LUND UNIVERSITY
School of Economics and Management

A Marriage Market for Economists

Stability and Optimality

A Bachelor's Essay in the Field of Microeconomics and Market Design

Department of Economics, Lund University

NEKH*02*-VT*19*

29th of May *2019*

Author: Ludvig Andersson

199410148374

Supervisor: Professor Tommy Andersson

Contents

Abstract

For a labor market, such as the one for economists, an employer often faces the difficulty of distinguishing attainable from unattainable candidates. Offers, rejections and the recruiting process overall is costly, and participants of the market need to choose who to pursue wisely. EconMatch is a site aiming to provide suggestions on what candidate is the best to pursue for a university, based on a matching algorithm and the market's preferences. Some concerns, however, regarding EconMatch, the algorithm it is based on, its performance on the labor market and possibilities for strategic behavior have been expressed. However, the essay shows that potential strategic conduct does not seem worthwhile because of how the algorithm is constructed, and even less so when considering the non-binding property of the matching. Simulations show that market size seem to have a negative impact on the probability of achieving the highest preference match for candidates. This is likely due to competition and increasingly complex combinatorics. However, considering the algorithm in the bigger picture may be more appropriate. This essay proposes to consider EconMatch as a tool for communicating information, rather than a traditional matching assignment, thus, reducing asymmetric information and facilitating credible signaling.

Keywords: EconMatch, Matching Theory, Strategy, Algorithm, Asymmetric Information

Chapter 1

Introduction and Background

1.1 Market design and Matching Theory

Roth (2018) describes market design as, although being an ancient human activity, it is a relatively new part of economics striving to understand how the design of a market relates to its functioning. With this understanding economists could come closer to the building of new marketplaces or re-designing of already existing, but dysfunctional marketplaces. For markets such as commodity markets, money and price will do all the work of deciding who will get what, and commodities may be transacted without any knowledge of one's counterparty. In comparison, a matching market as described by Roth (2018) is a market in which *"you can't just choose what you want, even if you can afford it: you also must be chosen"*, i.e. every agent care with whom they're dealing with. Obviously, prices cannot do all the work in this case. Matching theory is a particular subject within market design dealing with these kinds of markets. Matching theory is a branch of discrete mathematics within game theory that aims to describe mutually beneficial assignments over time and how these are formed. There are different kinds of matching markets depending on the properties of the market. A first categorization of markets is the one-sided- and the two-sided matching market. One-sided implies that only one set of agents exist who are to be matched to each other within the set. Two-sided, however, means two disjoint set of agents who are to be matched to agents of the opposite set. Additionally, the matching can be divided according to conditions for capacity, where many-to-one matching implies that one side of the market has the possibility of matching to more than one agent, and one-to-one matching is a matching in which each agent can only match to one other agent. This essay will in particular treat so called one-to-one-two-sided matching markets, in which both sides of the matching choose and must be chosen, such as the labor market.

1.2 The Market for Economists — Thickness, Congestion and Safety

The theme of this essay is the one-to-one and two-sided matching labor market for new economists. The annual meeting of the *Allied Social Sciences Associations* (ASSA), held for a few days and organized by the *American Economic Association* (AEA), is amongst other things a venue during which new PhD economists meet university employers for interviews (American Economic Association, 2019). PhD economists (henceforth candidates) in different fields of economics has the opportunity to be interviewed for one or multiple positions at different universities during the days of which the event is held. Once the interviews have been held and the event is over, universities will determine which candidates to pursue for fly-outs[1]. However, there are some problems connected to this. While such a centralized market for economists helps make it thick, i.e. bringing together many participants, both employers and

[1] *A fly-out is when a candidate visits a university at its location for further interviews. Usually travel expenses incurred are paid by the university.*

job seekers, thick markets can however suffer from congestion (Roth, 2018). Congestion means that while there are a lot of participants, there is not enough time for effectively making, rejecting or accepting transactions. This congestion leads to a strategic problem for universities, which is the one of which candidates a university wants and which they can get (Roth, 2018). In episode 769 called *"Speed Dating for Economists"* from the podcast Planet Money (2017) this issue is discussed. In the episode, a representative of a university tells the reporters that the university must not only decide which candidates they want, but also which they can get. That is, sometimes, giving a great interview and being highly recommended, actually reduces your chances of being hired for a certain position. The reason for this is simple. A fly-out, a rejection of an offer or the recruiting procedure overall is costly, so if a university believes a candidate to be highly desirable for other, more prestigious, universities then that university might assume that candidate unattainable and thus not pursue him or her. This is obviously a problem, since that candidate might very well be interested in the university that now won't pursue him or her because of being "too good". There's also another side of the same problem, which may be more intuitive. While some universities neglect attainable but "too good" candidates, some pursue good but unattainable candidates, wasting resources. These problems originate from candidates not being able to credibly signal their interest to universities, which in turn leads to the herculean task for universities of determining a specific candidate's attainability. The consequence of this information not being communicated is inefficiency on the market, such as advantageous matches left unrealized and, thus, less advantageous matches realized. The strategical game and decision-making as a result from congestion continues when offers are sent out. More often than not, offers are not likely to be on the table for more than a limited time which may be because of congestion. If a university expects to have to make several offers to fill a single position, but only can make one at a time, then by sending out many offers in a short amount of time gives the opportunity to catch candidates before they have committed to other positions (Roth, 2018). Another reason, according to Roth (2018), is one less beneficent, which is to pressure candidates into committing before they have time to receive another preferable offer. These deadlines and offers being not in sync clearly is a problem for both parties. Candidates are deprived of the opportunity to consider multiple offers, and the negative externality for other employers experiencing a very short time during which candidates are available. It's clear that, as it is, the sets of strategies for participants is large and the congestion leads to the lack of another important property of a functional market, i.e. that of safety. Safety refers to a market in which agents feel secure to make decisions based on their best interests, rather than attempting to game a flawed system (Roth cited in article from Harvard Business School by Nobel, 2010). Gaming a flawed system may include strategies such that some universities might, for example, find that more talented candidates can be found if you search amongst those who "fell through the cracks" of the event (Roth, 2018).

1.3 EconMatch

One site aiming to address these types of issues is EconMatch. EconMatch is a recommender system that suggests which candidates an employer should consider pursuing after the above mentioned interviews at ASSA (EconMatch, 2019). The recommender system is based on a modified version of an algorithm called the Deferred Acceptance Algorithm. The modification, according to EconMatch (2019) themselves, is to help accommodate for the fact that universities and candidates have evolving preferences as they learn more about each other in stages. The algorithm and the modification will be thoroughly presented in the following chapters.

1.3.1 EconMatch?

At the moment, however, it is clear that EconMatch is not overall understood among users or others interested in the subject. Several questions can be found concerning what it is, how it works and its implications, on forums such as Economics Job Market Rumors (2018) and the American Economic Association forum (2019). Some concerns belong to the subject of data privacy and security, e.g. encryption of passwords on the site. Others, which are the ones of concern for this essay, are more related to the market in which EconMatch works, and the algorithm which is used. A post written by Thayer Morrill and Umut Dur (2019) in the American Economic Association forum will serve as an example for the concerns regarding market and algorithm. Morrill and Dur begins by clearly stating that the post is not to be confused with a complete analysis of the issue and continue by presenting their standpoint, that any system proposed for a job market matching needs to be carefully analyzed in every detail. The opinion is partly based on a quote by 2012 Nobel Laureate in Economics, Alvin Roth: *"Market design involves a responsibility for detail, a need to deal with all of a market's complications, not just its principle [sic] features"*. Morrill and Dur recognizes the problem addressed by EconMatch as legitimate, i.e. the issue for universities of distinguishing unattainable candidates from attainable ones. They present the nature of the recommender system's problem by considering two cases, of which only the second is actually relevant, since the first one is regarding binding assignments. They state that, although being a suggestion of which candidate to pursue, it's not clear whether it is a good suggestion or not. The problem is that of partial participation, i.e. that not every university and every candidate participates in the matching system. They wrap up their post by the concluding remarks that they do not recommend people to participate in the match since the issue of partial participation implies that people would not know the best way to rank their options or whether to follow the system's recommendation, which may be hurtful for the users of EconMatch.

1.4 Purpose and Method

The aim of this essay is to, first of all, analyze what EconMatch is in terms of the algorithm it is based on, and their own description of their purpose. The description of how EconMatch works is based on their presentation of the algorithm, since no formal code or otherwise of the algorithm is available. Further, the essay will also strive to answer and give perspective on the questions concerning whether to participate or not, based on potential issues of strategic behavior and the impact of market size. To determine impact of market size, I have coded a simulation program which works according to how EconMatch itself describes its algorithm. In this program I simulate different market sizes and perform at most one thousand simulations, each with randomized preference lists, for each market size. The results are analyzed in terms of who gets what, i.e. do candidates generally obtain, according to their preferences, good matches? How does market size affect this result[2]? Additionally, in this essay, a theoretical approach will be used for determining the possible implications of the specific market analyzed, and the fact that EconMatch uses a modified algorithm. Already existing studies on incentives and possibilities for strategic behavior in a matching market based on a stable mechanism are applied to EconMatch, to determine the relevance of these incentives, and their potential implications.

[2]Not to be confused with the analysis in Roth and Peranson (1999) which showed that the potential benefit for any agent to be non-truthful in terms of preferences tends to zero when increasing the market size, while holding the length of preference lists constant.

Chapter 2

The One-to-One Matching Model

This chapter introduces first of all notations and definitions for the matching market. These will allow for a formal presentation and analysis of a matching market and its structure, the properties of a matching algorithm and lastly the results of this essay. Some background of the Deferred Acceptance Algorithm will then be briefly presented, and concluding this chapter is a step-by-step summary of the run of this algorithm. Thus, the chapter will form the one-to-one matching model of this essay.

2.1 Notations and Definitions

There are two finite and disjoint sets of agents: a set $C = \left\{C_1^{FS}, ..., C_\alpha^{FS}\right\} = \{c_1, ..., c_m\}$ of candidates and a set $U = \left\{U_1^{FS}, ..., U_\beta^{FS}\right\} = \{u_1, ..., u_n\}$ of universities. Each set U and C can be divided into subsets called field-sets (Definition 1.2) C_α^{FS} and U_β^{FS} where α and β denotes the number of field-sets. Each $c \in C$ has a strict linear ordering $>_c$ over $U \cup \{c\}$, i.e. every element is comparable, no agent is indifferent between two or more elements and the ordering satisfies transitivity, i.e. if $u \ >_c \ u'$ and $u' \ >_c \ u''$ then $u \ >_c \ u''$ due to transitivity. These preferences are ordered within a preference list $P(c)$. Each $u \in U$ also has a strict linear ordering $>_u$

over $C \cup \{u\}$ which indicates preferences of u, which provides the preference list $P(u)$. These notations $(C,\ U,\ (>_i)_{i\in C\cup U})$ will together define a one-to-one matching market. Note that the strict linear ordering $>_c$ over $U \cup \{c\}$ allows for c to be matched with him or herself rather than to any $u \in U$, which is what Roth (2007), Coles (2009), Zanardo (2016) and many more denotes being unmatched. For example, $c >_c u$ is to be interpreted as c preferring to be unmatched rather than being matched to u. For any $u \in U$ and $c \in C$ a candidate is said to be acceptable if $c >_u u$ and a university is acceptable if $u >_c c$, respectively. For the case of EconMatch, there is no possible way for a candidate/university to find an agent of the opposite set unacceptable or acceptable, unless it's mutual, i.e. $c_i >_{c_i} u_j \Leftrightarrow u_j >_{u_j} c_i$, which is a result of how the matching system is constructed. This property will be called mutual- unacceptability or acceptability in this essay.

Denote a specific market for universities and candidates by the triple $(C,\ U,\ P)$, where C and U is the set of candidates and universities, respectively. P denotes the set of preference lists:

$$P = P(C,\ U) = \left\{P(C_1^{FS}), ..., P(C_\alpha^{FS}),\ P(U_1^{FS}), ..., P(U_\beta^{FS})\right\}$$
$$= \{P(c_1), ..., P(c_n),\ P(u_1), ..., P(u_m)\}$$

2.1.1

Definition 1.1 (Matching) A matching μ, as defined by Roth (2007), is a one-to-one correspondence from $C \cup U \rightarrow C \cup U$ where:

$$\mu(c) \in U \cup \{c\}\ \ for\ all\ c \in C \tag{1a}$$

$$\mu(u) \in C \cup \{u\} \ for \ all \ u \in U \tag{1b}$$

$$\mu(\mu(i)) = i \ for \ all \ i \in C \cup U \tag{1c}$$

The matching μ provides as its output the identity of the input's partner, i.e. $\mu(i)$ refers to agent i's partner. (1a) and (1b) formally states that for every candidate $c \in C$ a match needs to be either an agent of the opposite set or alternatively, him or herself. Likewise, for every university $u \in U$ a match has to be either an agent of the opposite set or a match with itself. Also, as stated in (1c), if c is u's match, then u must be c's match.

Definition 1.2 (Field-set) To make certain assumptions in terms of correlation in preference lists more intuitive, and help make possible a more in-depth analysis of the market, as well as come closer to a real life situation of EconMatch, I will define so called field-sets. As its name implies the intuition behind it is that certain candidates specialize in certain fields, and certain universities search for candidates in certain fields. Some of these fields may overlap (e.g. Macro and Finance for Macro/Finance) and some are disjoint (e.g. Asian agriculture economics and Nordic microeconomics). Each disjoint set C and U can be divided into one or multiple field-sets, C_α^{FS} and U_β^{FS}, with α and β denoting the number of field-sets. Note that, field-sets themselves do not overlap, such that $C_i^{FS} \subseteq C_j^{FS}$. However, a field-set may be defined by an existing overlap in fields of research.

The conditions for field-sets are as follows:

$$\forall C^{FS} \in C;\; C^{FS} \subseteq C \tag{2a}$$

$$\forall P(c) \in P(C^{FS});\; P(c)' \in P(C^{FS}) = \varnothing \tag{2b}$$

$$u \in P(C^{FS}) \Leftrightarrow C^{FS} \in P(u) \tag{2c}$$

$$\alpha = \beta \wedge \neg\Box(|U| = |C|) \wedge \neg\Box(\left|U_i^{FS}\right| = \left|C_i^{FS}\right|) \tag{2d}$$

Condition (2a) states that any field-set belonging to a main set is a subset of the latter. In the simplest of cases one field-set would be equal to the main set, which would be the equivalent of there not being any field-sets. Condition (2b) is important as it defines which agents belong to what field-set. It states that every preference list belonging to a set of preference lists within a field-set need to contain the same elements. Basically, every agent within a field-set need to have preferences over the same agents from the opposite set. However, this does not mean that they have the same preferences. Condition (2c) states that if university u is to be found in the preference lists of the candidates $c \in C^{FS}$ then these candidates must be found in the preference list of that university $P(u)$ as well, which is a result of the mutual-unacceptability and acceptability mentioned above. Given condition (2d), the number of field-sets in both U and C is the same, though the number of agents in the main sets, or two corresponding field-sets, need not be equal. This may imply that a field set C^{FS} or U^{FS} might be empty, depending on which main-set lacks agents within that certain field. Conditions are written for candidates, but they apply symmetrically for universities. See below for a visual representation of an arbitrary matching market $(C,\ U,\ P)^{\theta}$. Note that a field-set is not contained within one box in itself, but rather the frames of two separate boxes. Additionally, the last field-set of universities is empty, leaving candidates such as m and others in that field-set to only list n in their preference lists.

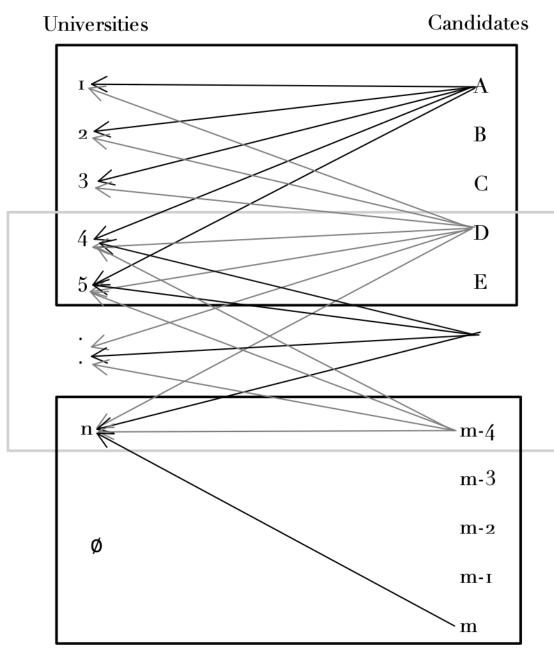


Figure 2.1: Matching Market $(C,\ U,\ P)^{\theta}$

2.2 The Deferred Acceptance Algorithm

In the U.S., during the early 1900's, the market in which graduates from medical school, i.e. job-seekers, meet hospitals, the employers, was in great part decentralized

(The Royal Swedish Academy of Sciences, 2012). Scarcity of medical students in this market led to a Prisoner's Dilemma problem in which hospitals offered employment increasingly early, and sometimes even several years before the student's graduation. Offers were sent to students who had yet to prove their qualifications, and even before students knew what branch of medicine they would like to practice (The Royal Swedish Academy of Sciences, 2012). In response to the issues hampering the market, a centralized market mechanism was adopted in 1951, which in large part resolved the problems (Roth, 1984). The important properties of this centralized market mechanism was, however, not known until the work of David Gale and Lloyd Shapley in 1962 (Roth, 1984). Roth (1984) shows that this market mechanism, called the NIMP (National Intern Matching Program) algorithm, is closely related to that presented by Gale and Shapley, called the Deferred Acceptance Algorithm, introduced more than ten years later, in *The American Mathematical Monthly* in 1962. The Deferred Acceptance Algorithm was in part presented in the context of a two-sided marriage matching market, in which men and women each having preferences over individuals of the opposite set, were to be matched with each other (Gale & Shapley, 1962). The run of the algorithm was based on having agents on one side of the market make proposals to the opposite set in order of preferences. The proposal-receiving side would in turn hold the most preferred proposal, which becomes their tentative match, and reject the rest. The tentative matches become the final matches, i.e. the algorithm stops, once no new proposals are made (Gale & Shapley, 1962). Using this algorithm, Gale and Shapley (1962) showed that there always exists at least one stable assignment (stable such that there are no incentives for divorce or infidelity) in such a market, and given strict preferences, there always exist, for each side of the market, an optimal stable matching for those agents belonging to that side. Optimality was defined as, the optimal stable matching being at least as good as any other stable matching for that side of the market (Roth, 2007).

Below is a step-by-step summary of the candidate proposing Gale-Shapley Deferred Acceptance Algorithm (henceforth DA-algorithm), in the context of a market for candidates and universities as presented above in Notations and Definitions. The summary is influenced by Roth's (2007) description of the algorithm with the exclusion of a step 0, arbitrarily break ties, that would only be useful if any preferences were not strict, which is not the case for the EconMatch application.

Every $i \in C \cup U$ is initially unmatched and free.

[Step 1.a] Every $c \in C$ proposes to its most preferred $u \in U$ among those for which $u \succ_c c$ holds. If no such $u \in U$ exist, c is matched to him or herself, i.e. unmatched.

[Step 1.b] Every $u \in U$ that received one proposal or more reviews and rejects all but one $c \in C$, namely, the proposal the most preferred. In such a case c and u are now tentatively matched to each other

…

[Step t.a] Every $c \in C$ proposes to its most preferred $u \in U$ among those for which $u \succ_c c$ holds, and who has not rejected him or her. If no such $u \in U$ exist, c is matched to him or herself, i.e. unmatched.

[Step t.b] Every $u \in U$ that received one proposal or more reviews the new proposals and its current tentative match and rejects all but one $c \in C$, namely, the proposal the most preferred. In such a case c and u are now tentatively matched to each other.

The algorithm ends when no candidate $c \in C$ is rejected, and the tentative matching becomes the final result.

Scheme 1: The Deferred Acceptance Algorithm — Step by Step

Chapter 3

Theoretical framework and related literature

In this chapter the theoretical framework for analyzing EconMatch will be presented. The theorems, first presented, will provide insight about important properties of a stable mechanism, i.e. the DA-algorithm, and the market in which it performs. Further, some strategic moves conditioned upon how a matching market is structured and how a stable mechanism is constructed, as well as limits and incentives for these strategic behaviors will be presented to allow for an analysis and discussion of strategic acting in the EconMatch matching.

3.1 Theorems

Below is a presentation of existing theorems that are relevant to the case of EconMatch and will be of use for analyzing the matching conducted by EconMatch. Some theorems are presented in terms of men and women, which was how they originally were introduced. However, for this essay it should be understood as candidates and universities, respectively. A marriage is to be interpreted as a "match" or an "assignment" and further, a marriage market as a one-to-one and two-sided matching market.

3.1.1

Theorem 1 (Gale and Shapley, 1962; Roth, 2007) *There always exist a stable set of marriages within a marriage market.*

As defined by Roth (2007), a matching μ is said to be stable if:

$$\forall i \in C \cup U; \ \mu(i) \geqslant_i \ i \tag{3a}$$

and,

$$\nexists (c,\ u) \in C \times U \ such\ that\ c >_u \ \mu(u)\ and\ u >_c \ \mu(c) \tag{3b}$$

Condition (3a) is a result of individual rationality. Assume on the contrary $i \geqslant_i \ \mu(i)$, then i would prefer being unmatched. However, as there is the option of remaining unmatched, by not attending/giving an interview, individual rationality is a minimum requirement for a matching to not be dissolved. Condition (3b) states that there is no pair $(c,\ u)$ that would rather be matched to one another, but is not. If on the contrary, $\exists i \in C \cup U\ for\ which\ i >_i \ \mu(i)$, or a pair $(c,\ u)$ that would prefer being matched to each other, but is not, i would be a blocking individual and $(c,\ u)$ would form a blocking pair. If there does exist a blocking individual or pair, the matching is not stable (Roth, 2007). The proof that there always exist a stable assignment in a marriage market follows from the observation that the DA-algorithm always stops, as no man proposes twice to the same woman, and the matching produced through this algorithm is itself stable (Roth, 2007), since no blocking pair (or blocking individual) can arise from the procedure.

Theorem 2 (Gale and Shapley, 1962; Roth, 2007) *In the case of strict preferences for all men and women, there always exist a man-optimal stable match, and a women-optimal stable match. The man-optimal stable match is achieved under man-proposing DA-algorithm, and symmetrically for the women-optimal stable match.*

Denote a candidate-optimal stable matching μ^C, and a university-optimal stable matching μ^U, resulting from the DA-algorithm. With this theorem, under the candidate-proposing DA-algorithm, optimality implies that the resulting match $\mu^C(c) \geqslant_c \mu(c)$ where $\mu(c)$ is any other stable matching. If we inverse the roles, i.e. having the universities proposing instead, then symmetrically we would achieve the university-optimal stable matching. Also, worth noting is that any stable matching μ is Pareto efficient with respect to all participating agents, i.e. there is no matching λ such that:

$$\lambda(i) \geqslant_i \ \mu(i) \ for\ all\ i \ \in C \cup U \tag{4a}$$

and,

$$\lambda(k) >_k \ \mu(k) \ for\ some\ k \in C \cup U \tag{4b}$$

To be convinced, assume the contrary. Suppose that $\lambda(c) >_c \ \mu(c) \ for\ some\ c \in C$ according to (4b). Because of individual rationality $\lambda(c) \in U$, denoted u. From (4a) we get that $c \geqslant_u \ \mu(u)$. Since $\lambda(c) \neq \mu(c)$ then $\lambda(u) \neq \mu(u)$, implying $c >_u \ \mu(u)$, which contradicts the property of stability of μ as c and u in that case would form a blocking pair. Considering optimum purely from the perspective of the proposing candidates (and not both sets of agents), μ^C will only be strongly Pareto efficient relative any other stable match (Erdil & Ergin, 2017). Once allowing for unstable matchings, Roth (2007) shows in an example that it is possible to achieve an unstable matching ν such that $v(c) >_c \ \mu^C(c)$ for some $c \in C$ and $v(c) \sim_c \ \mu^C$ for other $c \in C$, which shows that the weak Pareto optimality cannot be strengthened to strong Pareto optimality once considering only the utility of the proposing party and allowing unstable matchings. Gale and Sotomayor (1990) also proves that there is no matching, stable or not, that is an improvement from the man-optimal stable matching for all men, i.e. also suggesting only weak Pareto efficiency in this case. Additionally, a stable matching μ only implies Pareto efficiency if indifferences are not incorporated in the model (Erdil & Ergin, 2017). However, for this paper non-strict preferences will not be relevant.

Theorem 3 (Knuth, 1976) *The set of stable marriages is a distributive lattice, for which the maximum element is the man-optimal stable match and the minimum element is the women-optimal stable match, with respect to the strict partial order of men.*

In this lattice of stable assignments, the candidate-optimal stable match will be the worst stable match for universities and vice versa. In a market with preferences that leads to $\mu^C = \mu^U$, then this is the only one unique stable match possible. A result of this theorem is that the set of unmatched men and women is the same at every stable matching (McVitie and Wilson, 1970 cited in Roth, 2007).

Theorem 4 (Roth, 2007) *No stable matching mechanism exists for which stating the true preferences is a dominant strategy for every agent. Anyone who does not match with an agent to whom they would be matched with at his or her optimal stable match can potentially manipulate a stable mechanism.*

Roth (2007) considers a case in which a woman can, by manipulating through truncation, turn the man-optimal stable matching into becoming a woman-optimal stable matching, simply by truncating her list below her woman-optimal stable match partner. A woman can, however, only benefit from lying if she has more than one stable partner. This kind of truncation of lists will not be possible for the case of EconMatch. However, Teo, Sethuraman and Tan (2001) shows that a woman can not only manipulate a stable mechanism by truncating her list, but potentially also by permuting her preference list, although this does not always guarantee her woman-optimal stable partner.

Theorem 5 (Roth and Sotomayor, 1990 cited in Roth, 2007) *Under any stable mechanism in a marriage market with strict preferences, and more than one stable matching, then at least one agent can profitably misrepresent his or her preferences assuming others tell the truth.*

Roth (2007) explains that any agent that matches with a partner less preferred than his or her optimal stable matching partner, can manipulate by truncating the list right after his or her optimal stable partner, though it is possible to construct the mechanism such that one side of the market can never do better than to state their true preferences.

Theorem 6 (Dubins and Freedman, 1981) *No agent of the proposing party can do better by misrepresenting his or her true preferences under the Gale-Shapley DA-algorithm, i.e. it is a weakly dominant strategy for any proposer to be truthful.*

The Gale-Shapley DA-algorithm is a mechanism, as mentioned in Theorem 5, such that one side of the market, the proposers, can never do better by not being truthful. Dubins and Freedman (1981) further shows that no collusion of several proposers under the DA-algorithm, each misreporting preferences, can lead to an improvement for all proposers with respect to their true preferences.

3.2 Strategy and Cheating

There are several papers to be found analyzing incentives for acting strategically within a matching market based on a stable mechanism such as the DA-algorithm. Acting strategically means that an agent itself or in coalition acts in a way such that he or she may achieve a better outcome than in the case of truthful acting. Below is a presentation of some of these. Note that by Theorem 6 these strategies will be purely from the perspective of the non-proposing party.

- *Manipulating by permutation of preference lists (single-agent).*

Manipulation through permutation implies that an agent of the non-proposing party is able to permute its preference list to potentially achieve a more desirable matching partner. A simple example will show how this could be achieved.

Table 3.1: Candidates' truthful preference lists (proposing party)

Agents/Ranks	Pref. 1	Pref. 2	Pref. 3	Pref. 4
c_1	u_1	u_4	u_2	u_3
c_2	u_1	u_3	u_2	u_4
c_3	u_2	u_3	u_1	u_4
c_4	u_2	u_4	u_1	u_3

Table 3.2: Universities' truthful preference lists (non-proposing party)

Agents/Ranks	Pref. 1	Pref. 2	Pref. 3	Pref. 4
u_1	c_3	c_2	c_1	c_4
u_2	c_1	c_4	c_3	c_2
u_3	c_2	c_3	c_1	c_4
u_4	c_4	c_1	c_3	c_2

Working through these preference lists along the candidate-proposing DA-algorithm, the final result of this matching is

Table 3.3: Resulting match

Pairwise matching	u_1:[c_2]	u_2:[c_4]	u_3:[c_3]	u_4:[c_1]

Now, let's assume that u_1 finds it in its interest to state a non-truthful preference list, i.e. to permute its truthful one. It turns out that this university could change its preference list such that

$$c_3 >_{u_1} c_1 >_{u_1} c_2 >_{u_1} c_4$$

And through this achieve a final result

Table 3.4: Final result manipulated execution

Pairwise matching	u_1:[c_3]	u_2:[c_4]	u_3:[c_2]	u_4:[c_1]

By letting c_1 and c_2 change place in its truthful preference list, u_1 now got to match with its most preferred candidate, c_3. The explanation is rather simple. From the truthful matching result, we see that u_3 matches with u_1's most preferred candidate, and vice versa. Thus, by ranking u_3's most preferred lower than truthfully so, u_1 is able to let go of c_2, who proposes to u_3, which in turn lets go of c_3, who proposes to u_1. Garg and Vaish (2017) defines an inconspicuous optimal manipulation as being stable with respect to the truthful preferences and being nearly identical to the true preference list. More formally, the manipulated list should be derived from the true preference list by moving at most one element. Above example turns out to be an inconspicuous optimal manipulation.

- *Manipulation through an accomplice*

Bendlin and Hosseini (2019) presents a new manipulation strategy which they call manipulation through an accomplice. The strategy is that a non-proposing agent teams up with an accomplice from the proposing party who manipulates on its behalf, to obtain a better match for the non-proposing agent. The authors state that the single-non-proposing-agent strategies, such as the one described above, may be limited such that there might be other, more desirable, matches available, should the agent get help from the proposing side. Obviously, the strategy should not be regretful and result in a worse situation for the accomplice, i.e. the accomplice matching with a less preferred partner, with manipulation, than under the proposer-optimal stable matching with truthful preferences. If that would be the case, then the strategy will fall altogether since there would not exist a willing accomplice. Bendlin and Hosseini (2019) introduces an algorithm (Algorithm 1) which returns the best non-regretful permutation strategy for an accomplice manipulating for a non-proposing agent. The authors prove in their Theorem 2 that, using Algorithm 1, the strategy of manipulating through an accomplice can in some instances, for the non-proposing agent, lead to a more preferred allocation than the single-non-proposing-agent manipulation.

- *Manipulation when having a capacity greater than one.*

Aziz, Seedig and von Wedel (2015) considers not only the case of manipulation in a one-to-one matching market but also manipulation in a many-to-one matching market, i.e. a market in which one of the sets has the possibility of matching to more than one from the opposite set. The authors show that the potential benefit from a manipulation comes from that the manipulating agent receives proposals from agents under a manipulated execution, otherwise not received, so called new proposals. This, in turn, is only achieved by the misrepresenting of preferences by the manipulating agent, as everyone else is assumed to be truthful. Further, the only way misrepresentation can cause new proposals, is if it leads to the manipulating agent rejecting some proposals it would not reject during a truthful run of the DA-algorithm. However, as a rejection in itself does not directly lead to a new proposal, it needs to trigger a chain of rejections which leads to new proposals for the manipulating agent. Comparing these conditions, although these being presented for a many-to-one matching market, to the example above, it's easy to see that this is exactly the case when a non-proposing agent manipulates through permutation in a one-to-one matching market as well. The authors provide an example of a many-to-one matching market where one non-proposing agent has a capacity greater than one, and can successfully manipulate the algorithm. They state that, not only does there exist incentives for manipulation, but also, given full insight into other agents' preference lists there exists efficient algorithms to find such a successful misreport.

- *Manipulation through truncation.*

Manipulation through truncation as potentially successful has been proved, as seen in Theorem 4 and 5. I will now show a simple example taken from Roth (2007) that illustrates the strategy (this example, however, replaces men by candidates and women by universities). Assume two candidates and two universities with the corresponding preference lists

Table 3.5: Candidates' truthful preference lists (proposing)

Agents/Ranks	Pref. 1	Pref. 2
c_1	u_1	u_2
c_2	u_2	u_1

Table 3.6: Universities' truthful preference lists (non-proposing)

Agents/Ranks	Pref. 1	Pref. 2
u_1	c_2	c_1
u_2	c_1	c_2

The final match result from the DA-algorithm would then be according to Table 3.7 below, which is the candidate-optimal stable matching.

Table 3.7: Final match result

Pairwise matching	u_1:[c_1]	u_2:[c_2]

Now let's assume that u_1 instead chooses to report c_1 as unacceptable, i.e. simply truncating its list after its most preferred c_2. Then c_1 would not be able to propose to u_1, and would therefore propose to u_2 already in the first iteration. Now $u_2$would reject c_2 and hold c_1 without further ado since it is its most preferred, forcing c_2 to propose to u_1. This would lead to the following matching result

Table 3.8: Final result from manipulated execution

Pairwise matching	u_1:[c_2]	u_2:[c_1]

Which is the university-optimal stable matching. This truncation leads to both u_1 and u_2 being better off by turning the matching into becoming university-optimal (as if the universities were proposing) rather than candidate-optimal.

3.2.1 Limits and Incentives for Strategic Behavior

In Roth and Peranson (1999) as well as Roth (2007) it can be understood that what determines whether there exist possibilities for manipulating one's preference list in order to achieve a better outcome is if that agent has more than one stable matching. This means that in the lattice of stable matchings, this agent needs to have at least two different partners depending on which matching within the lattice is produced. Therefore, to determine incentives and potentially successful manipulations, the number of stable matchings is a relevant starting point. Roth and Peranson (1999) states that, one factor which strongly influences the size of the set of stable matchings is that of the correlation of preferences among proposers and non-proposers, respectively. The size of the set of stable matchings decreases as correlation increases, and if perfectly correlated, only one unique stable matching exists, i.e. $\mu^C = \mu^U$. Roth and Peranson (1999) further shows that the size of the market in relation to the length of preference lists also plays a critical role for the size of the set of stable matchings. The authors consider a case in which preferences are uncorrelated, and so the set of stable matchings is initially large. Further, to make the case more realistic, they state that a proposer cannot possibly be interviewed by every non-proposer, and every non-proposer cannot possibly conduct interviews for every proposer. This implies limited lengths on preference lists. Now, assume k to be the number of agents a proposer can fit into his or her preference list, and n to be the number of agents in the total market. With this, Roth and Peranson (1999) shows that even with completely uncorrelated preferences, as k/n becomes small, the size of the set of stable matchings becomes small. Teo, Sethuraman and Tan (2001) further show that once only complete preference lists are allowed, and thus ruling out the possibility for truncation and so for a non-proposer of remaining single, significantly reduces the ability of a non-proposer to change the outcome by cheating. In an example where six out of the eight existing non-proposers could successfully truncate their lists, the authors show that once only misrepresentation through order reversal is allowed, no non-proposer could successfully cheat. Teo, Sethuraman and Tan (2001) ran the DA-algorithm 1,000 times for a market size of $n = 8$ and found the following result.

Table 3.9: Result from Teo, Sethuraman and Tan (2001)

Number of women who benefited	0	1	2	3	4	5	6	7	8
Number of observations	740	151	82	19	7	1	0	0	0

As can be seen from above table, in 74 % of the runs the proposer-optimal solution is the non-proposers' only option, no matter how they cheat. The authors took the same approach for a larger scale of $n = 100$ and found that the average number of non-proposers who benefited from cheating was 9.515%. The conclusion was that the chances that a typical non-proposer is, first of all, able to collect perfect information of everyone else's preferences (which is of course needed), and further perform a successful manipulation are slim.

Chapter 4

Results and Analysis

In this chapter, as its name implies, results and analyses of these will be presented. First of all, how EconMatch works, in terms of algorithm, modification and its output, is shown. In section 4.2 I will show what the modification, as described by EconMatch, as well as the introduction of field-sets for this essay, may or may not imply for the stable mechanism based on the theorems above. Additionally, section 4.2 will provide an analysis of potential strategic behavior for participants in EconMatch both in technical terms, and based on the nature of EconMatch and the recommender system. Lastly, section 4.3 will present the method of the simulation program and how it is built. The results of the simulation will then finally be presented and analyzed.

4.1 How does EconMatch work?

The result from the recommender system, on which EconMatch is based, is for the universities a ranked list of matches in order of likely match quality, instead of the original single-set-of-pairs-matching resulting from a non-modified DA-algorithm (EconMatch, 2019). The modification is easily described with a simple example of a market consisting of three candidates and three universities.

Table 4.1: Candidates' preference lists (proposing)

Agents/Ranks	Pref. 1	Pref. 2	Pref. 3
c_1	u_1	u_2	u_3
c_2	u_1	u_2	u_3
c_3	u_2	u_3	u_1

Table 4.2: Universities' preference lists (non-proposing)

Agents/Ranks	Pref. 1	Pref. 2	Pref. 3
u_1	c_1	c_2	c_3
u_2	c_3	c_2	c_1
u_3	c_1	c_3	c_2

Table 4.3: First run results

Pairwise matching	u_1:[c_1]	u_2:[c_3]	u_3:[c_2]

Running the Deferred Acceptance Algorithm on the basis on above preference lists results in the above first run results, i.e. this is the result from the non-modified candidate-proposing DA-algorithm. Now, however, EconMatch removes from each preference list the agent they matched with in previous run(s) and re-run the algorithm. Below are the following results from the second and the third run.

Table 4.4: Second run results

Pairwise matching	u_1:[c_2]	u_2:[c_1]	u_3:[c_3]

Table 4.5: Third run results

Pairwise matching	u_1:[c_3]	u_2:[c_2]	u_3:[c_1]

At this point EconMatch will provide the corresponding ranked list to each university, which in this case would be according to the table below.

Table 4.6: Ranked lists for universities

Rank/Agents	u_1	u_2	u_3
1	c_1	c_3	c_2
2	c_2	c_1	c_3
3	c_3	c_2	c_1

The interpretation for these lists are, for example for u_1: *You should pursue c_1, however, should this for some reason not work out (because of evolving preferences), pursue c_2 etc.*

4.2 Theory

In this section stability and optimality for the case of EconMatch is analyzed. Additionally, the potential for successful strategic acting on the basis of the structure of the market, and the nature of the recommender system is analyzed. The section will provide insight on the concerns of EconMatch, in terms of strategy and the issue of partial participation.

4.2.1 The Algorithm, Stability and Optimality

EconMatch uses a modified candidate-proposing DA-algorithm. It is therefore necessary to take a look at whether this modification has any implications on the previously proved properties of the original DA-algorithm. Additionally, having for the case of this essay introduced field-sets, it is also appropriate to analyze the possible implications of these.

The Impact of the Modification

The modification that EconMatch introduces allows for presenting not only one single-pair matching, but rather a ranked list on expected matching quality for the participating universities, as shown in Table 4.6. On EconMatch's (2019) site they provide information on the modification, such that, once pairs has been established in the first run they, from each and every one's preference list, delete the matching partner and continues with a second run. The modification is rather intuitive, since it gives a more realistic dynamic to the matching, allowing for evolving preferences.

Assume a market that in the first run of the one-to-one correspondence $C \cup U \rightarrow C \cup U$ allows every agent to, in its preference list, rank every agent of the opposite set. Further, assume a fairly low number of participants for the sake of simplicity, e.g. $n = 3 \; and \; m = 3$. The preference lists and first run results are as follows:

Table 4.7: Candidates

Agents/Ranks			
c_1	u_1	u_2	u_3
c_2	u_2	u_3	u_1
c_3	u_1	u_3	u_2

Table 4.8: Universities

Agents/Ranks			
u_1	c_2	c_3	c_1
u_2	c_1	c_3	c_2
u_3	c_3	c_2	c_1

Table 4.9: First run results

Pairwise matching	u_1:[c_3]	u_2:[c_1]	u_3:[c_2]

For the second run I refer back to Table 4.7 and Table 4.8, and delete (shown in **bold**) from each preference list the match from the first run such that:

Table 4.10: Candidates, second run

c_1	u_1	$\mathbf{u_2}$	u_3
c_2	u_2	$\mathbf{u_3}$	u_1
c_3	$\mathbf{u_1}$	u_3	u_2

Table 4.11: Universities, second run

u_1	c_2	$\mathbf{c_3}$	c_1
u_2	$\mathbf{c_1}$	c_3	c_2
u_3	c_3	$\mathbf{c_2}$	c_1

Table 4.12: Second run results

Pairwise matchning	u_1:[c_1]	u_2:[c_2]	u_3:[c_3]

Working through the modified preference lists above, the second run results are according to Table 4.12.

For instability in the second run, a blocking pair needs to be found. Referring back to Equation (3b), a situation in which $\exists(c,\ u) \in C \times U$ *such that* $c >_u \ \mu(u)$ *and* $u >_c \mu(c)$ holds true, is the only possible way to break stability. We can immediately confirm that u_3, c_1 and c_2 cannot be part of a blocking pair, respectively, since they all got their most preferred partner in the second run. For the remaining u_1, u_2 and c_3 we can simply see that there is no mutual higher preference between two non-matched agents. Note that the match from previous run is irrelevant for the second run, since should any agent prefer their first run match, then luckily that match will be interpreted as the best expected match quality in the EconMatch modified DA-algorithm anyhow. The mutual deletion of agents is practically the same as mutual unacceptability introduced in Notations and Definitions above.

Although above example does not serve as a general proof, every run and iteration is according to the DA-algorithm, which in itself produces stable assignments. Thus, Theorem 1 holds true in this modification, and since it remains a stable mechanism DA-algorithm, it follows that Theorem 2 holds true, i.e. the outcome of the second run is still candidate-optimal. Since the result remains to be the maximum element of the distributive lattice of stable matchings, i.e. being candidate-optimal, the remaining Theorems also holds true. Though a non-mutual truncation, i.e. non-mutual unacceptability, is impossible.

From above, it follows that the introduction of mutual unacceptability does not have an impact on the properties of the stable mechanism of the DA-algorithm.

The Impact of Field-Sets

For the labor market for economists it is most likely unrealistic to assume that every university interviews every candidate. As the market gets thicker, congestion likely increases, leading to that both candidates and universities need to prioritize and choose wisely which interviews to take or give. Additionally, all candidates are probably not doing research in the same field, and all universities are likely not searching to fill positions in the same field either. Because of these factors, a more realistic approach is to introduce field-sets defined in Definition 1.2. However, it's appropriate to understand what the implications are for a regular matching market, such as the marriage market, to be divided into field-sets, and what this in turn may or may not imply for the properties of a stable mechanism.

First of all, referring to Equation (2b) every candidate in a certain field-set, say C_1^{FS}, has at least one thing in common with the other candidates in that field-set, which is that they all have preferences over the same universities. Likewise, for the agents of the corresponding field-set for the universities, U_1^{FS}, $P(u \in U_1^{FS}) = P(U_1^{FS})$ needs to hold. However, since research fields may overlap, $C_1^{FS} \cup U_1^{FS} \rightarrow C_1^{FS} \cup U_1^{FS}$ will not necessarily be the entire truth for C_1^{FS}[1]. Suppose a market, $(C,\ U,\ P)^{\omega}$, with three field-sets on each side of the market[2], i.e. $C_{1-3}^{FS}\ and\ U_{1-3}^{FS}$, where $\{c_1,\ c_2\} \in C_1^{FS},\ \{c_3\} \in C_2^{FS}\ and\ \{c_4,\ c_5\} \in C_3^{FS}$ and symmetrically for universities.

From the perspective of the candidates (symmetrically for universities) the following one-to-one correspondence per field-set is faced:

$$C_1^{FS} \cup U_1^{FS} \cap U_2^{FS} \rightarrow C_1^{FS} \cup U_1^{FS} \cap U_2^{FS}\ for\ C_1^{FS}$$

$$C_2^{FS} \cup U_1^{FS} \cap U_2^{FS} \cap U_3^{FS} \rightarrow C_1^{FS} \cup U_1^{FS} \cap U_2^{FS} \cap U_3^{FS}\ for\ C_2^{FS}$$

$$C_3^{FS} \cup U_2^{FS} \cap U_3^{FS} \rightarrow C_3^{FS} \cup U_2^{FS} \cap U_3^{FS}\ for\ C_3^{FS}$$

[1]As mentioned above, some field-sets can be empty, though for the sake of simplicity and without loss of generality, assume no empty field-sets.

[2]Referring back to market in Figure 2.1, it would correspond to only the two upper boxes.

Now, for thought, let's cut the boundaries and expand into a regular $C \cup U \to C \cup U$ matching market, $(C,\ U,\ P)^{\varepsilon}$. Suppose preference lists according to:

Table 4.13: Candidates preference lists

Agents/Ranks					
c_1	u_2	u_1	u_3	N/A	N/A
c_2	u_1	u_2	u_3	N/A	N/A
c_3	u_3	u_4	u_2	u_1	u_5
c_4	u_4	u_5	u_3	N/A	N/A
c_5	u_5	u_4	u_3	N/A	N/A

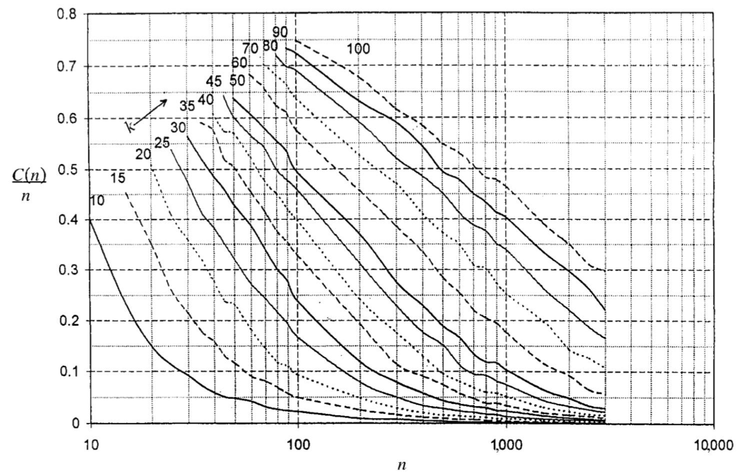
For universities, the preference lists correspond according to Equation (2c), such that, if a candidate has u_1 in his or her list, then u_1 has that candidate in its list etc.

In this case N/A means that no answer is given for this position, e.g. c_1 only went to interviews for u_1, u_2 and u_3. The realized one-to-one correspondence in both $(C,\ U,\ P)^{\omega}$ and $(C,\ U,\ P)^{\varepsilon}$ will actually be exactly the same. The reason, is that field-sets, in terms of correspondence, only leads to some sort of allowed mutual truncation of preference lists. On the basis of the theorems above, this theoretical EconMatch market has no implication on stability and optimality, both because truncation is "allowed" in the originally introduced stable marriage market on which the theorems are based. Also, since this form of truncation is actually mutual for all, i.e. c_1 will not rank u_4 and u_5, but nor will they rank c_1. The mutual truncation or unacceptability is shown above to have no implication on the stable mechanism's properties of stability and optimality. Field-sets could, however, have an impact on competition amongst agents. Candidates in mixed fields seem to expect greater competition, e.g. c_3, since they compete with all candidates within their own field-set as well as those in the overlapping fields of research. Although, they face a greater supply of possible universities, which may neutralize the increased competition.

4.2.2 The Algorithm and Strategy

Strategy, Stable Matchings and Correlation

Having shown above that the algorithm used by EconMatch does not break any of the important properties of a stable mechanism, this section will analyze the level of strategy-proofness for EconMatch. Above, I introduced a few different strategies to achieve a better outcome. All of these strategies has got at least one thing in common though, which is that the level of their potential is dependent on the size of the set of stable matchings. The smaller the size, the smaller the possibilities for successful strategic behavior. Further, some factors are more or less directly connected to the set of stable matchings, such as correlation of preference lists and the number of participants relative the length of preference lists. For the ASSA event of January 2017 about 1,800 candidates attended in search for a job (Planet Money, Speed Dating for Economists, 2017). Interviews usually take between 30 and 60 minutes (Cawley, 2018). To my knowledge, there is no information available for how many interviews an average candidate usually attends. However, what can be understood from episode 769 from Planet Money, is that interviews are not held in one place, candidates usually have to walk, run or use other means of transportation to get from one interview to another, limiting the possible amount of interviews. A certain candidate, being followed around by Planet Money on the episode, had a total of around 20 interviews over the three days of the event, i.e. around 6 to 7 interviews per day. Assuming interviews of between 30 and 60 minutes, it seems like a reasonable amount of interviews. Suppose that this candidate could represent the average candidate participating in the matching conducted by EconMatch, this would imply that they would on average have a preference list of the approximate length of 20. Suppose that there are $n = 1,800$ agents in the market, and that preference lists has the length of 20. Below is a graph showing that, when comparing the candidate-proposing algorithm with the university-proposing algorithm, the number of candidates receiving different stable matches $C(n)$ in the two versions of the algorithm[3], develops asymptotically towards zero when increasing the number of participants.



Graph 1: Size of the set of stable matchings as a fraction of n for different lengths of preference lists (k) (Uncorrelated preference lists) (Roth and Peranson, 1999)

From Graph 1 above, it's clear that for $k = 20$ and $n = 1,800$ the size of the set of stable matchings is small. Note that this graph is showing for uncorrelated preference lists. As mentioned above, adding some correlation to preference lists will further reduce the size of the set of stable matchings.

For the case of EconMatch, truncation of preference lists, and permutation with a capacity greater than one is not possible. There is simply no option of truncating preference lists in the way presented in Theorem 4 and 5. Additionally, all agents in the matching has a capacity of one, and no more. This leaves us with single-agent manipulation through permutation and manipulation through an accomplice. The strategy of using an accomplice of the proposing party seem somewhat far-fetched for

[3]Remember the lattice structure of the set of stable matchings w.r.t. to the partial order of men in Theorem 3. It's not necessary to analyze every possible stable matching, only the maximum and the minimum element of the lattice. If $\mu^C = \mu^U$ then there is only one unique stable matching.

the case of a labor market such as this. It's hard to see how a university could find that *one* candidate with a certain preference list, who is willing to permute his or her preference list for the benefit of the university, without regret. For the case of single-agent manipulation through permutation, as Teo, Sethuraman and Tan (2001) shows, beneficial manipulation, once only allowing complete preference lists, is reduced significantly. Not only is it not guaranteed that the manipulator achieves his or her optimal match, but also, it is likely not even possible to perform a manipulation with benefit at all, thus instead the manipulator is left in an unchanged or worse situation. For the case of field-sets, these probably have an impact on correlation of preference lists. Suppose, on the contrary, that only one field-set per main set existed, which would imply only one field of economics was applied to and searched for. In such a market the correlation would probably be high, since every university can interpret grades and recommendations, and every candidate probably knows which universities are the most prestigious ones. However, when assuming the number of field-sets to be greater than one, correlation should decrease, since not only grades, recommendations and reputation is of interest. All these equal, candidates and universities will be ranked in greater extent according to in which field they act. For example, a top-grade candidate in Macro/Finance may be ranked highly for universities within Macro/Finance, but lower for universities searching for pure Macro or pure Finance, resulting in a decreased preference list correlation within main sets C and U[4].

All things considered, a probable decrease in correlation due to field-sets does increase the set of stable matchings, and thus, increase potential strategy benefits. However, limited length on preference lists in relation to market size reduces the set of stable matchings. Additionally, note that every strategic move in practice means that the manipulating agent or coalition of agents need to have complete and true

[4]This impact on correlation should in some degree reduce the competition mentioned before, since agents fit for each other research field-wise are likely to rank each other higher than they rank others, all else equal.

information on every other agent's preferences, a condition which should directly reduce chances of successful cheating. Technically, it seems unlikely for an agent to succeed in the permutative manipulations presented.

Bad but True vs. Good but False

The strongest argument, however, for why it seems unlikely for a participant to act strategically, is possibly the fact that the matching is not binding and so leaving EconMatch with a minimal responsibility. Suppose a truthful run of the EconMatch modified DA-algorithm on the basis of the preferences in Table 3.1 and Table 3.2[5]. The results are the following ranked lists.

Table 4.14: Ranked list provided to universities.

University 1	University 2	University 3	University 4
Candidate B	Candidate D	Candidate C	Candidate A
Candidate A	Candidate C	Candidate B	Candidate D
Candidate C	Candidate A	Candidate D	Candidate B
Candidate D	Candidate B	Candidate A	Candidate C

Now suppose that University 1 finds it in its interest to provide a non-truthful preference list, such that $C > A > B > D$.

Then, the following ranked list would instead be provided to the universities.

Table 4.15: Ranked lists provided to universities with manipulated run.

University 1	University 2	University 3	University 4
Candidate C	Candidate D	Candidate B	Candidate A
Candidate A	Candidate C	Candidate C	Candidate D
Candidate B	Candidate A	Candidate A	Candidate C
Candidate D	Candidate B	Candidate D	Candidate B

Now University 1 and University 3 got, on their first position, their most preferred

[5] The names of the agents are changed for the sake of easier reading, since the names in greater extent will be used in coherent text here. University 1 = u_1 and Candidate A = c_1 etc.

candidates. However, nor Candidate C or Candidate B is in any way obliged to accept an offer from University 1 or 3, respectively, since the matching is non-binding. The other universities, i.e. 2 and 4, received unchanged ranked lists, except the inverse of the two last positions for University 4. Going back to the truthful ranked lists, University 1 got the message to pursue Candidate B, and not until third place, Candidate C. This information is based on the candidates' truthful preferences, University 1's true preferences, and the other universities' preferences for those candidates. So, what actually happened for University 1 when it manipulated, is that it basically falsified the information given to it, i.e. turning bad but true news into good but false news. Since the matching is non-binding, every university is free to pursue whichever candidate they would like, the matching is purely a way to inform universities of their best expected match quality. In this case University 1 would, given the manipulation, pursue Candidate C, however, that candidate knows very well that University 1 is second last on his or her preference list and would possibly delay its response in the wait for a better offer, which may result in University 1 having to settle for either an empty position or a less preferred candidate.

As mentioned in the introduction above, Thayer Morrill and Umut Dur expressed concerns regarding the issue of partial participation. The issue was presented with the help of two examples which are described below.

Assume that there are three universities (1, 2 and 3) and three candidates (A, B and C). All candidates have preferences such that $1 > 2 > 3$. University 1 either wants to hire A, or no one, however, only they know this information. Universities 2 and 3 both rank $A > B > C$. Now, suppose that University 1 do not participate in the match. The matching result will be: A assigned to 2, and B assigned to 3. The question whether A and/or B should accept now arises. The authors conclude that neither should accept, since A should wait to see if he or she gets an offer from 1, in which case B would get an offer from 2 (and possibly C getting his or her first offer from 3, however not stated in the post), which would make both (all) candidates better off than with the resulting match. Morrill and Dur further shows an example of two universities and two candidates in which one university can, by opting out of the match, obtain its most preferred candidate. The concerns expressed by Morrill and Dur can in large part be addressed with the argument of the matching being purely informative, non-binding and only representing best expected match quality. Every agent in this market, is absolutely free to pursue whatever candidate or university it would like, and a candidate is seldom hired on the spot after the first interview at ASSA, i.e. the match informs on what candidate to pursue for further interviews (so called fly-outs). For the explicit example presented by Morrill and Dur, it's hard to see the harm for Candidate A to accept. First of all, it has to be noted that the situation, as it is, implies that the candidates were able to rank a university (University 1) not participating in the match, which is not possible in EconMatch. Further, what is accepted or not is an invitation for further interviews and not a commitment such as an employment offer. Since University 1 does not participate in the match, no candidate could ever see or rank that university, so for candidates preferring a university not participating in the match, as well as for universities not participating in the match, they simply have to approach each other the traditional way. For candidates, the matching is an opportunity to credibly signal their interest and attainability for a certain university. Universities receive information on what candidates are attainable given their preferences, the candidates' preferences and other universities' preferences. Should a university for some reason want to take their chances on a candidate not in the ranked list, they are free to do so. It seems like the market with EconMatch in the worst case remains unchanged, i.e. the worst case scenario is that all agents chooses to approach each other outside the centralized matching, which is an unchanged situation. For the ones participating, however, information is being communicated while retaining privacy.

4.3 Simulation and Experiments

Above was shown that the algorithm, although modified, together with a market, likely divided into so called field-sets, can be analyzed as a regular DA-algorithm. The properties of stability as well as optimality holds, which allows for an analysis of the possibilities of and incentives for strategic behavior on the basis of earlier research. In this section, insight and possibly an answer, will be provided for the question of impact of market size. Do candidates in general obtain good matches according to their preferences? How does market size affect the probability for good matches?

4.3.1 Method

Considering that for this theme and essay real life experiments are, to say the least, hard to conduct, these questions were approached through a simulation. The program was based on Python programming language and simulated matching markets according to what is described above about EconMatch, in terms of algorithm, and the ASSA event, in terms of market structure.

Generating Agents

For generating agents, the *"player"* is asked to input the number of candidates and universities separately[6]. I assume that the number of candidates is greater than that of the universities. More specifically, I arbitrarily chose universities to be 60% of the number of candidates. The effect of this is that 40% of the candidates will remain unmatched, since the matching is structured as a one-to-one matching. The ratio between number of universities and candidates is held constant[7] when increasing

[6]As opposed to the input being part of the main code in the editor, it is an input through the console.

[7]Note that, e.g. for candidates, an increased market size means more competitors, but also greater supply of universities and by keeping the ratio constant, the competition from this source is also kept constant.

market size.

Generating Preference Lists

Having populated the market with universities and candidates, the following step is to populate every agent's preference list. For each candidate's preference list, the source of population is the number of universities chosen above, and vice versa. The preference lists provided to each agent is randomly constructed and contains every agent of the opposite set, i.e. any form of truncation, random or not, is not allowed.

The Runs of the Algorithm

At the same time as when the number of agents is decided, the number of runs should be decided. One run is to be interpreted as "one year" in the sense that after one run, a complete ranked list, such as the one provided to the universities in a real situation, is constructed. For market sizes where $m < 300$, 1,000 runs are performed, i.e. simulating results for 1,000 years of the ASSA event with EconMatch. For any market size where, on the contrary, $m > 300$, only 100 runs are performed due to calculation complexity and time restraints[8]. For any market size where $m < 300$ I compared the result from 100 runs with that of 1,000 runs and found no difference.

The Output

The output provided and analyzed is a list, e.g. $[134,\ 123,\ 145, ..., 0]$, which is to be interpreted as over the 1,000/100 years of the run of the EconMatch algorithm, 134 candidates received their first choice, 123 received their second choice etc. Note that,

[8]The regular Deferred Acceptance Algorithm is solvable in polynomial time, and has in worst case scenario a time complexity of $O(n^2)$. The EconMatch algorithm, incurring a multiple of these regular steps, obviously takes more time. This could possibly be the reason that, for this program, the time to do 100 runs of 1000 candidates and 600 universities amounted to a bit more than 13 days. Being an optimization problem, it could likely be solved so that it could be run in less time, however not in the scope of this essay.

for example 1,000 years (runs) of 100 candidates each year (run) implies 100,000 candidates in total. The absolute number of candidates receiving their different choices will, all else equal, increase as market size increases. However, by dividing this number with the number of total candidates the share of candidates will instead be achieved. The output will only be based on the first match, i.e. the first position of the ranked list (See Appendix .2 "First Position Matters" for an explanation of this).

4.3.2 The Results

Below is shown, for different market sizes, the share of candidates receiving the 1st, 2nd, 3rd, 4th and 5th preferred choice in the match, respectively.

Table 4.16: Who gets what — results for $m > n$

No. Runs	Market Size	1st choice	2nd choice	3rd choice	4th choice	5th choice
1,000	10:6	14.7%	10.8%	11.7%	9.0%	7.1%
1,000	20:12	8.2%	6.4%	6.05%	5.85%	5.2%
1,000	30:18	4.7%	5.1%	4.8%	4.03%	4.83%
1,000	50:30	2.78%	3.16%	2.94%	2.56%	3.02%
1,000	100:60	1.54%	1.516%	1.368%	1.411%	1.454%
1,000	120:72	1.285%	1.312%	1.214%	1.216%	1.183%
1,000	150:90	0.968%	1.042%	0.995%	0.933%	0.961%
1,000	175:105	0.836%	0.842%	0.882%	0.925%	0.833%
1,000	200:120	0.770%	0.748%	0.781%	0.717%	0.718%
1,000	225:135	0.697%	0.673%	0.660%	0.643%	0.681%
1,000	250:150	0.603%	0.591%	0.600%	0.617%	0.601%
1,000	275:165	0.541%	0.526%	0.545%	0.554%	0.554%
100	300:180	0.477%	0.537%	0.437%	0.483%	0.447%
100	600:360	0.233%	0.272%	0.232%	0.240%	0.252%

From Table 4.16 above it can be seen that the share of candidates receiving a certain match decreases as market size increases.

In a case in which we keep $m = n$ the following results are given:

Table 4.17: Who gets what — results for $m = n$

No. Runs	Market Size	1st choice
1,000	1:1	100%
1,000	2:2	75%
1,000	3:3	65%
1,000	4:4	57%
1,000	5:5	53%
1,000	10:10	43%
1,000	20:20	31%
1,000	50:50	26%
1,000	100:100	20%
100	300:300	18%

From above table the probability of receiving the highest-preference match also seem to decrease when market size increases in the case of $m = n$.

The graph below shows that in the market where $m = n$, the curve dives drastically in the start, and eventually evens out. Having not been able to perform for larger sets of agents, due to running time, it is not clear whether it develops asymptotically towards a certain point above zero, or if it tends to zero for $m = n \to \infty$[9].

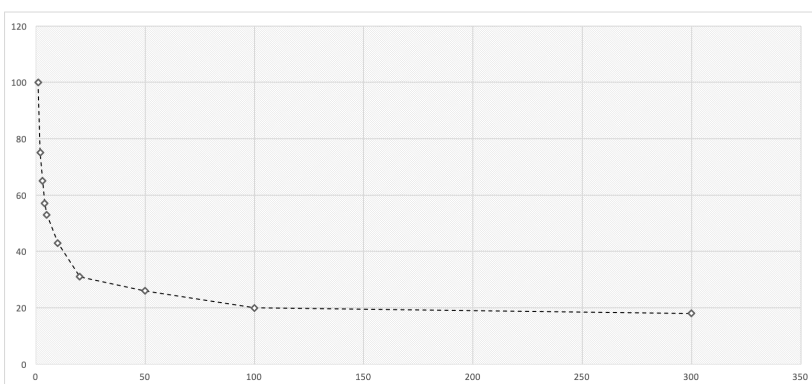


Figure 4.1: Probability of First Choice Match — $m = n$

[9] A possible approach for answering this is presented at the end of this section.

One possible explanation for this behavior of the curve is that of combinatorial explosion[10]. When increasing the number of agents, the number of potential combinations amongst them increases even more rapidly. As a simple example, in the table below, consider an increasing number of agents, from 1 to 5, and the growth of possible combinations of their preference lists. For every permutation of one preference list, there is a number of permutations for every other preference list.

Table 4.18: Agents and combinations of preference lists

m	$(m!)^m$
1	1
2	4
3	216
4	331,776
5	24,883,200,000

The rapid growth of combinations leads to a rapidly decreasing probability for a certain outcome, which could explain what is seen in Figure 4.1, Table 4.17 and Table 4.16 above.

Comparing the results from Table 4.16 and Table 4.17, there is an evident difference in probabilities even at market sizes where the number of candidates are equal. This result could be because of the impact of competition, which in turn is connected to correlation. In the case of *100:100* the chances of two or more candidates competing for one certain university is smaller than in the case of *100:60*, which is also a result of combinatorics. When $m = n$, the risks of correlating preference lists is lower than if $m > n$, which has a direct impact on the amount of competition faced by the candidates. Comparing Figure 4.1 above with Figure 4.2 below, it seems like the probability for receiving a first match develops in the same manner, whether $m = n$ or $m > n$. However, the curve in Figure 4.2 is drastically scaled down, which could

[10] A combinatorial explosion is, in mathematics, the expression of the rapid growth of the complexity of a problem, due to how the combinatorics in the problem is affected by the input.

be explained by increased competition.

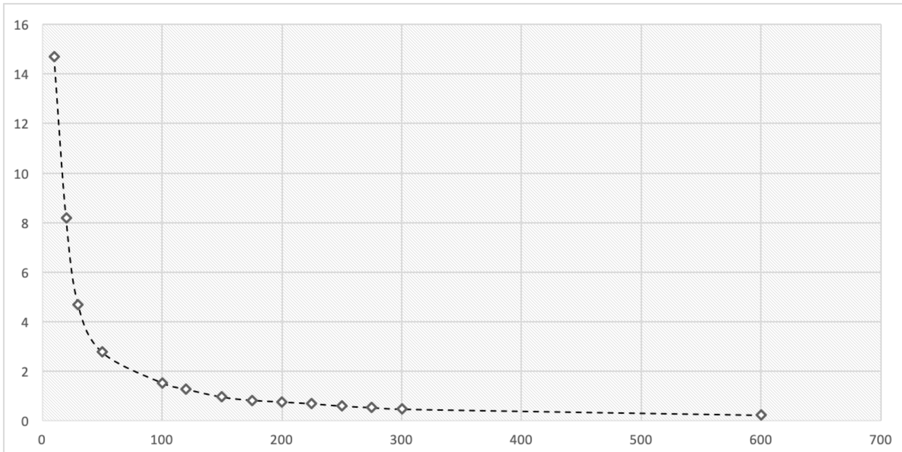


Figure 4.2: Probability of First Choice Match — $m > n$

If for Figure 4.1 and Figure 4.2 we allow for a description of the curve in the form of a function, it turns out that the best fitting function is a power function.

$$y = \frac{90.194}{n^{.311}} \; for \; m > n$$

and,

$$y = \frac{0.2233}{n^{.965}} \; for \; m = n$$

Above functions fits well for the range analyzed. However, nothing can be said for any range larger or smaller than that.

Considering that $\lim\limits_{x \to \infty} \frac{1}{\sqrt[2]{x}} = 0$, it may be the case that the probability of receiving the highest preference match will tend to zero in infinity, rather than develop asymptotically towards a point above zero.

Size Matters

In the results above it does not seem like an increased market size is beneficial when measuring the share of candidates receiving good matches according to their preference lists. The impact of combinatorial explosion when increasing market size seems to lead to a reduced probability of achieving a highly preferred match. Additionally, increased correlation, leading to increased competition, seem to have at least as great an impact when it comes to good matches or not. The source of the impact from increasing market size, and correlation is that of combinatorics. On the basis of the results from the simulation above, it does seem like EconMatch may be more beneficial for candidates on a smaller market.

Remarks

In the simulation, agents need to have preferences over all agents of the opposite set. The result of this is that the length of all preference lists is the same as the number of agents of the opposite set. Although not being realistically correct, it will not incur any issues for the main result as the ratio between universities and candidates is kept constant. Additionally, randomized preference lists could possibly overestimate the probability of achieving a certain match. Somewhat weighted preference lists would likely illustrate a more realistic result. However, weighted preference lists is practically just a way of adding correlation to preference lists, which, as mentioned above, increases competition and lowers the likelihood of highest-preference matches.

Chapter 5

Conclusion and Discussion

5.1 Conclusion

EconMatch's modified algorithm is constructed in a way such that important properties like stability and optimality continues to hold true, since it remains a stable mechanism. Elements and terms such as field-sets, mutual acceptability and mutual unacceptability were, in this essay, added to better model the market in which EconMatch performs. However, these turned out to have no impact on the stability or the optimality of the algorithm. Field-sets could, however, affect the amount of correlation seen in preference lists since it adds subjectivity to agents' preferences.

The possibilities for strategic actions in a matching market is directly related to the size of the set of stable matchings. The set of stable matchings is in turn connected to the length of preference lists in relation to market size, and correlation of preference lists within sets. In a realistic setting, preference lists are limited, which lowers the possibilities of successful strategic behavior. However, since field-sets, lowering correlation, are introduced, these will likely counteract each other on some level. All things considered, it does seem unlikely for an agent to successfully manipulate and act strategically. The added subjectivity in preference lists will likely not eliminate the objectivity in preferences, such as grades and so forth. Further, to act strategically, an agent needs to have complete and true information of others' preferences, which also is a condition reducing the potential for successful strategic behavior. The non-binding and purely informative property of EconMatch only leaves an agent acting strategically, with false information. The freedom that agents remain to have in the matching, and so the impossibility of locking another agent in the matching, should reduce the incentives for strategic behavior significantly.

What may seem somewhat counterintuitive, is that of the negative impact that market size has on the matching. If considering probability for good matches a reasonable measure of expected candidate-utility, EconMatch seem to perform better in smaller markets. Additionally, the ratio between the number of candidates and universities, and thus competition, seem to have just as great, if not greater, an impact on the probability of achieving the highest-preference match.

5.2 Discussion, Application and Further Research

5.2.1 Discussion

Earlier research on matching markets has, to my knowledge, only studied matchings as binding assignments. In such a setting, concerns on strategic behavior is certainly more relevant. It seems, however, that EconMatch, as an informative and non-binding assignment, keeps getting mistaken for a binding match when individuals interested in the subject express their concerns.

The result that market size has a negative impact on probability of achieving the highest-preference match is, of course, unfortunate for candidates. However, in the spirit of Kaldor-Hicks[1], it is probably more appropriate to get some perspective of the

[1] An outcome is a Kaldor-Hicks improvement if those that are worse off, are less worse off than the ones better off are better off.

situation. When looking at the case in the bigger picture, stability and optimality is likely of greater importance than first choice matches. Stability implies that, no two agents prefer each other, but aren't matched to each other. Optimality implies that the optimal stable matching will be strongly Pareto-efficient with respect to all participants, and weakly Pareto-efficient with respect to the proposing party. When only considering stable matches, which is rational since a match would likely not persist otherwise, the outcome of the stable mechanism is actually strongly Pareto-efficient for all sets and subsets.

The feature of being non-binding and giving agents full freedom leads to a reduced relevance of the concern on partial participation. Clearly, the algorithm would perform better, in terms of information, had everyone participated. However, partial participation only leads to that some agents are required to approach each other in the traditional manner, which is exactly like a market without EconMatch. Therefore, it is not clear that EconMatch in any way could "worsen" the existing situation, but rather potentially make it better. Full participation or not, the resulting match informs universities of attainable and unattainable candidates. A candidate, let's say a top-candidate, gets the opportunity of credibly signaling his or her interest for a university, that otherwise possibly would reject him or her based on too good qualifications. The information provided may just as well be bad news, as well as good news, however, it's information nonetheless which reduces asymmetric information and in turn possibly congestion.

5.2.2 Application and Further Research

Application

A matching as a tool for communicating information among agents in a market has not, to my knowledge, been analyzed before. Based on this essay, I believe that other markets suffering from asymmetric information and a lack of credible signaling, could benefit from something such as EconMatch. It is certainly hard, without any form of structure, to retrieve information on the entire markets' preferences. With the DA-algorithm as an informative tool, agents could, now being better informed, make better decisions, waste less resources and credibly signal their interest for one another, while retaining privacy.

Further Research

Hopefully, more research will be done when it comes to matching algorithms for communicating information. For further research, the simulation done in this essay should be optimized so that time complexity is not a restraint for, still realistically, large market sizes. Making the code more flexible would further increase the possibilities of a more extensive and in-depth analysis of the market and the algorithm. Additionally, assuming EconMatch to be in the game for some time, real empirical data could in addition be analyzed for further research and possibly provide new results.

.1 Appendix

.1.1 The Python Program Code

THE RUNNING CODE

```
from daa_functions_v3 import *
import time

output_file = "output_result_" + str(time.time()) + ".txt"

nbr_of_runs = int(input("Number of runs: "))
if nbr_of_runs == '':
   nbr_of_runs = 1
nbr_of_students = int(input("Number of students: "))
nbr_of_universities = int(input("Number of universities: "))

deferred_acceptance(output_file, nbr_of_runs,
nbr_of_students,nbr_of_universities)
```

THE CODE FOR FUNCTIONS

```
from random import randint
from matching import Player
from matching import HospitalResident
import pandas as pd
import copy
import time

'''
Generate dictionary with key = player prefix+nbr and value = [ list of
randomized preferences of the others_prefix ]
e.g key = s1, value = ['u1', 'u2', 'u3']
'''
def generate_dictionaries(nbr_of_players, nbr_of_other_players, prefix,
others_prefix):

   players_dict = {}

      players_list = list(range(1, int(nbr_of_other_players)+1))

   for i in range(1, int(nbr_of_players)+1):

            players_list_copy = players_list.copy()

      temp_pref_list = []

      dict_key = prefix + str(i)

      for j in range(1, int(nbr_of_other_players)+1):

         temp_pref_list.append(others_prefix +
str(players_list_copy.pop(randint(0 , len(players_list_copy) - 1))))

      players_dict[dict_key] = temp_pref_list

   return players_dict

"""
Returns True until all lists are empty
"""
def prefs_empty(dict1, dict2):
   if len(dict1) == 0 and len(dict2) == 0:
      return True
   else:
```

```
      return False

def deferred_acceptance(output_file, nbr_of_runs, nbr_of_students,
nbr_of_universities):

   choice_nbr_place = []

   for y in range(0, nbr_of_students):
      choice_nbr_place.append(0)

   for x in range(1, nbr_of_runs+1):
      counter = 1
      print("Round {}/{}".format(x, nbr_of_runs))


      round_result = []


      students_obj_list = []
      students_result_dict = {}
      students_dict = generate_dictionaries(nbr_of_students,
nbr_of_universities, "s", "u")
      students_dict_ro = copy.deepcopy(students_dict)

      univs_obj_list = []
      univs_result_dict = {}
      univs_dict = generate_dictionaries(nbr_of_universities,
nbr_of_students, "u", "s")
      univs_dict_ro = copy.deepcopy(univs_dict)

      for k , v in students_dict.items():
         students_obj_list.append(Player(k , list(v).copy() , 1))
         students_result_dict[k] = []

      for k , v in univs_dict.items():
         univs_obj_list.append(Player(k , list(v).copy() , 1))
         univs_result_dict[k] = []

      stud_panda = pd.DataFrame.from_dict(students_dict)
      univ_panda = pd.DataFrame.from_dict(univs_dict)

      with open(output_file , 'a') as f:
         f.write("-------------- Round number {} --------------".format(x))
         f.write("\nStudents' preflists\n\n\n")

      stud_panda.to_csv(output_file , mode = 'a' , index = False)

      with open(output_file , 'a') as f:
         f.write("\nEnd of students' preflists\n\n\n")
         f.write("\nUnivs' preflists\n\n\n")

      univ_panda.to_csv(output_file , mode = 'a' , index = False)

      with open(output_file , 'a') as f:
         f.write("\nEnd of Univs' preflists\n\n\n")

      while not prefs_empty(students_dict , univs_dict):

         hr = HospitalResident(suitors = students_obj_list , reviewers =
univs_obj_list)
```

```
        match = hr.solve()

        round_result.append("Iteration " + str(counter) + " match: " +
str(match) + "\n")

        for i in students_obj_list:
           if str(i.matching) != 'None':
              students_result_dict[i.name].append(str(i.matching))

           if str(i.matching) in students_dict[i.name]:
              if counter == 1:

choice_nbr_place[students_dict_ro[i.name].index(str(i.matching))] += 1
              students_dict[i.name].remove(str(i.matching))

           if not students_dict[i.name]:
              del students_dict[i.name]

        for i in univs_obj_list:
           if i.matching != []:

univs_result_dict[i.name].append(str(i.matching)[1:len(str(i.matching)) -
1])

           if str(i.matching)[1:len(str(i.matching)) - 1] in
univs_dict[i.name]:

univs_dict[i.name].remove(str(i.matching)[1:len(str(i.matching)) - 1])
           if not univs_dict[i.name]:
              del univs_dict[i.name]

        univs_obj_list = []
        students_obj_list = []

        for k , v in students_dict.items():

           students_obj_list.append(Player(k , v.copy()))

        for k , v in univs_dict.items():
           univs_obj_list.append((Player(k , v.copy())))

        counter += 1

     '''
     Below is the code for formatting the output to be written to file.
     '''
     stud_result_panda = pd.DataFrame.from_dict(students_result_dict)
     univ_result_panda = pd.DataFrame.from_dict(univs_result_dict)

     with open(output_file , 'a') as f:
        f.write("\n\nStudents matching results\n\n\n")

     stud_result_panda.to_csv(output_file , mode = 'a' , index = False)

     with open(output_file , 'a') as f:
        f.write("\n\nEnd of Students matchings\n\n\n")
        f.write("\n\nUniversities matching results\n\n\n")

     univ_result_panda.to_csv(output_file , mode = 'a' , index = False)
```

```
      with open(output_file , 'a') as f:
         f.write("\n\nEnd of Universities matchings\n\n\n")

      with open("round_by_round-" + output_file , 'a') as f:
         for round in round_result:
            f.write(round)
            f.write("\n\n")
   print("List of fördelning av resultat (index noll är antal som fått
förstaval osv...):\n {}".format(
      choice_nbr_place))

   for i in range(0, len(choice_nbr_place)):
      print("Amount of students who got their nr {} choice:
{}%".format(i+1, choice_nbr_place[
         i]/(nbr_of_students*nbr_of_runs)*100,))
```

.2 Appendix

.2.1 First Position Matters

Only the first match will be analyzed in the output. The reason is that of compressed ranked lists and mutual deletion. Assume the following preferences:

Table 1: Candidates' preference lists

Agents/Ranks					
c_1	u_1	u_2	u_3	u_4	u_5
c_2	u_2	u_1	u_3	u_4	u_5
c_3	u_1	u_3	u_2	u_4	u_5
c_4	u_2	u_3	u_1	u_4	u_5
c_5	u_3	u_2	u_1	u_5	u_4

Table 2: Universities' preference lists

Agents/Ranks					
u_1	c_2	c_1	c_3	c_4	c_5
u_2	c_2	c_3	c_1	c_4	c_5
u_3	c_3	c_1	c_2	c_4	c_5
u_4	c_1	c_2	c_3	c_4	c_5
u_5	c_1	c_3	c_2	c_4	c_5

With these preferences, the following first, second, third and fourth round results are given:

Table 3: Matching results

Pairwise Matching 1st	c_1:[u_1]	c_2:[u_2]	c_3:[u_3]	c_4:[u_4]	c_5:[u_5]
Pairwise Matching 2nd	c_2:[u_1]	c_3:[u_2]	c_1:[u_3]	c_5:[u_4]	c_4[u_5]
Pairwise Matching 3rd	c_3:[u_1]	c_1:[u_2]	c_2:[u_3]	N/A	N/A
Pairwise Matching 4th	N/A	c_4:[u_2]	c_5:[u_3]	c_1:[u_4]	c_3:[u_5]

Below are hypothetical ranked lists provided to candidates. These are hypothetical in the sense that, in real life, candidates does not receive ranked lists.

Table 4: Hypothetical ranked lists provided to candidates

Position/Agents	c_1	c_2	c_3	c_4	c_5
1	u_1	u_2	u_3	u_4	u_5
2	u_3	u_1	u_2	u_5	u_4
3	u_2	u_3	u_1	u_2	u_3

Looking at the table above it says that c_4*'s* and c_5*'s* third ranked position is u_2 and u_3, respectively, provided in the fourth round. However, from the perspective of u_2 and u_3, their third ranked position is c_1 and c_2, respectively, provided in the third round. Now suppose analyzing what choices candidates get according to their preferences, based on the third position. This would give the result that at least two candidates (c_4 and c_5) got their first choices, which is obviously not the case since c_4 and c_5 actually received these matches in the fourth round. The reason for this effect is that, under some instances, some agents' only possible matching under a stable mechanism is that of being matched to themselves, leading to some unrealized matches, which in turn is a result of the mutual deletion occurring at every round. Additionally, the ranked lists are compressed and doesn't allow empty positions, which results in that agents receive their respective positions at different rounds. By only analyzing the first round match, where no mutual deletion has occurred, this effect is eliminated.

Bibliography

[1] American Economic Association. About the AEA Annual Meeting, 2019. https://www.aeaweb.org/conference/about [Accessed 24th of April]

[2] Aziz, H, Seedig, H.G, and von Wedel, J.K. On the Susceptibility of the Deferred Acceptance Algorithm. In *Autonomous Agents and Multiagent Systems*, 14, May, 2015. https://pdfs.semanticscholar.org/5de3/9c2772faac8b9629e20631a70847fbb15f78.pdf [Accessed 15th of April]

[3] Bendlin, T and Hosseini, H. Partners in Crime: Manipulating the Deferred Acceptance Algorithm through an Accomplice, 2019. https://www.aaai.org/Papers/AAAI/2019/SA-BendlinT.397.pdf [Accessed 15th of April]

[4] Cawley, J. A GUIDE AND ADVICE FOR ECONOMISTS ON THE U.S. JUNIOR ACADEMIC JOB MARKET, September 2018. https://www.aeaweb.org/content/file?id=869 [Accessed 15th of May]

[5] Coles, P. Optimal Truncation in Matching Markets, July 2009. http://www.people.hbs.edu/pcoles/papers/ot.pdf [Accessed 15th of April]

[6] Dubins, L.E and Freedman, D.A. Machiavelli and the Gale-Shapley Algorithm. *The American Mathematical Monthly*, Vol. 88(No. 7):485–494, 1981. http://www.eecs.harvard.edu/cs286r/courses/fall09/papers/dubbinsfreedman.pdf [Accessed 11th of April]

[7] EconMatch. EconMatch FAQ, 2019. http://www.econmatch.com/Test/faq.html [Accessed 17th of April]

[8] Economics Job Market Rumors,2018.Forum thread in Economics Job Rumors. https://www.econjobrumors.com/topic/econmatch [Accessed 15th of April]

[9] Erdil, A and Ergin, H. Two-Sided Matching with Indifferences, August 2017. https://eml.berkeley.edu/~hie/research/matching-with- indifferences.pdf [Accessed 16th of April]

[10] Gale, D and Shapley, L. College Admissions and the Stability of Marriage. *The American Mathematical Monthly*, Vol. 69(No. 1):9–15, 1962. http://www.eecs.harvard.edu/cs286r/courses/fall09/papers/galeshapley.pdf [Accessed 17th of April]

[11] Gale, D and Sotomayor, M. Some Remarks On the Stable Matching Problem. *Discrete Applied Mathematics*, 11:223–232, 1985. https://core.ac.uk/download/pdf/82447873.pdf [Accessed 19th of April]

[12] Knuth, D.E. Mariages stables et leur relation avec d'autres problèmes combinatoires. In *Mariages Stables*, pages 88–93. Les Presses de l'Université de Montréal, 1976. https://www-cs-faculty.stanford.edu/~knuth/mariages-stables.pdf [Accessed 4th of April]

[13] Morrill, T and Dur, U. Should you participate in EconMatch?, 2019. Post in the American Economic Association forum. https://www.aeaweb.org/forum/918/should-you-participate-in-econmatch [Accessed 17th of April]

[14] Nobel, C. How to Fix a Broken Marketplace, November 2010. https://hbswk.hbs.edu/item/how-to-fix-a-broken-marketplace [Accessed 17th of April]

[15] Planet Money. Episode 769: Speed Dating for Economists, May 2017. https://www.npr.org/sections/money/2017/05/05/527087730/episdoe-769-speed-dating-for-economists?t=1556483195199 [Accessed 14th of May]

[16] Roth, A.E. Marketplaces, Markets, and Market Design. *American Economic Review*, 108(7):1609–1658, 2018. http://cramton.umd.edu/market- design/roth-marketplaces-markets-and-market-design.pdf [Accessed 15th of April]

[17] Roth, A.E. The Evolution of the Labor Market for Medical Interns and Residents: A Case Study in Game Theory. *Journal of Political Economy*, 92:991–1016,1984. https://dash.harvard.edu/bitstream/handle/1/29410143/evolut.pdf [Accessed 15th of April]

[18] Roth, A.E. Deferred Acceptance Algorithms: History, Theory, Practice, and Open Questions. *International Journal of Game Theory*, pages 537–569, 2008. https://web.stanford.edu/~alroth/papers/GaleandShapley.revised.IJGT.pdf [Accessed 15th of April]

[19] Roth, A.E and Peranson, E. The Redesign of the Matching Market for American Physicians: Some Engineering Aspects of Economic Design, February 1999. https://www.nber.org/papers/w6963.pdf [Accessed 15th of April]

[20] Teo, C-P, Sethuraman, J and Tan, W-P. Gale-Shapley Stable Marriage Problem Revisited: Strategic Issues and Applications. *Management Science*, 47(9), 2001. http://www.columbia.edu/~js1353/pubs/tst-ms01.pdf [Accessed 17th of April]

[21] The Royal Swedish Academy of Sciences. Stable matching: Theory, evidence, and practical design, 2012. https://www.nobelprize.org/uploads/2018/06/popular-economicsciences2012.pdf [Accessed 14th of April]

[22] Vaish, R and Garg, D. Manipulating Gale-Shapley Algorithm: Preserving Stability and Remaining Inconspicuous. In *International Joint Conference on Artificial Intelligence*, 2017. https://www.ijcai.org/proceedings/2017/0062.pdf [Accessed 15th of April]

[23] Zanardo, E. Matching in the Smallest Large Market, January 2016. http://www.columbia.edu/~ez2197/Matching.pdf [Accessed 15th of April]