit capacities, that is, they can be matched with multiple partners, we refer to it as **many-to-one matching** or **many-to-many matching**. A simple variant of the Deferred Acceptance Algorithm still computes a stable matching by making q_i copies of agent i with the same preference list, where q_i is the capacity of agent i .

We need to define preferences over subsets when agents can be matched with multiple partners. A natural way to extend from the preferences over individuals gives the responsive preferences.⁶ An agent's preferences are called **responsive** to his/her preferences over individuals if for any two matchings that differ in only one partner, he/she prefers the matching containing the more preferred partner. Formally, the definition is given as following.

Definition 1.3.1 (Responsive Preferences). *The preference over sets of agents is responsive (to the preference over individual agents) if, whenever $\mu'(m) = \mu(m) \cup \{w\} \setminus \{w'\}$ and $w \notin \mu(m)$, then $\mu'(m) \succ_m \mu(m)$ if and only if $w \succ_m w'$.*

⁵We will use these terms, “weak preferences”, “non-strict preferences”, “preferences with ties” and “preferences with indifferences”, interchangeably.

⁶There are other types of preferences over subsets, such as substitutable preferences (see for instance Roth (1984a)).

In the above definition, we can simply replace m by w and w by m to define responsive preferences for the women. With responsive preferences, a stable matching is still Pareto efficient in the many-to-one matching markets.

Theorem 1.3.2. *In any many-to-one matching market with strict preferences, a pairwise stable matching is always Pareto efficient.*

Proof. Consider a many-to-one matching example of matching students S to colleges C . Consider any stable matching μ . If μ is not Pareto optimal, then there is μ' such that no one gets worse off and at least one gets better off. Assume without loss of generality that s gets better off (such a student always exist since no student is worse off and students have strict preferences). s should be matched with some school c' in μ' , i.e., $s \in \mu'(c')$. Then we must have $|\mu(c')| = q_{c'}$, since otherwise (c', s) will be a blocking pair in μ . Hence there is $s' \in \mu(c')$ s.t. $s \succ_{c'} s'$. Consider the following two cases:

- Case 1: If $|\mu(s)| = 0$, then (c', s) is a blocking pair in μ . This contradicts the assumption that μ is pairwise stable.
- Case 2: If $s \in \mu(c)$, then we have $c' \succ_s c$ and (c', s) is a blocking pair in μ . This again contradicts the assumption that μ is pairwise stable.

Therefore, in many-to-one matching markets with strict preferences, a pairwise stable matching is always Pareto efficient. $\square$

However, this is no longer true for the many-to-many matching markets as illustrated in the following example adapted from Roth and Sotomayor (1990).

Example 1.3.3 (Stability no longer implies Pareto efficiency in many-to-many matching markets.). *There are four men and four women, each with a capacity of two, with responsive preferences as follows:*

$$\begin{aligned}
 m_1 : \quad & \{w_1, w_2\} \succ \{w_1, w_3\} \succ \{w_1, w_4\} \succ \{w_2, w_3\} \succ \{w_2, w_4\} \succ \{w_3, w_4\} \\
 & \succ \{w_1\} \succ \{w_2\} \succ \{w_3\} \succ \{w_4\} \\
 m_2 : \quad & \{w_1, w_2\} \succ \{w_2, w_4\} \succ \{w_2, w_3\} \succ \{w_1, w_4\} \succ \{w_1, w_3\} \succ \{w_3, w_4\} \\
 & \succ \{w_2\} \succ \{w_1\} \succ \{w_4\} \succ \{w_3\}
 \end{aligned}$$

$$\begin{aligned}
m_3 : \quad & \{w_3, w_4\} \succ \{w_3, w_1\} \succ \{w_3, w_2\} \succ \{w_4, w_1\} \succ \{w_4, w_2\} \succ \{w_1, w_2\} \\
& \succ \{w_3\} \succ \{w_4\} \succ \{w_1\} \succ \{w_2\} \\
m_4 : \quad & \{w_4, w_3\} \succ \{w_4, w_2\} \succ \{w_4, w_1\} \succ \{w_3, w_2\} \succ \{w_3, w_1\} \succ \{w_2, w_1\} \\
& \succ \{w_4\} \succ \{w_3\} \succ \{w_2\} \succ \{w_1\} \\
w_1 : \quad & \{m_4, m_3\} \succ \{m_4, m_2\} \succ \{m_4, m_1\} \succ \{m_3, m_2\} \succ \{m_3, m_1\} \succ \{m_2, m_1\} \\
& \succ \{m_4\} \succ \{m_3\} \succ \{m_2\} \succ \{m_1\} \\
w_2 : \quad & \{m_3, m_4\} \succ \{m_3, m_1\} \succ \{m_3, m_2\} \succ \{m_4, m_1\} \succ \{m_4, m_2\} \succ \{m_1, m_2\} \\
& \succ \{m_3\} \succ \{m_4\} \succ \{m_1\} \succ \{m_2\} \\
w_3 : \quad & \{m_1, m_2\} \succ \{m_2, m_4\} \succ \{m_2, m_3\} \succ \{m_1, m_4\} \succ \{m_1, m_3\} \succ \{m_3, m_4\} \\
& \succ \{m_2\} \succ \{m_1\} \succ \{m_4\} \succ \{m_3\} \\
w_4 : \quad & \{m_1, m_2\} \succ \{m_1, m_3\} \succ \{m_1, m_4\} \succ \{m_2, m_3\} \succ \{m_2, m_4\} \succ \{m_3, m_4\} \\
& \succ \{m_1\} \succ \{m_2\} \succ \{m_3\} \succ \{m_4\}
\end{aligned}$$

Consider the following matching:

$$\mu = \begin{pmatrix} m_1 & m_2 & m_3 & m_4 \\ w_2, w_3 & w_1, w_4 & w_4, w_1 & w_3, w_2 \end{pmatrix}$$

μ is stable but not Pareto optimal, since it is dominated by the matching μ' that gives each agent his or her third choice set of agents.

$$\mu' = \begin{pmatrix} m_1 & m_2 & m_3 & m_4 \\ w_1, w_4 & w_2, w_3 & w_3, w_2 & w_4, w_1 \end{pmatrix}$$

1.3.2 Weak Preferences

In addition to the multi-unit capacities, another important variant which is often independently introduced is weak preferences. Weak preferences can arise in practice for various reasons, for instance, the other side of the market is too large such that it is very costly for agents to give a strict ranking of each potential partner.

In the one-to-one matching with strict preferences, the Pareto efficiency property is a direct implication from a stable solution. In implementing the Deferred Acceptance Algorithm, ties in preferences would create ambiguities in the choices of making proposals, rejections and acceptances. A common practice in these cases is

to break all the indifference classes according to an exogenously and randomly fixed strict ordering of the individuals for both sides. Then, one can apply the Deferred Acceptance Algorithm with respect to the strict preference profile derived from the original one. Since the breaking of indifferences does not switch the rankings of any two individual in separate indifference classes for any preference profile, the outcome is also stable with respect to the original preference structure. The stability property can therefore be preserved through this simple and intuitive tie-breaking procedure.

However, the practice of tie-breaking is not cost-free. With ties in preferences, stability no longer guarantees Pareto efficiency. Employing a random tie-breaking rule when faced with indifferences can cause a significant loss of efficiency by returning a matching that is not Pareto efficient with respect to the original preference structure, as illustrated in the following example.

Example 1.3.4 (Inefficiency from tie-breaking.). *There are two men m_1, m_2 and two women w_1, w_2 , each with unit capacity. Their preference profiles are given as:*

$$\begin{array}{ll} m_1 : w_1 & w_1 : m_1 \sim m_2 \\ m_2 : w_1 \sim w_2 & w_2 : m_2 \succ m_1 \end{array}$$

Consider a random tie-breaking that gives the preference of m_2 as: $w_1 \succ w_2$ and the preference of w_1 as: $m_2 \succ m_1$, then the only stable matching after tie-breaking is:

$$\mu = \begin{pmatrix} m_1 & m_2 & \emptyset \\ \emptyset & w_1 & w_2 \end{pmatrix}$$

However, this matching is not Pareto efficient with respect to the preferences before tie-breaking since the following matching is a Pareto improvement:

$$\mu' = \begin{pmatrix} m_1 & m_2 \\ w_1 & w_2 \end{pmatrix}$$

Comparing μ and μ' , a notable difference is that in μ there is only one matched pair while in μ' there are two matched pairs.

In the presence of ties and/or multi-unit capacities, to ensure both fairness and efficiency, a natural solution benchmark is **Pareto stability** which ensures both stability and Pareto efficiency. The Deferred Acceptance Algorithm with tie-breaking

cannot be directly applied to generate a Pareto stable matching, and alternative algorithms need to be developed. A critical characterization of Pareto efficiency, which is the key to construct Pareto stable mechanisms, is related to the non-existence of augmenting paths and cycles in the corresponding matching graph. For a given matching, an augmenting path or cycle can be easily found using a network flow approach. By reassigning matches according to the augmenting paths and cycles, we are essentially making Pareto improvement to the current matching.⁷

In a many-to-one matching with ties, we have another important observation which helps to construct a Pareto stable algorithm, that is, any Pareto improvement to a stable assignment can preserve stability. This immediately implies an algorithm to find a Pareto stable matching proposed by Erdil and Ergin (2008): starting from any stable assignment, keep making Pareto improvements by eliminating augmenting paths and cycles until none remain, and the resulting matching will be both stable and Pareto efficient. In applications to school-student matching, this algorithm can return a student optimal matching, which is not Pareto dominated by any other stable matching, that is, no student can improve his/her matching without getting any other student worse off. This result greatly improves the welfare of students compared to the results from Deferred Acceptance Algorithm with tie-breaking procedures (Abdulkadiroğlu et al., 2009).

In the more generalized many-to-many matching with ties, the same stability preserving result may not hold. If only one side of the market has ties, Pareto improvement can still preserve stability and therefore the above algorithm can generate a Pareto stable matching. However, when both sides of a market have ties, Pareto improvement to a stable assignment need not preserve stability and thus the above algorithm fails to work. This problem cannot be solved by a careful selection of Pareto improvements either. We study the existence and computation issues of Pareto stable matchings in this generalized problem in Chapter 2 and further look into more specific market design problems as well as applications in Chapter 3 and Chapter 4.

⁷We will discuss these in details and define them formally in the rest of the thesis, particularly in Chapters 2 and 3.

1.4 Roommates Problem

Besides the two-sided matching problem, another important type of matching problem is called the one-sided matching problem, also known as the “problem of the roommates”. It was first discussed in Gale and Shapley (1962) as an example. It is one of the oldest but least well understood problems in matching theory literature.

The classic roommates problem involves a set of students $S = \{s_1, s_2, \dots, s_n\}$ to be paired with each other based on their preferences over each other $\succ_s$. It asks the following question: “Is there a stable way to assign $2M$ students into M roommate pairs?”, where a set of assignments are called **stable** if there are no two students who are not roommates prefer each other to their actual roommates. Gale and Shapley (1962) show that while a stable matching always exists in a two-sided market, a stable pairing need not exist in a one-sided market like the roommates problem. The following example is adapted from Gale and Shapley (1962)’s original example.

Example 1.4.1 (A stable assignment may not exist in the roommates problem.). *Consider four students s_1, s_2, s_3 and s_4 , where s_1 ’s top choice is s_2 , s_2 ’s top choice is s_3 , s_3 ’s top choice is s_1 , and s_1, s_2 and s_3 all rank s_4 last. There are three possible assignments but none of them is stable.*

$$\mu_1 = \begin{pmatrix} s_1 & s_3 \\ s_2 & s_4 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} s_1 & s_2 \\ s_3 & s_4 \end{pmatrix} \quad \mu_3 = \begin{pmatrix} s_1 & s_2 \\ s_4 & s_3 \end{pmatrix}$$

In this setup, no matter who is assigned to s_4 , he/she prefers the other two students to s_4 and is the top choice of one of these two students. Hence, this person will want to move out and one of the other two will be willing to take him/her in.

The above negative result has resulted in a relative sparse economics literature on the classic roommates problem. Because of the possibility of non-existence of a stable matching, there have been little economics studies on finding and analyzing solutions of this important one-sided matching problem. Nevertheless, there is a relatively large computer science literature on it. From a computational perspective, Irving (1985) and Gusfield and Irving (1989) present a polynomial-time algorithm which finds a stable roommates assignment when one exists. Later, a necessary and sufficient condition for the existence of a stable assignment is established by Tan

(1991). Taking a more general perspective, Irving and Manlove (2002) relax the assumption of strict preferences and study the stable roommates problem with ties.

Among the small economics literature, Roth and Sotomayor (1990) mention the roommates problem only as an example. Chung (2000) extends Tan (1991)'s results to a sufficient condition for the existence of a stable assignment when preferences are weak. Most recently, Morrill (2010) approaches the classic roommates problem with a different solution concept. He argues that the traditional notion of stability is too restrictive in the sense that an agent cannot unilaterally dissolve his/her partnership in the roommates problem. Unlike the marriage problem, roommates face an additional physical constraint that they must have a room to live in. Two students may prefer each other to their current assignments, but neither of them has the right to evict their current roommates. In this case, bilateral agreement is required to dissolve. Since an agent will only get involved in a change only if the new assignment does not make him/her worse off, any deviation from the original set of assignments must be a **Pareto improvement**. In Example 1.4.1, while one of the other two will be willing to take him/her in, this student is not able to take him/her in since no one will want to change his/her assignment to s_4 . Therefore, Morrill (2010) suggests that **Pareto efficiency**⁸ instead of stability is the natural equilibrium solution concept for the roommates problem. When an assignment is Pareto optimal, no agent can be better off without hurting anyone else and hence the existing assignment will not be disturbed.

Similar to the two-sided matching problem, variants of the roommates problem include having more than two agents to share a room and introducing ties in preferences. These variants however have rarely been studied in the literature. We study Pareto efficiency in more generalized roommates problem with these variants in Chapter 5 of this thesis.

⁸We will use “Pareto efficiency” and “Pareto optimality” interchangeably.

1.5 Thesis Outline

This thesis studies two types of matching problems built upon the marriage problem and the roommates problem respectively: the many-to-many matching with ties and the generalized roommates problem with multi-unit capacities as well as weak preferences, both of which are generalizations of the previously studied matching problems in literature. For the former problem, the main question is about the existence and computation of a Pareto stable matching. We also study the computation issues and applications of a practical matching problem with one-sided homogenous preferences. For the latter, the main question is about the existence and computation of a Pareto efficient roommates assignment.

The main contributions of this thesis are threefold. First, we show that a Pareto stable matching exists for every many-to-many matching market with ties by developing an efficient algorithm to find such a matching. Second, for many-to-many matching markets with one-sided homogeneous preferences, we propose two competing Pareto stable matching mechanisms and further test them in a practical application of course allocation with simulations, which quantitatively demonstrates the efficiency loss of existing mechanisms. Third, in the generalized roommates problem we show that a Pareto efficient assignment always exists and can be computed efficiently.

The rest of this thesis is organized as follows.

In Chapter 2, we propose and study the many-to-many matching problem with ties, where agents can be matched with multiple partners and their preferences are not necessarily strict. We adapt the solution concept of Pareto stability, which requires both stability and Pareto efficiency, to ensure fairness and efficiency in a matching outcome. We show that a Pareto stable matching always exists by developing a polynomial-time algorithm that computes such a matching.

In Chapter 3, we examine a related practical many-to-many matching market design problem where one side of the market has homogeneous preferences, which captures many real life assignment problems such as the university student-course matching market. Our novel contributions to the practical market design literature include two competing Pareto stable matching mechanisms. We examine various

properties of these two mechanisms including group stability and strategyproofness.

In Chapter 4, we propose a new design to the course allocation system at Nanyang Technological University (NTU) using the theories developed in the previous two chapters. With unique course matching data, we quantify and compare the welfare and efficiency gains from our proposed mechanisms compared to the existing mechanism.

In Chapter 5, we study the generalized roommates problem with N students to be assigned to M rooms and rooms having different capacities. This generalized problem better captures many real life one-sided matching problems, for instance finding roommates with different room sizes, and grouping project teams with a relatively flexible team size. When agents have strict preferences, we show that a Pareto efficient assignment can be computed from the Random Serial Dictatorship algorithm. When ties are introduced, we develop a polynomial-time efficient algorithm to compute a Pareto efficient assignment.

In Chapter 6, we conclude this thesis with a discussion on the main findings and implications. Some future research directions are also discussed.

Chapter 2

Two-Sided Many-to-Many Matching with Ties

2.1 Introduction

In a two-sided many-to-many matching problem, two disjoint sets of agents are to be matched with each other, where agents have multi-unit capacities and preferences over the possible partners on the other side. Practical examples of two-sided many-to-many matching markets are pervasive, such as peer-to-peer social lending (Chen and Ghosh, 2011), assigning shared scientific resources to scientists (Wulf, 1993), assigning players to sports teams (Albergotti, 2010), part-time job-worker matching¹ and university course-student matching (Budish and Cantillon, 2012; Budish and Kessler, 2014). Many-to-many matching problems are generalizations of the multi-unit assignment problems (Budish and Cantillon, 2012; Kojima, 2013) in which preferences are one-sided.

Compared with the one-to-one and many-to-one matching problems, the generalization to multi-unit capacities on both sides of the market is nontrivial as the properties and structures of many-to-many matchings behave rather differently. The presence of a few many-to-many capacities may change the matching outcome for all agents completely (Echenique and Oviedo, 2006). Under responsive preferences, a

¹An example is Matchimi (<http://matchimi.com/>), a Singapore-based start-up company that automatically matches part-time jobs to job seekers according to their preferences.

pairwise stable matching need not be group stable and need not even be Pareto efficient (Roth and Sotomayor, 1990). While the many-to-one matching problems have been extensively studied and successfully applied in the market design literature, both theoretical and empirical works on the many-to-many matching market design have remained elusive. It is therefore important to study the generalized many-to-many matching problems to better understand and design these marketplaces.

The literature on two-sided matching problems mostly assumes strict preferences. For example, in a many-to-many matching problem with strict preferences, Roth (1984a) shows that the set of pairwise stable matchings is nonempty with substitutable preferences, and that one-sided optimal stable matchings exist. Blair (1988) proves that the set of pairwise stable matchings forms a lattice structure. Its properties have been later investigated by Alkan (2001) and Alkan (2002). Martinez et al. (2004) presents an algorithm that finds all pairwise stable matchings. Sotomayor (2011) presents examples in which the set of group stable matchings is empty. Echenique and Oviedo (2006) later give conditions under which the group stable matching set is nonempty and can be approached through an algorithm. The main reason underlying this assumption of strict preferences is that the introduction of weak preferences, also known as preferences with ties or indifferences, dramatically changes the properties and structures of the set of stable matchings. It consequently leads to a series of negative results. For instance, men or women-optimal stable matchings are no longer well-defined (Roth and Sotomayor, 1990), and stable matchings need not all have the same cardinality. In addition, arguably more importantly, stability no longer guarantees Pareto efficiency (Abdulkadiroğlu and Sönmez, 2003; Sotomayor, 1999; Erdil and Ergin, 2006, 2008; Abdulkadiroğlu et al., 2009).² The computer science literature includes results showing that finding a maximum cardinality stable matching becomes computationally hard (Iwama et al., 1999; Manlove et al., 2002), and much work has focused on finding approximation solutions (Iwama et al., 2007; McDermid, 2009). While the assumption of strict preferences is key to

²For example, there are two students s_1, s_2 and two courses c_1, c_2 , where s_1 strictly prefers c_1 to c_2 , but all others are indifferent amongst their possible partners. The matching $(s_1, c_2), (s_2, c_1)$ is stable, but not Pareto efficient since s_1 can be reassigned to c_1 and s_2 to c_2 without making anyone worse off.

maintain many theoretically important and attractive properties of the set of stable matchings, there are various practical matching markets in which agents are not able to strictly rank their prospective partners for a variety of reasons. For instance, it may be costly or even unrealistic to have strict preferences when the size of the other side of the market is extremely large or when agents have incomplete information about the other side of the market. Therefore, practical market designers should take this “detail” of the market³, that is, weak preferences, into consideration when designing matching market mechanisms.

The generalization to multi-unit capacities on both sides and the market design “detail” of weak preferences together have been overlooked in both the theoretical two-sided matching literature and the empirical matching market design literature. This chapter fills in the gap by addressing both of them in a single generalized framework — **many-to-many matching with ties** — from both theory and practice. It provides an efficient algorithm to compute a desirable matching outcome in such generalized frameworks.

Before designing the algorithm, we examine which solutions are desirable in a many-to-many matching problem with ties. Pairwise stability as a well-established solution concept in two-sided matching problems captures fairness and equity by taking preferences of both sides into account. When ties are introduced, it is easier to achieve pairwise stability since the set of possible blocking pairs is smaller. However, a pairwise stable matching is not necessarily a desirable solution any more because of it may sacrifice efficiency. As a consequence, on top of stability, a desirable matching outcome should be Pareto efficient, which to some extent qualifies the overall efficiency of a matching. Pairwise stability and Pareto efficiency, together called **Pareto stability** as suggested by Sotomayor (2011), provide a natural solution benchmark for matching markets in the presence of ties.

The next natural question is whether a Pareto stable matching always exists, and if it exists, how to find one efficiently. This question has been studied in Erdil and Ergin (2006, 2008) for the many-to-one matching problem with ties. They pro-

³Roth and Peranson (1999) first bring up the importance of “details” in market design as compared to pure theory work. They put it as “Perhaps the first rule of any design effort is that ‘details matter’.”

pose a stability-preserving Pareto-improving algorithm by eliminating augmenting paths and cycles upon a stable matching. However, this algorithm fails to work in many-to-many matching frameworks, in which Pareto improvement does not necessarily preserve stability. New algorithms therefore need to be designed for the more generalized many-to-many matching problem with ties. Based on the idea of Roth and Vate (1990), we construct an efficient algorithm that computes a Pareto stable many-to-many matching in the presence of ties. Our result immediately implies the existence of such a matching. Our algorithm improves the results of Erdil and Ergin (2006, 2008) for many-to-one matching, and is fundamentally different from their algorithm.

The remainder of this chapter is organized as follows. Section 2.2 formally defines the generalized many-to-many matching model with ties. Section 2.3 discusses the solution concepts. A technical algorithm that computes a Pareto stable matching in general frameworks is presented in Section 2.4. We conclude in Section 2.5.

2.2 Model

We take the student-course matching as a practical example for the many-to-many matching model, and use the terms “students” and “courses” to describe the model. Our model and technical results work for generic many-to-many matching markets besides the student-course matching.

We consider a two-sided many-to-many matching market with a set of students $\mathcal{S}$ and a set of courses $\mathcal{C}$. Throughout, we will use $s \in \mathcal{S}$ to denote a student, $c \in \mathcal{C}$ to denote a course and $x, y, z \in \mathcal{S} \cup \mathcal{C}$ to denote any individual agent (a student or a course). Each agent $x \in \mathcal{S} \cup \mathcal{C}$ has a **capacity** of $q_x \in \mathbb{N}$, which is the maximum number of agents on the opposite side that can be matched to x . In other words, q_s denotes the maximum number of courses that s can take and q_c is the maximum number of students that c can enroll. The presence of capacities allows us to assume, without loss of generality, that $|\mathcal{S}| = |\mathcal{C}| = n$, as dummy agents with $q_x = 0$ can be added to the market.

Each student $s \in \mathcal{S}$ has a preference ranking $\succsim_s$ over individual courses and

remaining unmatched, where $c_1 \succ c_2$ means that s strictly prefers c_1 to c_2 , and $c_1 \sim c_2$ means that s is indifferent between them.⁴ We use $\sim$ to represent **ties** in a preference relation. We say s **weakly prefers** c_1 to c_2 if either $c_1 \succ c_2$ or $c_1 \sim c_2$, and denote it as $c_1 \succeq c_2$. Let P_s denote the set of **acceptable** courses to s , which are the courses more preferred over remaining unmatched. Every two courses in P_s are comparable and the preference is assumed to be transitive. The preference relation $\succeq_s$ and the set of acceptable courses P_s give a partial preference list of s . In other words, s does not want to be matched with any course that is not on the list P_s . For example, a possible preference list for s is $\succeq_s$: $c_1 \sim c_2 \succ c_3 \sim c_5$. Here, s is indifferent between c_1 and c_2 , prefers either of them to c_3 and c_5 , amongst which he/she is indifferent, and finds all other courses unacceptable. We use $\succeq_S = (\succeq_s)_{s \in \mathcal{S}}$ to denote the **preference profile** for students over individual courses. The preference profile for courses over individual students $\succeq_C = (\succeq_c)_{c \in \mathcal{C}}$ is defined similarly. We use $\mathcal{E} = \{(s, c) \mid s \in P_c, c \in P_s\}$ to denote the set of mutually acceptable pairs. A two-sided many-to-many matching market with ties can then be represented by a tuple $(\mathcal{S}, \mathcal{C}, (q_s)_{s \in \mathcal{S}}, (q_c)_{c \in \mathcal{C}}, \succeq_S, \succeq_C)$, where every agent has an integral capacity and a (possibly nonstrict) preference list over agents on the other side of the market.

The preference relations $\succeq_s$ and $\succeq_c$ defined above are over individuals. Because agents can have capacities greater than one, we need to extend the preference relations over subsets. We assume that preferences over subsets are **responsive with a lattice structure**. That is, given a subset $C \subseteq \mathcal{C}$ and two courses $c, c' \notin C$, $C \cup \{c\} \succeq_s C \cup \{c'\}$ if and only if $c \succeq_s c'$.⁵ In other words, for any two sets that differ in only one course, s prefers the one that contains the more preferred course and is indifferent between them if he is indifferent between the two courses. Responsiveness is a natural way of relating preferences over subsets to preferences over individuals and it is compatible with mild complementarities and substitutabilities. The preferences of courses over subsets of students are defined similarly. In addi-

⁴More precisely, there is a subscription s in the notations, i.e., $\succ_s$ and $\sim_s$. For simplicity we omit the subscription whenever it is clear from the context.

⁵Note that it is allowed c or $c' \sim \emptyset$. In particular, $\emptyset \succ_s c$ for any $c \notin P_s$, $c \succ \emptyset$ for any $c \in P_s$ and $c \sim c'$ for any $c, c' \notin P_s$.

tion, the responsive preference is transitive, that is, if $C_1 \succ_s C_2$ and $C_2 \succ_s C_3$, then $C_1 \succ_s C_3$. Note that this preference for subsets only constitutes a partial order, and more precisely, a complete distributive lattice. That is, two alternatives are comparable only if they are an ancestor-descendant relation in the lattice.⁶ This model of preferences with multi-unit capacities has been used in Erdil and Ergin (2006). It is simple, because agents only need to express preferences for individuals, and is arguably natural for settings in which the benefit from a partner to an agent does not depend on the agent's remaining partners.

Given the preferences and capacities of all the agents, our objective is to establish a multi-unit pairing between the students and the courses, called a **many-to-many matching** (or an *assignment*, or an *allocation*). A **matching** is the outcome of a market and is denoted by $\mu = (\mu_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$, where $\mu_{sc} = 1$ means that s and c are matched and $\mu_{sc} = 0$ otherwise. For any $x \in \mathcal{S} \cup \mathcal{C}$, we use $\mu(x)$ to denote the set of individuals matched to x in the matching μ . A (direct) **mechanism** is a systematic procedure, possibly with randomness, that selects a matching for each market. A **feasible matching** is one that satisfies the following conditions: $\sum_c \mu_{sc} \leq q_s$ and $\sum_s \mu_{sc} \leq q_c$, and $\mu_{sc} = 1$ only if $(s, c) \in \mathcal{E}$, that is, s and c are mutually acceptable. All matchings considered in this chapter are feasible.

2.3 Solution Concepts

Given the two-sided many-to-many matching market with ties defined above, the next question is which matching is desirable. While it is not possible to satisfy the demand of all agents, we hope to provide a fair and efficient solution that is in favor of most of the agents. In this section, we examine several solution concepts from different perspectives in terms of efficiency and fairness.

⁶The responsive preference defined in Roth and Sotomayor (1990) is a complete ranking for all compatible subsets, unlike ours, which forms a lattice structure. In Appendix 2.6.1, we give an example of responsive preference with a lattice structure.

2.3.1 Pairwise Stability

In many two-sided matching models, for instance, student placement and school choice (Sönmez and Ünver, 2011), one desirable property of a matching is called **elimination of justified envy**. That is, whenever a student s prefers the allocated course c of another student s' , s should not rank higher than s' in the preference list of c . Consider the following example.

Example 2.3.1. *There are two students $\mathcal{S} = \{s_1, s_2\}$ and two courses $\mathcal{C} = \{c_1, c_2\}$ each with $q_x = 1$ for all $x \in \mathcal{S} \cup \mathcal{C}$. Their preference profiles are given as:*

$$\begin{array}{ll} s_1 : c_1 \succ c_2 & c_1 : s_2 \succ s_1 \\ s_2 : c_1 \succ c_2 & c_2 : s_2 \succ s_1 \end{array}$$

There are two possible matchings:

$$\mu_1 = \begin{pmatrix} s_1 & s_2 \\ c_1 & c_2 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} s_1 & s_2 \\ c_2 & c_1 \end{pmatrix}$$

In μ_1 , s_2 is not assigned to his first choice c_1 , which prefers her over s_1 . In such a case, s_2 can appeal about the outcome:

“I am more eligible for the course c_1 . Why was I not assigned to c_1 ?”

Hence, a fairer outcome is to match c_1 with s_2 as in μ_2 . Now s_1 is not assigned to her first choice c_1 , but she can reason as follows:

“I did not get the course, but all those who got it are (more) eligible.”

Equivalently, if a course (c_2) has not been assigned to a student (s_2) who has a higher priority, then the student should be assigned to another course that she prefers more.

The above example illustrates a solution concept called (pairwise) stability, which was first proposed by Gale and Shapley (1962) in the contexts of marriage problem and college admission. It is a baseline solution in two-sided matching markets. Theoretically, if the market outcome is unstable, there is an agent or pair of agents who have the incentive to circumvent the matching. Empirically, Roth (2002) summarizes evidence in medical markets and health markets showing that stable mechanisms have mostly succeeded while unstable mechanisms have mostly failed. Therefore,

producing a stable matching is an important criterion for a successful clearinghouse. The formal definition of stability in a many-to-many matching market with ties is given as follows.

Definition 2.3.2 (Pairwise Stability). *Fix a market $(\mathcal{S}, \mathcal{C}, (q_s)_{s \in \mathcal{S}}, (q_c)_{c \in \mathcal{C}}, \succsim_{\mathcal{S}}, \succsim_{\mathcal{C}})$. We say that a feasible matching $\mu = (\mu_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$ is (pairwise) stable if there is no blocking pair. A blocking pair is a mutually acceptable pair $(s, c) \in \mathcal{E}$ with $\mu_{sc} = 0$ that satisfies one of the following conditions:*

- $\sum_c \mu_{sc} < q_s$ and $\sum_s \mu_{sc} < q_c$;
- $\sum_c \mu_{sc} < q_s$ and $\exists s'$ with $\mu_{s'c} = 1$, such that $s \succ_c s'$;
- $\sum_s \mu_{sc} < q_c$ and $\exists c'$ with $\mu_{sc'} = 1$, such that $c \succ_s c'$;
- $\exists s'$ and c' with $\mu_{sc'} = 1$ and $\mu_{s'c} = 1$, such that $c \succ_s c'$ and $s \succ_c s'$.

Note that both members of a blocking pair are able to strictly improve their assignments respectively by matching with each other (and possibly breaking some of the current assignments). A stable matching always exists, and can be found using a variant of the Deferred Acceptance Algorithm (Gale and Shapley, 1962), by making q_x copies for each individual $x \in \mathcal{S} \cup \mathcal{C}$ with the same preference lists and breaking ties randomly.

2.3.2 Pareto Efficiency

Besides fairness, another important aspect in a desirable solution is social welfare, which measures the overall efficiency of a matching outcome. Consider the following two examples.

Example 2.3.3. *There are two students $\mathcal{S} = \{s_1, s_2\}$ and two courses $\mathcal{C} = \{c_1, c_2\}$ each with $q_x = 1$ for all $x \in \mathcal{S} \cup \mathcal{C}$. Their preference profiles are given as:*

$$\begin{array}{ll} s_1 : c_1 & c_1 : s_1 \sim s_2 \\ s_2 : c_1 \sim c_2 & c_2 : s_1 \sim s_2 \end{array}$$

Consider the following two matchings:

$$\mu_1 = \begin{pmatrix} s_1 & s_2 \\ \emptyset & c_1 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} s_1 & s_2 \\ c_1 & c_2 \end{pmatrix}$$

μ_1 is stable. However, a more efficient matching is μ_2 , where both s_1 and s_2 are assigned to a course. Note that μ_2 is also stable.

Example 2.3.4. There are two students $\mathcal{S} = \{s_1, s_2\}$ and two courses $\mathcal{C} = \{c_1, c_2\}$ with $q_x = 1$ for all $x \in \mathcal{S} \cup \mathcal{C}$. Their preference profiles are given as:

$$\begin{array}{ll} s_1 : c_2 \succ c_1 & c_1 : s_1 \sim s_2 \\ s_2 : c_1 \sim c_2 & c_2 : s_1 \sim s_2 \end{array}$$

Consider the following two matchings:

$$\mu_1 = \begin{pmatrix} s_1 & s_2 \\ c_1 & c_2 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} s_1 & s_2 \\ c_2 & c_1 \end{pmatrix}$$

While both μ_1 and μ_2 are stable, μ_2 is more desirable in the sense that s_1 is matched to a course she likes more (c_2) and the assignments of other individuals remain at the same preference level.

In the two examples above, the second matching dominates the first one in the sense that someone's matching is strictly improved while no one else is worse off. This property is captured by the notion of Pareto efficiency. For a Pareto inefficient matching, certain changes to the matching may result in some individuals being better off with no individual worse off, therefore leading to an efficiency improvement. The formal definition of Pareto efficiency is given as follows.

Definition 2.3.5 (Pareto Efficiency). Fix a market $(\mathcal{S}, \mathcal{C}, (q_s)_{s \in \mathcal{S}}, (q_c)_{c \in \mathcal{C}}, \succsim_{\mathcal{S}}, \succsim_{\mathcal{C}})$. Given a feasible matching $\mu = (\mu_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$, we say that $\mu' = (\mu'_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$ is a Pareto improvement of μ if for all $x \in \mathcal{S} \cup \mathcal{C}$, $\mu'(x) \succeq_x \mu(x)$ and $\mu'(x) \succ \mu(x)$ for at least one x . A matching μ is Pareto efficient if it does not have any Pareto improvement.

2.3.3 Pareto Stability

In the original Gale-Shapley stable marriage model with strict preferences, stability implies Pareto efficiency. However, when ties are introduced, stability no longer guarantees Pareto efficiency. Furthermore, even if preferences are strict, in many-to-many matching models with responsive preferences, a stable matching need not be

Pareto efficient.⁷ As a consequence, in a many-to-many matching market with ties, both “many-to-many” and “ties” can contribute to potential Pareto inefficiency in a stable outcome. A natural solution concept in this market is Pareto stability (Sotomayor, 2011), which ensures both fairness and efficiency in an outcome and is defined as follows.

Definition 2.3.6 (Pareto Stability). *A feasible matching μ is Pareto stable if it is both (pairwise) stable and Pareto efficient.*

2.4 Computing a Pareto Stable Matching

We have defined a Pareto stable matching as a desirable outcome in a many-to-many matching market with ties. The next question is about the existence of such a Pareto stable matching and how to compute one if it exists.

For a many-to-one matching model with ties, a Pareto stable matching always exists and can be computed using the algorithm of Erdil and Ergin (2006, 2008). The algorithm is based on two observations. First, a matching has a Pareto improvement only if the corresponding assignment graph has an augmenting path or cycle.⁸ Second, and more critically, any Pareto improvement to a stable matching preserves stability. These observations immediately imply an algorithm to find a Pareto stable matching: starting from any stable matching, keep making Pareto improvements by eliminating augmenting paths and cycles until none remains, and the resulting matching is Pareto stable.

In a many-to-many matching market, if only one side has ties, the same stability preserving result still holds.

Claim 2.4.1. *In a many-to-many matching market, if only one side has ties, Pareto improvement preserves stability.*

However, when both sides of the market have ties, we observe that the second critical property fails. That is, a Pareto improvement to a stable matching need not preserve stability, as illustrated by the following example.

⁷See Example 1.3.3 in Chapter 1.

⁸Augmenting path and cycle are formally defined in Section 2.4.1

Example 2.4.1 (Pareto improvement does not preserve stability). *There are three students $\mathcal{S} = \{s_1, s_2, s_3\}$ and three courses $\mathcal{C} = \{c_1, c_2, c_3\}$ with $(q_{s_1}, q_{s_2}, q_{s_3}) = (1, 2, 1)$ and $(q_{c_1}, q_{c_2}, q_{c_3}) = (1, 2, 1)$. Their preference profiles are given as:*

$$\begin{array}{ll} s_1 : c_1 \sim c_2 & c_1 : s_1 \sim s_2 \succ s_3 \\ s_2 : c_1 \succ c_2 \succ c_3 & c_2 : s_1 \sim s_2 \succ s_3 \\ s_3 : c_2 & c_3 : s_1 \sim s_2 \succ s_3 \end{array}$$

Consider the following matching:

$$\mu_1 = \begin{pmatrix} s_1 & s_2 & s_3 \\ c_1 & c_2, c_3 & c_2 \end{pmatrix}$$

μ_1 is stable. It has a Pareto improvement where s_2 strictly improves her assignment and no one is worse off, resulting in the following matching:

$$\mu_2 = \begin{pmatrix} s_1 & s_2 & s_3 \\ c_2 & c_1, c_3 & c_2 \end{pmatrix}$$

However, μ_2 is unstable since (s_2, c_2) is a blocking pair.

The example above shows that the approach of starting with an arbitrary stable matching and making Pareto improvements does not work, because this need not preserve stability. Thus, all previous approaches computing Pareto stable matchings in variant models fail to work. Furthermore, for the given stable matching μ_1 in the above example, there is only one Pareto improvement μ_2 . Thus, the problem cannot be solved by a careful selection of Pareto improvements.

It is thus unclear whether a Pareto stable matching exists in the general many-to-many matching markets with ties. In this section we give an affirmative answer to this question by presenting an algorithm which computes a Pareto stable matching. We first provide a characterization of Pareto efficiency, and then describe the algorithm.

2.4.1 Characterization of Pareto Efficiency

Given the connection between matching and network flow, it is not surprising that the existence of augmenting paths and cycles in an assignment is closely related to

whether it can be improved, that is, its Pareto efficiency. The main difference in the context of stable matching is that nodes have preferences in addition to capacities. Thus, augmenting paths and cycles must improve not only the size of a matching, but also its quality, as determined by node preferences. Examples of an augmenting path and path are shown in Figure 2.1, with the solid lines representing the existing matching and the dashed lines representing the potential Pareto improvement. Their formal definitions are given as follows.

Figure 2.1: *Augmenting Path and Cycle*

Definition 2.4.2 (Augmenting Path). *Fix a market $(\mathcal{S}, \mathcal{C}, (q_s)_{s \in \mathcal{S}}, (q_c)_{c \in \mathcal{C}}, \succsim_{\mathcal{S}}, \succsim_{\mathcal{C}})$. Given a feasible matching $\mu = (\mu_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$, we say that $[s_0, c_1, s_1, \dots, c_l, s_l, c_{l+1}]$ is an augmenting path if (i) $\sum_c \mu_{s_0 c} < q_{s_0}$ and $\sum_s \mu_{s c_{l+1}} < q_{c_{l+1}}$, (ii) $\mu_{s_k c_k} = 1$ and $\mu_{s_{k-1} c_k} = 0$ for all k , and (iii) $c_{k+1} \succsim_{s_k} c_k$ and $s_{k-1} \succsim_{c_k} s_k$.*

The first condition states that the capacities of s_0 and c_{l+1} are not exhausted. The second condition states that pairs alternatively are not and are in the current matching μ along the path. The last condition ensures that we are able to achieve a Pareto improvement by reassigning matches according to the augmenting path. That is, removing all pairs (s_k, c_k) and matching all pairs (s_k, c_{k+1}) produces a

feasible matching, which is a Pareto improvement over μ , where no one is worse off and s_0 and $c_{\ell+1}$ are better off.

Definition 2.4.3 (Augmenting Cycle). *Fix a market $(\mathcal{S}, \mathcal{C}, (q_s)_{s \in \mathcal{S}}, (q_c)_{c \in \mathcal{C}}, \succsim_{\mathcal{S}}, \succsim_{\mathcal{C}})$. Given a feasible matching $\mu = (\mu_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$, we say that $[s_1, c_2, s_2, \dots, c_\ell, s_\ell, c_1, s_1]$ is an augmenting cycle if (i) $\mu_{s_k c_k} = 1$ and $\mu_{s_k c_{k+1}} = 0$ for all k (where $c_{\ell+1} = c_1$) (ii) $c_{k+1} \succsim_{s_k} w_k$ and $s_{k-1} \succsim_{c_k} s_k$, and at least one of these preferences is strict.*

We can match all pairs (s_k, c_{k+1}) and unmatched all pairs (s_k, c_k) in an augmenting cycle to get a Pareto improvement. For a given matching, an augmenting path or cycle can be found easily using a network flow approach.

The following lemma characterizes the relation between Pareto efficient matching and augmenting path and cycle. The proof in Erdil and Ergin (2008) for the many-to-one matching problem remains valid in the many-to-many matching framework.

Lemma 2.4.4. *A feasible matching is Pareto efficient if and only if it has no augmenting path or cycle.*

2.4.2 A Pareto Stable Matching Algorithm

Our algorithm builds on the idea of Roth and Vate (1990), who provide an alternative to the deferred acceptance algorithm to compute a stable one-to-one matching. Their algorithm can be interpreted as follows. Assume that all women are present at the beginning, and men “arrive” one by one. We start with the empty matching. When a new man m arrives, match him to a most preferred woman w with whom he forms a blocking pair, if any; if this woman was already matched to a man m' , set m' free and consider him as the next arriving man; the algorithm runs iteratively until all men have arrived. Because every woman which changes its matching in this process gets a strict improvement and no woman ever becomes worse off, the algorithm terminates, and the final matching is stable, because by construction the matching at every man’s arrival is stable.

In our algorithm, all students and courses are initially available. Courses are with *full* capacities and students are with *null* capacity. We consider all students one by one and increase their capacities unit by unit. When the capacity of a student is

increased by one, we do a sequence of reassignments such that the resulting matching satisfies the following invariants (with respect to the current considered capacities):

- *Stability preserving*: it is always stable.
- *Courses improving*: the matching of any course does not become worse off.
- *No augmenting cycle*: it does not contain any augmenting cycle.

An important idea in our algorithm to derive Pareto efficiency is that in the process of reassignments, no augmenting cycles have ever been introduced in the matching. However, we allow the existence of augmenting paths. The key component of our algorithm is a subroutine for eliminating augmenting paths while preserving stability and not introducing any augmenting cycle. Having constructed a matching that is stable and contains no augmenting cycles, we apply the subroutine to eliminate augmenting paths in a stability preserving fashion, which finally yields a Pareto stable matching as characterized by Lemma 2.4.4. The high-level structure of the algorithm is described as follows.

PARETO-STABLE ALGORITHM

1. Initialization:

- There are no assigned matches between $\mathcal{S}$ and $\mathcal{C}$.
- All courses have their full capacities available.
- Let $\mathbf{d} = (d_s)_{s \in \mathcal{S}}$ be a virtual capacity vector of students;
initially $d_s = 0$ for $s \in \mathcal{S}$.

2. While there is $s \in \mathcal{S}$ with $d_s < q_s$, run INCREASE-CAP($\mathbf{d}$).

3. While there is an augmenting path P , run ELIMINATE-PATH(P).

4. Return the final matching μ .

Note that in the algorithm, we always maintain the invariant that the algorithm contains no augmenting cycles. Why do we need such a condition, whereas it is allowed to have augmenting paths? Observe that the reason that a Pareto improvement may not preserve stability is that the path or cycle corresponding to the Pareto improvement contains a matched pair (s, c) where both s and c are also matched to a less preferred agent, say c' and s' . When the match (s, c) is removed in

the reassignment process of the augmenting path/cycle, even though s and c could receive better partners in the path or cycle, they will prefer to be matched to each other instead of c' and s' respectively. For augmenting path, however, we can always start reassignment from one side of the path (say, the student), and stop proceeding along the path when we reach such a course c (then (s', c) is unmatched and the process restarts). In this stability-preserving process, a course becomes strictly better off. However, for the pair (s, c) in an augmenting cycle, we would need to release both (s', c) and (s, c') to preserve stability. That is, we would no longer have the monotonically improving property for courses' assignments, which is critical to the analysis of the algorithm.

Note that in the algorithm, $\mu = (\mu_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$ and $(d_s)_{s \in \mathcal{S}}$ are global variables in both subroutines. The first subroutine, INCREASE-CAP, increases the virtual capacity of a student by one and does a number of reassignments to ensure the three invariants listed above (in particular, it guarantees that the assignment is stable for the increased virtual capacity vector). The second subroutine, ELIMINATE-PATH, eliminates all possible augmenting paths to derive a Pareto efficient assignment in a stability preserving fashion. After all augmenting paths have been eliminated, by Lemma 2.4.4, the returned assignment is Pareto stable.

While the algorithm may look a bit complicated, the fact that no courses ever get worse off in the process implies a simple, but critical, structure of the algorithm: we iteratively do a sequence of reassignments to improve courses' assignments while preserving stability and containing no augmenting cycle. If at any moment in the algorithm a course's assignment gets strictly improved, no matter at which stage the algorithm is, we terminate that thread immediately and go to Step (2) of the main algorithm to repeat the process given the current virtual capacity vector $\mathbf{d}$.

Next we describe the two subroutines in detail in the following subsections. All discussions are with respect to the considered virtual capacity vector. In the algorithm, for any (augmenting) cycle C and a pair $(s, c) \in C$, we use $C \setminus \{(s, c)\}$ to denote the path by removing pair (s, c) from C .

2.4.2.1 Capacity Increment

The first subroutine that increases virtual capacities of the students is the following.

INCREASE-CAP(**d**)

1. Pick an arbitrary student s with $d_s < q_s$
2. Let $d_s \leftarrow d_s + 1$, i.e., increase the virtual capacity of s by one
3. Let $S = \{c \mid (s, c) \text{ is a blocking pair}\}$
4. Let $T = \{c \in S \mid s \text{ prefers } c \succeq c' \text{ for any } c' \in S\}$
5. If $T = \emptyset$ (i.e., there is no blocking pair), return
6. Otherwise
 - (a) If there exists $c \in T$ such that adding match (s, c) does not introduce any augmenting cycle
 - pick such a course c'
 - add match (s, c')
 - (b) Otherwise
 - pick an arbitrary $c' \in T$
 - let C be a potential augmenting cycle by adding (s, c')
 - let $P = \left[s \xrightarrow{C \setminus \{(s, c')\}} c' \right]$ be the path from s to c' through $C \setminus \{(s, c')\}$
 - run ELIMINATE-PATH(P)
 - (c) If c' (defined either in Step (6.a) or (6.b)) is over-matched (i.e., matched to more than $q_{c'}$ neighbors)
 - let s' be a least preferred student matched to c' where deleting (s', c') does not introduce an augmenting cycle
 - delete match (s', c')
 - let $d_{s'} \leftarrow d_{s'} - 1$
 - return
 - (d) Otherwise, return

When the virtual capacity of s is increased by one, there might be some blocking pairs, among which the subroutine tries to match s to one that he prefers most ($c' \in T$ in the above description). However, this could introduce potential augmenting cycles (Step 6(b)). Instead of matching s and c' directly, the subroutine

considers a potential augmenting cycle C incurred by (s, c') and tries to do reassignments according the other path from s to c' along the cycle. Finally, if c' is over-matched, then we delete one of her least preferred assignments without incurring any augmenting cycles and delete the virtual capacity of that student by one. This guarantees that the assignment remains stable, and the assignment of c' strictly improves. The existence of s' in Step 6(c) is guaranteed by the following lemma.

Lemma 2.4.5. *Given a stable matching without augmenting cycles, for any course c , let $S \subseteq \mathcal{S}$ be the subset of students matched to c to whom c is least preferred. Then there is $s \in S$ such that deleting match (s, c) does not introduce any augmenting cycle.*

2.4.2.2 Augmenting Path Elimination

Consider a given stable assignment, assume there is an augmenting path $P = [s_0, c_1, s_1, \dots, c_\ell, s_\ell, c_{\ell+1}]$, where (s_i, c_i) is in the assignment and (s_i, c_{i+1}) is not. It is possible that an individual x (either a student or a course) or a pair (x, y) appears more than once in P . In this subsection, when we refer to an individual $x \in P$ or a pair $(x, y) \in P$, we denote the corresponding one at that position of P .

Truncation. Before describing the subroutine, we will first consider a truncation process, which deletes some pairs in a given augmenting path according to different appearances of the same agent and will be used in the subroutine.

TRUNCATE-PATH(P)

1. **while** one of the following "if" conditions holds

- If there is s such that $P = [\dots, s, c_1, \dots, c_2, s, \dots]$ and s weakly prefers c_1 to c_2
 - truncate $P = [\dots, s, \text{---}(c_1, \dots, c_2, s, \text{---}) \dots]$
- If there is c such that $P = [\dots, c, s_1, \dots, s_2, c, \dots]$ and c weakly prefers s_2 to s_1
 - truncate $P = [\dots, c, \text{---}(s_1, \dots, s_2, c, \text{---}) \dots]$

2. **Return** path

It can be seen that if $\text{TRUNCATE-PATH}(P)$ is executed, by the rules of the truncation, no pair (x, y) can appear more than once after truncation. However, it is still possible that an individual appears more than once (e.g., when s strictly prefers c_2 to c_1 , we do not truncate the two occurrences of s). The truncation process is necessary in our algorithm; in particular, it is important to the analysis of termination of the algorithm.

In $\text{TRUNCATE-PATH}(P)$, if a truncation is executed at x (a student or a course), we denote by $\Gamma(x)$ the truncated path. That is, $\Gamma(x) = [s, c_1, \dots, c_2, s]$ if $x = s$, and $\Gamma(x) = [c, s_1, \dots, s_2, c]$ if $x = c$.

We have the following observations.

Proposition 2.4.6. *For any given augmenting path P , if a truncation is executed at x , then $\Gamma(x)$ forms a cycle and every individual involved is indifferent between its two neighbors in the cycle.*

Proposition 2.4.7. *For any given augmenting path P , if a truncation is executed at x and x strictly prefers its one neighbor to the other for one occurrence of x in the truncation, then x still strictly prefers one neighbor to the other after truncation.*

Lemma 2.4.8. *For any given augmenting path P , $\text{TRUNCATE-PATH}(P)$ returns an augmenting path as well.*

Elimination. We next describe the subroutine to eliminate augmenting paths while preserving the three invariants listed at the beginning of the section. Note that for any augmenting path, its one side must be a student and the other side must be a course. The subroutine starts from the student side and considers pairs one by one. Hence, for any student-course pair in the path, the objective is to match them; and for any course-student pair in the path, the objective is to unmatched them.

The subroutine tries to add and delete matches one by one along pairs in the path P . If the current considered pair is a student-course pair (i.e., $e = (s, c)$), the subroutine matches them if it does not introduce any augmenting cycle. If the assignment of c is strictly improved (i.e., the condition in Step (3.b) or (3.c) is satisfied), the subroutine terminates. Note that at this point the subroutine may not completely eliminate the augmenting path, however, the overall assignment of

ELIMINATE-PATH(P)

1. Assume $P = [s^*, c_1, s_1, \dots, c^*]$
2. Let $e = (s^*, c_1)$ be the first pair on path P
3. while $e \neq \emptyset$
 - If e is not a match (i.e., $e = (s, c)$)
 - if adding match (s, c) does not introduce an augmenting cycle
 - (a) add match (s, c)
 - (b) if c is not over-matched, return
 - (c) if c strictly prefers s to a current partner
 - * let s' be a least preferred student matched to c where deleting (s', c) does not introduce an augmenting cycle (by Lemma 2.4.5, such s' exists)
 - * delete match (s', c) and let $d_{s'} \leftarrow d_{s'} - 1$
 - * return to Step 2 of the main algorithm PARETO-STABLE to run INCREASE-CAP
 - (d) else let e be the next pair after (s, c) in P
 - otherwise
 - (e) let $C = [s, c'_1, s'_1, \dots, c'_k, s'_k, c, s]$ be such a potential cycle if adding (s, c)
 - (f) expand $P = \left[s^*, \dots, s, c'_1 \xrightarrow{C \setminus \{(s, c)\}} s'_k, c, \dots, c^* \right]$
 - (g) truncate $P = \left[s^*, \dots, \text{TRUNCATE-PATH} \left(s, c'_1 \xrightarrow{C \setminus \{(s, c)\}} s'_k, c, \dots, c^* \right) \right]$
 - (h) let e be the first pair returned by the TRUNCATE-PATH
 - If e is a match (i.e., $e = (c, s)$)
 - if deleting match (c, s) does not introduce an augmenting cycle
 - (i) delete match (c, s)
 - (j) let e be the next pair after (c, s) in P
 - otherwise
 - (k) run the above Steps (e,f,g,h)
 - (switching the notations of s and c (except s^* and c^*))

the course gets strictly improved and the process restarts at the capacity increment

stage. If matching s and c will introduce a potential augmenting cycle, instead of adding the match directly, the subroutine takes a “detour” and considers the other path from s to c along the cycle and expands it to the path P (Step 3(f); by the following Lemma 2.4.9, it is a valid expansion). Then the subroutine will do a truncation from s to the end of the path P and restarts the process by considering the first pair returned by the truncation (its first individual must be s). The subroutine performs similarly if the considered pair is a course-student pair.

We first establish the following observations.

Lemma 2.4.9. *The expansion of path P in Step (3.f) is a well-defined augmenting path.*

We have the following key claim, which implies that the subroutine always terminates.

Lemma 2.4.10. *The subroutine ELIMINATE-PATH(P) terminates in finite number of steps for any augmenting path P .*

2.4.3 Analysis of the Algorithm

Again, the high level structure of the algorithm is to increase capacities of students and eliminate augmenting paths. While the algorithm may look involved, as the virtual capacity is not always monotonically increasing (e.g., in Step 6(c) of INCREASE-CAP and Step 3(c) of ELIMINATE-PATH, we actually need to reduce the virtual capacities) and two subroutines may call each other, there is a simple, but crucial, idea behind the algorithm: the assignments of courses keep improving (this is the exact reason that we do not want to introduce any augmenting cycle in the course of the algorithm). Therefore, at any moment of the algorithm, if a course’s assignment gets improved (e.g., Step 6(c) of INCREASE-CAP and Step 3(b), 3(c) of ELIMINATE-PATH), the algorithm will abandon the current subroutine and restart the whole process (i.e., capacity increment and augmenting path elimination) starting from the current virtual capacity vector. Since every course can improve her assignment at most n^2 times (as her capacity is at most n and every unit capacity can be improved at most n times), the whole algorithm will terminate.

It is easy to see that the three invariants listed at the beginning of the section are maintained in the course of the algorithm. Indeed, the last two (no augmenting cycle and courses not worse off) hold trivially as they are guaranteed by the algorithm itself. For stability, in the subroutine INCREASE-CAP, when increasing the virtual capacity of s by one, we try to match s with a most preferred course c where (s, c) forms a blocking pair. If c is not over-matched, then the resulting assignment is still stable. Otherwise, we delete a match (s', c) where s' is a least preferred student matched to c and reduce the virtual capacity of s' by one (Step (6.c) of INCREASE-CAP); this implies that the resulting assignment is still stable with respect to the new capacity vector. For the second subroutine ELIMINATE-PATH, stability comes from the definition of augmenting path and the fact that when we delete a match (c, s) , we know that s must be a least preferred student matched to c and c was over-matched (otherwise, when we add the match right before (c, s) , the assignment of c gets strictly improved and the subroutine will run Step (3.b) or (3.c) to terminate). Therefore, the final returned assignment is stable.

When the algorithm PARETO-STABLE terminates, by its rule there is no augmenting path. By the invariant that there is no augmenting cycle, we know that the returned assignment is Pareto efficient. This yields the following result.

Theorem 2.4.11. *For the generalized many-to-many matching market with ties, a Pareto stable matching always exists and can be computed by the algorithm PARETO-STABLE in polynomial time.*

2.4.4 Multiple Preferences

Our PARETO-STABLE algorithm continues to work for settings with such multiple preferences. In general, for each individual x , the other side of the market is divided into (not necessarily disjoint) partitions $S_1(x), S_2(x), \dots, S_\ell(x)$, where x has a preference (again, can be incomplete and have ties) and a capacity q_{xk} for each partition $S_k(x)$. Further, x has a universal capacity q_x that bounds the total number of partners that can be matched to x among all partitions. Observe that there are now two types of capacity constraints for each individual: a universal one and a local one for each partition. Without loss of generality, we can assume that $q_x \geq q_{xk}$ for

any k . Note that there could be no relation between q_x and $\sum_k q_{xk}$. The preference model discussed in the previous sections corresponds to the special case when there is only one partition (i.e., $\ell = 1$).

In this extension, the preference of every agent is restricted to every partition. While partitions are not necessarily disjoint, we assume that the preference lists of all partitions of an individual are disjoint. Hence, for a given assignment $\mu = (\mu_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$, s and c form a blocking pair if both of them strictly prefer each other to one of their assigned partners in the same partition (or if they have remaining capacities). The preferences over subsets are restricted to every partition as well and defined similarly.

Our objective is again to find a Pareto stable matching, which can be computed by the same algorithm described above. Observe, however, that the preference of every agent is essentially with respect to each of its partitions. Hence, the definition of augmenting path and cycle will be changed accordingly, i.e., for any node in the path/cycle, its two neighbors must be from the same partition. Note that an agent can still appear more than once in a path/cycle, but different appearances may correspond to different partitions. The algorithm described above continue to apply here, where all executions are restricted to be within each of the partitions respectively.

2.5 Conclusion

In the many-to-many matching problem with ties, agents can be matched with multiple partners and their preferences are not necessarily strict. This problem is a generalization of many previously studied two-sided matching problems and it has significant implications for many real life two-sided matching market designs. Despite the wide applications in real life, the many-to-many matching problem with ties have been overlooked in both theoretical and empirical market design literature. This chapter represents the very first comprehensive study on this fundamental problem and is devoted to improving social efficiency and welfare.

For the many-to-many matching problem with ties, we propose a solution con-

cept called Pareto stability, which requires both stability and Pareto efficiency in a matching outcome, to ensure fairness and efficiency. We show that a Pareto stable matching always exists by developing an efficient algorithm that computes such a matching. This positive result provides a systematic procedure to find a fair and efficient solution, which can be widely applied in many two-sided matching markets.

2.6 Appendix

2.6.1 Responsive Preference with Lattice Structure

We give an example to show a responsive preference with a lattice structure. Assume that the preference of s over individuals is $\succsim_s$: $(c_1 \succ c_2 \sim c_3 \succ c_4 \sim c_5)$ and its capacity is 3, then the lattice structure of m 's responsive preference is shown in Figure 2.2. Each node in the figure denotes a feasible matching. For example, the node " $\{1,2,3\}$ " means that s is matched with $\{c_1, c_2, c_3\}$.

Figure 2.2: *Lattice Structure of Responsive Preference.*

2.6.2 Proofs

We present the omitted proofs for this chapter.

Proof of Claim 2.4.1

Proof. Without loss of generality, we assume that only courses have ties in their preferences. Let μ be a stable matching and μ' be one derived from μ through Pareto improvement. Then for all $x \in \mathcal{S} \cup W$, $\mu'(x) \succeq_x \mu(x)$, and at least one preference is strict.

Assume that μ' is not stable and (s, c) is a blocking pair. Let

$$\mu(s) = \{c_1, c_2, \dots, c_{q_s}\} \text{ and } \mu'(s) = \{c'_1, c'_2, \dots, c'_{q_s}\},$$

and let

$$\mu(c) = \{s_1, s_2, \dots, s_{q_c}\} \text{ and } \mu'(c) = \{s'_1, s'_2, \dots, s'_{q_c}\}.$$

(We can add $\emptyset$ to the list if one is not fully matched.) We can enumerate the indices such that $c'_i \succeq_s c_i$, for all $i = 1, 2, \dots, q_s$ and $s'_j \succeq_c s_j$, for all $j = 1, 2, \dots, q_c$. Note that $c_i =_s c'_i$ if and only if c_i and c'_i are the same course. Since (s, c) is a blocking pair for μ' , we know that $c \succ_s c'_{q_s}$ and $s \succ_c s'_{q_c}$.

Case 1: s and c are not matched in μ . Then $c \succ_s c'_{q_s} \succeq_s c_{q_s}$ and $s \succ_c s'_{q_c} \succeq_c s_{q_c}$, implying that (s, c) is a blocking pair for μ , a contradiction.

Case 2: s and c are matched in μ . Since the assignment of c is not worse off in μ' and $s \notin \mu'(c)$, there is $s' \in \mu'(c)$ such that $s' \notin \mu(c)$ and $s' \succeq_c s$. Consider s' ; as all students have strict preferences and s' is not worse off in μ' , there is $c' \in \mu(s')$ such that $c \succ_{s'} c'$. Hence, (s', c) is a blocking pair for μ , a contradiction.

Hence, μ' has no blocking pairs, and the claim follows. $\square$

Proof of Lemma 2.4.5

Proof. Consider any student $s_1 \in \mathcal{S}$; assume that deleting (s_1, c) introduces an augmenting cycle C_1 (otherwise, we are done); let s_2 be the other student incident to c in C_1 . Note that c is matched to both s_1 and s_2 and weakly prefers s_1 to s_2 (due to the fact that C_1 is an augmenting cycle if deleting (s_1, c)). Hence, $s_2 \in S$. Next consider deleting (s_2, c) and again assume that it introduces an augmenting cycle C_2 ; let s_3 be the other student incident to c in C_2 . By the same argument, $s_3 \in S$; we

may continue with this argument — if none of these matches can be deleted, then we get a loop $s_1, s_2, \dots, s_r, s_{r+1} = s_1$, where deleting (s_i, c) introduces an augmenting cycle C_i containing (s_{i+1}, c) , for $i = 1, \dots, r$. Note that c is indifferent between all $s_1, s_2, \dots, s_r$. Then consider the following big cycle

$$C = \left[s_1 \xrightarrow{C_1 \setminus \{(s_1, c), (s_2, c)\}} s_2 \xrightarrow{C_2 \setminus \{(s_2, c), (s_3, c)\}} s_3 \cdots s_r \xrightarrow{C_r \setminus \{(s_r, c), (s_1, c)\}} s_1 \right]$$

Note that it is possible that an edge appears more than once in C . For each s_i , $i = 2, \dots, r+1$, let c'_{i-1} and c_i be the other course (not c) incident to s_i in cycle C_{i-1} and C_i , respectively. Notice that each s_i weakly prefers c'_{i-1} to c and weakly prefers c to c_i , i.e., s_i weakly prefers c'_{i-1} to c_i . Further, if one of these two preferences is strict, then s_i strictly prefers c'_{i-1} to c_i . Therefore, C is an augmenting cycle, a contradiction. (Note that the fact that there is an individual whose assignment can be strictly improved from C follows from the fact of augmenting cycles of each $C_1, \dots, C_r$.) $\square$

Proof of Proposition 2.4.6

Proof. We will only prove the claim for the first case when $\text{TRUNCATE-PATH}(P)$ is executed at a student; the argument for the second case is similar. Assume that the given augmenting path $P = [\dots, s, c_1, \dots, c_2, s, \dots]$ is truncated between the two occurrences of s . By the rule of truncation, s weakly prefers c_1 to c_2 ; by the rule of augmenting path P , all individuals weakly prefer his/her unmatched neighbor to matched neighbor. Hence, $[s, c_1, \dots, c_2, s]$ forms a cycle and everyone is indifferent between its two neighbors (otherwise s strictly prefers c_1 to c_2 , it is an augmenting cycle, which contradicts to the fact that no augmenting cycle ever appears in the course of the algorithm). $\square$

Proof of Proposition 2.4.7

Proof. We will only prove the claim when x is a student s ; the argument for course is similar. Consider the augmenting path $P = [\dots, c_1, s, c_2, \dots, c_3, s, c_4, \dots]$ and a truncation is executed at s . Assume that s strictly prefers c_2 to c_1 . Since P is an augmenting path, we know that s weakly prefers c_4 to c_3 . By Proposition 2.4.6, s is indifferent between c_2 and c_3 . Therefore, after truncation s strictly prefers one

neighbor c_4 to the other c_1 . The same argument holds if the strict preference occurs at the second occurrence s (i.e., s strictly prefers c_4 to c_3). $\square$

Proof of Lemma 2.4.8

Proof. Again we will only prove the claim for the first case of $\text{TRUNCATE-PATH}(P)$ and the second case follows similarly. For the path $P = [\dots, s, c_1, \dots, c_2, s, \dots]$ with the truncation for the middle of the two occurrences of s , if the first s is the beginning of path P , then certainly after truncation it is still a valid augmenting path. Otherwise, we can write P as $[\dots, c_0, s, c_1, \dots, c_2, s, c_3 \dots]$ (note that the end of the path must be a course). Notice that s weakly prefers c_1 to c_0 , and c_3 to c_2 . Further, we have s is indifferent between c_1 and c_2 by Proposition 2.4.6. Hence, s weakly prefers c_3 to c_0 , which implies the desired result. $\square$

Proof of Lemma 2.4.9

Proof. Let $P_1 = [s^*, c_1, s_1, \dots, c', s]$ and $P_2 = [c, s', \dots, c^*]$, where c' is the course before s and s' is the student after c in P . Then the original augmenting path can be written as $P = [P_1, s, c, P_2]$. By the fact that P is an augmenting path, we know that s weakly prefers c to c' and c weakly prefers s to s' . Let $C' = [s'_1 \xrightarrow{C \setminus \{s, c\}} c'_k]$, then the extended path (denoted by P') is $P' = [P_1, s, C', c, P_2]$. By the fact that C is an augmenting cycle if adding (s, c) , we know that s weakly prefers c'_1 to c and c weakly prefers s'_k to s . Therefore, s weakly prefers c'_1 to c' and c weakly prefers s'_k to s' ; this implies that the expanded path P' is a well-defined augmenting path. $\square$

Proof of Lemma 2.4.10

We will prove the second subroutine ELIMINATE-PATH always terminates.

Proposition 2.6.1. *In the course of the subroutine $\text{ELIMINATE-PATH}(P)$, for each considered pair e in the augmenting path P , we can always reach a different pair (i.e., redefine e by Step (3.d), (3.h), or (3.j)) with different starting individual. That is,*

- if $e = (s, c)$, there is c' such that the subroutine matches (s, c') (not introducing any augmenting cycle) and next moves to $e = (c', \cdot)$;
- if $e = (c, s)$, there is s' such that the subroutine deletes (c, s') (not introducing any augmenting cycle) and next moves to $e = (s', \cdot)$.

Proof. We will only prove the claim for the case when $e = (s, c)$; the same argument extends for $e = (c, s)$. Assume that adding $(s, c_1) \triangleq (s, c)$ introduces an augmenting cycle C_1 (otherwise, we are done); let c_2 be the other course incident to s in C_1 . Note that s weakly prefers c_2 to c_1 and c_2 weakly prefers s to her assignment in C_1 . Next the subroutine expands path P with $\left[s, c_2 \xrightarrow{C_1 \setminus \{(s, c_1)\}} c_1 \right]$, and consider adding (s, c_2) . Again assume that it introduces an augmenting cycle C_2 ; let c_3 be the other course incident to s in C_2 . We may continue with this argument; if none of these matches can be added, then we get a loop $c_1, c_2, \dots, c_r, c_{r+1} = c_1$, where adding (s, c_i) introduces an augmenting cycle C_i containing (s, c_{i+1}) , for $i = 1, \dots, r$. Note that s is indifferent between all $c_1, c_2, \dots, c_r$. Then consider the following big cycle

$$C = \left[c_1 \xrightarrow{C_1 \setminus \{(s, c_1), (s, c_2)\}} c_2 \xrightarrow{C_2 \setminus \{(s, c_2), (s, c_3)\}} c_3 \cdots c_r \xrightarrow{C_r \setminus \{(s, c_r), (s, c_1)\}} c_1 \right]$$

Note that C is available before the subroutine arrives at edge $(s, c) = (s, c_1)$ and it is possible that an edge appears more than once. For each c_i , $i = 2, \dots, r+1$, let s'_{i-1} and s_i be the other student (not s) incident to c_i in cycle C_{i-1} and C_i , respectively. Notice that each c_i weakly prefers s to s'_{i-1} and weakly prefers s_i to s , *i.e.*, c_i weakly prefers s_i to s'_{i-1} . Further, if one of the two preferences is strict, then c_i strictly prefers s_i to s'_{i-1} . Therefore, C is an augmenting cycle, a contradiction to the invariant that the algorithm will never produce any augmenting cycle in the process. (Note that the fact that there is an individual whose assignment can be strictly improved from C follows from the fact of augmenting cycles of each $C_1, \dots, C_r$.) $\square$

We are now ready to prove the lemma.

Proof of Lemma 2.4.10. Note that if the condition in Step (3.b) or (3.c) is satisfied, *i.e.*, the assignment of a course gets strictly improved, the subroutine returns and terminates. Hence, we will assume without loss of generality that in the course of $\text{ELIMINATE-PATH}(P)$, all courses involved are already fully-matched and are indifferent between their adjacent neighbors in path P .

Assume to the contrary that $\text{ELIMINATE-PATH}(P)$ does not terminate. By Proposition 2.6.1, we know that the subroutine will not get stuck at any specific node. This implies that the subroutine will keep changing the statuses of pairs (*i.e.*,

either matched or unmatched) through Step (3.a) and (3.i). Since the assignments of all courses are kept at the same level, and the assignments of all students will not get worse off (by the definition of augmenting path)⁹, we can divide the subroutine into *stages* where it moves from one stage to another if there is a student whose assignment get strictly improved. Since the subroutine does not terminate, it eventually gets into the last stage where no student will be able to improve his assignment. In other words, all individuals (students and courses) are indifferent between their new assigned partner(s) and old partner(s) onwards.

Consider a moment when the subroutine is at the last stage, and let

$$P^* = [x^*, y^*, \dots, w^*]$$

be the current remaining augmenting path (*i.e.*, after expansions and truncations in previous stages), where $e = (x^*, y^*)$ is the current considered pair (can be either (s, c) or (c, s)) and c^* is the last course of the augmenting path. Since it is guaranteed that the subroutine will never introduce any augmenting cycle, at this moment there is no augmenting cycle. In the rest of the proof we will restrict on the subroutine starting from this moment running on P^* .

For the initial augmenting path P^* , we set all pairs on it to be *unmarked*. In the process of the subroutine when P^* is updated, we mark/unmark pairs according to the following rules:

1. If the status of (x, y) is changed (become matched or unmatched), *mark* (x, y) .
2. Recursively do the following: If a pair (x, y) is marked, *mark* all pairs in $\Gamma(y)$ (recall that $\Gamma(y)$ is the truncated path at a specific occurrence of y in the path P^*).
3. If P^* is expanded, *unmark* all expanded pairs.

Roughly speaking, the sign of a pair, marked or unmarked, denotes whether the subroutine has reached that pair or not in the path P^* . In particular, if the subroutine

⁹In the subroutine ELIMINATE-PATH, we will do a sequence of reassignments, e.g., first unmatched (c, s) then match (s, c') . Precisely speaking, the assignment of s first gets worse off then gets better off. Here saying s does not get worse off or gets strictly better off, we mean the overall assignment of s by combining these two consecutive reassignments.

reaches to the last pair of P^* , all pairs have to be marked.

Let E^* denote a subset of pairs where $(x, y) \in E^*$ if in the process of running ELIMINATE-PATH, the subroutine cannot change the status of (x, y) because otherwise it will bring a potential augmenting cycle. Note that (x, y) can be either (s, c) or (c, s) . Certainly $E^* \neq \emptyset$. Let $(x_1, y_1), (x_2, y_2), \dots, (x_\ell, y_\ell)$ be the order of pairs that are included into the subset E^* in the subroutine (note that $|E^*|$ is finite as the number of pairs is finite) and $C_1, C_2, \dots, C_\ell$ be corresponding potential augmenting cycles. Note that $(x_i, y_i), (x_{i+1}, y_{i+1}) \in C_i$ for $i = 1, \dots, \ell - 1$ (indeed, (x_{i+1}, y_{i+1}) is the reason that why the subroutine cannot move along with $\left[x_i \xrightarrow{C_i \setminus \{(x_i, y_i)\}} y_i \right]$ to reach y_i).

We have the following observations.

Claim 1. Consider any $(x, y) \in E^*$ and the moment when the subroutine is about to change its status but cannot do so because of a potential augmenting cycle C . Then x is indifferent between its two neighbors in C and the other neighbor right before it in P^* .

Proof. We will only prove the claim for the case $(x, y) = (s, c)$; the argument is similar when $(x, y) = (c, s)$. Let c_1 be the course before s in the augmenting path P^* and c_2 be the other course incident to s in C . By the definition of augmenting path P^* and augmenting cycle C , we know that s weakly prefers c to c_1 and c_2 to c ; hence s weakly prefers c_2 to c_1 . If s strictly prefers c_2 to c_1 , then by the rule of ELIMINATE-PATH, s is able to strictly improve his assignment, which contracts to the assumption that we are at the last stage where no one can improve his assignment anymore.

Claim 2. Consider any $(x, y) \in E^*$ and the moment when the subroutine is about to change its status but cannot do so because of a potential augmenting cycle C . Let y' be an individual who is able to strictly improve its assignment in $C = [x, \dots, x', y', x'', \dots, y, x]$ (by the above claim, $y' \neq x$). By the rule of the subroutine, we will expand the augmenting path to be

$$P^* = \left[\dots, x \xrightarrow{C \setminus \{(x, y)\}} x', y', x'' \xrightarrow{C \setminus \{(x, y)\}} y, \dots, c^* \right]$$

and all pairs between x and y are unmarked. Then all pairs after (x', y') (inclusive)

in the current P^* are always unmarked from this moment through the course of the subroutine.

Proof. We will prove the claim for the first pair (x_1, y_1) added into E^* ; the proof for the rest of pairs can be done in a similar way by induction. Initially all pairs in P^* are unmarked. The subroutine follows pairs in P^* one by one — changes their status and makes them marked — until the point when we get to (x_1, y_1) . At this moment all pairs after (x_1, y_1) in P^* are still unmarked. Then the subroutine expands P^* according to the potential augmenting cycle $C_1 = [x_1, \dots, x', y', x'', \dots, y_1, x_1]$ and unmarks all expanded pairs.

We will first show that (x', y') is always unmarked. Since y is an individual who is able to strictly improve its assignment, we cannot change the status of (x', y') (*i.e.*, get it marked) directly, because otherwise its assignment will be strictly improved. Hence, the only way to mark (x', y') is through the second rule above by marking all pairs in a truncated cycle $\Gamma(z)$, where z is the node whose truncation contains (x', y') . By Proposition 2.4.6, we know that all individuals in $\Gamma(z)$ are indifferent between their two neighbors; this implies that $(y', x'') \notin \Gamma(z)$. Since (x', y') and (y', x'') are consecutive pairs in P^* , the only way to separate them is to truncate at y' , *i.e.*, $z = y'$. By Proposition 2.4.7, however, after such truncation, y' still strictly prefers its one neighbor to the other, which implies that it is still able to strictly improve its assignment, a contradiction.

Next consider any pair (x_0, y_0) after (x', y') in P^* , *i.e.*,

$$P^* = [\dots, x', y', x'', \dots, x_0, y_0, \dots, c^*]$$

since (x', y') is always unmarked, again the only way to mark (x_0, y_0) is through a truncated cycle. But that cycle has to include (x', y') , which is impossible.

We consider the following walk according to pairs $(x_1, y_1), (x_2, y_2), \dots, (x_\ell, y_\ell)$ in E^* : start from y_1 following the direction of $C_1 \setminus \{(x_1, y_1)\}$ until we get to y_2 ; next start from y_i following the direction of $C_i \setminus \{(x_i, y_i)\}$ until we get to y_{i+1} for $i = 2, \dots, \ell - 1$; finally start from y_ℓ following the direction of $C_\ell \setminus \{(x_\ell, y_\ell)\}$ until we get to the first y_k , where $(x_k, y_k) \in E^* \cap C_\ell$ (note that such (x_k, y_k) must exist, otherwise, the subroutine can reach y_ℓ , which contradicts to the above claim).

Therefore, it forms a big cycle

$$C^* = \left[y_k \xrightarrow{C_k \setminus \{(x_k, y_k)\}} y_{k+1} \xrightarrow{C_{k+1} \setminus \{(x_{k+1}, y_{k+1})\}} y_{k+2} \rightarrow \cdots \rightarrow y_\ell \xrightarrow{C_\ell \setminus \{(x_\ell, y_\ell)\}} y_k \right]$$

Note that by the above claim, all pairs in the walk are unmarked. Further, it can be seen that C^* is an augmenting cycle, which contradicts to the fact that the algorithm never introduces an augmenting cycle. This completes the proof of the lemma. $\square$

Chapter 3

Two Competing Pareto Stable Matching Mechanisms

3.1 Introduction

In the practical many-to-many matching markets, we notice that some markets have homogeneous preferences on one side, especially in object or resource allocation problems. The homogeneous preferences can be considered as a priority ranking (possibly with ties) for agents on the other side. A motivating example is the course allocation problem in which courses rank students based on their study years. We refer to this specific problem with homogeneous preferences on one side of the market as **many-to-many matching with one-sided homogeneous preferences**. The homogeneity property has been investigated in other settings as well. For instance, for resource allocation with multi-unit demand, Kojima (2013) shows that stability is compatible with strategyproofness or efficiency if and only if the priority structure is essentially homogeneous.

In this problem, we are interested in finding a Pareto stable matching as for the more generalized many-to-many matching problem with ties. Even though the technical PARETO-STABLE algorithm presented in Chapter 2 still works for this problem as it is a special case of the generalized many-to-many matching problem with ties, the one-sided homogeneous preferences allow us to design additional Pareto stable matching mechanisms which are easier to implement in practice.

In this chapter, we propose two competing Pareto stable mechanisms known as the **Pareto-improving draft mechanism** and the **Pareto-improving dictatorship mechanism**.¹ These two mechanisms are built on the simple draft and the dictatorship mechanisms in literature. The draft mechanism is used in the course allocation at Harvard Business School and is studied by Budish and Cantillon (2012). The dictatorship mechanism is used in many applications (with some variations) such as office allocation for professors, NYC school choice system and Columbia and Harvard housing allocation (Che and Kojima, 2010; Kojima and Manea, 2010; Pathak and Sethuraman, 2011), just to name a few.

Both mechanisms compute a Pareto stable matching which is also (one-sided) group stable. Note that Pareto stability and one-sided group stability cannot coexist in general. However, with one-sided homogeneous preferences, we show that Pareto stability implies group stability for the opposite side. This additional property of group stability is attractive. For instance, in the application of course allocation, student-sided group stability eliminates incentives for post-allocation swaps among the students since there is no group of students that each can strictly improve his/her matchings by swapping courses among themselves. This property further guarantees the efficiency of a Pareto stable matching and improves the welfare of the students.

With the two competing Pareto stable matching mechanisms, our next question is: What are the comparative advantages of the two Pareto stable matching mechanisms? Which mechanism is better? An important property of a market mechanism is strategyproofness which makes it strategically simple for the market participants. The Pareto-improving dictatorship mechanism can be implemented in a truthful manner, which makes it a dominant strategy for one side to reveal their true preferences. This significantly simplifies the strategic considerations for the agents on one side of the market. The property of one-sided strategyproofness is intuitive in that the Pareto-improving dictatorship mechanism is built on the random serial dictatorship mechanism with augmenting paths and cycles elimination after each allocation. Despite its attractive strategic properties, the Pareto-improving dicta-

¹We will refer to them as the “draft mechanism” and the “dictatorship mechanism” for short when it is clear from the context.

torship mechanism brings out potential fairness issues by giving too much priority to the lucky agents who get a high random order while callously disregarding the preferences of those who belong to the same priority group but receive a low random order. As a result, high priority agents may get all their most preferred choices while the low priority ones may end up with nothing. Therefore, there is a tradeoff between non-callousness and strategyproofness of the Pareto-improving draft and dictatorship mechanisms.

The remainder of this chapter is organized as follows. Section 3.2 defines the model with one-sided homogeneous preferences. Section 3.3 discusses the solution concepts. Two new competing Pareto stable mechanisms are presented and compared in Section 3.4. We conclude in Section 3.5.

3.2 Model

We take the course allocation at universities as an example to describe the many-to-many matching model with one-sided homogeneous preferences while it can be applied to various cases with one side of the market having homogeneous preferences over the other side, in particular many resources allocation problems. The market is represented by a tuple $(\mathcal{S}, \mathcal{C}, (q_s)_{s \in \mathcal{S}}, (q_c)_{c \in \mathcal{C}}, \succsim_{\mathcal{S}}, \succsim_{\mathcal{C}})$, consisting of:

- a finite set of students: $\mathcal{S} = \{s_1, s_2, \dots, s_n\}$;
- a finite set of courses: $\mathcal{C} = \{c_1, c_2, \dots, c_m\}$;
- two **capacity** vectors $(q_s)_{s \in \mathcal{S}}$ and $(q_c)_{c \in \mathcal{C}}$ such that $q_s \in \mathbb{N}$ and $q_c \in \mathbb{N}$ denote the quotas of student s and course c , respectively;
- a **preference profile** for students over individual courses $\succsim_{\mathcal{S}} = (\succsim_s)_{s \in \mathcal{S}}$ such that $\succsim_s$ is a preference relation (possibly with ties) over individual courses and remaining unmatched. When such a relation is extended over group of courses it satisfies the following two properties known as **responsiveness** (Roth, 1985):²

- whenever $c, c' \in \mathcal{C}$ and $C \subseteq \mathcal{C} \setminus \{c, c'\}$, $C \cup \{c\} \succsim C \cup \{c'\}$ if and only if $c \succsim c'$,

²By an abuse of notation, we will denote a singleton without $\{\}$.

- whenever $c \in \mathcal{C}$ and $C \subseteq \mathcal{C} \setminus c$, $C \cup \{c\} \succsim C$ if and only if $c \succsim \emptyset$, which denotes the remaining unmatched option;
- a preference profile for courses over individual students $\succsim_{\mathcal{C}} = (\succsim_c)_{c \in \mathcal{C}}$ such that $\succsim_c$ is a preference relation (possibly nonstrict) over individual students and remaining unmatched. We have $\succsim_c = \succsim_{c'} \forall c, c' \in \mathcal{C}$, indicating that courses have **homogeneous preferences**. Similarly, the preference relation is extended over group of students by the responsiveness property:
 - whenever $s, s' \in \mathcal{S}$ and $S \subseteq \mathcal{S} \setminus \{s, s'\}$, $S \cup s \succsim S \cup s'$ if and only if $s \succsim s'$,
 - whenever $s \in \mathcal{S}$ and $S \subseteq \mathcal{S} \setminus s$, $S \cup s \succsim S$ if and only if $s \succsim \emptyset$, which denotes the remaining unmatched option.

Note that the market is defined similarly as the generalized many-to-many matching market with ties in Chapter 2 expect that the courses now have homogeneous preferences over the students.

A **matching** is the outcome of a market and is defined by a function $\mu : \mathcal{S} \cup \mathcal{C} \rightarrow 2^{\mathcal{S}} \cup 2^{\mathcal{C}}$ such that for each student $s \in \mathcal{S}$, $\mu(s) \in 2^{\mathcal{C}}$ with $|\mu(s)| \leq q_s$, for each course $c \in \mathcal{C}$, $\mu(c) \in 2^{\mathcal{S}}$ with $|\mu(c)| \leq q_c$, and $c \in \mu(s)$ if and only if $s \in \mu(c)$. A (deterministic direct) **mechanism** selects a matching for each market.

3.3 Solution Concepts

The central solution concepts of (pairwise) stability, Pareto efficiency and Pareto stability discussed in Chapter 2 are still relevant in this problem with one-sided homogeneous preferences and they can be defined in the same way as in Chapter 2. While these solution concepts aim to balance the interests of agents on both sides of the market, in some matching markets the interests of one side of the market is more important. This can be due to either the other side of the market does not truly have a preference or the objective is to maximize welfare of one side of the market. For instance, in the example of student-course matching, it is more critical to ensure that the matching outcome is for the benefits of all the students while the “preferences” of the courses can be compromised in some sense. In this section, we further propose solution concepts which address welfare of one side of the market.

3.3.1 One-Sided Pareto Efficiency

In the definition of Pareto efficiency (see Definition 2.3.5 in Chapter 2), we consider the social welfare of both sides of the market. A related notion is **one-sided Pareto efficiency**, which only considers the social welfare of one side of the market. In the general many-to-many matching markets and the application of course allocation, both definitions are reasonable and may find their applications. In the example of course allocation, sometimes the priorities of courses are set to mainly reflect students' need to take the courses. In such a case, student-sided Pareto efficiency may better capture social efficiency. However, in the applications where the preferences of courses are set by individual departments or lecturers, two-sided Pareto efficiency might be a better solution concept. Formally, student-sided Pareto efficiency is defined as following.

Definition 3.3.1 (Student-Sided Pareto Efficiency). *Fix a market $(\mathcal{S}, \mathcal{C}, (q_s)_{s \in \mathcal{S}}, (q_c)_{c \in \mathcal{C}}, \succsim_{\mathcal{S}}, \succsim_{\mathcal{C}})$. Given a feasible matching $\mu = (\mu_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$, we say that $\mu' = (\mu'_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$ is a student-sided Pareto improvement of μ if for all $s \in \mathcal{S}$, $\mu'(s) \succsim_s \mu(s)$ and $\mu'(s) \succ \mu(s)$ for at least one s . A matching μ is student-sided Pareto efficient if it does not have any student-sided Pareto improvement.*

3.3.2 One-Sided Group Stability

The notion of (pairwise) stability captures fairness in terms of a student-course pair. However, students may still end up with an unsatisfactory matching in the simple one-to-one matching market with strict preferences. Consider the following example.

Example 3.3.2. *There are two students $\mathcal{S} = \{s_1, s_2\}$ and two courses $\mathcal{C} = \{c_1, c_2\}$ each with $q_x = 1$ for all $x \in \mathcal{S} \cup \mathcal{C}$. Their preference profiles are given as:*

$$\begin{array}{ll} s_1 : c_2 \succ c_1 & c_1 : s_1 \succ s_2 \\ s_2 : c_1 \sim c_2 & c_2 : s_2 \succ s_1 \end{array}$$

Consider the following two matchings:

$$\mu_1 = \begin{pmatrix} s_1 & s_2 \\ c_1 & c_2 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} s_1 & s_2 \\ c_2 & c_1 \end{pmatrix}$$

μ_1 is both stable and Pareto efficient. However, since s_1 prefers s_2 's assignment and s_2 prefers s_1 's assignment, they can swap courses with each other, resulting in μ_2 . Although the swapping is not a Pareto improvement (when taking courses' welfare into consideration), it improves both students' welfare. Therefore, when students' welfare is our main concern, μ_2 is more desirable.

To capture the issue illustrated by the above example, we define a property called **(one-sided) group stability**. If a matching is (student-sided) group stable, there is no group of students such that each of them can be strictly better off through reassignment within the group. The formal definition is given as follows.

Definition 3.3.3 (Group stability). *Fix a market $(\mathcal{S}, \mathcal{C}, (q_s)_{s \in \mathcal{S}}, (q_c)_{c \in \mathcal{C}}, \succsim_{\mathcal{S}}, \succsim_{\mathcal{C}})$. Given a feasible matching $\mu = (\mu_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$, we say that μ is blocked by a coalition $S \subseteq \mathcal{S}$ if all members in S are able to get a strictly better matching by swapping matchings within S . Formally, consider the submarket formed by S and $\mathcal{C}$ with capacity $q_c - |\{s \in \mathcal{S} \setminus S \mid c \in \mu(s)\}|$ for each $c \in \mathcal{C}$, there is a feasible matching μ' such that $\mu'(s) \succ_s \mu(s)$ for all $s \in S$. We say a matching is (student-sided) group stable if it is not blocked by any coalition.*

In the definition above, the capacity of each $c \in \mathcal{C}$ is the remaining capacity of c for S . If a matching is group stable, then there is no way for any subset of students to swap courses within the group to improve everyone's matching. In other words, there always exists a student in the coalition whose assignment cannot be improved. Then this student has no incentive to form the coalition and the original matching is group stable.

In a many-to-one matching market, it is well known that group stability is equivalent to pairwise stability under responsive preferences.³ Note that the notion of group stability defined in Roth and Sotomayor (1990) considers the improvement of both sides. However, such equivalence result does not hold for one-sided group stability even in one-to-one matching, as illustrated by Example 3.3.2. More generally, we have the following impossibility result.

Claim 3.3.1. *There is an instance in which (pairwise) stability and (one-sided) group stability cannot hold simultaneously even if all individuals have unit capacity.*

³Refer to Lemma 5.5 in Roth and Sotomayor (1990).

Note that if a matching is student-sided Pareto efficient, it is also student-sided group stable. Thus, the above claim also implies that (pairwise) stability and (one-sided) Pareto efficiency cannot coexist in general. This fact is also illustrated by the following example adapted from Roth and Sotomayor (1990).

Example 3.3.4. *There are three students $\mathcal{S} = \{s_1, s_2, s_3\}$ and three courses $\mathcal{C} = \{c_1, c_2, c_3\}$ each with $q_x = 1$ for all $x \in \mathcal{S} \cup \mathcal{C}$. Their preference profiles are given as:*

$$\begin{array}{ll} s_1 : c_2 \succ c_1 \succ c_3 & c_1 : s_1 \succ s_2 \succ s_3 \\ s_2 : c_1 \succ c_2 \succ c_3 & c_2 : s_3 \succ s_2 \succ s_1 \\ s_3 : c_1 \succ c_2 \succ c_3 & c_3 : s_1 \succ s_2 \succ s_3 \end{array}$$

Consider the following stable matching:

$$\mu = \begin{pmatrix} s_1 & s_2 & s_3 \\ c_1 & c_3 & c_2 \end{pmatrix}$$

This is in fact the student-optimal stable matching. Consider the welfare of students only, there is another matching which leaves s_2 no worse off while benefits both s_1 and s_3 :

$$\mu' = \begin{pmatrix} s_1 & s_2 & s_3 \\ c_2 & c_3 & c_1 \end{pmatrix}$$

So μ is not student-sided Pareto efficient (group stable). Since μ is the student-optimal stable matching, then μ' is a student-sided Pareto improvement for any stable matching of this instance. In general there are some matchings that all students like at least as well as the stable matching, and that some students prefer.

In markets with one-sided homogeneous preferences, for instance, the course allocation application, the two stability notions can coexist. Furthermore, in such a framework, one-sided Pareto efficiency implies two-sided Pareto efficiency.⁴

Proposition 3.3.5. *In a many-to-many matching market, if one side of the market (e.g., courses) have homogeneous preferences, then a Pareto stable matching is group stable for the other side (e.g., students).*

⁴See more discussions in Sections 3.4.1 and 3.4.2.

The property of group stability is important and attractive in designing Pareto stable course allocation mechanisms, because it eliminates post-allocation exchanges among the students. The above proposition implies that Pareto stable matching mechanisms are always free from such post-allocation exchanges, therefore they can guarantee further efficiency and improve the welfare of students in terms of saving their effort in swapping courses with each other after the centralized matching outcome.

3.4 Pareto Stable Matching Mechanisms

Even in this specific type of market with one-sided homogeneous preferences, Pareto improvement does not necessarily preserve stability, as shown in Example 2.4.1 in Chapter 2. Nevertheless, the PARETO-STABLE algorithm presented in Chapter 2 still computes a Pareto stable matching for markets with one-sided homogeneous preferences as they are special cases. In this section, based on the homogeneity property of preferences, we present two new Pareto stable matching mechanisms which are easier to implement in practice.

We first characterize the relation between one-sided Pareto efficient matching and one-sided augmenting path and cycle in the following lemma. The proof in Erdil and Ergin (2008) for the many-to-one matching problem remains valid in the many-to-many matching framework with one-sided homogeneous preferences.

Lemma 3.4.1. *A feasible matching is student-sided Pareto efficient if and only if it has no student-sided augmenting path or cycle.*

The student-sided augmenting paths and cycles only consider the welfare of students, i.e., an augmenting path/cycle only requires that no students are worse off and at least one student is strictly better off. This is similar to the definition in Erdil and Ergin (2008) and slightly different from the ones defined earlier in Section 2.4.1 in which welfare takes both sides of the market into account. Formally, they are defined as following.

Definition 3.4.2 (Student-sided Augmenting Path). *Fix a market $(\mathcal{S}, \mathcal{C}, (q_s)_{s \in \mathcal{S}}, (q_c)_{c \in \mathcal{C}}, \succsim_{\mathcal{S}}, \succsim_{\mathcal{C}})$. Given a feasible matching $\mu = (\mu_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$, we say that $[s_0, c_1, s_1, \dots, c_\ell, s_\ell, c_{\ell+1}]$*

is an student-sided augmenting path if (i) $\sum_c \mu_{s_0 c} < q_{s_0}$ and $\sum_s \mu_{s c_{\ell+1}} < q_{c_{\ell+1}}$, (ii) $\mu_{s_k c_k} = 1$ and $\mu_{s_{k-1} c_k} = 0$ for all k , and (iii) $c_{k+1} \succ_{s_k} c_k$.

Definition 3.4.3 (Student-sided Augmenting Cycle). *Fix a market $(\mathcal{S}, \mathcal{C}, (q_s)_{s \in \mathcal{S}}, (q_c)_{c \in \mathcal{C}}, \succ_{\mathcal{S}}, \succ_{\mathcal{C}})$. Given a feasible matching $\mu = (\mu_{sc})_{s \in \mathcal{S}, c \in \mathcal{C}}$, we say that $[s_1, c_2, s_2, \dots, c_\ell, s_\ell, c_1, s_1]$ is an student-sided augmenting cycle if (i) $\mu_{s_k c_k} = 1$ and $\mu_{s_k c_{k+1}} = 0$ for all k (where $c_{\ell+1} = c_1$) (ii) $c_{k+1} \succ_{s_k} s_k$, and at least one of these preferences is strict.*

3.4.1 Pareto-Improving Draft Mechanism

In this subsection, we present a mechanism called the **Pareto-improving draft mechanism**, which incorporates the draft mechanism (Budish and Cantillon, 2012) with the student-sided augmenting paths/cycles elimination process after each allocation. The mechanism is described as follows (assume there are L priority levels).

PARETO-IMPROVING DRAFT MECHANISM

Initialization:

- Students have null virtual capacities.
- Courses have full capacities available.

For $\ell = 1, \dots, L$, consider students in the ℓ -th priority group:

- While there are students whose virtual capacities are smaller than their capacities, consider these students one by one in a random order (with respect to their considered virtual capacities):
 1. Increase the student's virtual capacity by one unit.
 2. The student receives her most preferred course among the remaining available courses (breaking ties randomly).
 3. After the above assignment, consider all students who have been assigned courses (including those in higher priority groups) and all courses, eliminate student-sided augmenting paths/cycles until there is none left.

The Pareto-improving draft mechanism first considers students according to their priorities, and among students who are in the same priority group, assigns one course

at a time over a series of rounds, according to the students' preferences. With ties in students' preferences and in the courses' preferences, we further perform a series of augmenting paths/cycles eliminations after each allocation to derive Pareto efficiency. Specifically, in the mechanism, we use student-sided augmenting paths/cycles elimination to ensure that each considered student, among all feasible matchings, is matched with her best possible assignment while not hurting any of the previously matched students. By the rule of the mechanism and the one-sided Pareto efficiency characterization, the mechanism actually returns a student-sided Pareto efficient matching. Furthermore, the following theorem says that the matching is Pareto stable and group stable for the students.

Theorem 3.4.4. *The Pareto-improving draft mechanism outputs a Pareto stable matching, which is also group stable for the students.*

3.4.2 Pareto-Improving Dictatorship Mechanism

Another important aspect in determining a matching is strategic considerations. We use the standard definition of **strategyproofness** in the literature, that is, a mechanism is strategyproof if truth-telling is a dominant strategy for all agents. While it is well known that there is one-sided strategyproofness for one-to-one and many-to-one matching models (Roth and Sotomayor, 1990), the strategyproofness only applies to the side with unit capacity, i.e., the “one” side. Hence, in the general many-to-many matching model, we cannot expect to have a one-sided strategyproof mechanism that always generates a stable matching. In general, when there are ties, no Pareto stable matching mechanism is one-sided strategyproof, even for one-to-one matching markets, as shown in the following example adapted from Erdil and Ergin (2008)).

Example 3.4.5. *There are three students $\mathcal{S} = \{s_1, s_2, s_3\}$ and three courses $\mathcal{C} = \{c_1, c_2, c_3\}$ with $q_x = 1$ for all $x \in \mathcal{S} \cup \mathcal{C}$. Their preference profiles are given as:*

$$\begin{array}{ll} s_1 : c_2 \succ c_3 \succ c_1 & c_1 : s_1 \succ s_2 \succ s_3 \\ s_2 : c_2 \succ c_3 \succ c_1 & c_2 : s_3 \succ s_1 \sim s_2 \\ s_3 : c_1 \succ c_2 \succ c_3 & c_3 : s_3 \succ s_2 \succ s_1 \end{array}$$

There are two Pareto stable matchings:

$$\mu_1 = \begin{pmatrix} s_1 & s_2 & s_3 \\ c_2 & c_3 & c_1 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} s_1 & s_2 & s_3 \\ c_3 & c_2 & c_1 \end{pmatrix}$$

If the Pareto-improving draft mechanism outputs μ_1 , s_2 has an incentive to misrepresent his/her preference and submit $c_2 \succ c_1 \succ c_3$. Then the mechanism will always output μ_2 and s_2 is better off. On the other hand, if the mechanism outputs μ_2 , then s_1 has an incentive to misrepresent his/her preference and submit $c_2 \succ c_1 \succ c_3$. Therefore, there is no Pareto stable matching mechanism that is strategyproof for the students.

In the less generalized environment with homogeneous preferences, it is natural to ask whether there is a Pareto stable mechanism that is one-sided strategyproof. Unfortunately, the Pareto-improving draft mechanism proposed in Section 3.4.1 is not strategyproof for the students, as the following example shows.

Example 3.4.6. There are two students $\mathcal{S} = \{s_1, s_2\}$ and three courses $\mathcal{C} = \{c_1, c_2, c_3\}$ with $q_{s_1} = q_{s_2} = 2$ and $q_c = 1$ for all $c \in \mathcal{C}$. Their preference profiles are given as:

$$\begin{aligned} s_1 : c_1 &\sim c_2 & c_1 : s_1 &\sim s_2 \\ s_2 : c_3 &\succ c_2 & c_2 : s_1 &\sim s_2 \\ & & c_3 : s_1 &\sim s_2 \end{aligned}$$

There are two Pareto stable matchings:

$$\mu_1 = \begin{pmatrix} s_1 & s_2 \\ c_1, c_2 & c_3 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} s_1 & s_2 \\ c_1 & c_2, c_3 \end{pmatrix}$$

If the Pareto-improving draft mechanism outputs μ_2 , s_1 can misreport his preference to be $c_2 \succ_{s_1} c_1$. Then the mechanism will always output μ_1 (although μ_2 is still a Pareto stable matching with respect to the true preferences) and s_1 is better off. Similarly, if the mechanism outputs μ_1 , then s_2 can benefit by manipulation ($c_2 \succ_{s_2} c_3$).

It is intuitively sensible that the Pareto-improving draft mechanism is not strategyproof for the students. The mechanism allocates courses to students one-at-a-time.

By strategically ranking an over-demanded course (c_2 in the above example) first, a student can secure a position in this course in the initial round and later get the other (more) preferred courses which are not over-demanded (c_1 for s_1 and c_3 for s_3 in the above example).

Nonetheless, with homogeneous preferences, a one-sided strategyproof Pareto stable matching mechanism does exist. An alternative Pareto stable mechanism, called the **Pareto-improving dictatorship mechanism**, combines the random serial dictatorship mechanism and student-sided augmenting paths/cycles elimination after each allocation. We next describe the mechanism (assume there are L priority levels for all students).

PARETO-IMPROVING DICTATORSHIP MECHANISM

Initialization:

- Students have full capacities available.
- Courses have full capacities available.

For $\ell = 1, \dots, L$, consider students in the ℓ -th priority group:

- Consider all these students one by one in a random order:
 1. The student receives her most preferred courses among the remaining available courses (under capacity constraint and breaking ties arbitrarily).
 2. After the above assignments, consider all students who have been assigned courses (including those in higher priority groups) and all courses, eliminate student-sided augmenting paths/cycles until there is none left.

Similar to the draft mechanism, here we use student-sided augmenting paths/cycles elimination to derive Pareto efficiency. The key difference is that it starts with students having full capacities and assigns courses *all-at-once* while the draft mechanism starts with students having null virtual capacities and assigns courses to students *one-at-a-time*. The Pareto-improving dictatorship mechanism also returns a Pareto stable matching and it is strategyproof for the students.

Theorem 3.4.7. *The Pareto-improving dictatorship mechanism outputs a Pareto stable matching, which is also group stable for the students. Furthermore, it is strategyproof for the students.*

3.4.3 Draft versus Dictatorship

The Pareto-improving draft and dictatorship mechanisms both satisfy the aforementioned “good” properties including pairwise and group stability, and one-sided and two-sided Pareto efficiency. In addition, the Pareto-improving dictatorship mechanism is strategyproof for the students. Incentive compatibility has been an important condition in various market design problems. However, one should also note that constraints always come with costs. The cost with the dictatorship mechanisms is known as “**callousness**” as described in Budish and Cantillon (2012).

Definition 3.4.8 (Callous). *An anonymous mechanism is called callous if there exists $n \in \{2, \dots, m\}$ and a random priority draw, such that a strictly positive measure of students get their n th choosing time before another set of students get their $(n - 1)$ th choosing time.*

The following example illustrates how the draft mechanism can potentially solve the callousness issue by avoiding severely unfair allocations.

Example 3.4.9. *There are two students $\mathcal{S} = \{s_1, s_2\}$ and two courses $\mathcal{C} = \{c_1, c_2\}$ with $q_{s_1} = q_{s_2} = 2$ and $q_{c_1} = q_{c_2} = 1$. Their preference profiles are given as:*

$$\begin{array}{ll} s_1 : c_1 \succ c_2 & c_1 : s_1 \sim s_2 \\ s_2 : c_2 \succ c_1 & c_2 : s_1 \sim s_2 \end{array}$$

The Pareto-improving dictatorship mechanism assigns both courses to either student who has a higher ranking after the initial random tie breaking, which will be either one of the followings:

$$\mu_1 = \begin{pmatrix} s_1 & s_2 \\ c_1, c_2 & \emptyset \end{pmatrix} \quad \mu_2 = \begin{pmatrix} s_1 & s_2 \\ \emptyset & c_1, c_2 \end{pmatrix}$$

But a fairer allocation would be to allocate each student one course:

$$\mu_3 = \begin{pmatrix} s_1 & s_2 \\ c_1 & c_2 \end{pmatrix}$$

which is exactly the output of the draft mechanism.

As a variant of the dictatorship mechanisms, the Pareto-improving dictatorship mechanism brings up callousness issue, which can be bad for welfare. Although the Pareto-improving draft mechanism solves the callousness issue, it is not strategyproof for students. This tradeoff between non-callousness and strategyproofness makes neither of them perfect for the course allocation problem. In Chapter 4, we use simulations based on real data to compare the two competing mechanisms to get a sense of which is better, if not perfect.

Budish and Cantillon (2012) have documented for the elective course allocation at Harvard Business School (HBS) under strict preferences, the callousness behavior harms efficiency in the sense that the welfare costs of using strategyproof dictatorship are much larger than the welfare costs of manipulability. As a consequence, Budish (2012) suggests the need for second-best alternatives to strategyproofness, for instance, strategyproofness in large markets.

3.4.4 Some Remarks on Strategyproofness in Large

Many stable matching mechanisms work quite well in practice even though theoretically they are not strategyproof and can be manipulated in various ways, such as, manipulation via pre-arranged matches, manipulation via preference lists and manipulation via capacities. One important theoretical support for such phenomena is that many markets of interests can be modeled as large markets.

In two-sided matching markets, several studies have analyzed the strategyproofness of matching mechanisms in large markets. The well-known Gale-Shapley Deferred Acceptance Algorithm becomes increasingly hard to manipulate as the number of participants increases, see for instance, Roth and Peranson (1999), Immorlica and Mahdian (2005) and Kojima and Pathak (2009). In particular, Kojima and Pathak (2009) examine the scope of manipulations of the student-optimal stable matching mechanism in the many-to-one matching market of students and colleges, where students have fixed length preferences. They find that the expected percentage of colleges that have incentives to manipulate their preferences converges to zero as the number of colleges approaches infinity, while the length of the preference lists

remains the same .

Strategyproofness in large has received much attention in other areas of economics besides two-sided matching as well. Most of these studies show either that the gains from misrepresenting one's preferences converges to zero or that the equilibrium behavior converges to truth-telling. For example, Roberts and Postlewaite (1976) show that in a pure exchange economy, the Walrasian mechanism becomes increasingly difficult to manipulate as the market becomes large under some regularity conditions. Similarly, in the context of double auctions, several studies have shown that the equilibrium behavior converges to truth-telling as the number of traders increases under various informational structures (see Cripps and Swinkels (2006) and Fudenberg et al. (2007) for example). A recent study by Kojima and Manea (2010) on the probabilistic serial mechanism (Bogomolnaia and Moulin, 2001) for the random assignment problem establishes a stronger result that truth-telling is an exactly dominant strategy in a finitely large market.

In practical applications such as course allocation, it usually involves a large amount of students and courses. Motivated by previous studies on strategic behavior in large markets, it would be an interesting future direction to explore the incentive properties of large many-to-many matching markets with ties. In large markets such as course allocation, the lack of strategyproofness of the Pareto-improving draft mechanism could be a potentially minor issue since students need to acquire a large amount of information in order to benefit from manipulation.

3.5 Conclusion

In this chapter, we examine a practical many-to-many matching market design problem where one side of the market has homogeneous preferences, which captures many real-life assignment problems such as the course-student matching market in many universities. We show that in this specific type of market, Pareto stable matchings have additional nice properties such as group stability. Our novel contributions to the practical market design literature include two new competing Pareto stable matching mechanisms, known as the Pareto-improving draft and dictatorship

mechanisms. The Pareto-improving dictatorship mechanism is strategyproof for the students, however, it is callousness by giving too much priority to the students with high ranks. These two mechanisms are easy to implement in practice and can be potentially applied to many real-life market design problems.

3.6 Appendix

This appendix presents the omitted proofs in this chapter.

3.6.1 Proofs

Proof of Claim 3.3.1

Proof. This can be seen by the following example. There are three students $\mathcal{S} = \{s_1, s_2, s_3\}$ and two courses $\mathcal{C} = \{c_1, c_2\}$ with $q_x = 1$ for all $x \in \mathcal{S} \cup \mathcal{C}$. Their preference profiles are given as:

$$\begin{aligned} s_1 : c_1 \succ c_2 & \quad c_1 : s_2 \succ s_3 \succ s_1 \\ s_2 : c_2 \succ c_1 & \quad c_2 : s_1 \succ s_2 \\ s_3 : c_1 & \end{aligned}$$

Consider the following four allocations:

$$\begin{aligned} \mu_1 &= \begin{pmatrix} s_1 & s_2 & s_3 \\ c_2 & c_1 & \emptyset \end{pmatrix} & \mu_2 &= \begin{pmatrix} s_1 & s_2 & s_3 \\ c_1 & c_2 & \emptyset \end{pmatrix} \\ \mu_3 &= \begin{pmatrix} s_1 & s_2 & s_3 \\ \emptyset & c_2 & c_1 \end{pmatrix} & \mu_4 &= \begin{pmatrix} s_1 & s_2 & s_3 \\ c_2 & \emptyset & c_1 \end{pmatrix} \end{aligned}$$

μ_1 is pairwise stable. However, it can be seen that $\{s_1, s_2\}$ is not group stable as both of them can get better off by swapping the assigned courses, which results in μ_2 . In μ_2 , s_3 and c_1 form a blocking pair, which enforces the allocation to μ_3 . Now s_1 and c_2 form a blocking pair, which transforms the allocation to μ_4 , in which s_2 and c_1 are a blocking pair and the matching returns back to μ_1 . Note that another possible allocation is $(s_1, c_2), (s_3, c_1)$; in this case, s_2 and c_1 form a blocking pair and the allocation will again return to the first one. Hence, for the considered instance, it does not admit an allocation that is both pairwise stable and group stable. $\square$

Proof of Proposition 3.3.5

Proof. Assume without loss of generality that $\mathcal{C}$ has homogeneous preferences over S and a Pareto stable assignment $\mu = (\mu_{sc})$ is not S -side group stable. Then there is a subset $S \subseteq \mathcal{S}$ such that all members in S can strictly improve his allocation among reassignments inside S ; denote the resulting new matching by μ' . Let $S = \{s_1, s_2, \dots, s_k\}$, and denote the assignment of each agent $x \in \mathcal{S} \cup \mathcal{C}$ in μ and μ' by $\mu(x)$ and $\mu'(x)$, respectively. Since μ is dominated by μ' , we have $\mu'(s) \succ_s \mu(s)$ for all $s \in S$. Note that the assignments of all students that are not in S remain the same in μ and μ' .

Consider any student $s \in S$. Let $\mu(s) = \{c_1, c_2, \dots, c_{q_s}\}$ with $c_1 \succsim_s c_2 \succsim_s \dots \succsim_s c_{q_s}$ and $\mu'(s) = \{c'_1, c'_2, \dots, c'_{q_s}\}$ with $c'_1 \succsim_s c'_2 \succsim_s \dots \succsim_s c'_{q_s}$, where q_s is the capacity of s . We can insert a copy of $\emptyset$ if s is not fully matched in the two matchings. Then due to responsive preferences, we have $c'_i \succsim_s c_i$ for all i and at least one preference is strict.

As all courses have the same preference, we can assume without loss of generality that the preference of courses is complete over students (otherwise, those students who are unacceptable will never be matched in any feasible assignment). Consider the following two cases about the structure of the homogeneous preference of $\mathcal{C}$ over $\mathcal{S}$.

Case 1. Courses are indifferent among all the students in $S = \{s_1, s_2, \dots, s_k\}$. We consider the exclusive-or structure of the two matchings μ and μ' .

We first show that no course is worse off in μ' . Assume otherwise that there is a course c_1 who is worse off in μ' . Since c_1 is indifferent between all students in S , we know that her number of assignments in μ' is less than that in μ , i.e., $|\mu'(c_1)| < |\mu(c_1)|$. Hence, there must exist a student s_1 such that s_1 breaks up the matching with c_1 in μ and is matched to a new course c_2 in μ' in which $c_2 \succ_{s_1} c_1$ (due to responsive preferences). Next consider c_2 . If all students matched to c_2 in μ are already matched to her in μ' , i.e., $\mu(c_2) \subset \mu'(c_2)$, then we know that c_2 does not exhaust her capacity in μ . Since μ is a stable matching, we know that s_1 is fully matched in μ and weakly prefers all his

assigned partners to c_2 ; this implies, in particular, $c_2 =_{s_1} c_1$. Hence, as s_1 improves his assignment in μ' , there must exist another course c'_2 with $c'_2 \succ_{s_1} c_1$ such that $c'_2 \in \mu'(s_1) \setminus \mu(s_1)$ and c'_2 is fully matched in μ . For such a case, we switch the name of c_2 and c'_2 . Therefore, there is a student s_2 such that s_2 breaks up the matching with c_2 in μ and is matched to a new course c_3 in μ' in which $c_3 \succ_{s_2} c_2$.

We continue with the argument. As the number of students is finite, eventually we will have a loop: $c_\alpha, s_\alpha, c_{\alpha+1}, s_{\alpha+1}, \dots, c_\beta, s_\beta, c_\alpha$ in which s_i breaks the matching with c_i in μ and is matched to c_{i+1} in μ' for all i (where $c_{\beta+1} = c_\alpha$) and $c_{i+1} \succ_{s_i} c_i$. (Note that it is possible that $\alpha = 1$, i.e., the loop goes back to the first course c_1 .) If at least one of the preferences is strict, then we have a Pareto improvement among these agents, this contradicts the fact that μ is Pareto stable. Hence, all these preferences are tight; in such a case, we can actually still use the old matchings in μ (i.e., remove all (s_i, c_{i+1}) and add (s_i, c_i) in μ'). Then we can continue with the same analysis on the exclusive-or structure of the two matchings μ and μ' , and eventually derive a contradiction.

Therefore, we know that μ' is a Pareto improvement over μ (as all students in S are better off while no course and any other students are worse off). This leads to a contradiction to the assumption that $\mu = (\mu_{sc})$ is Pareto efficient.

Case 2. There is at least one strict preference over two students in S , say, without loss of generality, $s_2 \succ s_1$. Similar to the above argument, there is a course $c_3 \in \mu'(s_2) \setminus \mu(s_2)$ such that $c_3 \succ_{s_2} c$ for some $c \in \mu(s_2)$. As μ is a stable matching, c_3 must be fully matched in μ ; thus, there is a man s_3 with $s_3 \succ_{c_3} s_2$ (again, due stability of μ) who breaks the matching with c_3 in μ and is matched to a new course c_4 in μ' . As s_3 also improves his assignment in μ' , we can again use the same analysis as above to show that there is a Pareto improvement, which contradicts to the Pareto stability of μ (note that all courses c_α considered in the process weakly prefers s_α to $s_{\alpha-1}$).

Therefore, when $\mathcal{C}$ have homogeneous preferences over $\mathcal{S}$, a Pareto stable assignment must be $\mathcal{S}$ -side group stable. $\square$

Proof of Theorem 3.4.4

Proof. Denote the matching computed by the mechanism by μ . By the rule of eliminating student-sided augmenting paths/cycles in the mechanism and the characterization of Pareto efficiency Erdil and Ergin (2006), μ is immediately student-sided Pareto efficient. Then there is no (two-sided) augmenting path under the matching μ since no s (i.e., student) can be better off. We claim that there is no (two-sided) augmenting cycle either. Assume otherwise that there is an augmenting cycle $[s_1, c_2, s_2, \dots, c_\ell, s_\ell, c_1, s_1]$, where $\mu_{s_k c_k} = 1$ and $\mu_{s_k c_{k+1}} = 0$ for all k (where $c_{\ell+1} = c_1$). Then according to the definition and the fact that no s can get strict improvement, there must be some c getting better off; assume without loss of generality that c_1 gets better off, i.e., $s_\ell \succ s_1$. Due to the homogeneity property of all c 's preferences and the fact that no c gets worse off, we can derive that $s_1 \succsim s_2 \succsim s_3 \succsim \dots \succsim s_\ell \succ s_1$, which is a contradiction. Therefore, by Lemma 2.4.4, μ is (two-sided) Pareto efficient.

Next we show that μ is stable. The rule of the mechanism implies that for any two students s_1 and s_2 , if s_1 has a higher priority than s_2 , then s_1 does not envy any course assigned to s_2 (i.e., all courses assigned to s_1 are at least as good as any course assigned to s_2). Otherwise, when considering augmenting paths/cycles for the iteration of s_1 , we would match a better course to s_1 . Hence, if there is a blocking pair (s, c) , where c strictly prefers s to one of her assignments s' (note that s' cannot be $\emptyset$ due to augmenting paths/cycles elimination at the iteration of s), i.e., $s \succ s'$. By the rule of the mechanism, s must be in a higher priority level than s' ; and by above discussion, s does not envy any course assigned to s' , which contradicts the assumption that s and c are a blocking pair. Hence, the mechanism always generates a stable matching.

Now we have proved the matching outcome is Pareto stable, then the student-sided group stability directly follows from Proposition 3.3.5.

This completes the proof of the theorem. □

Proof of Theorem 3.4.7

Proof. The proofs of Pareto stability and student sided group stability are the same

as the one for Theorem 3.4.4.

To the end of the strategyproofness, it can be seen that for each considered student, among the remaining available courses, we allocate him/her the best possible courses. Thus, the student has no incentive to lie. In the later augmenting paths/cycles eliminations, while the assignment of the student can be changed, a simple but critical invariant holds: Given the assignments of all previously considered students, we always allocate the best possible courses to the student. Therefore, it is a dominant strategy for the student to submit his/her true preference. This completes the proof of the theorem. □

Chapter 4

Improving Efficiency in Course Allocation

4.1 Introduction

Course allocation is a classic many-to-many matching problem in which a set of courses are to be allocated to a set of students who have multi-unit demand. Allocating courses equitably and efficiently has proven to be a challenging market design problem and there has been little in the literature to address this problem due to a variety of difficulties. First, students do not have proprietary rights over courses; therefore, buying or selling courses through money transfer is strictly prohibited. Second, the use of an unauthorized computer program to gain unfair advantage over other students in securing courses is repugnant. Third, students may have preferences not only for individual courses but also for combinations of courses, and these preferences may include ties (i.e., indifferences); thus, they may exhibit complicated strategic behavior. Finally, because most applications involve a large number of students and courses and are under a strict deadline to produce a desirable allocation, an efficient computation is critical.

In practice, there are two main types of course allocation mechanisms employed in educational institutions.

- Preference-ranking mechanisms, e.g., the draft mechanism at Harvard Business School (HBS), in which students submit ordinal preferences for courses.

Budish and Cantillon (2012) show that the draft mechanism is manipulable in theory and manipulated in practice; but, interestingly, the mechanism outperforms the strategyproof alternative, which implies that strategyproofness has both benefits and costs. They further propose a proxy draft mechanism that is shown to generate better efficiency. Kominers et al. (2010) propose a new proxy mechanism that simplifies students' strategic decision by directly incorporating their manipulation strategy into the mechanism. The mechanism is Pareto efficient and resistant to strategic manipulations observed in the extant data. However, other unobserved manipulations may still exist.

- Bidding mechanisms, e.g., the mechanism at the Ross School of Business at the University of Michigan (UMBS), in which students bid for courses. In the UMBS mechanism, bids submitted by students play a dual role—to infer the preferences of both students and courses. These two roles can easily conflict and result in unnecessary efficiency loss. Sönmez and Ünver (2010) propose an alternative Gale-Shapley Pareto-dominant mechanism that asks students to submit their preferences for courses in addition to bids. The mechanism is confirmed to have superior efficiency in both field and laboratory studies by Krishna and Ünver (2008). However, the mechanism is not strategyproof, which can prompt additional concerns about efficiency loss.

Note that while the common objectives of course allocation mechanisms are efficiency and equity, the practice of using different mechanisms in different educational institutions indicates that the current mechanisms are neither well understood nor satisfactory.

In this chapter, we examine the elective course allocation system for around 13,000 undergraduate students at Nanyang Technological University (NTU) in Singapore and explore its potential improvement. In 2010, more than 45 percent of the courses offered are over-demanded.¹ The scarcity problem presents a challenge to allocate the over-demanded courses fairly and efficiently.

¹This number is a conservative estimate, because it is calculated based on the number of vacancies of each course and the number of students who rank the courses first in their preference lists.

In NTU's current mechanism, students submit strict preferences for individual courses in different categories and courses have predetermined preferences (with ties) for students, which are essentially the priorities of each student. A centralized mechanism then determines allocations by considering student-course pairs in an order based on the priority structure of course over students and students' submitted preferences. This mechanism runs separately for courses of different types. This is a preference-ranking mechanism and is similar to the Boston Student Assignment Mechanism (Abdulkadiroğlu and Sönmez, 2003).²

We first identify inefficiencies and then propose improvement to the current system. The NTU's mechanism does exhibit some nice properties. An allocation generated through this mechanism is both pairwise stable (i.e., there is no student-course blocking pair) and student-sided group stable (i.e., there is no group of students such that each of them can strictly improve his/her assignment by swapping courses among themselves). Note that the latter is a simple implication of the strict preferences of students and a homogeneity property exhibited by the students' priorities. However, the mechanism is not strategyproof in theory and is manipulated in practice. As shown in a field survey of over 1,200 students at NTU (see Figure 4.1), 19 percent indicate that they would manipulate their preferences.³ Furthermore, the practice of breaking ties arbitrarily at random (as courses' preferences for students have ties) may result in severe efficiency loss. A similar issue was recently addressed for many-to-one matchings (Erdil and Ergin, 2006, 2008; Abdulkadiroğlu et al., 2009). In addition, the practice of dividing a course's total number of vacancies into different types and then allocating courses of different types separately introduces additional constraints into the problem and therefore can cause efficiency loss as well.

A critical ingredient in NTU's mechanism is that students are enforced to submit strict preferences. Indeed, much of the literature on many-to-many matching markets assumes strict preferences. In practice, however, there are various matching markets in which agents are not able to strictly rank their prospective partners for a variety of reasons (e.g., incomplete information). Our survey has revealed that 76

²The details of the mechanism are deferred to Section 4.3.

³See Appendix 4.7.2 for a detailed description of the survey and the corresponding results.

1. 65% of the students are not satisfied with the current course allocation system.
2. 76% of the students are indifferent between two or more courses that they are interested in.
 - 75% of these students would like to have two or three levels in their preference structure, where they are indifferent between the courses in the same level.
3. 75% of the students are satisfied with five courses on preference lists, while the rest 25% would like to have longer preference lists.
4. 81% of the students submit courses to the system according to their true preferences.

Figure 4.1: *Survey Results for 1,200 Students at NTU.*

percent of the students would like to express ties in their preference lists as shown in Figure 4.1. Therefore, we propose a new design which allows students to express weak preferences, in the hope of satisfying more demand and increasing overall efficiency. On top of the survey statistics, another important motivation for allowing students to express weak preferences originates from the observation that indifference can arise even from strict preferences, in terms of the sequence of taking the bundle of one's preferred courses. We notice that a main characteristic that distinguishes the elective courses from the major courses is that there is no prerequisite requirement among the elective courses. Therefore, a student can register them at any stage of study program. For a student who has strict preferences among elective courses, she actually can be indifferent about the order in which the courses are taken. That is, she can be indifferent between taking her most preferred or second most preferred course in the current semester.⁴

⁴Since the priority of students are mainly determined by their study years and more senior students have higher priority, it is more difficult for a junior student to get her most preferred course if it is (highly) over-demanded. By planning "strategically" about the sequence of taking

The prospect of improving welfare by incentivizing agents to report weak preferences has recently been investigated by Fragiadakis and Troyan (2013) with lab experiments in a many-to-one matching framework. They assume that students have strict common ordinal preferences but different cardinal preferences, and try to improve welfare by incentivizing students to report goods they value similarly as indifferent. Their results suggest that the proposed mechanisms significantly improve welfare despite sacrificing strategyproofness.

By proposing to introduce ties into students' preferences and combine the allocation process of two types of courses, we have a many-to-many matching market that include ties on both sides of the market and one side of the market has homogeneous preferences. This type of market has been examined in Chapters 2 and 3. We have proposed the general PARETO-STABLE algorithm as well as two competing Pareto stable matching mechanisms to find a Pareto stable matching outcome. We therefore can apply these two mechanisms to the new design of NTU course allocation system to explore the potential efficiency and welfare gains.

We further quantify efficiency improvement from the proposed new design to the course allocation system as well as compare the two competing Pareto stable matching mechanisms in view of the callousness of the dictatorship mechanism. With the real course allocation data from three academic years at NTU, we run simulations in different environments by introducing ties in students' preferences using the draft and dictatorship mechanisms. Ties are introduced to the students' submitted strict preference lists based on the surveyed distribution of ties in a preference list.⁵ We compare the performance of the current mechanism and our proposed mechanisms based on some simple welfare measures, including the total number of allocations, the total number of unassigned students and the average rank of the courses in a student's allocated bundle. Our simulation results show that (1) the overall efficiency in terms of the total number of allocations and the total number of unassigned students can be significantly improved when allowing ties in students' preferences;

the bundle of the preferred courses, that is, expressing weak preferences over this bundle, she can have higher chances of getting at least one course in this bundle as compared to getting none of them in the current semester.

⁵See Appendix 4.7.2 for details.

(2) the draft mechanism outperforms the dictatorship mechanism in terms of the total number of allocations, the average rank and the total number of unassigned students, even though it is not strategyproof for the students; and (3) the overall efficiency in terms of total number of allocations can be improved by almost 12 percent with the dictatorship mechanism and ties introduced to students' preferences.

The remainder of this chapter is organized as follows. Section 4.2 defines the course allocation problem at NTU. Section 4.3 discusses the current NTU course allocation mechanism and its inefficiencies. Section 4.4 proposes a new design to the NTU system. Simulation setups and results are discussed in Section 4.5. We conclude this chapter in Section 4.6.

4.2 Course Allocation Problem

In course allocation, while it is natural to assume that students have preferences over courses, educational institutions usually decide a priority ranking of individual students for each course. For instance, a course may take prior considerations for final year graduating students who need the course to fulfill graduation requirements, or those who failed the course in the preceding semesters and need to take the course again in the current semester. The priority ranking can be considered as a course's implicit preference for students.

We focus on the elective course allocation system at NTU.⁶ There are two types of elective courses for general education requirement that are open to students from all departments: Prescribed Electives (PE) and Unrestricted Electives (UE). A course can fall into both types. The current course allocation system at NTU solicits preferences from both students and courses to decide allocations.

Students submit two separate *strict* and non-overlapping preference lists over individual courses to the system, with up to five courses for PE and UE, respec-

⁶The detailed description of the curriculum structure is presented in Appendix 4.7.1. Specifically, a student's curriculum structure consists of two parts: the major core courses required by each department and the elective courses for general education requirement that are open to students from all departments. While the demand for major core courses can be satisfied in most cases, this is not the case for elective courses because of the competition and scarcity issues.

tively. Students in the current system are only allowed to submit preferences lists over individual courses. From a practical point of view, with a large number of courses and respective schedules, specifying preferences over subsets may result in complicated comparisons of alternatives and lengthy preference lists, making the system inapplicable. By taking preference lists over individual courses as input, the current system implicitly assumes responsiveness in students' preferences. Besides, the elective curriculum is designed to minimize complementarities and substitutabilities. Complementarities may occur among courses with prerequisite relations, which cannot be registered in the same semester.⁷ As a result, within one semester the main concern of every student is over the set of individual courses. Therefore, the standard assumption of responsive preferences used in the matching models in Chapters 2 and 3 is justifiable for the students' preferences in the course allocation context. For courses in both types, a student can only put it in one of the lists. That is, she needs to pre-specify whether to take this course under PE or under UE. This introduces constraints in the matching in that students are not allowed to have preference rankings among the courses across different types. Furthermore, the practice of forcing students to submit strict preferences can be considered as a way of tie breaking, because a recent survey has revealed that 76 percent of students would like to express ties in their preference lists.

The implicit preferences of courses over individual students, which are essentially the priorities of the students, are formed according to the following hierarchy (from highest to lowest):

1.
 - Students with only PE courses remaining to fulfill, if the course is a PE type.
 - Students with only UE courses remaining to fulfill, if the course is a UE type.
 - Students with only PE or UE courses remaining to fulfill, if the course is both a PE and UE type.
2. Final year students.
3. Students of special programs (e.g., accelerated bachelor degree).

⁷For example, one must take the course "Microeconomics Principles" before being allowed to take the course "Industrial Organization".

4. Year 3 students.
5. Year 2 students.
6. Year 1 students.

With such a preference setting of the courses, some of the students are at the same level on the priority lists. Therefore, there are ties in the courses' preference lists. From the viewpoint of courses, they only care about the interests of individual students and do not necessarily have preferences over subsets of students. In particular, the preferences of courses for individual students are according to the students' need to take the courses, rather than their identities and scores. Therefore, responsiveness is also justifiable in the courses' preferences. In addition, it can be observed that all the courses have the same preferences for students and we say the courses have homogeneous preferences. In particular, all of the courses (PE or UE) can list all those students who have only PE or UE courses to fulfill in the first priority. This is because a student with only PE (UE) courses to fulfill will not submit any UE (PE) courses in her preference list. The individual rationality property of feasible matchings allows us to unify the preferences of courses.

Observation 4.2.1. *The preferences of courses over students are homogeneous.*⁸

Each course has a pre-specified capacity constraint due to resource limitations, for example, the size of a classroom. If a course is in both PE and UE, its capacity will be pre-divided for PE and UE, respectively, in an exogenous manner. Every student has a capacity constraint as well, which sets an upper bound on the number of courses registered within one semester. Specifically, except for final year graduating students, a student will be allocated, at most, one course for PE and one course for UE in the current system.

Given the preferences and capacity constraints of both students and courses, the system allocates courses to students using a mechanism, which runs for PE and UE separately and sequentially, first for PE, followed by UE.

⁸In practice, some courses may have special predetermined preferences for students (for example, the course "Economics of Manufacturing" takes prior consideration of students majoring in Material Sciences and Engineering). In such cases, some vacancies are reserved in advance by the corresponding departments, and hence the preferences of courses are still homogeneous in the system.

4.3 NTU's Course Allocation Mechanism

The NTU's current course allocation mechanism considers students in each priority level one by one. It runs separately and sequentially for PE and UE: first for PE, followed by UE. The NTU mechanism is described as follows (assume there are L priority levels).

NTU MECHANISM

For $\ell = 1, \dots, L$: consider students in the ℓ -th priority group.

- Consider the courses in their preference lists sequentially.

That is, for $k = 1, \dots, 5$,

- Consider all student-course pairs in which the student is at the considered priority level ℓ and the course is as her k -th choice, assign courses to students amongst these pairs subject to the capacity constraints, with ties in priorities broken randomly.

The mechanism is simple to implement and quite similar to the Boston Student Assignment Mechanism as described in Abdulkadiroğlu and Sönmez (2003). The main difference is that the Boston Student Assignment Mechanism runs simultaneously for all students instead of running separately for students in different priority groups.

The NTU's mechanism does exhibit some nice properties. A matching generated through this mechanism is stable and student-sided group stable. A stable outcome ensures a certain level of fairness among students in the sense that no blocking pairs can upset the structure of a matching. It creates a balance in the competition among students and their priorities in each course. However, a serious shortcoming of the Boston Student Assignment Mechanism and the NTU mechanism is that students with high priorities at specific schools lose their priorities unless they list these schools as their top choices.⁹ It gives very strong incentives to students to misrepresent their preferences by improving ranks of courses in which they have high priority.

⁹For instance, there are three students s_1, s_2, s_3 and two courses c_1, c_2 with unit capacity each; all students prefer c_1 to c_2 and the two courses are indifferent among the students. Assume we

This mechanism causes welfare loss for students who reveal their true preferences and whose first choices are over demanded. In such a case, if the student does not get his/her first choice, then he/she checks the second choice. Even if the second choice is not so popular, the chance of getting the second choice becomes smaller since some vacancies (or even all the vacancies) have been taken up in the first round. Therefore, with the current mechanism, it makes a big difference for a student to specify his/her first choice strategically. Sometimes, a student can become better off if he/she sacrifices the first choice in the true preference list and misrepresent the second choice as the first choice. This kind of manipulation over preferences can provide the student higher chances of getting his/her second choice. As a consequence, students are forced to play very complicated course registration games, and often, misrepresenting their preferences is their best interest. As a consequence, truth-telling is not a dominant strategy and students can easily manipulate their preferences. Indeed, in a recent field survey (see Figure 4.1) of NTU's current course allocation system, 19 percent indicated that they did not place course preferences truthfully. Manipulation is a key challenge to the mechanism, and may result in severe efficiency loss.

4.3.1 Inefficiency of the Mechanism

While the NTU mechanism guarantees fairness to some extent by considering the preferences of students and courses, there are a large number of students who are not assigned any course, which results in considerable efficiency loss. Table 4.1 gives a summary of course allocation statistics for the three academic years: 2010, 2011 and 2012. Percentage is measured as the total number of allocations over the total vacancies. Note that the allocation statistics were collected after certain manual adjustments, which have already corrected some of the inefficiency from the mechanism. Inefficient course allocations result in appeals and complaints from unsatisfied students, and can even cause deferral of graduation for final year students. Due to the inefficiency of the NTU mechanism, many unsatisfied students and program co-

break ties in the order of $s_1 \succ s_2 \succ s_3$; then s_3 does not get any assignment if behaving truthfully. However, if he manipulates his preference to $c_2 \succ c_1$, then he can get c_2 in the first round.

ordinators spend a tremendous amount of time and effort manually seeking for better matchings. Despite this effort, some students can still end up with an unfavorable outcome in which no course is registered.

Table 4.1: *Course Registration Statistics.*

	Course	Vacancies	Students	Allocations	Percentage
2010	PE	8,118	9,660	6,844	84.31
	UE	14,178	10,672	8,366	59.01
	PE+UE	22,296	13,471	15,210	68.22
2011	PE	10,092	8,546	6,919	68.56
	UE	15,342	8,836	9,128	59.50
	PE+UE	25,434	11,321	16,047	63.09
2012	PE	9,305	8,157	6,729	72.32
	UE	16,260	8,731	7,170	44.10
	PE+UE	25,565	11,541	13,899	54.37

Percentage is measured as the total number of allocations over the total vacancies. For instance, for the 2010 data, the PEs and UEs have 8,118 and 14,178 vacancies while 9,660 and 10,672 students demand them, respectively. The PEs are approximately 15.96% over demanded whereas the UEs are approximately 32.85% under demanded. Overall, the combination of PEs and UEs are 9.66% under demanded. The data indicate that the current systems resulted in an allocation of 6,844 PEs and 8,366 UEs. About 29.15 percent and 21.61 percent of the students who requested PEs and UEs failed to get any allocation, respectively. In total, there is over 25% efficiency loss. Note that in the 2011 data, the number of UE allocations is larger than that of students, because a considerable amount of students get more than one UE registered in the process of manual adjustments after the implementation of the mechanism.

4.4 A New Design

Survey statistics show that 65 percent of students find the current system unsatisfactory,¹⁰ indicating that better mechanisms should be designed to improve the welfare of the students. We propose a new design to the course allocation system trying to address the two main inefficiency issues identified.

First, in the survey we find that 76 percent of the students think that they should be allowed to indicate ties in their preferences. In practice, while students may have strict preferences for some courses, such strictness is not very sensitive in the sense that most students are usually only concerned with whether one of her desired courses is registered, but not exactly which one. Breaking ties arbitrarily at random (as courses' preferences have ties) and forcing students to submit strict preference lists (making the students to break ties in their preferences) may result in severe efficiency loss. As a consequence, we consider introducing ties into students' preferences, with the objective of reducing the flaws in the current system and improving overall efficiency and welfare of the students.

Second, students submit two separate preference lists, one for PE and one for UE.¹¹ The constraint of separating the allocation process of two types of courses introduces additional efficiency loss since students are not allowed to compare courses between different types. For a course's perspectives, it does not really matter whether a student takes it as a PE type or a UE type. As for a student, sometimes he/she may prefer to get two UEs than one UE and one PE because he/she has more UEs remaining to take. In the proposed new design, we therefore combine the allocation of these two types of courses and allow for comparisons among courses of different types.

With the proposed new design, we have formulated a many-to-many matching market with one-sided homogeneous preferences. Therefore, the two competing Pareto stable matching mechanisms proposed in Chapter 3 can be applied to com-

¹⁰See Appendix 4.7.2 for the details.

¹¹Our PARETO-STABLE algorithm continues to work for such a setup with such multiple preferences, as shown in Section 2.4.4 of Chapter 2. The NTU course allocation application corresponds to the case with two partitions (i.e., $\ell = 2$).

pute a Pareto stable outcome.¹²

4.5 Simulations

We examine the performance of the competing Pareto stable improving draft and dictatorship mechanisms with real course allocation data for three years (2010, 2011 and 2012) from NTU. In particular, we quantify the efficiency and welfare gains for students with the proposed new mechanisms as well as compare the two competing mechanisms, in terms of the total number of allocations, the total number of unassigned students and the average rank of the courses in a student's assigned bundle.

4.5.1 Data and Key Assumptions

The dataset consists of (a) the number of vacancies for each course ; (b) students' *strict* preferences over up to five PE and five UE courses, separately; and (c) course allocation results from NTU's current mechanism after the manual adjustment process (summarized in Table 4.1). The input information for our simulations is specified as follows:

- Courses' preferences: we assume that courses' homogeneous preferences for students are based only on the students' study years. That is, all courses strictly prefer final year students to penultimate year students, and so on. Courses are indifferent among students in the same study year. This assumption is for the simplicity of simulations.
- Courses' capacities: the above (a) provided by the dataset.
- Students' preferences: the above (b) provided by the dataset.¹³

¹²Note that the more generalized PARETO-STABLE algorithm proposed in Chapter 2 is also applicable. However, here we want to compare the performance of the two competing Pareto stable matching mechanisms using real course matching data.

¹³The current NTU mechanism is non-strategyproof and in the course allocation survey 19% of the students have revealed being strategic when submitting their preferences. In the simulations, we take the reported preferences of the students as their true preferences into the mechanisms for two reasons. First, it is almost implausible to obtain the true preferences of over 10,000 students

- Students' capacities: either one or two, see the specific setup in the simulations described below.

Notice that while ties are natural in courses' preferences, the submitted preferences of the students are forced to be strict in the current system. Before determining how to introduce ties, let us first take a closer look at the distribution of the lengths of students' preference lists, which are depicted in Figure 4.2. For both PE and UE, it can be seen that the length of preferences is almost symmetrically distributed, and the average lengths are 3.00 and 3.07, respectively. On average approximately 23% and 25% students list full five courses while around 25% and 24% students list only one course in their preference lists. In total, more than 60% students have preference lists of no more than three for both PE and UE. Ties will make no difference to preference lists with only one course. For preference lists no longer than three, a maximum of two ties can be introduced. Further, in the recent course registration survey (see Figure 4.1 Question 3), 35 percent of students preferred two levels of preferences and 23 percent preferred three levels, representing a major portion of all surveyed students and over 76 percent of whom preferred to have ties.¹⁴

In the simulations, ties are introduced to students' preferences based on the empirically observed distribution of the tie structures in students' preference profiles from the course allocation survey. Given the average lengths and survey information, in our simulations we consider two scenarios: two or three levels in students' preference lists.

- Two levels means that there is one strict ' $\succ$ ' and the preference list of a student is divided into two levels at random.

- Three levels means that there are two strict ' $\succ$ ' and the preference list of a student
-
- since some of them have already graduated when we conducted this study. Second, our main objective in the simulations is to globally examine the performance of our proposed mechanisms in terms of the number of matched student-course pairs while the cost of manipulation is not in our scope.

¹⁴We note that the current limit of a maximum of five courses for PE/UE preference lists does not impose a significant constraint against students indicating all of the courses that they like, as the survey results show that 75 percent of the students are satisfied with the upper limit of five courses (see Figure 4.1 Question 4).

Figure 4.2: *Distribution of the lengths of students' PE and UE Preference Lists.*

is divided into three levels at random.

Note that a student is indifferent between the courses in the same level. For example, if a student has preference $A \succ B \succ C \succ D \succ E$, then $A \sim B \succ C \sim D \sim E$ is a (random) realization with two levels and $A \succ B \sim C \succ D \sim E$ is a (random) realization with three levels.

Since the draft mechanism is not strategyproof for the students, the following comparisons are made under the assumption that the students report their true preferences when the draft mechanism is used. In practice, the non-strategyproofness of the draft mechanism could be a minor issue in a large course allocation system as in the aforementioned remarks on strategyproofness in large (see Section 3.4.4 in Chapter 3).

Next we describe the simulation environments and results. Due to the randomness involved in introducing ties into students' preferences, five independent computations are performed for each of the simulation environments, and the percentage is the average of the five experiments.¹⁵ For the rest of the paper, percentages are measured as the total number of allocations over the total number of vacancies, unless otherwise specified.

¹⁵The variations in the simulations are very small, therefore, five simulations are sufficient to demonstrate the results.

4.5.2 Simulation Environment I

In the first set of simulations, we examine the efficiency improvement from allowing ties in students' preferences while maintaining the other constraints (mainly in terms of students' capacities) in NTU's current system. We consider the following three scenarios for the first set of simulations.

1. Pure PE. We only consider PE courses. Each student has unit capacity for PE and we only consider her preference list for PE courses.
2. Pure UE. We only consider UE courses. Each student has unit capacity for UE and we only consider her preference list for UE courses. Note that the first two scenarios are essentially many-to-one matchings and the allocations for PE and UE courses are run separately and independently.
3. Combined PE+UE. We consider the PE and UE courses being pooled together. Note that if a course falls into both the PE and UE categories, in NTU's current system, the course's total capacity is manually divided into two parts for PE and UE, respectively. In our simulations, we merge the two separate capacities, making it a single capacity for each such course. Each student still has two separate preferences, one for PE and one for UE. The capacity of a student is in the format of $1 + 1$. In other words, a student can get at most one PE course and at most one UE course, and it is possible that a student is allocated two courses in total. Note that this scenario precisely captures the capacity requirements for the students in NTU's current system, and is the one by which we quantitatively evaluate the performance of our mechanisms when introducing ties into students' preferences.

In these three scenarios, students essentially have unit capacities.¹⁶ In such cases, one can notice that the draft and dictatorship mechanisms are equivalent. In practice, we run simulations using both mechanisms and the results do appear to be the same as expected, except for some small deviations due to the randomness nature of the mechanisms. Therefore, we do not distinguish these minor differences

¹⁶In the third scenario, a student with two separate preferences can be treated as two students, among whom one has preferences consisting of PE only and the other has preferences consisting of UE only.

caused purely by the randomness in the algorithms, and instead focus on the effects of allowing ties into students' preferences.

Table 4.2: *Simulation Results I: Total Number of Allocations*

	Course	Maximum matching	NTU's current mechanism	Pareto- Stable without ties	Pareto- Stable with 3 levels	Pareto- Stable with 2 levels
2010	PE	7,140	6,615	6,647	6,720	6,852
		87.95%	81.49%	81.88%	82.78%	84.41%
	UE	8,958	8,142	8,162	8,295	8,515
		63.18%	57.43%	57.57%	58.51%	60.06%
	PE+UE	16,536	14,990	15,105	15,312	15,670
		74.17%	67.23%	67.75%	68.68%	70.28%
2011	PE	7,725	7,045	7,150	7,342	7,368
		76.55%	69.81%	70.85%	72.75%	73.01%
	UE	8,041	7,495	7,537	7,626	7,784
		52.41%	48.85%	49.13%	49.71%	50.74%
	PE+UE	15,803	14,526	14,627	14,833	15,190
		62.13%	57.11%	57.51%	58.32%	59.72%
2012	PE	7,308	6,726	6,762	6,838	7,005
		78.54%	72.28%	72.67%	73.49%	75.28%
	UE	8,075	7,556	7,574	7,672	7,819
		49.66%	46.47%	46.58%	47.18%	48.09%
	PE+UE	15,412	14,283	14,350	14,530	14,855
		60.29%	55.87%	56.13%	56.84%	58.11%

Table 4.2 summarizes the simulation results in terms of the total number of allocations under four different mechanisms, including maximum matching, NTU's mechanism, Pareto-Stable without ties and Pareto-Stable with ties (2 levels and 3 levels).¹⁷ We first compute a maximum cardinality matching (i.e., one with the maximum number of assigned pairs) from the set of all mutually acceptable pairs, which gives a theoretical upper bound on the size of all feasible matchings. Note that the total number of students and course vacancies do not qualify, as no feasible matching can match the bound. For PE, UE and combined PE+UE, the size of a maximum matching in 2010 is 7,140, 8,958 and 16,536, which takes 87.95 percent, 63.18 percent and 74.17 percent of the total vacancies, respectively.¹⁸

In the second step, we run NTU's current mechanism with respect to the submitted preferences. Note that the statistics differ slightly from those in Table 4.1, because we consider slightly simplified preferences for the courses and do not include the manual adjustment process. In 2010, about 81.49 percent of PE vacancies, 57.43 percent of UE vacancies and 67.23 percent of the combined PE+UE vacancies are allocated. We will compare the performance of our mechanisms to these statistics.

In the third step, we employ the Pareto stable draft/dictatorship mechanism on the submitted preferences (with mandatory tie-breaking in students' preferences). In comparison with the allocation results using NTU's current mechanism, for 2010 we see a 0.39 percent improvement for PE, a 0.14 percent improvement for UE and a 0.52 percent improvement for combined PE+UE. Note that NTU's current mechanism also generates a Pareto stable matching with respect to the submitted strict students' preferences. The marginal improvement in the number of allocations comes from the differences in the implementations of the algorithms.

The last step is to employ the Pareto stable draft/dictatorship mechanism with ties introduced in students' preferences, which resemble the true preferences of the students if they are allowed to express weak preferences. In the column "Pareto-Stable with 3 levels", the draft/dictatorship mechanism is employed with three random levels on students' preferences. That is, each student's preference has at most

¹⁷It will be interesting to theoretically examine the improvement in the cardinality of matchings when ties are allowed. We leave this as future work.

¹⁸We will mainly explain the 2010 results, because the results are robust all over the three years.

three levels. Here we observe improvements of 1.29 percent, 1.08 percent and 1.45 percent for PE, UE and combined PE+UE, respectively, compared to NTU's current mechanism. With the 2 levels preferences, we observe improvements of 2.92 percent, 2.63 percent and 3.05 percent for PE, UE and combined PE+UE, respectively. These results suggest that introducing ties into students' preferences can improve the overall welfare of students.¹⁹

We also observe improvements in the total number of allocations after introducing ties into students' preferences for 2011 and 2012. In 2011, there are 3.20 percent, 1.89 percent and 2.61 percent improvements for PE, UE and combined PE+UE, respectively. The improvements for 2012 are 3.00 percent, 1.62 percent and 2.24 percent. In summary, for the combined PE+UE scenario, after introducing ties into students' preferences, over the three years we see an average improvement of 2.63 percent. This translates into roughly 639 more student-course matchings every year which, compared to the allocation results from NTU's current mechanism, significantly improves overall students' social welfare. The statistics of the total number of allocations for combined PE+UE are also depicted in Figure 4.3.

In addition to improvements in the total number of allocations, Table 4.3 reports the corresponding statistics on the number of unassigned students for combined PE+UE.²⁰ We observe a decrease in the total number of unassigned students of 2.87 percent, 2.18 percent and 2.42 percent for the three years, respectively, where unassigned students refer to those who are not allocated to any course. The statistics of the total number of unassigned students for combined PE+UE are also depicted in Figure 4.4.

¹⁹Note that for the PE and UE scenarios, students have unit capacity, the allocations are therefore many-to-one matchings. For the third step, the simulations are equivalent to Erdil and Ergin's algorithm Erdil and Ergin (2008), which computes a Pareto stable matching for many-to-one markets with *one-sided* indifference. For last step, the simulations are equivalent to Erdil and Ergin's algorithm Erdil and Ergin (2006), which computes a Pareto stable matching for many-to-one markets with *two-sided* indifference.

²⁰The percentages in this table are calculated as the total number of unassigned students over the total number of students.

Table 4.3: *Simulation Results I: Number of Unassigned Students for PE+UE*

	NTU's current mechanism	Pareto- Stable without ties	Pareto- Stable with 3 levels	Pareto- Stable with 2 levels
2010	1,866 13.85%	1,766 13.11%	1,679 12.46%	1,479 10.98%
2011	883 7.80%	859 7.59%	779 6.88%	636 5.62%
2012	1,007 8.73%	970 8.40%	885 7.67%	728 6.31%

4.5.3 Simulation Environment II

For the combined PE+UE scenario in the first set of simulations, we explicitly set the capacities of students to be $1 + 1$. This is in accordance with NTU's current system and helps us to compare the simulation results to the current mechanism in quantifying the efficiency loss from forcing students to submit strict preferences. In practice, however, students may get more than one course from a category (either PE or UE) after the manual adjustment period. This fact is illustrated by the statistics in Table 4.1, where the number of allocations for UE is larger than the number of students in 2011. Indeed, the $1 + 1$ capacity assumption is mainly for the purpose of a balanced and fair allocation as NTU's current mechanism considers students level by level according to the preferences of the courses, and is unnecessary in our mechanism as we consider all students at different levels simultaneously. In practice, the $1+1$ capacity is not a sharp constraint and is purely set to achieve certain equity in the outcome. This additional constraint however can result in significant inefficiency in course allocation. Therefore, in the following simulations we maintain an overall capacity of two for each student, but without any constraint on the types of the two courses. That is, a student can be allocated any two courses (either one PE and one UE, or two PEs, or two UEs) from her preference list. This

proposed change allows for practical situations where students may have strong preferences of taking courses from one category. For instance, consider a student with 1 PE and 5 UE courses to take in the next three semesters, then the student certainly prefers two UEs to one PE and one UE. An overall capacity of two is still maintained to ensure a balance of fairness among students in different priority levels, by avoiding allocating almost all of the courses to senior students while junior students receive no courses.

When students have multi-unit capacities, the draft and dictatorship mechanisms lead to different matching outcomes. The main source of such differences is that in the draft mechanism students are allocated courses to students one-at-a-time while in the dictatorship mechanism students are allocated courses all-at-once up to their capacities. On top of analyzing the efficiency improvement from introducing ties, we focus on the comparisons between the draft and dictatorship mechanisms.

Similar to the first set of simulations, we consider three scenarios: pure PE, pure UE, and combined PE+UE. Since students can get two courses from the same category and in practice they may have preferences for the two categories, we randomly concatenate a student's PE and UE preference lists with ties into one single list using one of the following rules for the combined PE+UE scenario, and students are uniformly distributed among the following three types.

1. If a student prefers PE to UE, the new single preference is the PE preference followed by the UE preference.
2. If a student prefers UE to PE, the new single preference is the UE preference followed by the PE preference.
3. If a student is indifferent between PE and UE, the new single preference is formed by randomly merging the PE and UE preferences, while preserving the same ordering for the PE and UE courses.

To introduce ties into students' preferences, for the PE and UE scenarios, the preference lists are randomly divided into two levels, where students are indifferent among the courses in the same level. For the combined PE+UE scenario, the concatenated preference lists are divided into four levels: one strict ' $\succ$ ' in PE's preference, one strict ' $\succ$ ' in UE's preference, and one strict ' $\succ$ ' in the concatenation of

the two preferences. If a student is indifferent between the two categories, the three strict ' $\succ$ ' are placed at random.

Table 4.4 summarizes the simulation results in terms of the total number of allocations from the draft and dictatorship mechanisms when students have an overall capacity of two and ties in preferences. The draft mechanism consistently outputs more matched student-course pairs when compared with the dictatorship mechanism.

Table 4.4: *Simulation Results II: Total Number of Allocations*

	2010		2011		2012	
Course	Draft	Dictatorship	Draft	Dictatorship	Draft	Dictatorship
PE	7,452	7,386	8,798	8,690	8,255	8,185
	91.80%	90.98%	87.18%	86.11%	88.72%	87.96%
UE	10,046	9,926	10,479	10,313	10,399	10,243
	70.86%	70.01%	68.30%	67.22%	63.95%	63.00%
PE+UE	17,354	16,979	17,803	17,473	17,308	17,015
	77.83%	76.15%	70.00%	68.70%	67.70%	66.56%

In addition to the total number of allocations, we compare the draft and dictatorship mechanisms by two other measures: the average rank²¹ and the total number of unassigned students. The statistics are presented in Tables 4.5 and 4.6, respectively. Note that Table 4.6 only refers to the PE+UE scenario and the percentages are calculated as the total number of unassigned students over the total number of students. We observe that the draft mechanism results in a lower average rank

²¹The idea of using average ranks as a simple measure of welfare is motivated by Budish and Cantillon (2012). To be precise, the average rank statistics here are calculated as the average rank of the courses in a student's assigned bundle based on the ranks in the *strict* preferences because it is impossible to get the true real preferences with ties. As a consequence, these average rank statistics are not the real average ranks. However, for the purpose of comparisons between two mechanisms, they are already sufficient.

and significantly lower percentage of unassigned students when compared with the dictatorship mechanism.

Table 4.5: *Simulation Results II: Average Rank*

	2010		2011		2012	
Course	Draft	Dictatorship	Draft	Dictatorship	Draft	Dictatorship
PE	1.73	1.93	1.72	1.88	1.64	1.81
UE	1.96	1.96	1.83	1.95	1.78	1.95
PE+UE	1.85	1.85	1.72	1.79	1.66	1.74

Table 4.6: *Simulation Results II: Number of Unassigned Students for PE+UE*

	2010		2011		2012	
PE+UE	Draft	Dictatorship	Draft	Dictatorship	Draft	Dictatorship
Number	682	3,104	357	1,241	450	1,518
Percentage	5.06%	23.04%	3.15%	10.96%	3.90%	13.15%

From all these statistics, we find that the draft mechanism outperforms the dictatorship mechanism in terms of the total number of allocations, the average rank and the total number of unassigned students. More interestingly, the dominant relationship is especially significant in terms of the number of unassigned students. With the draft mechanism, only 5.06 percent, 3.15 percent and 3.90 percent of the students are not assigned to any course for the three years, respectively, as compared to 23.04 percent, 10.96 percent and 13.15 percent with the dictatorship mechanism. These statistics quantify the negative effects on welfare from the callousness issues in the dictatorship mechanism.

Overall, if compared with the total number of allocations from NTU's current mechanism in Table 4.2, there is approximately 11.65 percent, 12.83 percent and

11.84 percent improvement in terms of the total number of allocations for the three years, respectively, with the dictatorship mechanism. These statistics are also illustrated in Figure 4.3. The corresponding number of unassigned students are summarized in Figure 4.4. As a result of its callousness nature, the dictatorship mechanism results in a dramatic increase in the total number of unassigned students, which is even much higher compared to the NTU's current mechanism. The draft mechanism outperforms the others with the smallest percentage of unassigned students.

Figure 4.3: *Summary of Simulation Results: Total Number of Allocations*

4.6 Conclusion

In this chapter, we study an empirical market design of course allocation, which can be modeled as a many-to-many matching market with one-sided homogeneous preferences. We first discuss the inefficiencies of the current course allocation mechanism at NTU and propose improvement to it. Using unique course matching data, we

Figure 4.4: *Summary of Simulation Results: Percentage of Unassigned Students*

quantify and compare the welfare and efficiency gains from my proposed mechanisms compared to the existing mechanism. My results immediately suggest that the commonly used practices of tie-breaking in preferences can cause significant welfare loss and therefore ties should be taken into account in practical market designs. My proposed mechanisms resolve the inefficiency from tie-breaking in two-sided matching markets and can potentially be extended to other types of markets.

4.7 Appendix

4.7.1 NTU Curriculum Structure and Course Registration Process

This appendix introduces the curriculum structure and course registration process of NTU. At NTU, an undergraduate needs to fulfill both the Major Requirement and the General Education Requirement (GER). The Major Requirement includes major core courses, which are compulsory courses to satisfy the program requirements, and

major prescribed electives, which are courses for specialization in a particular degree program. The GER is the curriculum requirement for broadening study, which covers key fields of knowledge for all students. It constitutes about 25 percent to 40 percent of the total curriculum workload and is divided into 3 classes of studies:

1. GER CORE: these include courses related to Human Resources Management, Communication Skills and Singapore Studies.
2. GER Prescribed Electives (PE): the courses represent the key fields of knowledge broadly relevant to all professions and are categorized into 3 sub-areas of studies:
 - (a) Arts, Humanities and Social Sciences
 - (b) Business and Management
 - (c) Science, Technology and Society
3. GER Unrestricted Electives (UE): these are courses chosen by students to broaden their learning experience. They may cover any area offered by the various departments, including, e.g., modern languages, entrepreneurship, music, and drama courses.

A course can fall into different categories simultaneously. For example, the course Principle of Economics can be in both PE and UE. There is a minimum academic units requirement for each category of courses for students to fulfill to meet their graduation requirement. The curriculum structure is shown in Figure 4.5.

An online Student Automated Registration System (STARS) is currently used for course registration at NTU. The information of the courses (e.g., time schedules and vacancies) is first released. The registration takes place over three phases. In the first phase, students register for the Major Requirement (major core and major prescribed electives) and GER CORE courses at their pre-specified date and time slot. These courses can be registered successfully as long as there are vacancies available (on a “first come, first serve” basis). Almost all students are able to register for their desired major courses and GER CORE courses in the first phase.²²

²²A major reason is that most registrants of a major course are those from the department that offers the course; therefore, every department can easily manage their offered major courses for its own students.

The second phase decides allocations of PE and UE courses by a centralized mechanism described in Section 4.3. The first two phases take place before a semester starts. During the first two weeks at the beginning of a semester, there is another and final phase where students can submit appeals for courses that they are keen to take, drop courses and add courses (provided vacancy availability). The appeals in this phase are handled manually by program coordinators from departments on a case by case basis. To ensure that certain special and urgent appeals are fulfilled, some courses may reserve a few vacancies for this final phase.

4.7.2 Survey Questions and Results

We conducted an online questionnaire survey among 1,200 undergraduate students at NTU in March 2012. The sample size is about 10% of the total population of undergraduate students. The main objective of this survey is to find out how the students would like to specify and reveal their preferences if some constraints in the current system were relaxed. We believe, by law of large numbers, the random sample that we surveyed is able to approximately capture the opinions of the whole population in large.

The survey questions and their corresponding responses are summarized as followings.

1. Are you satisfied with the current course allocation system?

- Yes — 35%
- No — 65%

2. Did you encounter a situation where you feel *indifferent* between two or more courses that you are interested in (i.e., among these courses, you do not strictly prefer one to another)?

- Yes — 76%
- No — 24%

3. Using A, B, C, D and E to denote the 5 courses that you are interested in. Then what is your usual *true* preference structure? Use " $\succ$ " to denote you strictly prefer one course to another and use " $\sim$ " to denote that you are indifferent between two courses.

(Note: Your preference structure may vary with the courses available. In this question, you only need to give a general answer. For example, you like A most, like B less than A, like C equally to B, like D less than C, and like E equally to B. Then your preference will look like: $A \succ B \sim C \succ D \sim E$.)

A ____ B ____ C ____ D ____ E (Put either " $\succ$ " or " $\sim$ " in the spaces.)

- $A \succ B \succ C \succ D \succ E$ — 24%
- $A \sim B \succ C \succ D \succ E$ — 12%
- $A \succ B \sim C \succ D \succ E$ — 7%
- $A \succ B \succ C \sim D \succ E$ — 1%
- $A \succ B \succ C \succ D \sim E$ — 6%
- $A \sim B \sim C \succ D \succ E$ — 8%
- $A \sim B \succ C \sim D \succ E$ — 4%
- $A \sim B \succ C \sim D \succ E$ — 4%
- $A \succ B \sim C \sim D \succ E$ — 3%
- $A \succ B \sim C \succ D \sim E$ — 4%
- $A \succ B \succ C \sim D \sim E$ — 17%
- $A \sim B \sim C \sim D \succ E$ — 1%
- $A \sim B \sim C \succ D \sim E$ — 1%
- $A \sim B \succ C \sim D \sim E$ — 1%
- $A \succ B \sim C \sim D \sim E$ — 5%
- $A \sim B \sim C \sim D \sim E$ — 2%

4. What is the number of courses you would like to put on your preference list?

- Five — 75%
- Ten — 13%
- Others, please specify. — 12%

5. Do you submit preferences over courses to the system according to your true preferences?

- Yes — 81%
- No — 19%

Figure 4.5: *Curriculum Structure at NTU.*

Chapter 5

Pareto Efficiency in Generalized Roommates Problem

5.1 Introduction

Most literature on the roommates problem so far has focused on the case with $2M$ students to be paired into M rooms and each room has exactly two students. More generalized situations can be where students can be assigned to multiple roommates or stay in single rooms and where students have preferences over both roommates and rooms. In generalized cases with three or more students in a room (with strict preferences), the stable roommates problem becomes NP-complete (Ng and Hirschberg, 1991; Huang, 2007). Another variant is to allow for weak preferences.

Motivated by the economics and computer science literature so far, especially Morrill (2010), we consider the generalized roommates problem, with the possible variants mentioned above. Our focus is on the Pareto optimal solution concept. For these variants, we ask the following questions: Does there always exist a Pareto optimal solution? If yes, how to compute one efficiently? In particular, we answer these questions separately for the cases with strict preferences and with ties in preferences.

Finding an efficient assignment in the roommates problem is important in practice. Currently many universities assign students to dormitories randomly, which usually results in an inefficient situation where students are unsatisfied with the allocations and therefore move out from the university dormitories. This is a waste

of resources and also causes significant welfare losses to students. Having a bad roommate has in fact a dire consequence on a person's life. Sacerdote (2001) shows that roommate peer influences makes a deep impact on one's grade point average as well as their social choices with a unique data set measuring peer effects among college roommates at Dartmouth College. A separate study conducted by the University of North Carolina identifies roommates as one of the major stress inducers and it finds that roommate conflicts ranks consistently among the top five reasons for school dropouts. There have also been many incidents whereby roommate disagreements have led to violence and in some extreme isolated cases, and even death.¹ To curb these issues, many of the colleges and universities have been outsourcing and depending on websites that help facilitate and expedite the process of matching roommates with one another, leaving the decision of pairing roommates to the students themselves. University partnerships with roommate matching websites are also established to ease the process of dormitory application.²

The roommates problem can also model many interesting one-sided matching problems, for instance, grouping lab partners, project teammates, police officers on patrols, pilots on flights and sports players in teams. The way in which agents are paired or grouped may affect the final result in sports competitions and the efficiency of teamwork in other settings. The kidney exchange problem has also been modeled as a roommates problem (Roth et al., 2005). Furthermore, the roommates problem can find potential applications in other problems with similar settings, especially centrally coordinated programs such as holidays home exchanges³, and centralized pairing methods used in chess competitions (Kujansuu et al., 1995).

In this chapter, we study Pareto efficiency in the generalized roommates problem, which provides an alternative perspective as opposed to the over-emphasized stability solution concept and reaches out to more general applications. The results suggest existence of Pareto efficient assignments in the general problem and

¹http://www.huffingtonpost.com/2012/08/07/colleges-roommates-gpa-drop-out_n_1752853.html

²Some examples of roommate matching websites are: Roomsurf.com, Roomster.com, Roomiematch.com.

³<http://www.exchangeholidayhomes.com/>

introduce algorithms to compute such assignments. Our proposed more general Pareto-Improving Roommate Swap Algorithm is however not strategyproof for the students.

The remainder of this chapter is organized as follows. Section 5.2 formally defines the generalized roommates problems. Section 5.3 discusses the existence of Pareto efficient assignments in cases without and with ties and present the algorithms. Section 5.4 examines the strategic implications of the proposed algorithms. We conclude in Section 5.5.

5.2 Problem Definition and Solution Concept

A generalized roommates problem consists of a tuple $(\mathcal{S}, \mathcal{R}, q_{\mathcal{R}}, \succsim_{\mathcal{S}})$, where we wish to assign N students $\mathcal{S} = \{s_1, s_2, \dots, s_N\}$ to M rooms $\mathcal{R} = \{r_1, r_2, \dots, r_M\}$. Each room has an integral **capacity** $q_{r_i} \in \mathbb{N}, i \in \{1, 2, \dots, M\}$, which denotes the maximum number of students that are allowed to stay in room r_i . We use $q_{\mathcal{R}}$ to denote the vector of room capacities. Rooms may have different capacities. Rooms with the same capacity are identical and students are indifferent between these rooms. For simplicity, we assume that the total capacities of the rooms are equal to the total number of students, i.e., $\sum_{i=1}^M q_{r_i} = N$, so that there is no shortage of the rooms, because virtual single rooms and virtual students who find no students acceptable can always be added.

Each student s_i has a **preference** ranking over the other $N-1$ students, denoted as $\succsim_{\mathcal{S}} = (\succsim_{s_i})_{s_i \in \mathcal{S}}$, where $s_j \succ_{s_i} s_k (i \neq j, k)$ means that s_i strictly prefers s_j to s_k , and $s_j \sim_{s_i} s_k$ means that s_i is **indifferent** between s_j and s_k .⁴ We use $\emptyset$ to denote being unmatched, that is, staying in a single room. The preferences are assumed to be complete and transitive. Since students can have multiple roommates, we extend the preferences over individual students to groups of students with the **max-min** criterion specified as follows⁵.

⁴By an abuse of notation, we will omit the subscript and use $\succ$ or $\sim$ to denote the preference relations whenever it is clear from the context.

⁵The definition of max-min preference here is different from the one for the two-side matching market, introduced by Baiou and Balinski (2000). In a two-sided matching market, the max-min

Definition 5.2.1 (Max-Min Preference). *For each s_i and any two subsets $A, B \subset \mathcal{S} \setminus \{s_i\}$, let $\min_{s_i}(A)$ and $\min_{s_i}(B)$ denote the least preferred student of s_i in the subset A and B , respectively. It is called max-min preference over subsets if the following holds:*

- If $\min_{s_i}(A) \succ_{s_i} \min_{s_i}(B)$, then $A \succ_{s_i} B$;
- If $\min_{s_i}(A) \sim_{s_i} \min_{s_i}(B)$, and⁶
 - $|\min_{s_i}(A)| < |\min_{s_i}(B)|$, then $A \succ_{s_i} B$;
 - $|\min_{s_i}(A)| = |\min_{s_i}(B)|$, then $A \sim_{s_i} B$;
 - $|\min_{s_i}(A)| > |\min_{s_i}(B)|$, then $B \succ_{s_i} A$;
- If $\min_{s_i}(B) \succ_{s_i} \min_{s_i}(A)$, then $B \succ_{s_i} A$.

In other words, each student cares about the least preferred roommate in a room and therefore wants to minimize the rank order and the number of the least preferred roommates in a room according to his/her preference list over individuals. In particular, when a student is not indifferent between the least preferred roommates in two subsets, he/she prefers the one with the more preferred least preferred roommate. Otherwise, he/she prefers the one with a smaller number of the least preferred roommates and he/she is indifferent between the two subsets if the number is equal. This type of preference is reasonable since students usually have problems in getting along with the least preferred person in a room. The max-min preference helps to minimize the possibility of conflicts in a room. Also, the max-min preference implies that students do not benefit from the positive externality by having a high rank ordered roommate as long as her least preferred roommates remain the same. We have the following observation saying that students do not get better offer by expanding the current nonempty roommates set.

preference is defined as: Taking the school choice problem as an example, use c to denote a school and S_1, S_2 to denote groups of students. For any $S_1, S_2 \subseteq S$ with $|S_1| \leq q_c$ and $|S_2| \leq q_c$, if (1) $|S_1| \geq |S_2|$ and c strictly prefers the least preferred agent in S_1 to the least preferred agent in S_2 ; or (2) $S_1 = S_2$, then $S_1 \succsim_c S_2$.

⁶There may be multiple least preferred roommates since the preferences can have ties. It reduces to the second case if all preferences are strict.

Observation 5.2.1. *For each s_i and any two nonempty subsets $A, B \subset \mathcal{S} \setminus \{s_i\}$, we have $A \succsim_{s_i} (A \cup B)$ and $B \succsim_{s_i} (A \cup B)$.*

Another commonly used assumption for preferences over subsets is called the *lexicographical preference*. For instance, in the cases of three persons in a room, let $A = \{s_1, s_2\}$ and $B = \{s_3, s_4\}$ be two possible roommates choice sets and $s_1 \succ_{s_i} s_2$, $s_3 \succ_{s_i} s_4$. We say s_i lexicographically prefers A to B if and only if either (1) $s_1 \succ_{s_i} s_3$ or (2) $s_1 \sim_{s_i} s_3$ and $s_2 \succ_{s_i} s_4$. The difference between lexicographical preference and max-min preference is illustrated in the following example.

Example 5.2.2 (Lexicographical preference vs. Max-min preference). *A student s_i has preference list over individuals as: $s_1 \succ_{s_i} s_2 \sim_{s_i} s_3$. Under max-min preferences, we have $\{s_1, s_3\} \succ_{s_i} \{s_1, s_2, s_3\}$; while under lexicographical preferences, we have $\{s_1, s_2, s_3\} \succ_{s_i} \{s_1, s_3\}$. Under max-min preferences, we have $\{s_2\} \sim_{s_i} \{s_1, s_3\}$; while under lexicographical preferences, we have $\{s_1, s_3\} \succ_{s_i} \{s_2\}$. Under max-min preferences, we have $\{s_1\} \succ_{s_i} \{s_1, s_2\}$; while under lexicographical preferences, we have $\{s_1, s_2\} \succ_{s_i} \{s_1\}$ (since $s_2 \succ_{s_i} \emptyset$).*

We next define an assignment in the generalized roommates problem and the solution concept of Pareto efficiency. Pareto efficiency is arguably the natural solution concept in the roommates problem as compared to stability. Because in the roommates problem, when some agents want dissolve a current assignment, they need to have a room to live in, while they can not simply force their current roommates to move out if the reassignment is not beneficial to their current roommates. Therefore, this physical constraint of a room makes Pareto efficiency the more appropriate solution concept.

Definition 5.2.3 (Assignment). *An assignment (or a matching) is an outcome of a roommates problem, and is defined by a function $\mu : \mathcal{S} \rightarrow 2^{\mathcal{S}}$ such that $\mu(s) \in 2^{\mathcal{S} \setminus \{s\}}$ and for each student $s' \in \mu(s)$, $\mu(s') = \mu(s) \cup \{s\} \setminus \{s'\}$. Every student is assigned to some other students or remain unassigned ($\mu(s) = \emptyset$ i.e., stay in a single room or assigned to oneself) and the assignments are symmetric.*

Definition 5.2.4 (Pareto Efficiency). *An assignment μ is Pareto efficient if there does not exist any Pareto improvement. We say an assignment μ' is a Pareto*

improvement to μ if for every student s , $\mu'(s) \succsim_s \mu(s)$ and $\mu'(s) \succ_s \mu(s)$ for at least one s .

5.3 Pareto Efficient Assignments

When all the agents have strict preferences, a Pareto efficient assignment always exists and can be computed using the **Random Serial Dictatorship (RSD) Algorithm**.

Theorem 5.3.1. *For any roommates problem with strict preferences, a Pareto efficient assignment always exists.*

Proof. For any given problem, we use the RSD algorithm to compute an assignment as following. We first assign every student a random priority. Under this priority structure, students sequentially choose their most preferred roommates given the capacity constraints of the currently available rooms. Assign the student with highest priority his/her most preferred roommates and remove them and the respective room from the queue. For the remaining students, assign the student with highest priority his/her most preferred roommates among those students who are unassigned. If there is no more acceptable roommates among the remaining students or only virtual single rooms left, this student will remain unassigned (or equivalently, stay in a virtual single room). Remove these students and the respective room from the queue. Repeat this process until no students remain in the queue. Since students do not benefit by expanding their current roommates sets (see Observation 5.2.1), rooms with smaller capacities ($q_r \geq 2$) will be occupied earlier as compared to rooms with larger capacities. Since the total capabilities of the rooms are equal to the total number of students, every student gets a room (or a virtual room) at the end of the algorithm.

We next show that the assignment returned by the RSD algorithm is Pareto efficient. Given an assignment from the RSD, if any student wants to improve his/her assignment, he/she must try to get unmatched with the least preferred partner in the current room and rematch with those students who have chosen before him/her (i.e., with higher priority) or those who have been chosen by other students with

higher priority. Since students have strict preferences over individuals, any this kind of reassignment will make another student with higher priority worse off, given the room capacity constraints and the max-min preferences over subsets. Therefore, there is no possible Pareto improvement and the assignment returned by the RSD algorithm achieves Pareto efficiency. $\square$

How about in a more general case where students have weak preferences? A natural practice to deal with weak preferences is to break ties randomly. We next show that when there are ties in preferences, the RSD algorithm with tie-breaking does not necessarily output a Pareto efficient assignment.

Proposition 5.3.2. *For any roommates problem with ties, the RSD algorithm with tie-breaking does not always output a Pareto efficient assignment.*

Proof. Assume there are two rooms each with a capacity of 2 and four students s_1 , s_2 , s_3 and s_4 with the following preferences:

$$\begin{aligned} s_1 : \quad & s_3 \sim s_2 \succ s_4 \\ s_2 : \quad & s_1 \succ s_3 \succ s_4 \\ s_3 : \quad & s_4 \succ s_2 \succ s_1 \\ s_4 : \quad & s_3 \succ s_2 \succ s_1 \end{aligned}$$

The RSD algorithm first assigns every student a random priority ranking, say, s_1 ranks first, followed by s_2 , then by s_3 and lastly s_4 . A random tie-breaking for s_1 's preference list gives $s_1 : s_3 \succ s_2 \succ s_4$. According to the preferences after tie-breaking, we then assign s_1 his/her most preferred roommate s_3 and remove s_1, s_3 from the queue. We next assign s_2 his/her most proffered roommate among the remaining students, which is s_4 . The algorithm terminates since every student is assigned. The outcome is $\mu = \{(s_1, s_3), (s_2, s_4)\}$. According to the true preferences with ties, the assignment $\mu' = \{(s_1, s_2), (s_3, s_4)\}$ is a Pareto improvement to μ since s_2, s_3 and s_4 are all better off while s_1 is not worse off. Therefore, μ is not Pareto efficient. $\square$

The next natural question is whether a Pareto efficient assignment exists for any generalized roommate problem with ties, and if it exists, how to find such a Pareto

efficient assignment. To answer these questions, we first define a blocking pair as following.

Definition 5.3.3 (Blocking Pair). *Given an assignment μ , (s_i, s_j) is a blocking pair if $s_j \succsim_{s_i} \min_{s_i}(\mu(s_i))$ and $s_i \succsim_{s_j} \min_{s_j}(\mu(s_j))$.*

That is to say, s_i weakly prefers s_j to his/her current least preferred roommate and vice versa.

Based on the ideas of the RSD algorithm and the Roommate Swap Algorithm proposed by Morrill (2010), we show that a Pareto efficient assignment always exists for any generalized roommates problem with ties by developing a Pareto efficient algorithm. We refer to it as the **Pareto-Improving Roommate Swap Algorithm**, which is presented as following.

PARETO-IMPROVING ROOMMATE SWAP ALGORITHM

Initialization:

- All students are available.
- Each student is represented by a vertex in an empty graph G_0 .
- Randomly rank all the students.
- Break ties in preferences randomly.

Among the available students, consider the student with highest priority ranking s_i , and we refer to s_i as the "choosing student":

- Assign s_i his/her most preferred roommates among the available students given the currently available room capacities, mark these students and the respective room as unavailable;
- Update the assignment graph to G_i as following:
 - Draw a solid edge between s_i and his/her least preferred roommate s_j ;
 - Among the currently assigned students, draw dashed edges for blocking pairs that involve only s_i 's and s_j 's;
- Run the Roommate Swap Algorithm on G_i .

The algorithm terminates when there is no more available student.

In each iteration of the algorithm, we update the assignment graph G_i by drawing edges among the choosing students s_i 's and their respective least preferred roommates s_j 's. In this way, each assigned room is reduced to two students, s_i and s_j , which becomes a two-roommates problem as commonly discussed in the roommates problem literature. Then we can apply the Roommate Swap Algorithm proposed by Morrill (2010) on the current assignment graph G_i to search for Pareto improvements among the choosing students s_i 's and their least preferred roommate s_j 's, which ensures that there is no possible Pareto improvement among these students.

We next show that such a consideration further ensures there is no possible Pareto improvement for the choosing students when we consider all the students given the assumption of max-min preferences over subsets.

Proposition 5.3.4. *The choosing students s_i 's cannot be strictly improved at the end of the algorithm.*

Proof. We show that for the choosing students s_i 's, we only need to consider their least preferred roommates s_j 's when searching for strict improvements in each iteration of the algorithm by the definition of the max-min preferences. We consider the following inductive steps in the implementation of the algorithm:

1. After the first iteration, s_1 gets his/her best possible roommates. Therefore, among the assigned students, there is no possible Pareto improvement for s_1 nor any of s_1 's roommate.
2. After the second iteration, both s_1 and s_2 are assigned. Among the currently assigned students, s_2 may get a strict improvement only if one of s_1 's roommates, say s_p , leaves s_1 and joins s_2 , while s_1 gets another roommate s_q and one of s_2 's roommates, say s_k , leaves. Then s_1 must be indifferent between s_p and s_q , which are both his/her least preferred roommates, because otherwise s_1 is able to get a Pareto improvement in the previous iteration. For s_2 , s_k must be his/her least preferred roommate and he/she strictly prefers s_p over s_k .
3. Inductively, we have similar reasoning for the remaining choosing students in each iteration.

We have shown that for each choosing student to get a Pareto improvement, it is always improving his/her least preferred roommate in the room. Therefore, we only need to consider the choosing students and their respective least preferred roommates in searching for a Pareto improvement. The Roommate Swap Algorithm on these two students in a room eliminates all such Pareto improvements for the choosing students. Therefore, the choosing students s_i 's cannot be strictly improved at the end of the algorithm. $\square$

With the above result, we are ready to show that the assignment output by the algorithm is Pareto efficient for all the students and the algorithm is polynomial-time in terms of the total number of students.

Theorem 5.3.5. *The Pareto-Improving Roommate Swap Algorithm computes a Pareto efficient assignment for any generalized roommates problem with ties in polynomial time.*

Proof. To prove that the algorithm returns a Pareto efficient assignment, we only need to show that no students can be involved in any Pareto improvement. We have shown that none of the choosing students s_i 's can be strictly improved in Proposition 5.3.4. For any non-choosing students to get a strict improvement, then it must change some of the roommates of a choosing student s_i by having s_p leave the room and s_q join the room. To ensure s_i is not worse off, we must have $s_q \succ_{s_i} \min(\mu(s_i))$ by the definition of max-min preferences. Then s_i gets a strict improvement in the process, a contradiction.

We next prove that the algorithm is polynomial-time. There are at most N choosing students in the worst case that all rooms are single rooms. Each iteration of the algorithm involves the following steps:

1. Assigning roommates to the choosing student and then marking the assigned students and rooms as unavailable is at worst $O(N)$.
2. Updating the assignment graph is at worst $O(N^2)$ since there are at most $N/2$ assigned rooms and therefore at most $\frac{N/2(N/2-1)}{2}$ dashed edges.
3. Making Pareto improvements with the Roommate Swap Algorithm is at worst $O(N^3)$.

Each iteration is therefore $O(N^3)$. The algorithm performs $O(N)$ many iterations of an $O(N^3)$ process. Therefore, it is at worst $O(N^4)$. $\square$

5.4 Strategic Implications

In this section, we investigate the strategic properties of the two Pareto efficient algorithms proposed. We define strategyproofness as following.

Definition 5.4.1 (Strategyproofness). *A dominant strategy is a strategy that is a best response to all possible strategies of the other agents. An assignment mechanism is strategyproof if it is a dominant strategy for each agent to reveal his/her preferences truthfully.*

When preferences are strict, the Pareto efficient RSD algorithm is strategyproof for the students as in its application to other types of markets.

Proposition 5.4.2. *The RSD algorithm is strategyproof for the students.*

Proof. For any random ranking of the students, let $s_1, s_2, s_3, \dots$ denote the students according to their order in choosing the roommates. The first student s_1 obtains the most favorite roommates for him/her when he/she tells the truth, so s_1 has no incentives to lie. The second student s_2 obtains his/her most favorite roommates among the remaining students when he/she tells the truth, so s_2 has no incentives to lie. This argument continues for s_3, s_4 and so on. So no choosing student has can benefit by misrepresenting his/her preferences. While for the non-choosing students, their preferences cannot change the outcome and do not really matter in the process. Therefore, no student has an incentive to misrepresent his/her preferences. $\square$

While the RSD algorithm is intuitively strategyproof because every student is getting the best available choices, the Pareto-Improving Roommate Swap Algorithm is no longer strategyproof for the students. In other words, it is not always a dominant strategy for the students to truthfully report their preferences.

Proposition 5.4.3. *The Pareto-Improving Roommate Swap Algorithm is not strategyproof for the students.*

Proof. Suppose there are four students, s_1, s_2, s_3 and s_4 , with preferences as following.

$$s_1 : s_3 \succ s_4 \succ s_2$$

$$s_2 : s_3 \succ s_4 \succ s_1$$

$$s_3 : s_2 \sim s_1 \sim s_4$$

$$s_4 : s_2 \succ s_1 \succ s_3$$

With four students, an assignment is completely determined by who s_1 is assigned to. There are three possible assignments as following:

$$\mu_1 = \begin{pmatrix} s_1 & s_3 \\ s_2 & s_4 \end{pmatrix} \quad \mu_2 = \begin{pmatrix} s_1 & s_2 \\ s_3 & s_4 \end{pmatrix} \quad \mu_3 = \begin{pmatrix} s_1 & s_2 \\ s_4 & s_3 \end{pmatrix}$$

To implement the Pareto-Improving Roommate Swap Algorithm, assume a random ranking gives s_3 the first place, followed by s_1 , and then s_2 and lastly s_4 . An arbitrary tie-breaking gives s_3 's preference as: $s_4 \succ s_2 \succ s_1$. We first assign s_4 to s_3 and then s_2 to s_1 , which gives the assignment μ_1 . Next we search for Pareto improvements using the Roommate Swap Algorithm. Notice that both μ_2 and μ_3 are Pareto improvements of μ_1 . How do we select a Pareto improvement then? Note that if s_1 misreports his/her preferences as $s_3 \succ s_2 \succ s_4$ while all other students submit their true preferences, then μ_2 is the only Pareto-improving assignment to μ_1 with respect to the submitted preferences. Similarly, if s_2 misreports his/her preferences as $s_3 \succ s_1 \succ s_4$ while all other students are truthful, then μ_3 becomes the assignment that Pareto improves μ_1 .

If all students are truthful, the algorithm selects either μ_2 or μ_3 as a Pareto improvement. If the outcome is μ_2 , then s_2 can become better off by misreporting his/her preferences as $s_3 \succ s_1 \succ s_4$. On the other hand, if μ_3 is the selected outcome, then s_1 can benefit by deviating and submitting $s_3 \succ s_2 \succ s_4$. Either way, the Pareto-Improving Roommate Swap Algorithm is not strategyproof for the students. $\square$

The intuition behind this result is that when there are multiple Pareto improvements, some may benefit one group of students more while others are more preferred by another group of students. These conflicting interests in different Pareto improvements create incentives for the students to misrepresent their preferences in a way

such that the less preferred Pareto improvements become implausible, that is, they are no longer Pareto improvements with respect to the submitted preferences.

5.5 Conclusion

We study a generalization of the roommates problem, in which N students need to be assigned to M rooms and rooms may have different capacities. In this problem, the traditional notion of stability in the classical two-sided matching problems is too restrictive, because multilateral agreement is required to dissolve a *status quo* assignment and consequently to be able to “block” the current assignment. A relevant solution concept is therefore Pareto efficiency. We show that a Pareto efficient assignment always exists in the generalized roommates problem with strict preferences and can be computed using the Random Serial Dictatorship Algorithm. However, when the preferences are not necessarily strict, the RSD algorithm does not guarantee Pareto efficiency. We further introduce an efficient Pareto-Improving Roommate Swap Algorithm to find a Pareto efficient assignment when preferences are weak. While the Random Serial Dictatorship Algorithm is strategyproof for the students, the Pareto-Improving Roommate Swap Algorithm is not.

Chapter 6

Discussion and Future Research

6.1 Conclusion and Implications

The seminal work of Gale and Shapley on stable marriage (Gale and Shapley, 1962) has inspired extensive investigations on its various variants. While in tradition economists tend to build up an solve models which are attractive in a stylized theoretical environment, in recent decades, market designers have been designing mechanisms which are able to address complications and details that constantly arise in practice.

We study the many-to-many matching problem with ties and the related practical market design problems. The complications in these problems arise from two aspects: the multi-unit capacities on both sides of the market and the weak preferences. Either of them can contribute to Pareto inefficiency in a stable matching outcome. Despite the wide applications in real life, these problems have been overlooked in both theoretical and empirical market design literature. This thesis fills in the gap from both theory and practice.

On the theory side, we develop an efficient algorithm that computes a Pareto stable matching for the generalized many-to-many matching markets with ties. This algorithm is general enough to be widely applied in various two-sided matching markets. For a more practical market with homogeneous preferences on one side, we propose two competing Pareto stable matching mechanisms, namely, the Pareto-improving draft and dictatorship mechanisms. These two mechanisms can

be easily implemented in practice and have comparative advantage in terms of strategyproofness and non-callousness. The Pareto-improving dictatorship mechanism is strategyproof, but it comes at the cost of callousness by resulting in extremely unfair matchings for low priority agents.

On the practice side, we propose a new design to the course allocation system at NTU by introducing ties into students' preferences and combining the allocation processes of two types of courses as refinements to the current system at NTU in order to improve overall efficiency. With the proposed Pareto-improving draft and dictatorship mechanisms, we run simulations on the real course allocation data from 2010 to 2012. Our simulation results show significant efficiency and welfare gains for the students with the proposed new designs. In total, we can see up to 2,597 more student-course matches for 2010 (3,263 for 2011 and 3,026 for 2012). This is equivalent to approximately 11.65 percent (12.83 percent for 2011 and 11.84 percent for 2012) improvement in total efficiency. These positive results provide further evidence on the cost of tie-breaking in practice and call for changes to NTU's current course allocation system.

Our simulation results comparing the Pareto-improving draft and dictatorship mechanisms further suggest that the draft mechanism outperforms the dictatorship mechanism in terms of the total number of allocations, the average rank and the total number of unassigned students, despite the fact that the former is non-strategyproof for the students.

Ties are realistic complications and details occurring in many matching markets with preferences, especially when individuals have incomplete information. Our work on the two-sided matching markets, following the studies of Erdil and Ergin (2008) and Abdulkadiroğlu et al. (2009), is devoted to improving social efficiency and welfare in the presence of ties. The models studied can potentially be applied to other applications with a similar setup. Our work considers a number of fundamental solution concepts, including pairwise and group stability, and one-sided and two-sided Pareto efficiency. We examine the existence and computation of these solution concepts. These results are of independent interest and may find applications in other many-to-many matching markets.

We also study the generalized roommates problem by considering multiple roommates and weak preferences. We show that even with these generalizations, Pareto efficient assignment still exists and can be computed efficiently. These results are important and have significant implications for the design of many practical one-sided matching markets.

6.2 Future Research

There are many other perspectives that deserve further attention in many-to-many matching markets with indifferences. For example, what if the preferences over subsets of acceptable partners are defined more generally and combinatorially? What if there are interdependence among students' preferences? For these models, does a Pareto-stable matching always exist and can it be computed efficiently? We leave these as future work.

For the problem with one-sided homogeneous preferences, our next step is to examine the strategic properties of the proposed Pareto stable matching mechanisms, because strategyproofness has been a desideratum in many practical market designs. The Pareto-improving draft mechanism is not strategyproof although it outperforms the Pareto-improving dictatorship mechanism based on our simulation results. So what are the costs of using this non-strategyproof mechanism, especially in a large matching market like the course-student matching? We can study this question both theoretically and experimentally. Identifying the costs of non-strategyproofness can provide insights on which one of my two competing mechanisms should be implemented in practice and in general the results can shed light on the importance of strategyproofness in practical market designs.

In the practical market design of course allocation, an interesting direction is to compare the course bidding mechanism with the matching mechanism. Which one is simpler for the students in the strategic sense? Which one gives larger social welfare? All these questions can be examined both theoretically and experimentally. In particular, experimental studies allow us to further test the new designs for the course-student matching market before it is put into implementation in practice.

When the new designs are tested or implemented, it may raise new theoretical questions, which could lead to progress in economic theory.

There are also many interesting directions for the generalized roommates problem. Our work is a first step to this generalized problem. There are many other questions which deserve future studies, such as efficiently determine whether a *status quo* assignment is Pareto efficient and find a Pareto improvement if it is not. While this problem has been studied in the two-roommates version, it may not be straightforward to carry over to the general cases as in theoretical computer science, the fine line between P and NP is often drawn between the numbers two and three. Note that the key ingredient in our Pareto-Improving Roommate Swap Algorithm is that the choosing student at each round is getting the best choices, we can therefore make sure that they will not be involved in any Pareto improvement. However, for a *status quo* assignment, we cannot identify these choosing students. We leave these as future work.

Bibliography

Abdulkadiroğlu, A., Pathak, P., Roth, A. E., and Sonmez, T. “Changing the Boston School Choice Mechanism.” NBER Working Papers 11965, National Bureau of Economic Research, Inc (2006).

Abdulkadiroğlu, A., Pathak, P. A., and Roth, A. E. “The New York City High School Match.” *American Economic Review*, 95(2):364–367 (2005a).

—. “Strategy-Proofness versus Efficiency in Matching with Indifferences: Redesigning the NYC High School Match.” *American Economic Review*, 99(5):1954–1978 (2009).

Abdulkadiroğlu, A., Pathak, P. A., Roth, A. E., and Sonmez, T. “The Boston Public School Match.” *American Economic Review*, 95(2):368–371 (2005b).

Abdulkadiroğlu, A. and Sönmez, T. “School Choice: A Mechanism Design Approach.” *American Economic Review*, 93(3):729–747 (2003).

Albergotti, R. “Why the NFL Draft Drives Economists Crazy.” *Wall Street Journal*, April 22 (2010).

Alkan, A. “On Preferences over Subsets and the Lattice Structure of Stable Matchings.” *Review of Economic Design*, 6(1):99–111 (2001).

—. “A Class of Multipartner Matching Models with a Strong Lattice Structure.” *Economic Theory*, 19(4):737–746 (2002).

Ashlagi, I. and Roth, A. E. “New Challenges in Multihospital Kidney Exchange.” *American Economic Review*, 102(3):354–59 (2012).

- Baiou, M. and Balinski, M. “Many-to-Many Matching: Stable Polyandrous Polygamy (or Polygamous Polyandry).” *Discrete Applied Mathematics*, 101:1–12 (2000).
- Blair, C. “The Lattice Structure of the Set of Stable Matchings with Multiple Partners.” *Mathematics of Operations Research*, 13(4):619–628 (1988).
- Bogomolnaia, A. and Moulin, H. “A New Solution to the Random Assignment Problem.” *Journal of Economic Theory*, 100(2):295 – 328 (2001).
- Budish, E. “Matching ‘versus’ Mechanism Design.” *ACM SIGecom Exchanges*, 11(2):4–15 (2012).
- Budish, E. and Cantillon, E. “The Multi-unit Assignment Problem: Theory and Evidence from Course Allocation at Harvard.” *American Economic Review*, 102(5):2237–2271 (2012).
- Budish, E. and Kessler, J. “Changing the Course Allocation Mechanism at Wharton.” *Working Paper* (2014).
- Bulow, J., Levin, J., and Milgrom, P. “Winning Play in Spectrum Auctions.” NBER Working Papers 14765, National Bureau of Economic Research, Inc (2009).
- Cechlarova, K. and Ferkova, S. “The Stable Crews Problem.” *Discrete Applied Mathematics*, 140(1-3):1–17 (2004).
- Che, Y.-K. and Kojima, F. “Asymptotic Equivalence of Probabilistic Serial and Random Priority Mechanisms.” *Econometrica*, 78(5):1625–1672 (2010).
- Chen, N. and Ghosh, A. “A Market Clearing Solution for Social Lending.” In *International Joint Conference on Artificial Intelligence (IJCAI)*, 152–157 (2011).
- Chen, Y. and Kesten, O. “From Boston to Chinese Parallel to Deferred Acceptance: Theory and Experiments on a Family of School Choice Mechanisms.” Discussion papers, research unit: Market behavior, Social Science Research Center Berlin (WZB) (2013).

- Chung, K.-S. “On the Existence of Stable Roommate Matchings.” *Games and Economic Behavior*, 33(2):206–230 (2000).
- Cramton, P. “Spectrum Auction Design.” *Review of Industrial Organization*, 42(2):161–190 (2013).
- Cripps, M. W. and Swinkels, J. M. “Efficiency of Large Double Auctions.” *Econometrica*, 74(1):47–92 (2006).
- Echenique, F. and Oviedo, J. “A Theory of Stability in Many-to-Many Matching Markets.” *Theoretical Economics*, 1(2):233–273 (2006).
- Erdil, A. and Ergin, H. “Two-Sided Matching with Indifferences.” *Working paper* (2006).
- . “What’s the Matter with Tie-breaking? Improving Efficiency in School Choice.” *American Economic Review*, 98(3):669–689 (2008).
- Fragiadakis, D. E. and Troyan, P. “Improving Welfare in Assignment Problems: an Experimental Investigation.” *Working paper* (2013).
- Fudenberg, D., Mobius, M. M., and Szeidl, A. “Existence of Equilibria in Large Double Auctions.” *Journal of Economic Theory*, 133(1):550–567 (2007).
- Gale, D. and Shapley, L. S. “College Admissions and the Stability of Marriage.” *American Mathematical Monthly*, 69(1):9–15 (1962).
- Guillen, P. and Kesten, O. “Matching Markets With Mixed Ownership: The Case for a Real Life Assignment Mechanism.” *International Economic Review*, 53(3):1027–1046 (2012).
- Gusfield, D. and Irving, R. W. *The Stable Marriage Problem: Structure and Algorithms*. Cambridge, MA, USA: MIT Press (1989).
- Huang, C.-C. “Two’s Company, Three’s a Crowd: Stable Family and Threesome Roommates Problems.” In *ESA’07 Proceedings of the 15th Annual European Conference on Algorithms*, 558–569 (2007).

- Immorlica, N. and Mahdian, M. “Marriage, Honesty, and Stability.” In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 53–62 (2005).
- Irving, R. W. “An Efficient Algorithm for the Stable Roommates Problem.” *Journal of Algorithms*, 6(4):577 – 595 (1985).
- Irving, R. W. and Manlove, D. F. “The Stable Roommates Problem with Ties.” *Journal of Algorithms*, 43(1):85 – 105 (2002).
- Iwama, K., Miyazaki, S., Morita, Y., and Manlove, D. “Stable Marriage with Incomplete Lists and Ties.” In *International Colloquium on Automata, Languages and Programming (ICALP)*, 443–452 (1999).
- Iwama, K., Miyazaki, S., and Yamauchi, N. “A 1.875-Approximation Algorithm for the Stable Marriage Problem.” In *ACM-SIAM Symposium on Discrete Algorithms (SODA)*, 288–297 (2007).
- Kojima, F. “Efficient Resource Allocation under Multi-unit Demand.” *Games and Economic Behavior*, 82(C):1–14 (2013).
- Kojima, F. and Manea, M. “Incentives in the Probabilistic Serial Mechanism.” *Journal of Economic Theory*, 145(1):106–123 (2010).
- Kojima, F. and Pathak, P. A. “Incentives and Stability in Large Two-Sided Matching Markets.” *Journal of Economic Theory*, 99(3):608–627 (2009).
- Kominers, S. D., Ruberry, M., and Ullman, J. “Course Allocation by Proxy Auction.” In *International Workshop on Internet and Network Economics (WINE)*, 551–558 (2010).
- Krishna, A. and Ünver, M. U. “Improving the Efficiency of Course Bidding at Business Schools: Field and Laboratory Studies.” *Marketing Science*, 27(2):262–282 (2008).
- Kujansuu, E., Lindberg, T., and Mäkinen, E. *The Stable Roommates Problem and Chess Tournament Pairings*. Julkaisusarja A. University of Tampere, Department of Computer Science (1995).

- Manlove, D. F., Irving, R. W., Iwama, K., Miyazaki, S., and Morita, Y. “Hard Variants of Stable Marriage.” *Theoretical Computer Science*, 276(1-2):261–279 (2002).
- Martinez, R., Masso, J., Neme, A., and Oviedo, J. “An Algorithm to Compute the Full Set of Many-to-Many Stable Matchings.” *Mathematical Social Sciences*, 47(2):187–210 (2004).
- McDermid, E. “A $3/2$ -Approximation Algorithm for General Stable Marriage.” In *International Colloquium on Automata, Languages and Programming (ICALP)*, 689–700 (2009).
- McMillan, J. “Selling Spectrum Rights.” *Journal of Economic Perspectives*, 8(3):145–162 (1994).
- Morrill, T. “The Roommates Problem Revisited.” *Journal of Economic Theory*, 145(5):1739 – 1756 (2010).
- Ng, C. and Hirschberg, D. S. “Three-Dimensional Stable Matching Problems.” *SIAM Journal on Discrete Mathematics*, 4(2):245–252 (1991).
- Pathak, P. A. and Sethuraman, J. “Lotteries in Student Assignment: An Equivalence Result.” *Theoretical Economics*, 6(1):1–17 (2011).
- Roberts, D. J. and Postlewaite, A. “The Incentives for Price-Taking Behavior in Large Exchange Economies.” *Econometrica*, 44(1):115–127 (1976).
- Roth, A. E. “Stability and Polarization of Interests in Job Matching.” *Econometrica*, 52(1):47–57 (1984a).
- . “The Evolution of the Labor Market for Medical Interns and Residents: A Case Study in Game Theory.” *Journal of Political Economy*, 92(6):991–1016 (1984b).
- . “The College Admissions Problem is not Equivalent to the Marriage Problem.” *Journal of Economic Theory*, 36(2):277–288 (1985).
- . “On the Allocation of Residents to Rural Hospitals: A General Property of Two-Sided Matching Markets.” *Econometrica*, 54(2):425–27 (1986).

- . “The Economist as Engineer: Game Theory, Experimentation, and Computation as Tools for Design Economics.” *Econometrica*, 70(4):1341–1378 (2002).
- Roth, A. E. and Peranson, E. “The Redesign of the Matching Market for American Physicians: Some Engineering Aspects of Economic Design.” *American Economic Review*, 89(4):748–780 (1999).
- Roth, A. E., Sönmez, T., and Ünver, M. U. “Kidney Exchange.” *The Quarterly Journal of Economics*, 119(2):457–488 (2004).
- . “Pairwise kidney exchange.” *Journal of Economic Theory*, 125(2):151–188 (2005).
- . “Efficient Kidney Exchange: Coincidence of Wants in Markets with Compatibility-Based Preferences.” *American Economic Review*, 97(3):828–851 (2007).
- Roth, A. E. and Sotomayor, M. A. O. *Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis*. Cambridge University Press (1990).
- Roth, A. E. and Vate, J. H. V. “Random Paths to Stability in Two-Sided Matching.” *Econometrica*, 58(6):1475–1480 (1990).
- Sacerdote, B. “Peer Effects with Random Assignment: Results for Dartmouth Roomates.” *Journal of Economics*, 116(2):681–704 (2001).
- Sönmez, T. and Ünver, M. U. “Course Bidding at Business Schools.” *International Economic Review*, 51(1):99–123 (2010).
- . *Matching, Allocation, and Exchange of Discrete Resources*, 781–852. Handbook of Social Economics. North-Holland (2011).
- Sotomayor, M. A. O. “Three Remarks on the Many-to-Many Stable Matching Problem.” *Mathematical Social Sciences*, 38(1):55–70 (1999).
- . “The Pareto-Stability Concept is a Natural Solution Concept for Discrete Matching Markets with Indifferences.” *International Journal of Game Theory*, 40(3):631–644 (2011).

- Tan, J. J. M. “A Necessary and Sufficient Condition for the Existence of a Complete Stable Matching.” *Journal of Algorithms*, 12(1):154–178 (1991).
- Vulkan, N., Roth, A. E., and Neeman, Z. *The Handbook of Market Design*. Oxford University Press (2013).
- Wulf, W. A. “The Collaboratory Opportunity.” *Science*, 261(5123):854–855 (1993).