The Compromise Algorithm in the Swedish PhD Admissions Problem

Dingquan Miao*

Abstract
What mechanism should be designed to allocate PhD applicants to universities in Sweden? We introduce the Swedish PhD admissions problem, and it is influenced by the college admissions problem (Gale and Shapley 1962) and the student placement problem (Balinski and Sönmez 1999). In order to “solve” this problem, we design a novel mechanism, namely the compromise algorithm. We propose three theorems from this algorithm, i.e. equivalence theorems. The equivalence theorems specify the equivalence relations among stability, worse and responsiveness. Additionally, we find a positive result that the number of fields determines the strategy-proofness of the algorithm; meanwhile, the student optimal stable matching and the university optimal stable matching can be treated as special cases of our model when we restrict the number of fields. Generally, the compromise algorithm generates a stable matching that falls in between the student optimal stable matching and the university optimal stable matching.

KEYWORDS: Matching, Swedish PhD Admissions Problem, Stability, Compromise Algorithm

Supervisor: Dr. Jens Gudmundsson
Date submitted: January 18, 2016
Date examined: January 26, 2016
Discussant: Josefin Kilman

* I would like to express my deepest gratitude to my supervisor Jens Gudmundsson, for his endless patience, constant encouragement and invaluable help. His guidance and support helped me overcome many problems and finish this thesis. I would like to thank my parents for their great confidence in me all through these years.
## Contents

1 **INTRODUCTION**  
2 **MODEL AND PROPERTIES**  
   2.1 PhD Admissions Problem  
   2.2 Matching and Properties of Matching  
3 **ALGORITHMS**  
   3.1 Gale-Shapley Deferred Acceptance Algorithm  
   3.2 Compromise Algorithm  
4 **WORSE**  
5 **RESPONSIVENESS**  
6 **EQUIVALENCE**  
7 **STRATEGY-PROOF**  
   7.1 A General Case  
   7.2 A Special Case  
8 **RESPECTING IMPROVEMENTS**  
9 **CONCLUSION**  
References
List of Tables

1. PREFERENCES FOR EXAMPLE 1 ........................................ 14
2. PREFERENCES FOR EXAMPLE 2 ........................................ 24
3. MATCHING RESULTS FOR STRATEGY-PROOFNESS .......... 25
4. PREFERENCES FOR EXAMPLE 3 ........................................ 27
5. MATCHING RESULTS FOR RESPECTING IMPROVEMENTS 29

List of Figures

1. STABILITY OF COMPROMISE ALGORITHM .......................... 30
1 INTRODUCTION

With the development of the science and technology civilisation, the number of postgraduate applicants increases rapidly. Nevertheless, research institutions and departments from higher education system are generally constrained by limited educational resources, such as research funding, grants, supervisions, and so on. The opportunity of engaging the PhD program exhibits scarcity property, and the admission of the PhD program is exceptionally competitive. In order to allocate these scarce resources more efficiently, it is appealing and meaningful to investigate the following problems: who should study PhD program, who is the right person for the research institution, what system should be designed to select qualified PhD candidates and so on. In this paper we study the Swedish PhD admission problem and design a mechanism for the admission system based on its distinctive attributes. The Swedish PhD program is a vital component of the national higher education system, and it has several idiosyncrasies: (1) In Sweden, each university posts PhD positions on its websites and admits PhD student individually. There is no centralised “admission centre” to deal with the national PhD admissions, and this may cause some potential shortcomings. For example, there is no unified requirement, and the degree of competence for the admission is unlikely to be the same for different universities, but all the admitted PhD candidates are able to access education resources from different universities, such as the U6 network, MIT and etc.\footnote{The U6 network consists of universities at Göteborg, Linköping, Lund, Stockholm, Umeå, Uppsala and Stockholm School of Economics. And MIT is the abbreviation of Research School of Management and IT, which is a co-operation between twelve Swedish universities.} If PhD candidates have the right to access the education resources equally among different academic networks within the country, then a centralised admission system can fully utilise the academic resources among universities, and it is more efficient to admit students through a centralised admission centre, e.g. a stu-
dent who is not qualified for one university might be acceptable for another university. (2) The Swedish PhD programs are considered as public goods for students; meanwhile, the programs select qualified candidates as employees for the universities, since PhD programs are mainly research-based in Sweden. Therefore, we may want to take both students’ and universities’ preferences into account. (3) The Swedish PhD program is very competitive, because of its reputation of high research standard. Meanwhile, Swedish universities consider PhD candidates as employees and pay them salaries, and this will cause two effects: (i) It assists PhD candidates to focus more on their research; (ii) The salary, on the other hand, limits the number of positions offered at each university in Sweden indirectly.

In contrast with the analysis of traditional economic markets that price level determines the equilibrium of the market, many real-world situations are not consistent with traditional theory (Roth 2007). For example, the school admission problem (Gale and Shapley 1962), the marriage problem (Gale and Shapley 1962), the housing allocation problem (Shapley and Scarf 1974), and the human organs for transplants problem (Roth, Sönmez and Ünver 2004) and so on. The allocation of these cases is no longer depending on price, and it should generate an “efficient outcome” under different norms rather than the traditional equilibrium theory. A question arises for how to allocate the resources of these markets, and the abstract theory developed in the early 1960s on the basis of the college admissions problem and the marriage problem (Gale and Shapley 1962). And the solution structured a well-known matching algorithm that assigns students to universities, namely the Gale-
Shapely “deferred acceptance” algorithm. More importantly, this deferred acceptance algorithm generates optimal stable matchings (Gale and Shapley 1962). A matching assigns one set of agents (e.g. students) to another set of agents (e.g. universities), and a matching is stable if it is individually rational and there is no blocking pair at that matching. More specifically, a matching matches two agents $A$ and $B$ with $\alpha$ and $\beta$, respectively; if no agent prefers unmatched option than her current matching, then this matching is individually rational; if no pair of agents prefers each other ($A$ and $\beta$, $B$ and $\alpha$) to their partners at this matching, then there is no blocking pair at this matching. An optimal stable matching is the stable matching for one agent that is at least as well off under this matching as the agent would be in any other stable matchings. And the deferred acceptance algorithm can be classified into two ways. Specifically, for the college admissions problem: either students propose to universities, namely the student proposing deferred acceptance (henceforth SP-DA) algorithm, or universities apply to students, that is, the university proposing deferred acceptance (henceforth UP-DA) algorithm. The first case leads to the student optimal stable matching, and the latter one produces the university optimal stable matching.

However, the empirical work of Gale-Shapley algorithm was not recognized until the 1980s, and it started from the medical markets (Roth 1984). It has been empirically verified the importance of the stability from two medical markets: the U.S. and the U.K.. In the U.S., a successful centralized clearinghouse, namely the National Resident Matching Program (NRMP), which was proved to produce stable matching. It was hypothesized that the success of the NRMP was intimately correlated to the stability of the matching it produced (Roth 1984). To test the “stability hypothesis” of medical markets in the U.S., a study was made in the U.K. in the early 1990s (Roth 1991). And this empirical work verified the “stability hypothesis”, i.e. for those regions that employed algorithms generate stable matching had turned out to be successful, while, for the others had broken down in
various ways. Along with medical markets, applications in the education markets have been extensively studied. Like the medical markets, the theoretically design of assignment mechanism in the education markets is closely related to the real world applications. After the famous Gale-Shapley college admissions problem, one real-life application of the Turkish student placement problem were introduced, namely the multi-category serial dictatorship mechanism (Balinski and Sönmez 1999). Balinski and Sönmez showed (1999) that the multi-category serial dictatorship and the Gale-Shapley college optimal stable mechanisms are equivalent. Another inspired paper (Abdulkadiroğlu and Sönmez, 2003) described mechanisms applied in the U.S. at that time, for instance, Boston student assignment mechanism and Columbus student assignment mechanism. Additionally, Abdulkadiroğlu and Sönmez (2003) explained some serious shortcomings of these two mechanisms, and they suggested two alternative mechanisms that offer practical solution for those drawbacks, namely the Gale-Shapley student optimal stable mechanism and the top trading cycles mechanism (Shapley and Scarf 1974). Along with theoretical studies, popular empirical studies of Gale-Shapley and Top Trading Cycles mechanisms suggest that the efficiency can be improved by replacing the Boston mechanism with either Gale-Shapley and Top Trading Cycles mechanism (Chen and Sönmez 2006). After Abdulkadiroğlu and Sönmez published their paper (2003), New York City high schools replaced it allocation mechanism with the student-proposing deferred acceptance mechanism (Abdulkadiroğlu, Pathak and Roth 2005).

In order to deal with idiosyncrasies mentioned in the Swedish PhD admission problem, a novel mechanism is introduced in this paper, namely the compromise algorithm. This algorithm has some desirable properties, e.g. it takes both the students and universities' preferences into account and generates stable matchings. This paper illustrates the Swedish PhD admissions problem in comparison with two well-known problems, specifically, the college admissions problem (Gale and Shapley 1962) and the Turkish student
placement problem (Balinski and Sönmez 1999). Sönmez (2015) explained in his mini-course video that the “no-tie” assumption in the Turkish student placement problem is acceptable. The “no-tie” assumption indicates universities’ preferences on students are strict, alternatively, it assumes students acquire different grades in each field. Since in the very unlikely case there is a tie, and the tie becomes relevant only for the border line students, e.g. if a university has 100 slots, only the tie occurs at 100th applicant matters. And the Turkish admissions system will assign the younger student with a higher score if two students have the same score on border line. However, in the Swedish PhD admissions problem, the PhD position for each university in Sweden is more limited, and “tie” becomes more sensitive than in the Turkish student placement problem. In order to mitigate the sensitivity of “tie” in the Swedish PhD admissions problem, we adopt the lexicographic preference orders over different fields rather than (strict) preference orders over students, and the lexicographic preferences impose a less restricted assumption on students’ grades, i.e. lexicographic preferences allow students to acquire same grades in some fields as long as there is no two students achieve same grades in every fields. Note that, the structure of preference relations of the Swedish PhD problem is a generalisation of the structure preference relations of the Turkish student placement problem (Balinski and Sönmez 1999).

We present three main contributions of this paper. First, this paper introduces a new matching mechanism, namely the compromise algorithm. This algorithm is designed for the Swedish PhD admissions problem, and it generates a stable matching. The essence of this algorithm is to start from the SP-DA algorithm, and then universities whose capacities are filled will collectively reject their least preferred students; the assignments subsequently continue iterating from SP-DA algorithm until the matching is not stable, and the algorithm selects the stable matching previously adjacent to that unstable matching. The compromise algorithm eventually produces a stable matching that falls in between the student optimal stable matching and the
university optimal stable matching. Besides the compromise algorithm, three important theorems are derived from the compromise algorithm, namely the equivalence theorems. The equivalence theorems suggest the equivalence relations among stability, worse and responsiveness. The equivalence theorems offer alternative inspections for the stability of the algorithm, which will be discussed in later section. Furthermore, one minor finding is obtained from the model, that is, if we restrict the number of fields to one, the compromise algorithm generates not only stable but also strategy-proof matching, moreover, this matching is equivalent to the student optimal student matching and the university optimal stable matching.

The paper is structured as follows. We introduce the basic structures and properties of the model in Section 2. And we review the Gale-Shapely deferred acceptance algorithm and present our compromise algorithm in Section 3. In Section 4, 5, and 6, we demonstrate the equivalence theorems with relevant concepts of worse and responsiveness, and then we show how they relate to the compromise algorithm. In Section 7, we examine the properties of strategy-proofness and discuss two related cases. We investigate whether the compromise algorithm respecting improvements in Section 8. We conclude in Section 9.

2 MODEL AND PROPERTIES

2.1 PhD Admissions Problem

There is a finite set of students $S = \{s_1, \ldots, s_n\}$, and a set of universities $U = \{u_1, \ldots, u_m\}$ is defined for university colleges and universities in Sweden. A binary preference relation $R_s$ on $U \cup \{s\}$ is defined for all $s \in S$ such that, for each $\{u, u'\} \subseteq U \cup \{s\}$, $u R_s u'$ if student $s \in S$ considers $u$ is at least as desirable as $u'$. Each student $s \in S$ has a strict preference relation $P_s$ on $U \cup \{s\}$ such that, for $\{u, u'\} \subseteq U \cup \{s\}$, $u P_s u'$
if and only if $uR_u$ and $u'R_u$. A vector of student preferences $P_S = (P_{s_1}, \ldots, P_{s_n})$ where $P_{s_i}$ is the strict preference relation for student $s_i \in S$. Both types of preference relations are rational, i.e. completeness and transitivity. There is a finite set of fields $F = \{f_1, \ldots, f_k\}$, e.g. $F = \{\text{microeconomics, macroeconomics, econometrics,} \ldots\}$. For each field $f \in F$, each student $s \in S$ has a grade $g^s_f \in \mathbb{R}$. Let $g^s = (g^s_f)_{f \in F} \in \mathbb{R}^F$ be the list of $s$’s grades and $G = (g^s)_{s \in S} \in \mathbb{R}^{F \times S}$ be the matrix consists of all students’ grades in all fields. Each university $u \in U$ has a (strict) preference $P_u$ over fields. This, together with the grade matrix $G$, induces a lexicographic preference relation $P^\text{lex}_u$ over all students and “no-student option”. The no-student option is ranked prior to those students who are unqualified for university $u$. For each $u \in U$, there exists a list of thresholds $g^u = (g^u_f)_{f \in F} \in \mathbb{R}^F$ such that, for all $s \in S$, $sP^\text{lex}_u u$ if and only if there is $f \in F$ such that $g^s_f > g^u_f$ and, for each $f' \in F$ such that $f'P_u f$, $g^s_{f'} = g^u_{f'}$. And $G^T = (g^u)_{u \in U} \in \mathbb{R}^{F \times U}$ is the threshold matrix consists of all universities’ thresholds in all fields. Generally, for each $u \in U$ and $\{s, s'\} \subseteq S \cup \{u\}$, $sP^\text{lex}_u s'$ if and only if there is $f \in F$ such that $g^s_f > g^{s'}_{f'}$ and, for each $f' \in F$ such that $f'P_u f$, $g^s_{f'} = g^{s'}_{f'}$. It is assumed that for any pair of students $\{s, s'\} \subseteq S$ there exists at least one field $f \in F$ such that $g^s_f \neq g^{s'}_{f'}$. In contrast to the student placement problem (Balinski and Sönmez 1999), the no tie assumption restricts students to possess different test scores in every skill category. We allow students to achieve same grades in some fields as long as there is no two students acquire same grades in every field. Each university $u \in U$ is constrained by its limited PhD position(s), and a vector of university capacities $q = (q_{u_1}, \ldots, q_{u_m})$ where $q_u \in \mathbb{N}_+$ is the capacity of university $u$.

2.2 Matching and Properties of Matching

A matching $\mu : S \to U \cup S$ matches students with universities. With some abuse of notation, a matching is $\mu$ such that, for each $s \in S$, $\mu(s) \in U \cup \{s\}$,
and for each $u \in U$, $\mu(u) = \{s \in S | \mu(s) = u\}$. Also, $|\mu(u)| \leq q_u$. If $\mu(s) = s$, then $s$ is not assigned a university at $\mu$. If $\mu(u) = \emptyset$, then $u$ has no student at $\mu$. The set of matchings is $\mathcal{M}$. A matching $\mu \in \mathcal{M}$ is individually rational if, for each student, the student finds his assigned university at least as desirable as the no-university option at $\mu$, meanwhile, the university assets the no-student option as the least preferred option among all students at $\mu(u)$.$^4$

DEFINITION 1 (Individual rationality) A matching $\mu \in \mathcal{M}$ is individually rational at $R$ if, for each $s \in S$ and $u \in U$,

$$\mu(s)R_s s, \text{ and } \forall s \in \mu(u), sP^\text{lex}_u u.$$  

Correspondingly, a matching is wasteful if, a student prefers $u$ to her current assigned university, and she acquires sufficient grades in terms of the lexicographical threshold of $u$, and she is not admitted by $u$.

DEFINITION 2 (Wastefulness) For a matching $\mu \in \mathcal{M}$ is wasteful, if there is $u \in U$ and $s \in S$,

$$uP_s \mu(s), \text{ and } |\mu(u)| < q_u, \text{ and } sP^\text{lex}_u u.$$  

A stable matching can be defined as no individual or any pair of agents can benefit from blocking or deviating from their current matchings. A student-university pair $(s, u) \in S \times U$ is a blocking pair for $\mu \in \mathcal{M}$ if, $uP_s \mu(s)$ and there is $s' \in \mu(u)$ such that $s^\text{lex}_u s'$.

DEFINITION 3 (Stability) For a matching $\mu \in \mathcal{M}$ is stable if,

- for each $u \in U$ and $s \in S$ are individually rational, and
- there is no blocking pair.

The list of students’ preferences and colleges’ preferences, and the capacities of colleges define the college admissions problem (Gale and Shapley, 1962),

---

$^4$This section is mostly guided by Gudmundsson’s working paper (2015).
and it is denoted as \((P_S, P_U, q)\). The student placement problem (Balinski and Sönmez, 1999) consists of students’ preferences, students’ test scores, and the capacities of colleges, which can be expressed as \((P_S, G, q)\). The Swedish PhD admissions problem reveals preferences by the lexicographical order on different fields of economics, and universities rank PhD applicants based on their grades in each field of economics. The Swedish PhD admissions problem can thus be defined as \((P_S, P_U^{\text{lex}}, q)\).

3 ALGORITHMS

In this section, we firstly review the classic Gale-Shapely Deferred Acceptance Algorithm (Gale and Shapley 1962), and then introduce the compromise algorithm, since the compromise algorithm builds on the SP-DA algorithm. Moreover, we demonstrate the compromise algorithm with comprehensive example.

3.1 Gale-Shapley Deferred Acceptance Algorithm

In order to illustrate our algorithm, we need to start from introducing the Gale and Shapley’s deferred acceptance algorithm (1962). The student proposing deferred acceptance (SP-DA) algorithm is one of the main contributions studied by them and it can be explained as: each student \(s \in S\) proposes to her most preferred university. Meanwhile, each university \(u \in U\) ranks its applicant(s), if any, in a list depends on student academic performance. Specifically, \(u\) temporarily keeps the best \(q_u \in \mathbb{N}\) applicants and rejects the rest if the number of applications exceeds \(q_u\), and it keeps all acceptable offer(s) if the number of applications does not exceed \(q_u\). Each student \(s \in S\) who gets rejected in the last step proposes to her next choice, and each university again tentatively accepts at most \(q_u\) applicant(s) based on the priority from both the new applicant(s) and previous ones, and re-
jects the rest. Student who gets rejected will not apply to the same university again, and student will be matched to the no-university option if she has been rejected by all the universities she finds acceptable, and this algorithm continues progressing until no students make further proposals. This algorithm yields a student optimal stable matching that is preferred by students, and it is the worst stable matching for universities (Roth 1985). Note that, a matching \( \mu \in \mathcal{M} \) is optimal if for each student \( s \in S \) is at least as well off under the matching \( \mu \) as she would be in any other stable matching.

### 3.2 Compromise Algorithm

As mentioned in the previous section, the SP-DA algorithm generates optimal stable matching for students, while it is an “adverse” stable matching to universities. In order to take both students’ and universities’ preferences into account, we design the compromise algorithm. The algorithm starts from running a SP-DA algorithm, and yields a student optimal stable matching. Secondly, for those universities who filled their capacities can reject their least preferred applicants, and the algorithm runs the SP-DA algorithm again, and we check if it generates a stable matching. The process continues iterating two preceding steps until the unstable matching \( \mu \in \mathcal{M} \) appears, and the algorithm selects the stable matching that previously adjacent to this unstable matching \( \mu \). The compromise algorithm is as follows.

**Step 1.** Set \( k = 0 \) as an initial point, and run the SP-DA algorithm and get the student optimal stable matching \( \mu_0 \).

**Step 2.** Each university such that, \( |\mu_k(u)| = q_u \), rejects its least preferred student among the students assigned to it at \( \mu_k \), that is, for \( u \in U \), \( s \in \mu_k(u) \) is rejected if, for each \( s' \in \mu_k(u) \), \( s' \not\in s \).

**Step 3.** Each rejected student continues with her application, and the application procedure follows the SP-DA algorithm and generates matching \( \mu_{k+1} \).
Step 4. If the matching $\mu_{k+1}$ is unstable, then select $\mu_k$ and the algorithm terminates. Otherwise, increase $k$ by 1, and return to step 2.  

The following example is constructed to illustrate our algorithm progressively.

EXAMPLE 1. Let $S = \{s_1, s_2, s_3, s_4, s_5, s_6, s_7\}$, $U = \{u_1, u_2, u_3\}$, $q = (q_{u_1}, q_{u_2}, q_{u_3}) = (3, 2, 1)$, $F = \{f_1, f_2, f_3\}$. The threshold matrix and grade matrix are as follows:

$$G^T = \begin{bmatrix} g_{f_1}^{u_1} \\ g_{f_2}^{u_2} \\ g_{f_3}^{u_3} \end{bmatrix} = \begin{bmatrix} g_{f_1}^{u_1} & g_{f_2}^{u_1} & g_{f_3}^{u_1} \\ g_{f_1}^{u_2} & g_{f_2}^{u_2} & g_{f_3}^{u_2} \\ g_{f_1}^{u_3} & g_{f_2}^{u_3} & g_{f_3}^{u_3} \end{bmatrix} = \begin{bmatrix} 83 & 85 & 80 \\ 81 & 82 & 80 \\ 84 & 83 & 85 \end{bmatrix}$$

$$G = \begin{bmatrix} g_{f_1}^{s_1} \\ g_{f_2}^{s_2} \\ g_{f_3}^{s_3} \\ g_{f_4}^{s_4} \\ g_{f_5}^{s_5} \\ g_{f_6}^{s_6} \\ g_{f_7}^{s_7} \end{bmatrix} = \begin{bmatrix} g_{f_1}^{s_1} & g_{f_2}^{s_1} & g_{f_3}^{s_1} \\ g_{f_1}^{s_2} & g_{f_2}^{s_2} & g_{f_3}^{s_2} \\ g_{f_1}^{s_3} & g_{f_2}^{s_3} & g_{f_3}^{s_3} \\ g_{f_1}^{s_4} & g_{f_2}^{s_4} & g_{f_3}^{s_4} \\ g_{f_1}^{s_5} & g_{f_2}^{s_5} & g_{f_3}^{s_5} \\ g_{f_1}^{s_6} & g_{f_2}^{s_6} & g_{f_3}^{s_6} \\ g_{f_1}^{s_7} & g_{f_2}^{s_7} & g_{f_3}^{s_7} \end{bmatrix} = \begin{bmatrix} 95 & 87 & 76 \\ 87 & 85 & 90 \\ 92 & 85 & 83 \\ 85 & 77 & 80 \\ 85 & 93 & 90 \\ 83 & 87 & 95 \\ 78 & 95 & 90 \end{bmatrix}$$

And students’ and universities’ preference relations are as follows:

---

5 We use “asterisk” to denote the iteration process and the number of asterisks equals to the number of times of iteration, e.g. Step 2∗ represents repeating step 2 once.

6 Table 1.c displays the preferences orderings only for acceptable students, i.e. omitting those students who acquire insufficient grades in comparison with thresholds for each university.
Step 1: In the light of the SP-DA algorithm, the admission procedure implements as follows. Firstly, each student applies to her most preferred university, for example, $s_1$, $s_2$, $s_3$, $s_4$ propose to $u_1$, and $s_5$, $s_6$, $s_7$ propose to $u_2$. Secondly, each university is limited by its capacity $q_u$ and it declines the unacceptable applications, for instance, $u_1$ rejects $s_4$ since $g^{s_4}_{u_1} < g^{u_1}_{f_4}$, $u_2$ rejects $s_7$ since $g^{s_7}_{f} < g^{u_2}_{f_7}$. Thirdly, $s_4$ and $s_7$ move to their next preferred universities, i.e. $s_4$ and $s_7$ both apply to $u_3$; $u_3$ rejects $s_4$, owing to $s_4$ is not qualified for $u_3$ either, meanwhile, $s_7$ is tentatively accepted by $u_3$. $s_4$ continues applying to her last option $u_2$, $s_6$ is then rejected by $u_2$, due to $s_5P^{lex}_{u_2} s_4 P^{lex}_{u_2} s_6$ and $q_{u_2} = 2$; and $s_6$ moves to her next preferred university $u_1$, and $u_1$ rejects $s_2$, since $s_1P^{lex}_{u_1} s_6P^{lex}_{u_1} s_3P^{lex}_{u_1} s_2$ and $q_{u_1} = 3$. $s_2$ then goes to her next preferred choice $u_2$. Lastly, $s_4$ is rejected by $u_2$, because she has been rejected by all universities, $s_4$ is assigned to the no university option, i.e. $\mu_0(s_4) = s_4$. The rest students are assigned to their tentative positions. Therefore, Step 1 generates student optimal stable matching $\mu_0 (\mu_k$, and $k = 0)$.

$$\mu_0 = \left( \begin{array}{c} u_1 \\ u_2 \\ u_3 \\ s_1, s_3, s_6 \\ s_2, s_5, s_7 \end{array} \right)$$
Step 2: Each university \( u \in U \) such that, \( |\mu_k(u)| = q_u \), rejects its least preferred applicant at \( \mu_0 \). For example, \( u_1 \) rejects \( s_3 \) and \( u_2 \) rejects \( s_5 \), since \( s_1 P_{u_1}^{lex} s_6 P_{u_1}^{lex} s_3 \) and \( s_2 P_{u_2}^{lex} s_5 \), \( u_3 \) rejects \( s_7 \) given \( q_{u_3} = 1 \).

Step 3: Each rejected student continues with her application, and the application procedure obeys the SP-DA algorithm. For example, \( s_3 \) applies to \( u_2 \), \( s_5 \) proposes to \( u_3 \) and \( s_7 \) proposes to \( u_1 \). And this generates the following matching \( \mu_1 \).

\[
\mu_1 = \begin{pmatrix}
  u_1 & u_2 & u_3 \\
  s_1, s_6, s_7 & s_2, s_3 & s_5
\end{pmatrix}
\]

Step 4: The matching \( \mu_1 \) is not blocked by any individual and there is no blocking pair exists, hence, \( \mu_1 \) is a stable matching. The algorithm continues and starts from \( \mu_1 \) (increase \( k \) by 1), and it returns to step 2.

Step 2*: Each university \( u \in U \) such that, \( |\mu_k(u)| = q_u \), rejects its least preferred applicant at \( \mu_1 \). For example, \( u_1 \) rejects \( s_6 \) and \( u_2 \) rejects \( s_2 \), since \( s_7 P_{u_1}^{lex} s_1 P_{u_1}^{lex} s_6 \) and \( s_3 P_{u_2}^{lex} s_2 \), \( u_3 \) rejects \( s_5 \) given \( q_{u_3} = 1 \).

Step 3*: Each rejected student continues with her application, and the application procedure follows the SP-DA algorithm. For instance, \( s_6 \) and \( s_5 \) propose to \( u_3 \) and \( u_1 \), respectively; and \( s_2 \) has no university to apply, because she has been rejected by all universities. And this generates the matching \( \mu_2 \).

\[
\mu_2 = \begin{pmatrix}
  u_1 & u_2 & u_3 \\
  s_1, s_5, s_7 & s_3 & s_6
\end{pmatrix}
\]

Step 4*: The matching \( \mu_2 \) is unstable with one blocking pair \( (s_2, u_2) \), i.e. \( s_2 P_{u_2}^{lex} u_2 \) and \( u_2 P_{s_2}^{lex} s_2 \). Therefore, the algorithm terminates and we select \( \mu_1 \) such that, \( \mathcal{M}(P_S, P_U^{lex}, q) = \mu_1 \).

PROPOSITION 1. The compromise algorithm generates a stable matching that falls in between the student optimal stable matching and university
optimal stable matching.

In the following three sections, we will introduce three important theorems for the compromise algorithm, i.e. the equivalence theorems. Based on these equivalence theorems, the algorithm can be converted into two alternative versions. And Section 4 will demonstrate the first possible option.

4 WORSE

In this section, we demonstrate the first equivalence theorem, and this theorem interprets the compromise algorithm in an alternative way. Before illustrating the theorem, we can define the relevant concept of this theorem. The university will not reject its current least preferred applicant if the rejection will replace with an even less preferred applicant in the next round for the university, otherwise, the university is worse off. Suppose a student \( s \in \mu_k(u) \) is the least preferred student among all students that are matched to \( u \) at round \( k \), then university \( u \in U \) is not going to reject \( s \) if a less preferred student \( s' \) is assigned to \( u \) at next round \( k + 1 \), otherwise, the university \( u \) is worse off.

**DEFINITION 4 (Worse)** A university \( u \) is worse off at \( k+1 \) than at \( k \) if,

- there exists \( s' \in \mu_{k+1}(u) \) such that, for all \( s \in \mu_k(u), sP^*_u s' \), or
- \( \mu_{k+1}(u) \subsetneq \mu_k(u) \)

In view of the compromise algorithm, for any \( u \in U \), and for some \( s' \in \mu_k(u) \) is the least preferred student at \( \mu_k(u) \) implies \( s' \not\in \mu_{k+1} \). If \( \mu_{k+1} \) assigns \( u \) a less preferred student than \( \mu_k \), then \( u \) is worse off at \( \mu_{k+1} \) than at \( \mu_k \). In addition, \( \mu_{k+1}(u) \subsetneq \mu_k(u) \) corresponds the case where student \( s \) who has passed the threshold grades of \( u \), and \( u \) does not get any new applicant at \( \mu_{k+1} \), then \( u \) is worse off at \( \mu_{k+1} \) if \( s \) is no longer an assigned applicant for
for some $\{g^u\}_{u \in S} > g^u$, $\mu_{k+1}(u) = \mu_k(u) \setminus \{s\}$. We should note that $\mu_{k+1}(u) \subsetneq \mu_k(u)$ implies the matching $\mu_{k+1}$ is wasteful.

The definition of worse is intimately related to the compromise algorithm, and it is one of the important building blocks for the following three theorems.

**THEOREM 1.** Given $\mu_0, \mu_1, \ldots, \mu_k$ is stable for $k \geq 0$, $\mu_{k+1}$ is stable if and only if no university is worse off at $\mu_{k+1}$ than at $\mu_k$.

**Proof of Theorem 1**

**Part 1:** Given the conditions from theorem 1, prove that if $\mu_{k+1}$ is stable, then no university is worse off at $\mu_{k+1}$ than at $\mu_k$.

Assume $\mu_{k+1}$ is stable, by contradiction, suppose $u \in U$ is worse off at $\mu_{k+1}$ than at $\mu_k$. By definition of worse, we have the following two cases.

**case 1:**

there exists $s' \in \mu_k^{-1}(u)$ such that, for each $s \in \mu_k^{-1}(u)$, $sP^uxs'$.

Let $s$ be the least preferred student at $\mu_k(u)$, i.e. $\mu_k(s) = u$, and $|\mu_k(u)| = q_u$. By the design of algorithm, $s$ proposes to a less preferred university until she is not rejected.

$\Rightarrow uP_s\mu_{k+1}(s)$.

Combine with the definition of case 1, i.e. $sP^uxs'$.

$\Rightarrow (s, u) \in S \times U$ is a blocking pair to $\mu_{k+1}$, which contradicts the assumption that $\mu_{k+1}$ is stable.

Therefore, if $\mu_{k+1}$ is a stable matching then no university is worse off at $\mu_{k+1}$ than at $\mu_k$.

**case 2:**

$\mu_{k+1}(u) \subseteq \mu_k(u)$

$\Rightarrow \mu_k(u) = \mu_{k+1}(u) \cup \{s\}$, since $s$ is the least preferred student at $\mu_k(u)$.

By the design of the algorithm, $u$ is wasteful at matching $\mu_{k+1}$ such that, $|\mu_{k+1}(u)| < q_u$. And $s$ proposes to a less preferred university until she is not
rejected.

\[ s P_u^{lex} \emptyset \text{ and } u P_s \mu_{k+1}(s) \]

\[ (s, u) \in S \times U \text{ is a blocking pair to } \mu_{k+1}, \text{ which contradicts the assumption that } \mu_{k+1} \text{ is stable.} \]

Again, if \( \mu_{k+1} \) is a stable matching then no university is worse off at \( \mu_{k+1} \) than at \( \mu_k \).

**Part 2**: Given the conditions from theorem 1, prove that if no university is worse off at \( \mu_{k+1} \) than at \( \mu_k \), then \( \mu_{k+1} \) is stable.

Firstly, introduce the following notations:

\( s_0^0 := \text{least preferred student at } \mu_{k+1}(u) \),
\( s_0^0 := \text{least preferred student at } \mu_k(u) \).

(I) Assume \( u \) is not worse off at \( \mu_{k+1} \) than at \( \mu_k \),

\[ s_0^{P_{lex}} s_0^0 \]

(II) Suppose \( \mu_{k+1} \) is not stable, and there is a blocking pair \( (s, u) \in S \times U \) at \( \mu_{k+1} \) such that,

\[ s P_u^{lex} s' \text{ and } u P_s \mu_{k+1}(s) \]

(III) From the known conditions of theorem 1, \( \mu_k \) is a stable matching.

\[ \Rightarrow \text{for any pair } (s, u) \in S \times U \text{ is not a blocking pair at } \mu_k \]

\[ \Rightarrow u P_s \mu_k(s), \text{ or } s P_u^{lex} s' \]

\[ \Rightarrow \mu_k(s) R_u u, \text{ or } s P_u^{lex} s' \]

Hence, there are two cases:

**case 1**: \( \mu_k(s) R_u u \)

\[ \therefore u P_s \mu_{k+1}(s) \text{ from (II)} \]

\[ \therefore \mu_k(s) R_u u P_s \mu_{k+1}(s) \]

By the design of algorithm, and \( s \notin \mu_{k+1}(s) \) and \( s' \in \mu_{k+1}(s) \),

\[ \Rightarrow s \text{ has been rejected by } u \text{ before } \mu_{k+1} \]

**case 1.1**: \( s \in \mu_k(u) \)
case 1.1.1:
\( s = s'' \)
\( \therefore s'R_u s'' \) from (I)
\( \therefore s'P_u s \)
\( \Rightarrow \) contradicts \( sP_u s' \) from (II)

**case 1.1.2:**
\( s \neq s'' \)
By the design of the algorithm, \( s \) has been rejected because \( u \) has received better proposals and is able to fill its capacity with better student at \( \mu_{k+1} \).
\( \Rightarrow s'R_u s \)
\( \Rightarrow \) contradicts \( sP_u s' \) from (II)

**case 2:**
\( s \notin \mu_k(u) \)
By the design of the algorithm, \( s \) has been rejected because \( u \) has received better proposals and is able to fill its capacity with better student at \( \mu_{k+1} \).
\( \Rightarrow s'R_u s \)
\( \Rightarrow \) contradicts \( sP_u s' \) from (II)

Hence, if no university is worse off at \( \mu_{k+1} \) than at \( \mu_k \), \( \mu_{k+1} \) is stable.

**case 2:**
\( s \sim P_u s' \)
\( \therefore u \) is not worse off at \( \mu_{k+1} \) than at \( \mu_k \), and \( s \sim P_u s' \) holds at \( \mu_{k+1} \)
\( \therefore (s, u) \in S \times U \) is not a blocking pair for \( \mu_{k+1} \)

Again, if no university is worse off at \( \mu_{k+1} \) than at \( \mu_k \), then \( \mu_{k+1} \) is stable.
\( \therefore \) For \( k \geq 0 \), \( \mu_k \) is stable, \( \mu_{k+1} \) is stable if and only if no university is worse off at \( \mu_{k+1} \) than at \( \mu_k \).
5 RESPONSIVENESS

Now, we proceed to the second equivalence theorem, similar with the previous one, it offers an intuitive translation of the compromise algorithm, and necessary concept is explained in this section. A responsive preference is applied when university’s capacity is greater than one, and it is a preference relation orders the sets of students rather than individual students, i.e. for any university u’s preference is responsive if, only one student is different under two different matchings, and it prefers the set consists more preferred student. Let $P_u^{LEX}$ denotes the responsive preferences to u’s lexicographic preference relation, $P_u^{lex}$.

**DEFINITION 5 (Responsiveness)** For a transitive preference relation, $P_u^{LEX}$, is responsive to $P_u^{lex}$ if,

$$\forall T \subseteq S, s \in S \setminus T \text{ and } s' \in S,$$

$$s P_u^{lex} s' \text{ and } s R_u^{lex} u \text{ implies } T \cup \{s\} P_u^{LEX} T \cup \{s'\}.$$ 

**THEOREM 2.** For $\mu \in U$, $\mu_k(u) P_u^{LEX} \mu_{k+1}(u)$ if and only if $u$ is worse off at $\mu_{k+1}$ than at $\mu_k$.

**Proof of Theorem 2**

**Part 1 :** Given the conditions from theorem 2, prove that $\mu_k(u) P_u^{LEX} \mu_{k+1}(u)$ if no university is worse off at $\mu_{k+1}$ than at $\mu_k$.

Assume for each $u \in U$ is not worse off at $\mu_{k+1}$ than at $\mu_k$. By definition of worse, we have the following two cases.

**case 1** $[s' \neq \emptyset]$ : 
there exists $s' \in \mu_{k+1}(u)$ such that, 
for each $s \in \mu_k^{-1}(u)$, $s P_u^{lex} s'$.
Let $s$ be the least preferred student at $\mu_k(u)$, i.e. $\mu(s) = u$, and $|\mu_k(u)| = q_u$.

By the definition of algorithm, and the definition of worse.

$\Rightarrow \mu_k(u) \setminus \{s\} = \mu_{k+1} \setminus \{s'\}$

Let $T$ be the complement of both sets, i.e.

$\Rightarrow T = \mu_k(u) \setminus \{s\} = \mu_{k+1} \setminus \{s'\}$
$\Rightarrow T \cup \{s\} = \mu_k(u)$ and $T \cup \{s'\} = \mu_{k+1}(u)$

$\therefore s^P_{u\rightarrow s'}$ and $s' \neq \emptyset$

$\therefore T \cup \{s\} P_u^{LEX} T \cup \{s'\}$

$\Leftrightarrow \mu_k(u) P_u^{LEX} \mu_{k+1}(u)$

Hence, $\mu_k(u) P_u^{LEX} \mu_{k+1}(u)$ if no university is worse off at $\mu_{k+1}$ than at $\mu_k$.

**Case 2** $|s'| = 0$:

$\mu_{k+1}(u) \subseteq \mu_k(u)$

$\Rightarrow \mu_k(u) = \mu_{k+1}(u) \cup \{s\}$, since $s$ is the least preferred student at $\mu_k(u)$.

By the design of the algorithm,

$\Rightarrow s^P_{u\rightarrow \emptyset}$,

By the definition of responsiveness,

$\Rightarrow \mu_{k+1}(u) \cup \{s\} P_u^{LEX} \mu_{k+1}(u) \cup \emptyset$

$\Rightarrow \mu_k(u) P_u^{LEX} \mu_{k+1}(u)$

Again, $\mu_k(u) P_u^{LEX} \mu_{k+1}(u)$ if no university is worse off at $\mu_{k+1}$ than at $\mu_k$.

**Part 2**: Given the conditions from theorem 2, prove that $u \in U$ is worse off at $\mu_{k+1}$ than at $\mu_k$ if $\mu_k(u) P_u^{LEX} \mu_{k+1}(u)$.

Firstly, introduce the following notations:

$s' :=$ least preferred student at $\mu_{k+1}(u)$,

$s :=$ least preferred student at $\mu_k(u)$.

Assume $\mu_k(u) P_u^{LEX} \mu_{k+1}(u)$.

By the design of algorithm, let $T = \mu_k(u) \setminus \{s\} = \mu_{k+1}(u)$.

$\Rightarrow T \cup \{s\} = \mu_k(u)$ and $T \cup \{s'\} = \mu_{k+1}(u)$

$\Rightarrow T \cup \{s\} P_u^{LEX} T \cup \{s'\}$
\[ \Rightarrow s^{Plex} s' \]

Suppose for each \( u \in U \) is not worse off at \( \mu_{k+1} \) than at \( \mu_k \),
\[ \Rightarrow s'^{Plex} s \]
\[ \Rightarrow \text{contradicts the assumption that } \mu_k(u) P^{LEX} \mu_{k+1}(u). \]
Hence, \( u \) is worse off at \( \mu_{k+1} \) than at \( \mu_k \) if \( \mu_k(u) P^{LEX} \mu_{k+1}(u) \).
\[ \therefore \mu_k(u) P^{LEX} \mu_{k+1}(u) \text{ if and only if } u \in U \text{ is worse off at } \mu_{k+1} \text{ than at } \mu_k. \]
We have now established the first two theorems, and we summarise all three equivalence theorems in next section.

6 EQUIVALENCE

The theorem 1 and theorem 2 have presented that there exist equivalence relationships among stability, worse and responsiveness. For example, theorem 1 states that \( \mu_k \) is stable and \( \mu_{k+1} \) is not stable is equivalent to saying \( u \in U \) is worse off at \( \mu_{k+1} \) than at \( \mu_k \), and theorem 2 implies \( u \) is worse off at \( \mu_{k+1} \) than at \( \mu_k \) is equivalent to \( \mu_k(u) P^{LEX} \mu_{k+1}(u) \). We can thus infer that there is an equivalence relationship between stability and responsiveness.

**THEOREM 3.** Given \( \mu_0, \mu_1, \ldots, \mu_k \) is stable for \( k \geq 0 \), for each \( u \in U \), \( \mu_{k+1} \) is not stable if and only if \( \mu_k(u) P^{LEX} \mu_{k+1}(u) \).

**Proof of Theorem 3**

For \( u \in U \), and based on the compromise algorithm, the following two theorems have been proved. From theorem 1, given \( \mu_k \) is stable for \( k \geq 0 \), \( \mu_{k+1} \) is stable if and only if no university is worse off at \( \mu_{k+1} \) than at \( \mu_k \).
Equivalently, given \( \mu_k \) is stable for \( k \geq 0 \), \( \mu_{k+1} \) is not stable if and only if for some \( u \) is worse off at \( \mu_{k+1} \) than at \( \mu_k \). From theorem 2, \( \mu_k(u) P^{LEX} \mu_{k+1}(u) \)
if and only if for some \( u \) is worse off at \( \mu_{k+1} \) than at \( \mu_k \). Therefore, by the above two theorems, given \( \mu_k \) is stable for \( k \geq 0 \), \( \mu_{k+1} \) is not stable if and only if \( \mu_k(u) \text{P}_{\text{LEX}} \mu_{k+1}(u) \).

The above theorems (Theorem 1, Theorem 2, and Theorem 3) are termed equivalence theorems, and the equivalence theorems offer us alternative ways to check the stability of the compromise algorithm. In some circumstances, it is more straightforward to verify if some university is worse off or has responsive preferences at two adjacent matchings. For example, according to theorem 1, the compromise algorithm can be interpreted as follows. The compromise algorithm runs the SP-DA algorithm first, and produces a student optimal stable matching. And then we check if each university such that, \( |\mu_k(u)| = q_u \), wants to reject its least preferred applicant in such a way that, rejection will not make it worse off at \( \mu_{k+1} \) than at \( \mu_k \). The algorithm thus can be illustrated as a series of iterative SP-DA processes with a terminating point such that, not all universities want to reject their least preferred applicants. We leave to the reader to verify that, by equivalence theorems, we can terminate the compromise algorithm at \( \mu_{k+1}(u) \), if \( \mu_k(u) \text{P}_{\text{LEX}} \mu_{k+1}(u) \).

7 STRATEGY-PROOF

7.1 A General Case

A mechanism is strategy-proof if reporting the true preference is a dominant strategy for everyone (Roth 1985). In consequence of the compromise algorithm, university can benefit from misrepresenting its preferences, but it is less obvious whether the matching \( \mathcal{M}(P_S, P^U_{\text{LEX}}, q) \) is immune to students’ manipulation. We can test whether the property of strategy-proof is tenable for compromise algorithm by following example.
EXAMPLE 2. This example is the same as EXAMPLE 1, except that $s_7$ untruthfully reports her preference orderings, and the new preference relations for $s_7$ is $u_2 \overset{\sim}{P}_s u_3, s_7 \overset{\sim}{P}_s u_1$. And students and universities’ preference relations are as follows:

$$
egin{array}{cccccccc}
P_{s_1} & P_{s_2} & P_{s_3} & P_{s_4} & P_{s_5} & P_{s_6} & \overset{\sim}{P}_{s_7} \\
\begin{array}{cccccc}
u_1 & u_1 & u_1 & u_1 & u_2 & u_2 \\
u_3 & u_2 & u_2 & u_3 & u_1 & u_3 \\
u_2 & u_3 & u_3 & u_2 & u_1 & u_3 & -
\end{array}
\end{array}
\quad
\begin{array}{cccc}
P_{u_1} & P_{u_2} & P_{u_3} & P_{u_4} \\
\begin{array}{cccc}
f_2 & f_1 & f_3 & s_7 \\
f_1 & f_2 & f_1 & s_5 \\
f_3 & f_3 & f_2 & s_6 \\
- & - & - & -
\end{array}
\end{array}
$$

(a) Students’ Preferences on $U$

(b) Universities’ Preferences on $F$

(c) Universities’ Lexicographic Preferences on $S$

Table 2: PREFERENCES FOR EXAMPLE 2

Step 1: Student $s_1$, $s_2$, $s_3$ and $s_4$ propose to university $u_1$, student $s_5$, $s_6$, $s_7$ propose to university $u_2$, and the SP-DA algorithm assigns each student with at most one university. Therefore Step 1 generates student optimal stable matching $\mu_0$ ($\mu_k$, and $k = 0$).\(^7\)

$$
\mu_0 = \begin{pmatrix}
u_1 & u_2 & u_3 \\
s_1, s_3, s_6 & s_2, s_5 & s_7
\end{pmatrix}
$$

Step 2: Each university such that, $|\mu_k(u)| = q_u$, rejects its least preferred applicant at $\mu_0$. For example, $u_1$ rejects $s_3$ and $u_2$ rejects $s_5$, since $s_4 \overset{\text{lex}}{P}_{u_1} s_6 \overset{\text{lex}}{P}_{u_3} s_3$ and $s_2 \overset{\text{lex}}{P}_{u_5} s_5$, $u_3$ rejects $s_7$ given $q_u = 1$.

Step 3: Each rejected student applies to her next preferred university, and the application procedure follows the SP-DA algorithm, and this generates the following matching $\mu_1$.

\(^7\)Student $s_4$ is assigned to the no university option, i.e. $\mu_0(s_4) = s_4$. 

24
\[ \mu_1 = \begin{pmatrix} u_1 & u_2 & u_3 \\ s_1, s_6 & s_2, s_3 & s_5 \end{pmatrix} \]

Step 4: The matching \( \mu_1 \) is unstable, since \( u_1 \) is worse off at \( \mu_1 \) than \( \mu_0 \) according to equivalence theorems, i.e. \( s_3^{P_{u_1}} u_1 \). The algorithm terminates and we select \( \mu_1 \) such that, \( \mathcal{M}(P_S(\tilde{P}_{s_7}), P^{lex}_U, q) = \mu_0 \).

The matching results of true and truthless preferences of \( s_7 \) are presented in Figure 3. From the results, \( s_7 \) is able to manipulate results of compromise algorithm by reporting the untruthful preferences. For example, at \( \mathcal{M}(P_S, P^{lex}_U, q) \), \( s_3, s_5 \) and \( s_7 \) are assigned to \( u_2, u_3 \) and \( u_1 \), respectively. while, at \( \mathcal{M}(P_S(\tilde{P}_{s_7}), P^{lex}_U, q) \), \( s_3, s_5 \) and \( s_7 \) move to \( u_1, u_2 \) and \( u_3 \), respectively. And other students’ matchings are unchanged. From Table 3(b), \( s_3, s_5 \) and \( s_7 \) are assigned to more preferred choices.

<table>
<thead>
<tr>
<th>( \mathcal{M}(P_S, P^{lex}_U, q) )</th>
<th>( s_1 )</th>
<th>( s_2 )</th>
<th>( s_3 )</th>
<th>( s_4 )</th>
<th>( s_5 )</th>
<th>( s_6 )</th>
<th>( s_7 )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( u_1 )</td>
<td>( u_2 )</td>
<td>( u_2 )</td>
<td>( u_3 )</td>
<td>( u_1 )</td>
<td>( u_1 )</td>
<td>( u_1 )</td>
<td>( u_1 )</td>
</tr>
</tbody>
</table>

| \( \mathcal{M}(P_S(\tilde{P}_{s_7}), P^{lex}_U, q) \) | \( u_1 \) | \( u_2 \) | \( u_1 \) | \( u_3 \) | \( u_2 \) | \( u_1 \) | \( u_3 \) |

(a) Matching Results for Students in EXAMPLE 1 and EXAMPLE 2

<table>
<thead>
<tr>
<th>( P_{s_3} )</th>
<th>( P_{s_5} )</th>
<th>( P_{s_7} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>( u_1 )</td>
<td>( u_2 )</td>
<td>( u_2 )</td>
</tr>
<tr>
<td>( u_2 )</td>
<td>( u_3 )</td>
<td>( u_3 )</td>
</tr>
<tr>
<td>( u_3 )</td>
<td>( u_1 )</td>
<td>( u_1 )</td>
</tr>
</tbody>
</table>

(b) Preferences for Some Students on \( U \)

Table 3: MATCHING RESULTS FOR STRATEGY-PROOFNESS
7.2 A Special Case

If we restrict the number of fields to 1, that is, the number of fields set $F$ only contains one element. Based on the structure of the universities’ preferences, all universities have the same preference orderings over students $S$. In this case, the student optimal stable matching is equivalent to the university optimal stable matching, and we can verify this matching is the unique strategy-proof and stable matching over the preference domain. Alcalde and Barberà (1994) formally proved this result, and the result states that if the preferences of one side of agents satisfy the top dominance condition, and the preferences of the other side of agents are unrestricted, then the matching rule generates a unique stable and strategy-proof matching on admissible preference profiles. For any pair of preferences $P^{lex}_u, P^{lex}_{0u}$ for university $u \in U$, and any two students $s$ and $s'$ that are preferred to $u$, if $s$ is the most preferred student at $P^{lex}_u$, and $s'$ is the most preferred student at $P^{lex}_{0u}$, then this type of preferences orderings is top dominance. When we limit the number of fields to 1, each university have only one possible order of preferences, and this order satisfies top dominance, meanwhile, the preferences of the students are unrestricted. No one can manipulate the matching results by misrepresenting their preferences when $|F| = 1$. More generally, the number of fields equals to total number of possible relations.

PROPOSITION 2. The compromise algorithm is generally not strategy-proof, however, there is one unique stable and strategy-proof matching if $|F| = 1$.

8 RESPECTING IMPROVEMENTS

An algorithm is respecting improvements if no student ever receives a less preferred assignment on account of she performs better in grades. Balinski and Sönmez (1999) show that the SP-DA algorithm is the unique stable
admissions mechanism respects improvements. It is indispensable to check the respecting improvements of the compromise algorithm, and we can test if the compromise algorithm respects improvements by the following example.

EXAMPLE 3. This example is the same as EXAMPLE 1, except that $s_5$ deliberately scores worse in $f_3$, and $g_{f}^{s_5} = ( g_{f_1}^{s_5} \ g_{f_2}^{s_5} \ g_{f_3}^{s_5} ) = ( 85 \ 93 \ 87 )$.

And students and universities’ preference relations are as follows:

Table 4: PREFERENCES FOR EXAMPLE 3

<table>
<thead>
<tr>
<th>$P_{s_1}$</th>
<th>$P_{s_2}$</th>
<th>$P_{s_3}$</th>
<th>$P_{s_4}$</th>
<th>$P_{s_5}$</th>
<th>$P_{s_6}$</th>
<th>$P_{s_7}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$u_1$</td>
<td>$u_1$</td>
<td>$u_1$</td>
<td>$u_2$</td>
<td>$u_2$</td>
<td>$u_2$</td>
<td>$u_2$</td>
</tr>
<tr>
<td>$u_3$</td>
<td>$u_2$</td>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_3$</td>
<td>$u_1$</td>
<td>$u_3$</td>
</tr>
<tr>
<td>$u_2$</td>
<td>$u_3$</td>
<td>$u_2$</td>
<td>$u_1$</td>
<td>$u_3$</td>
<td>$u_1$</td>
<td>$u_1$</td>
</tr>
</tbody>
</table>

(a) Students’ Preferences on $U$

<table>
<thead>
<tr>
<th>$P_{u_1}$</th>
<th>$P_{u_2}$</th>
<th>$P_{u_3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$f_2$</td>
<td>$f_1$</td>
<td>$f_3$</td>
</tr>
<tr>
<td>$f_3$</td>
<td>$f_2$</td>
<td>$f_1$</td>
</tr>
<tr>
<td>$f_1$</td>
<td>$f_2$</td>
<td>$f_3$</td>
</tr>
</tbody>
</table>

(b) Universities’ Preferences on $F$

<table>
<thead>
<tr>
<th>$p_{lex}^{u_1}$</th>
<th>$p_{lex}^{u_2}$</th>
<th>$p_{lex}^{u_3}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_7$</td>
<td>$s_1$</td>
<td>$s_6$</td>
</tr>
<tr>
<td>$s_5$</td>
<td>$s_3$</td>
<td>$s_2$</td>
</tr>
<tr>
<td>$s_1$</td>
<td>$s_2$</td>
<td>$s_7$</td>
</tr>
<tr>
<td>$s_5$</td>
<td>$s_6$</td>
<td>$s_7$</td>
</tr>
<tr>
<td>$s_3$</td>
<td>$s_4$</td>
<td>$-$</td>
</tr>
<tr>
<td>$s_6$</td>
<td>$s_5$</td>
<td>$s_4$</td>
</tr>
<tr>
<td>$s_2$</td>
<td>$s_6$</td>
<td>$-$</td>
</tr>
<tr>
<td>$-$</td>
<td>$-$</td>
<td>$-$</td>
</tr>
</tbody>
</table>

(c) Universities’ Lexicographic Preferences on $S$

Step 1: Student $s_1$, $s_2$, $s_3$ and $s_4$ propose to university $u_1$, student $s_5$, $s_6$, $s_7$ propose to university $u_2$, and the SP-DA algorithm assigns each student with at most one university. As a result, Step 1 generates student optimal stable matching $\mu_0$ ($\mu_k$, and $k = 0$).

$$\mu_0 = \begin{pmatrix} u_1 & u_2 & u_3 \\ u_1 & u_2 & u_3 \\ s_1, s_3, s_6 & s_2, s_5 & s_7 \end{pmatrix}$$

Step 2: Each university $u \in U$ such that, $|\mu_k(u)| = q_u$, rejects its least preferred applicant at $\mu_0$. For example, $u_1$ rejects $s_3$ and $u_2$ rejects $s_5$, since

---

8 Table 1.c displays the preferences orderings only for acceptable students, i.e. omitting those students who acquire insufficient grades in comparison with thresholds for each university.

9 Student $s_4$ is assigned to the no university option, i.e. $\mu_0(s_4) = s_4$. 
s_1 p_{u_1}^{lex} s_6 p_{u_1}^{lex} s_3 and s_2 p_{u_2}^{lex} s_5, u_3 rejects s_7 given q_{u_3} = 1.

**Step 3:** Each rejected student applies to her next preferred university, and the application procedure obeys the SP-DA algorithm, and this produces the following matching $\mu_1$.

$$\mu_1 = \begin{pmatrix} u_1 & u_2 & u_3 \\ s_1, s_6, s_7 & s_2, s_3 & s_5 \end{pmatrix}$$

**Step 4:** The matching $\mu_1$ is unstable, since $u_3$ is worse off at $\mu_1$ than $\mu_0$ according to equivalence theorems, i.e. $s_7 p_{u_1}^{lex} s_5$. The algorithm terminates and we select $\mu_1$ such that, $M(P_\ast(g_f^{s_5}), P_{U}^{p_{u_1}^{lex}}, q) = \mu_0$.

**Proposition 3.** The compromise algorithm does not respect improvements.

The matching results of respecting improvements are presented in Table 5. From the results, $s_5$ is assigned to her most preferred choice because of a lower grade in $f_3$, therefore, the compromise algorithm does not respect improvements. As a consequence of worse performance of $s_5$, students $s_3$, $s_5$ and $s_7$ are assigned to more preferred choices than before, e.g. $s_3$ moves from $u_2$ to $u_1$, $s_5$ moves from $u_3$ to $u_2$ and $s_7$ prefers new assigned $u_3$ to $u_1$ (Table 5(b)). The proposition 3 is consisting with the results from the student placement problem (Balinski and Sönmez 1999), that is, the SP-DA algorithm is the unique stable admissions mechanism respects improvements.
\[ \mathcal{M}(P_S, P_{U}^{lex}, q) \]
\[ \mathcal{M}(P_S(g_{f}^{s}), P_{U}^{lex}, q) \]

(a) Matching Results for Students in EXAMPLE 1 and EXAMPLE 3

\[
\begin{array}{ccccccccc}
 & s_1 & s_2 & s_3 & s_4 & s_5 & s_6 & s_7 \\
\hline
u_1 & u_2 & u_2 & u_0 & u_3 & u_1 & u_1 \\
u_1 & u_2 & u_1 & u_0 & u_2 & u_1 & u_3 \\
\end{array}
\]

(b) Preferences for Some Students on \( U \)

\[
\begin{array}{ccc}
P_{s_3} & P_{s_5} & P_{s_7} \\
\hline
u_1 & u_2 & u_2 \\
u_2 & u_3 & u_3 \\
u_3 & u_1 & u_1 \\
\end{array}
\]

Table 5: MATCHING RESULTS FOR RESPECTING IMPROVEMENTS

9 CONCLUSION

The Swedish PhD program is a vital component of national higher education system, meanwhile, it supports students with salaries. Even the compromise algorithm is neither strategy-proof (generally) nor respecting improvements, it produces stable matching that compromise the students and universities’ preferences. The most important desideratum for a matching is stability, and the compromise algorithm does lead to a stable matching \( \mathcal{M}(P_S, P_{U}^{lex}, q) \).

What makes this stable matching differ from other matchings? We can analyse the matching results produced from EXAMPLE 1 and compare with typical matchings generated from the Gale-Shapley deferred acceptance algorithm, i.e. the student optimal stable matching and the university optimal stable matching.
\[
\begin{array}{cccc}
\mu_s = \mu_0 & \mathcal{M}(P_S, P_U^{lex}, q) = \mu_1 & \mu_u \\
\mu_s = \mu_0 & \mathcal{M}(P_S, P_U^{lex}, q) = \mu_1 & \mu_u \\
\mu_s = \mu_0 & \mathcal{M}(P_S, P_U^{lex}, q) = \mu_1 & \mu_u \\
\mu_s = \mu_0 & \mathcal{M}(P_S, P_U^{lex}, q) = \mu_1 & \mu_u \\
\end{array}
\]

(a) Matching Results of EXAMPLE 1

(b) A Path of Stability

Figure 1: STABILITY OF COMPROMISE ALGORITHM

We use \(\mu_s\) to denote the student optimal stable matching, which is equivalent to \(\mu_0\) at the compromised algorithm. And \(\mu_u\) signifies the university optimal stable matching, and compromise algorithm generates stable matching \(\mathcal{M}(P_S, P_U^{lex}, q)\), i.e., \(\mu_1\) in EXAMPLE 1. The matching results can be seen from Figure 1, each university is assigned with different students at different matchings, and compromise algorithm produces a stable matching that follows the direction from the SP-DA algorithm to UP-DA algorithm (Figure 1(b)). Particularly, the SP-DA algorithm creates the stable matching \(\mu_s\) that is optimal for students, and the UP-DA algorithm produces the stable matching \(\mu_u\) that is optimal for universities, whereas the compromise algorithm generates a compromise stable matching that is between \(\mu_s\) and \(\mu_u\).

We should notice that there is one special case of the strategy-proofness. If the number of fields \(|F| = 1\), then compromise algorithm satisfies the top dominance condition (Alcalde and Barberà 1994), and it creates a unique stable and strategy-proof matching that is equivalent to the student optimal stable matching and the university optimal stable matching. Alternatively, if \(|F| = 1\), then \(\mu_s = \mathcal{M}(P_S, P_U^{lex}, q) = \mu_u\).

The equivalence theorems are major findings from the compromise algorithm, and they endow the algorithm with alternate options. Essentially, we
can determine whether to terminate the algorithm by checking its stability, worse or responsiveness.

This thresholds factor generalises the student placement problem, it allows each university to rank the no student option prior to some students at its preference orderings. The related issues of thresholds factor can be further studied, and one possible implication is students will not apply to those universities that she is not qualified, since she will be rejected for sure. Specifically, suppose each student has the symmetric information about the thresholds of each university, and she knows whether she is qualified for the basic requirements for each university before the application, and she will only apply to universities that she is both qualified and desired. Two questions arise from the above problem: What will change for the compromise algorithm under the above case? What if students access to asymmetric information set?

Lastly, this paper tries to design a mechanism that captures attributes of the real-world Swedish PhD admissions problem, and it aims to offer a solution such that, both students and universities are “satisfied” from the matching produced by the algorithm.
References


