Helsinki Center of Economic Research

Discussion Papers

The size of the core in school choice

Kristian Koerselman
University of Jyväskylä, FIER

and

Mikko A. A. Salonen
University of Helsinki and HECER

Discussion Paper No. 422
December 2017

ISSN 1795-0562

HECER – Helsinki Center of Economic Research, P.O. Box 17 (Arkadiankatu 7), FI-00014 University of Helsinki, FINLAND,
Tel +358-2941-28780, E-mail info-hecer@helsinki.fi, Internet www.hecer.fi

HECER
Discussion Paper No. 422

The size of the core in school choice*

Abstract

We study the determinants of the size of the core in the school choice problem using three years of data from a large higher education application clearinghouse. The clearinghouse uses a variation of the college-optimal stable mechanism (COSM) to assign applicants to slots in Finnish polytechnics. If the core is large, switching to a student-optimal stable mechanism (SOSM) could yield large improvements for applicants at a cost to schools. We however find that the core is either a singleton or very small each year. This suggests that the student/school trade-off is relatively unimportant within the set of stable matchings in Finnish polytechnic assignments. We show that the similarity of COSM and SOSM matchings is due to correlated school priorities, differing numbers of students and slots, and to students only applying to a small number of programs each. Because these properties are common to other higher education school choice problems, our conclusions are likely to generalize. In spite of the fact that Finnish polytechnics jointly only accept a third of applicants, accepted applicants' average matriculation exam grades are not much better than those of the median applicant. We attribute this to the low effective number of programs applied to, and suggest that details in the design of the application process affect the trade-off in match quality.

JEL Classification: C78, D82, C71

Keywords: school choice, singleton core, deferred acceptance, match quality

Kristian Koerselman

FIER
University of Jyväskylä
FI-40014 University of Jyväskylä
FINLAND

e-mail: kristian.w.koerselman@jyu.fi

Mikko A. A. Salonen

Department of Political and Economic Studies
University of Helsinki
P.O. Box 17 (Arkadiankatu 7)
FI-00014 University of Helsinki
FINLAND

e-mail: mikko.salonen@helsinki.fi

* We thank the Yrjö Jahnsson Foundation (grants: 6652, 6528, 6400) and the OP-Pohjola Group Research Foundation (grants: 201700153, 201600101, 201500156, 201300053) for financial support. We thank seminar participants at Autonomous University of Barcelona, Hannu Vartiainen, Hannu Salonen, Caterina Calsamiglia, Antonio Miralles, Angela Djupsjöbacka, and Benjamin Tello for guidance, comments, and insightful discussions. All of the remaining mistakes are our own.

1 Introduction

The school choice problem is the problem of allocating applicants (students) to indivisible slots in programs (schools) without using money. A large part of the school choice literature has focused on the theoretical properties of the different matching algorithms and on improvements to realized matchings. We know much less about how matching algorithms compare in practice, especially in large, real-world applications.

Gale and Shapley (1962) showed that in a marriage market, where each agent wants to find a pair, with strict preferences there always exists a stable matching, i.e. the size of the core is at least one. Furthermore, one can generally find many different stable matchings in a marriage market, with the man-optimal and women-optimal stable matching as extreme cases. In a school choice problem, where schools can accept more than one student, these two extreme matchings are the college-optimal and the student-optimal stable matching, respectively produced by the College-Optimal Stable Mechanism (COSM) and the Student-Optimal Stable Mechanism (SOSM).[1] Applicants are matched to Finnish polytechnic programs by a variation of COSM. The use of a college-optimal mechanism naturally poses the question whether the core is large and applicants' outcomes could be improved by using a student-optimal mechanism instead, or whether the core is small and the two extreme matchings are similar to each other.

Theoretical findings on the expected size of the core are mixed. Pittel (1989) showed that the size of the core is growing with market size when the market is balanced and preferences are randomly drawn. Eeckhout (2000), Clark (2006), Salonen and Salonen (2018), and Akahoshi (2014) formulate conditions on preference domains such that the core is guaranteed to be a singleton, but these conditions are so restrictive that they seem unlikely to hold in practice[2]. On the other hand, when submitted Rank Ordered Lists (ROLs) are shortened, for example because of nonzero marginal application costs or because of administrative restrictions, the size of the core converges to an upper bound (Roth and Peranson, 1999). Likewise, when the preferences of the agents are correlated the size of the core has an upper bound (Holzman and Samet, 2014). Ashlagi et al. (2017) showed that the core can shrink substantially by even a slight unbalancedness in a marriage market, i.e. when the number of men and women on the market are only slightly unequal. Empirical studies using real matching data on the differences between student- and college-optimal stable matchings have been carried out by Colenbrander (1996a,b), Peranson and Randlett (1995), and Roth and Peranson (1999). All of these studies were based on small datasets, and found little or no difference between the two extreme stable matchings.

We add to the literature by studying the size of the core and its determinants in a large higher education market. We find that the core is either a singleton or very small for each of the three years which our data cover. We further find that this is the case because in Finnish polytechnic applications, i) program priorities

[1] The algorithms are described in Appendix A.

[2] Eeckhout shows that the core is a singleton when the preferences on both sides of the market satisfy *single-peakedness*. Clark shows the same result for *no-crossing* property on preferences. This condition is also known as the single-crossing property. Salonen and Salonen (2018) show that these properties induce a singleton core in college admission problems. Akahoshi introduces an acyclicity condition and a capacity based condition.

are correlated, ii) the number of applicants vastly outnumbers the number of slots, and iii) applicants apply to a very small average number of programs.

We show that removing any single one of these empirical features from the clearing house data is insufficient to create a sizable core, and that adding a single one of these empirical feature to a simulated data set designed to create large differences between the COSM and SOSM matchings is sufficient to dramatically reduce the size of the core. Since these features are common to other education markets, the similarity of COSM and SOSM matchings is likely to be a general property in these markets.

Yet, when the core is a singleton, match quality in terms of average ranks of their matched pairs differs for applicants and schools. Majority of accepted applicants get matched to their most preferred program while the programs on average get matched with middle tier students when ranks are based on the field specific grade priorities. We attribute this phenomenon as a consequence of short ROLs listed by students.

In the next section we describe the institutional background of school choice in Finnish higher education. We introduce the data in Section 3 and the standard school choice model in Section 4. In Section 5 we show the effect of ROL restrictions on the size of the core, convergence of the two extreme matchings, and changes in average ranks. Our empirical results are shown in Section 6. We discuss the implications of our results in Section 7. Finally the descriptions of the algorithms and additional tables can be found in Appendix A.

2 Institutional Background

Finnish higher education is provided by polytechnics and universities, with students typically graduating with a Bachelor's degree from the former, and with a Master's from the latter. Some higher education programs have secondary enrollment rounds, but large majority of applicants enroll in August or September at the end of an application process which starts in March. Since university applications are handled in a decentralized manner, in this paper we concentrate on the larger group of polytechnic applicants, whose applications were processed through a centralized clearing house.

The application process starts in March when applicants can rank up to four programs in order of their preference. Entry exams are mostly held in May or June. Program priorities are represented by a composite score based on the student's matriculation grades, his entry exam result, and other program specific factors e.g. working experience in a relevant field, with extra points awarded for the first listed choice. Programs in different fields weight the different matriculation exam subjects differently, but programs within the same fields tend to use the same weights, and typically share a common entry exam as well.

Though entry exams play an important role in acceptance decisions, applicants must choose where to apply before learning their entry exam results. Applicants do however receive preliminary matriculation exam grades, with the final grades published in May. Applicants also have access to previous years' composite score cut-offs, and to the different programs' selection criteria.

After the entry exams have been graded, applicants are matched to programs through a centrally run COSM variation, receiving either an offer from a single program, or no offer at all. Applicants then either accept this offer, or reject

Table 1: Number of programs in each field and year

field	2011	2012	2013
pedagogy	7	7	7
fine arts and culture	62	59	48
agriculture, forestry and the environment	26	25	16
business information systems	15	14	13
social sciences and business administration	39	40	35
health and social care	121	124	123
engineering	137	131	127
tourism	33	32	26
total	440	432	395

it. A second round of offers is sent out by the programs themselves to make up for first-round rejections. Because the second round is decentralized, we concentrate on the first round in our analysis.

3 Data

We have data covering all applications to Finnish polytechnics that were made through the centralized clearing mechanism during the fall application rounds of years 2011, 2012, and 2013. For each applicant, the data contain the applicant's ROLs, i.e. the programs applied to in order of preference. For each of these applications, it also includes matriculation exam grades, entry exam scores, and the composite application score. We also have the final state of the actual matching algorithm, indicating which applicants received offers, and which did not.

Regrettably, we do not have information on the program identifiers and quota used in the actual application algorithm. We therefore use combinations of polytechnic name and program name as proxies of program identifiers and the number of simultaneous offers made by each program as a proxy for program quota. We can replicate about 98% of application decisions by applying a standard COSM algorithm to the composite scores. Throughout the paper, we will use our simulated COSM matching as the benchmark rather than the actual outcome of the matching.

Table 1 shows the number of programs in each field of study in each of the three years we have data. Programs in the fields of health, social care, and engineering jointly make up more than half of the programs. Table 2 contains information on the number of applicants, applications and slots in each application year. The central clearing house handles more than 50 000 applicants to Finnish polytechnics each year, with an average of between two and three applications per applicant. Only about 38% of applicants apply to the maximum number of four programs. The number of applicants vastly outnumbers the number of slots, with the number of slots available per applicant being approximately 0.31. The last column shows the average probability of receiving a study slot offer conditional on the number of programs applied to. Offer probabilities are only marginally larger for applicants who apply to larger number of programs, and few applicants are admitted to their fourth listed program.

Correlated preferences have been shown to be important in determining the

Table 2: Application statistics

	2011	2012	2013	
programs	440	432	395	
applicants	50894	50979	52665	
applications	140917	140485	142723	
slots	16655	16425	15210	
number of programs applied to				$\mathbb{P}$
1	10362	10508	11458	28.79%
2	10089	10308	11299	30.65%
3	11395	11291	10965	31.78%
4	19048	18872	18943	32.75%
accepted to fourth preference	496	466	405	

Application statistics for the fall application rounds of 2011, 2012 and 2013. Column $\mathbb{P}$ shows the offer probability conditional on number programs applied to averaged over the three years

size of the core. In a higher education setting, it seems natural that program priorities would be correlated with each other. An applicant who is desirable to one program, is likely to be desirable to another program as well. In our data, the mean pairwise correlation between programs' rankings of applicants' GPA is approximately 0.86. Part of this is because rankings are correlated across fields, part is because different programs with the same fields rank applicants identically. When we simulate rankings to be uncorrelated across fields, but perfectly correlated within them, the mean pairwise correlation is only 0.21. This shows that the overall correlation is largely driven by a correlation of priorities across fields.

Preferences can also be correlated between applicants, but since applicants only apply to up to four programs in our data, these correlations are hard to measure, and we refrain from quantifying them explicitly. We do however note that in spite of the fact that less than a third of the applicants receive an offer, across the grade distribution applicants apply to programs that are substantially harder for them to get in to than the median program. This suggests a positive correlation of preferences across applicants.

Applicants further fail to list safe choices. Applicants of all grades have a range of programs available to them that would double their offer probabilities compared to the mean offer probability of their actual application portfolio. Moreover, 62% of applicants fail to fill out all four programs on their application. Among those 62%, the second and third choices are not particularly safe either. Though these applicants could have added a safe fourth program in their application at low marginal cost, they did not.

A third type of correlation exists between program and applicant preferences, and this cross-correlation can also affect the size of the core. In the context of higher education applications, a positive cross-correlation can be seen as applicants preferring programs in which they have a comparative advantage. Somewhat surprisingly however, in our data applicants do not apply to programs in which they have unusually high grades, suggesting that the cross-correlation between preferences is not a contributing factor to a small observed core size in our data.

There is a concern that Finnish polytechnics applicants may strategize, especially since bonus points are awarded for the highest listed program, ties are broken based on ROLs, and the matching algorithm in use is COSM. Applicants however i) fail to apply to the maximum of four programs, ii) fail to list a safe option among the programs they apply to, and iii) fail to apply to programs in which they hold a comparative advantage, suggesting that many applicants genuinely prefer staying unmatched to being matched to a different program, and that their strategizing is not sophisticated. Hence, in our simulations we assume that the preferences are reported truthfully. We furthermore carry out a series of robustness checks where we replace applicant preferences by randomly drawn ones in Appendix A. The results remain substantially similar.

4 Model and methods

The school choice model consists of finite set of applicants $s \in S$ and finite set of programs $c \in C$. Each program c has a quota of $q_c > 0$. Applicants have strict preferences over the set of programs and demand a single indivisible study slot. If the preferences of an applicant s are $\prec_s$: $c, ..., s, ..., c'$, then applicant s prefers program c to all other programs and s prefers to be unmatched rather than given a study slot in program c'. Priorities of the programs are defined in the same way except programs demand at most their quota amount q_c of applicants and only consider applicants that applied there.

We denote the matching function by $\mu : S \to C \cup S$. Function μ matches an agent to himself or to an agent from the other set. An applicant is matched to himself when $\mu(s) = s$ and an applicant is matched to a program if $\mu(s) = c$. An algorithm produces a matching μ where i) an agent is matched to himself, or ii) an applicant s is matched to program c under μ and the set of applicants matched to program c includes applicant s as well under μ. We assume that applicants only care about being matched to a program and not about which slot in that program they are placed in. Furthermore, we assume that priorities of the programs are responsive (see Roth and Sotomayor, 1992, p. 128). That is, programs' priorities over different matchings corresponds to priorities over applicants, not groups of applicants.

A matching μ is individually rational if no agent is matched to an unacceptable counterpart, that is, for all $s \in S$ and $c \in C$ it holds that $\mu(s) \neq c$ if i) $c \prec_s s$, ii) $s \prec_c \varnothing$, or iii) s is acceptable to c but has no free capacity. A blocking pair (s, c') can be formed when under some algorithm applicant s and program c' are not matched, but both prefer each other to their matched pairs: $\mu(s) \prec_s c'$ and $s' \prec_{c'} s$ such that $\mu(s') = c'$ holds[3]. An algorithm satisfies *stability*, if participating in the algorithm is individually rational and there are no blocking pairs. An algorithm is *strategy-proof* if there is no incentive for the agents to misrepresent their true preferences. A matching produced by an algorithm is *Pareto-efficient* if no improvements to the matching can be made without making an applicant worse off.

We use the notation S' as the subset of applicants that are matched with a program in a stable matching. That is, $s \in S'$ if $\mu(s) = c$. We say a program c is matched with rank $r_c = r_{\mu(s)}$ applicants under an algorithm that produces μ, where $s \in S'$. If a program $\mu(s)$ is matched with the first and the third

[3]We use notation where program $c' \in C$ is some other program than $c \in C$. Likewise applicant $s' \in S$ is some other applicant than $s \in S$.

applicants on its list of priorities the rank r_c equals two. Similarly, an applicant s has rank $r_s = 1$ if he is matched to his most preferred program.

In the Finnish polytechnic clearing house, programs only express exact priorities over actual applicants. Realizations of r_c therefore become dependent on which applicants choose to apply to the individual program, and is bounded by the number of applicants to that program. Since field-specific matriculation grade averages are available for applicants to all programs, where relevant we therefore also report averages of r_c as defined on applicant grades. These can be seen as proxies of match quality from the programs' perspective, and are more easily generalizable to other markets. Since we want average ranks to be invariant to the aggregation level of schools, we weight by applicant rather than by program throughout.

As Irving and Leather (1986) demonstrated, calculating the size of the core is NP-hard. We therefore follow Roth and Peranson (1999) and Holzman and Samet (2014), and compare the extreme COSM and SOSM matchings to each other. We do this by looking at the number of applicants allocated differently, as well as the average ranks of agents under COSM and SOSM.

5 ROL restrictions and the size of the core

Next we illustrate how ROL restrictions reduce the size of the core, converge the two extreme matchings, and affect average ranks. By construction, the length of ROL for an applicant in an unrestricted case is the size of the set of programs $|C|$ when all programs are acceptable. If the length of ROLs are restricted by some number n the applicant can apply to at most $|C| - n$ programs.

To demonstrate how restricting the length of ROLs or introducing unbalancedness can affect the size of the core we present two extreme examples where we have tried to minimize the effect of ROL restrictions. By convergence, we mean how the matching produced by COSM comes closer to SOSM by setwise inclusions measured by the number of newly similarly matched pairs under the two algorithms. To clarify, let the set of applicants and programs that are matched to the same pairs under COSM and SOSM in the unrestricted ROL case be denoted by Z^0 and similarly when there is a restriction of one on ROLs by Z^1. By stability it holds that $Z^0 \in Z^1$ and the rate of convergence is $|Z^1| - |Z^0|$. Moreover, the set $z^1 = Z^1 \backslash Z^0$ includes the new pairs that are now matched with the same partner under the two algorithms when the ROL restriction is in place.

Example 1. Strict preferences for a balanced market form the following Condorcet cycle are displayed below. For applicants the cycle starts from the most preferred choice while for the programs it is simply reversed horizontally. All programs have a quota of one and the market is balanced with $k > 1$ agents on both sides. The preferences and priorities are displayed below. To clarify, for an applicant s_1 the preferences are $\prec_{s_1}: c_1, c_2, ..., c_{k-2}, c_{k-1}, c_k, s_1$.

$\prec_{s_1}$	$\prec_{s_2}$	$\prec_{s_3}$	$\cdots$	$\prec_{s_{k-2}}$	$\prec_{s_{k-1}}$	$\prec_{s_k}$
c_1	c_k	c_{k-1}	$\cdots$	c_4	c_3	c_2
c_2	c_1	c_k	$\cdots$	c_5	c_4	c_3
c_3	c_2	c_1	$\cdots$	c_6	c_5	c_4
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$
c_{k-2}	c_{k-3}	c_{k-4}	$\cdots$	c_1	c_k	c_{k-1}
c_{k-1}	c_{k-2}	c_{k-3}	$\cdots$	c_2	c_1	c_k
c_k	c_{k-1}	c_{k-2}	$\cdots$	c_3	c_2	c_1

$\prec_{c_1}$	$\prec_{c_2}$	$\prec_{c_3}$	$\cdots$	$\prec_{c_{k-2}}$	$\prec_{c_{k-1}}$	$\prec_{c_k}$
s_2	s_3	s_4	$\cdots$	s_{k-1}	s_k	s_1
s_3	s_4	s_5	$\cdots$	s_k	s_1	s_2
s_4	s_5	s_6	$\cdots$	s_1	s_2	s_3
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$
s_{k-1}	s_k	s_1	$\cdots$	s_{k-4}	s_{k-3}	s_{k-2}
s_k	s_1	s_2	$\cdots$	s_{k-3}	s_{k-2}	s_{k-1}
s_1	s_2	s_3	$\cdots$	s_{k-2}	s_{k-1}	s_k

Suppose that there is a restriction of n on the length of ROLs, and applicants can apply to $k - n$ programs each. It follows that at least $2n$ applicants are matched to their most preferred pair (naturally up to market size k). If k is odd, there are $2n + 1$ applicants matched to their most preferred pair. In addition, the rate of convergence to a singleton core remains constant when the length of ROLs is restricted further. With a ROL restriction of n, the average rank of applicants matched to programs is $n + 1$. When $k > 1$ COSM converges by a rate of two. That is, by further restricting the length of applicants' ROLs by one, two more applicant-program pairs are matched the same under SOSM and COSM. This example is a special case of a preference structure where the rate of convergence to a singleton core is constant.

To illustrate how restricting ROL length affects the core size let us look an example where $n = 3$ and $k = 7$. The first ROL restriction is marked with a slash, second with a backslash, and the third with a cross. When there are no restriction on ROLs in place we have $Z^0 = \{s_5 - c_4\}$. Note that when $n = 1$ we have $Z^1 = \{(s_5 - c_4), (s_2 - c_7), (s_6 - c_3)\}$ and $z^1 = \{(s_2 - c_7), (s_6 - c_3)\}$.

$\prec_{s_1}$	$\prec_{s_2}$	$\prec_{s_3}$	$\prec_{s_4}$	$\prec_{s_5}$	$\prec_{s_6}$	$\prec_{s_7}$
c_1	c_7	c_6	c_5	c_4	c_3	c_2
c_2	c_1	c_7	c_6	c_5	c_4	c_3
c_3	c_2	c_1	c_7	c_6	c_5	c_4
c_4	c_3	c_2	c_1	c_7	c_6	c_5
~~c_5~~	~~c_4~~	~~c_3~~	~~c_2~~	~~c_1~~	~~c_7~~	~~c_6~~
~~c_6~~	~~c_5~~	~~c_4~~	~~c_3~~	~~c_2~~	~~c_1~~	~~c_7~~
~~c_7~~	~~c_6~~	~~c_5~~	~~c_4~~	~~c_3~~	~~c_2~~	~~c_1~~

$\prec_{c_1}$	$\prec_{c_2}$	$\prec_{c_3}$	$\prec_{c_4}$	$\prec_{c_5}$	$\prec_{c_6}$	$\prec_{c_7}$
s_2	s_3	~~s_4~~	s_5	s_6	~~s_7~~	~~s_1~~
s_3	~~s_4~~	~~s_5~~	s_6	s_7	~~s_1~~	s_2
s_4	~~s_5~~	s_6	s_7	~~s_1~~	~~s_2~~	s_3
~~s_5~~	~~s_6~~	s_7	s_1	~~s_2~~	s_3	s_4
~~s_6~~	s_7	s_1	~~s_2~~	~~s_3~~	s_4	s_5
~~s_7~~	s_1	s_2	~~s_3~~	s_4	s_5	~~s_6~~
s_1	s_2	~~s_3~~	~~s_4~~	s_5	s_6	~~s_7~~

When $n = 3$ the set Z^3 produced by COSM has the same applicant-program pairs as matching μ produce by SOSM. That is, the core is a singleton and the average rank of applicants matched to programs is four. After this example, it should be clear that a constant convergence rate is a special case; typically the more competition there is among the programs in each round, the higher the rate of convergence. $\triangleleft$

Remark 1. A careful reader has already induced that the matching μ produced by SOSM under the ROL restriction of n can be generally written down as a partition of pairs produced by COSM under the iterative ROL restrictions:

$$\mu_{SOSM} = Z^0 \cup z^1 \cup \cdots \cup z^n = Z^n,$$

where n is bound to allow ROL length of at least 1. In our example we have $n < k$.

The next example shows why we should not report just the number of differently allocated agents, but also the average ranks of matched agents under COSM and SOSM. Furthermore, it shows why in some instances a short ROL by a single agent can dominate the entire matching to produce a singleton core.

Example 2. Suppose that strict preferences form two distinct Condorcet cycles: for applicants it starts from the first applicant's most preferred program and moves towards the least preferred program – for programs it starts from the least preferred applicant of the first program moving upwards and to the right. That is, for programs the priorities are simply moved left by one column. The market is balanced with k applicants and programs. Each program has a quota of one. The preferences and priorities are shown below.

$\prec_{s_1}$	$\prec_{s_2}$	$\prec_{s_3}$	$\cdots$	$\prec_{s_{k-2}}$	$\prec_{s_{k-1}}$	$\prec_{s_k}$
c_1	c_k	c_{k-1}	$\cdots$	c_4	c_3	c_2
c_2	c_1	c_k	$\cdots$	c_5	c_4	c_3
c_3	c_2	c_1	$\cdots$	c_6	c_5	c_4
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$
c_{k-2}	c_{k-3}	c_{k-4}	$\cdots$	c_1	c_k	c_{k-1}
c_{k-1}	c_{k-2}	c_{k-3}	$\cdots$	c_2	c_1	c_k
c_k	c_{k-1}	c_{k-2}	$\cdots$	c_3	c_2	c_1

$\prec_{c_1}$	$\prec_{c_2}$	$\prec_{c_3}$	$\cdots$	$\prec_{c_{k-2}}$	$\prec_{c_{k-1}}$	$\prec_{c_k}$
s_k	s_{k-1}	s_{k-2}	$\cdots$	s_3	s_2	s_1
s_{k-1}	s_{k-2}	s_{k-3}	$\cdots$	s_2	s_1	s_k
s_{k-2}	s_{k-3}	s_{k-4}	$\cdots$	s_1	s_k	s_{k-1}
$\vdots$	$\vdots$	$\vdots$	$\cdots$	$\vdots$	$\vdots$	$\vdots$
s_3	s_2	s_1	$\cdots$	s_6	s_5	s_4
s_2	s_1	s_k	$\cdots$	s_5	s_4	s_3
s_1	s_k	s_{k-1}	$\cdots$	s_4	s_3	s_2

It is straightforward to see that SOSM would produce a matching where all applicants are matched to their most preferred program and all programs would be matched to their least preferred applicant. Similarly for COSM, programs would get the most preferred applicants and applicants would be matched to their least preferred programs.

If there is a restriction of length n on the length of the ROLs for the applicants, this would result in omitting $k - n$ top choices for programs. Again, SOSM and COSM would match the proposal side with the best possible match when the ROL restriction is in place. Thus, even when the length of the ROLs would be limited to just two, matchings produced by COSM and SOSM would not contain any common pairs. However, the average rank of the applicants matched to programs is very low compared to the unrestricted case. Furthermore, there is no difference between the average rank of applicants matched to programs under COSM and SOSM when the ROL restrictions are in place illustrating that the core is indeed very small.

The core can shrink to a singleton even when the market is balanced and no restrictions on the length of the ROLs are set. To see this, suppose applicant s_k falsely reports his preferences such that only his most preferred program c_2 is acceptable. This would result COSM to run for $k(k-1)+1$ rounds, only to produce the exact same matching as SOSM. Clearly, the chaining of rejections and later proposals can have a dramatic effect on the size of the core. $\lhd$

6 Empirical results

We start by comparing the extreme COSM and SOSM matchings using the clearinghouse data. Summaries of matchings produced by COSM, SOSM, as well as by running a Top Trading Cycles algorithm (Shapley and Scarf, 1974) on top of the COSM matching are shown in Table 3. All algorithms are described in the appendix. Panel (1) shows properties of our replication of the actual matching. As we have seen before, less than a third of applicants receive a slot in each of the three years. The values of $\bar{r}_C$ show that programs accept applicants which they rank about 56th among their applicants on average, but this number is hard to interpret without further context. The grade-based program ranking $\bar{r}_{C(GPA)}$ shows the ranking among all applicants ranging from one to the total number of applicants. This measure is easier to interpret, and seems quite high, much closer to their average rank of all applicants than to the average rank of the top one third of applicants. On the other hand, $\bar{r}_S$ seems quite low, close to its theoretical minimum of 1. Especially the high program rankings seem surprising given the large number of applicants per slot.

Table 3: A comparison of matchings under COSM and two alternative mechanisms

	2011	2012	2013
applicants	50894	50979	52665
slots	16655	16425	15210
(1) *COSM*			
filled slots	16655	16425	15210
$\bar{r}_C$	55.54	55.88	56.19
$\bar{r}_{C(GPA)}$	17915.90	17924.99	17347.49
$\bar{r}_S$	1.27	1.27	1.25
(2) *SOSM*			
filled slots	16655	16425	15210
applicants better off	0	2	0
applicants worse off	0	0	0
$\bar{r}_C$	55.54	55.88	56.19
$\bar{r}_{C(GPA)}$	17915.90	17924.50	17347.49
$\bar{r}_S$	1.27	1.27	1.25
(3) *COSM + Trade*			
filled slots	16655	16425	15210
applicants better off	149	181	172
applicants worse off	0	0	0
envious applicants	9448	10371	10147
$\bar{r}_C$	56.92	57.52	57.89
$\bar{r}_{C(GPA)}$	17919.34	17935.34	17351.76
$\bar{r}_S$	1.26	1.26	1.23

COSM, SOSM, and a Top Trading Cycles algorithm run on top of COSM produce similar matchings when run on empirical data. Justifiably *envious applicants* is a measure of how many applicants can form a blocking pair

Panel (2) shows the results of running SOSM on the same data. In two out of three years, SOSM produces exactly the same matching as COSM, while two applicants switch programs in the third. As a consequence, mean rankings are almost exactly the same across the two matchings within each year. Note how the grade-based mean program ranking, $\bar{r}_{C(GPA)}$, counter-intuitively improves slightly under SOSM during year 2012. This is because the grade-based ranking is an imperfect proxy of the composite score used in the matching.

In Panel (3) we show the results of a Top Trading Cycles algorithm run on top of the COSM matching. Allowing applicants to trade their COSM-allocated slots with each other makes less than two hundred applicants better off in each year, but at the cost of about 10000 applicants' justified envy. From the programs' perspective, trading reduces match quality, but only by a few rankings out of approximately 50000. Mean applicant rankings are similarly improved by 0.01 to 0.02 rankings on average.

In Table 4 we show comparisons of COSM and SOSM matchings using the clearinghouse data, but with different adjustments made to the data. Panel (1) summarizes the first two panels of Table 3, with the COSM and SOSM matchings being identical to each other in two out of three years, and near-identical in 2012. In Panel (2), we replace program priorities with completely randomized priorities. COSM still produces the same matching as SOSM in two out of three years, with two applicants switching programs in 2011. Program rankings improve from about 55 to about 41 and the average applicant ranking deteriorates to about 2. The GPA-based program rankings are not informative in this specification, since grades are now uncorrelated with the program priorities used in the matching.

In Panel (3), we multiply the number of slots in each program by a fixed factor in such a way that the total number of slots equals the total number of applicants. This balances the market. Because applicants only apply to a limited number of slots, about a fifth of the slots remain unallocated in this specification. In all three years, COSM produces the same matching as SOSM. The average rankings of programs are only slightly better than random in this specification, while applicants' average rankings are even better than in the original matching.

In Table 5, we pad applicants' ROLs with the complete set of programs the applicants did not originally apply to added in random order. Since we do not have composite scores for programs applicants did not apply to, we instead use their field-specific matriculation exam GPAs to construct program preferences. A comparison of Panels (1) and (2) shows the effect of swapping out the composite-score based preference with the GPA-based one, while in Panel (3) we have also padded the ROLs to full length. The COSM matchings are identical to the SOSM matchings under grade-based program priorities in Panel (2), while they only differ marginally from each other when every student applies to every single program in Panel (3). The values of $\bar{r}_{C,COSM}$ and $\bar{r}_{C,SOSM}$ show that match quality deteriorates for programs when taking program preferences at face value. This is to be expected since program preferences become more correlated in this specification compared to the baseline. It is however also notable how much match quality improves when we compare the values of $\bar{r}_{C,COSM(GPA)}$ and $\bar{r}_{C,SOSM(GPA)}$ across panels. Part of the reason baseline programs are matched to such a bad selection of applicants in terms of GPA under the baseline matching is because they in practice do not end up selecting on it very strongly,

Table 4: Core size, deviations from empirical data (Part I).

	2011	2012	2013
(1) *Original matching*			
applicants	50894	50979	52665
slots	16655	16425	15210
filled slots	16655	16425	15210
differently allocated applicants	0	2	0
$\bar{r}_{C,COSM}$	55.54	55.88	56.19
$\bar{r}_{C,SOSM}$	55.54	55.88	56.19
$\bar{r}_{C,COSM(GPA)}$	17915.90	17924.99	17347.49
$\bar{r}_{C,SOSM(GPA)}$	17915.90	17924.50	17347.49
$\bar{r}_{S,COSM}$	1.27	1.27	1.25
$\bar{r}_{S,SOSM}$	1.27	1.27	1.25
(2) *Random program priorities*			
slots	16655	16425	15210
filled slots	16655	16425	15210
differently allocated applicants	2	0	0
$\bar{r}_{C,COSM}$	41.81	41.67	40.16
$\bar{r}_{C,SOSM}$	41.81	41.67	40.16
$\bar{r}_{C,COSM(GPA)}$	N/A	N/A	N/A
$\bar{r}_{C,SOSM(GPA)}$	N/A	N/A	N/A
$\bar{r}_{S,COSM}$	1.94	1.96	1.97
$\bar{r}_{S,SOSM}$	1.94	1.96	1.97
(3) *One slot per applicant*			
slots	50894	50979	52665
filled slots	40101	40368	42112
differently allocated applicants	0	0	0
$\bar{r}_{C,COSM}$	192.47	199.24	222.47
$\bar{r}_{C,SOSM}$	192.47	199.24	222.47
$\bar{r}_{C,COSM(GPA)}$	23204.94	23283.02	24019.45
$\bar{r}_{C,SOSM(GPA)}$	23204.94	23283.02	24019.45
$\bar{r}_{S,COSM}$	1.21	1.21	1.20
$\bar{r}_{S,SOSM}$	1.21	1.21	1.20

Original matching is the empirical matching produced by COSM and the *One slot per applicant* is a specification where the market is balanced.

but instead also select on whether the applicant takes the entry exam, and whether the applicant listed the program first on his ROL. For the same reason, match quality deteriorates for applicants in specification (2) as well if we take the applicants' ROLs at face value.

Effectively forcing each applicant to apply to each program in Panel (3) causes a large improvement in match quality from the programs' perspective. Mean applicant ranks deteriorate because applicants previously unmatched are now accepted by other programs, and to a lesser degree because the applicants they replaced are themselves re-matched at programs they prefer less. Applicant welfare is however hard to evaluate since different applicants are accepted into programs than under the baseline.

In Appendix A Table 8, we replicate Tables 4 and 5 , but with actual ROLs replaced with ROLs consisting of independent random draws from the set of all programs available in each year. Because exact program priorities are not known for counterfactual applications, we replace program priorities with grade-based priorities in all specifications, except for the specification in which program priorities are drawn randomly. The core is now a singleton in all specifications, and just like before, program and applicant mean ranks are affected in the expected directions and similar magnitudes when removing the correlation between pro-

Table 5: Core size, deviations from empirical data (Part II).

	2011	2012	2013
(1) *Original matching*			
applicants	50894	50979	52665
slots	16655	16425	15210
filled slots	16655	16425	15210
differently allocated applicants	0	2	0
$\bar{r}_{C,COSM}$	55.54	55.88	56.19
$\bar{r}_{C,SOSM}$	55.54	55.88	56.19
$\bar{r}_{C,COSM(GPA)}$	17915.90	17924.99	17347.49
$\bar{r}_{C,SOSM(GPA)}$	17915.90	17924.50	17347.49
$\bar{r}_{S,COSM}$	1.27	1.27	1.25
$\bar{r}_{S,SOSM}$	1.27	1.27	1.25
(2) *Grade-based program priorities*			
slots	16655	16425	15210
filled slots	16655	16425	15210
differently allocated applicants	0	0	0
$\bar{r}_{C,COSM}$	71.29	71.51	71.34
$\bar{r}_{C,SOSM}$	71.29	71.51	71.34
$\bar{r}_{C,COSM(GPA)}$	9992.65	9923.47	9006.37
$\bar{r}_{C,SOSM(GPA)}$	9992.65	9923.47	9006.37
$\bar{r}_{S,COSM}$	1.49	1.49	1.49
$\bar{r}_{S,SOSM}$	1.49	1.49	1.49
(3) *Full length ROL*			
slots	16655	16425	15210
filled slots	16655	16425	15210
differently allocated applicants	4	6	0
$\bar{r}_{C,COSM}$	5667.89	5555.64	5046.56
$\bar{r}_{C,SOSM}$	5668.02	5556.29	5046.56
$\bar{r}_{C,COSM(GPA)}$	5667.89	5555.64	5046.56
$\bar{r}_{C,SOSM(GPA)}$	5668.02	5556.29	5046.56
$\bar{r}_{S,COSM}$	8.72	8.39	8.57
$\bar{r}_{S,SOSM}$	8.72	8.39	8.57

Original matching is the empirical matching produced by COSM and the *Grade-based program priorities* means common program priorities within a study field. *Full length ROL* uses the empirical ROLs and adds random preferences such that all programs are acceptable for applicants

gram preferences, balancing the market, or simulating full-length ROLs. This suggests that our results are not an artifact of any specific pattern in submitted ROLs.

In summary, the core is commonly a singleton in the clearinghouse data, and remains typically a singleton when we replace program priorities with randomized, uncorrelated priorities, when we balance the market, and when we force each applicant to apply to each program. In spite of the extreme unbalancedness of the market, programs are matched to applicants who are on average not much better in terms of matriculation grades than if they would be matched with randomly drawn applicants. This can be explained both by the low number of programs each applicant applies to, and to the lack of selection on grades that programs seem to exercise in the first place.

The above results suggest that policy changes of a realistic magnitude are not sufficient to create substantial differences between COSM and SOSM matchings. This poses the question of how large policy changes would have to be to produce a substantial difference. For this purpose, we simulate an higher education market where the numbers of applicants and programs is equal to those in the clearinghouse dataset, but programs are equally large, the market is balanced, ROLs are complete, and preferences are completely independent random draws.

The first panel of Table 6, shows characteristics of the COSM and SOSM matchings on the simulated market. All applicants are matched to a program, and almost all applicants are matched to different programs under SOSM than under SOSM. Differences in rankings are large too, with programs being matched to much more preferred applicants under COSM, and applicants being matched to much more preferred applicants under SOSM. Since they are independently drawn, the correlation ρ between the different programs' priorities over applicants is 0 in this specification.

In Panel (2), we change program quota to be proportional to real quota of the programs, leaving the total number of slots unchanged from Panel (1). The correlation is however still very close to zero. All applicants are still matched but the average rank for applicants rises. This is due to the fact that small and large programs receive applications with equal probability. As can be seen from the table, this matching is favorable to the programs in terms of average ranks.

In Panel (3) we restore the uniform program sizes, but change how we draw program priorities so that they are perfectly correlated within field, but uncorrelated across fields. This increases the mean pairwise correlation of priorities between programs to 0.21. Though fewer applicants are now allocated differently between the two extreme stable matchings, their numbers still are substantial. Because program priorities are correlated with field in this specification, program rankings are substantially higher than in the previous specification, and applicant rankings substantially lower.

In Panel (4) we replace the simulated program priorities with the empirical ones. The mean pairwise correlation between programs is about 0.86 in the clearinghouse data. The much lower correlation in Panel (3) illustrates that the high empirical correlation is to a large extent due to correlated priorities across fields. Applicants are even better off than before, and programs are matched to applicants which they do not prefer substantially more than had they been selected at random. The number of differently allocated applicants is moderately small, and applicants are on average at most a few hundreds of ranks better off under SOSM than under COSM. For programs, the difference in average rankings is negligible between the algorithms.

In Table 7 we gradually remove slots from the balanced market of our simulated data set. As can be seen from a comparison of Panels (1) and (2), the core shrinks substantially even for a reduction of as little as 0.1% of the total number of slots in the market. This amounts to a mere 51 to 53 removed slots. Differences in mean rankings between the algorithms disappear almost completely. Applicants are on average substantially worse off than on the balanced market, and programs are substantially better off, regardless of algorithm.

Further reducing the number of slots to 99% of the number of applicants in Panel (3), and then to 90% of the number of applicants in Panel (4) further reduces the number of differently allocated applicants between algorithms, and improves program mean rankings at the expense of applicant rankings. Our simulations confirm the findings of Ashlagi et al. (2017) that even a small imbalance in the market can drive the core close to a singleton.

We next analyze the effect of restricting the length of ROLs on the size of the core. As stated by Ashlagi et al. (2017), restrictions on the length of the ROLs can be seen as introduction of indirect unbalancedness to the market.

Table 6: Core size, deviations from a simulated market.

	2011	2012	2013
(1) *Baseline*			
ρ	0.00	0.00	0.00
applicants	50894	50979	52665
slots	50894	50979	52665
filled slots	50894	50979	52665
differently allocated applicants	49328	49444	50967
$\bar{r}_{C,COSM}$	602.71	614.00	685.49
$\bar{r}_{C,SOSM}$	20104.28	20037.30	20723.16
$\bar{r}_{S,COSM}$	42.37	41.61	38.58
$\bar{r}_{S,SOSM}$	1.28	1.28	1.28
(2) *Unequally sized programs*			
ρ	0.00	0.00	0.00
slots	50894	50979	52665
filled slots	50894	50979	52665
differently allocated applicants	37103	36285	37484
$\bar{r}_{C,COSM}$	1182.78	1266.68	1387.45
$\bar{r}_{C,SOSM}$	4371.85	4398.77	4805.15
$\bar{r}_{S,COSM}$	39.34	36.48	33.83
$\bar{r}_{S,SOSM}$	10.51	10.49	9.79
(3) *Grouped program priorities*			
ρ	0.21	0.21	0.23
slots	50894	50979	52665
filled slots	50894	50979	52665
differently allocated applicants	10511	8255	7292
$\bar{r}_{C,COSM}$	17253.50	17959.65	18977.83
$\bar{r}_{C,SOSM}$	21308.51	21118.09	21667.16
$\bar{r}_{S,COSM}$	1.77	1.65	1.64
$\bar{r}_{S,SOSM}$	1.27	1.29	1.32
(4) *Grade-based program priorities*			
ρ	0.86	0.86	0.87
slots	50894	50979	52665
filled slots	50894	50979	52665
differently allocated applicants	197	51	8
$\bar{r}_{C,COSM}$	25178.94	25230.71	26081.85
$\bar{r}_{C,SOSM}$	25186.70	25232.35	26082.23
$\bar{r}_{S,COSM}$	1.32	1.33	1.28
$\bar{r}_{S,SOSM}$	1.25	1.30	1.28

Unequally sized programs means that the market is balanced but the number of slots in each program is grown proportionally relative to their original size. *Grouped program priorities* stands for identical program priorities within a study field. In *Grade-based program priorities* the simulated program priorities are changed back to the empirical ones. Correlation within program priorities is denoted by ρ

Hence, one can expect ROL restrictions to affect the size of the core as we saw in Section 5.

Simulations using random uniform preferences in a balanced market are shown for different length ROLs in Figures 1 and 2. Slightly restricting the length of ROLs has a milder effect on the size of the core than slightly unbalanced

Table 7: Core size, deviations from a balanced market with uniform preferences.

	2011	2012	2013
(1) *Baseline*			
applicants	50894	50979	52665
slots	50894	50979	52665
filled slots	50894	50979	52665
differently allocated applicants	49328	49444	50967
$\bar{r}_{C,COSM}$	602.71	614.00	685.49
$\bar{r}_{C,SOSM}$	20104.28	20037.30	20723.16
$\bar{r}_{S,COSM}$	42.37	41.61	38.58
$\bar{r}_{S,SOSM}$	1.28	1.28	1.28
(2) *0.999 slots per applicant*			
slots	50843	50928	52612
slots	50843	50928	52612
filled slots	50843	50928	52612
differently allocated applicants	153	13	87
$\bar{r}_{C,COSM}$	402.31	401.70	461.24
$\bar{r}_{C,SOSM}$	403.53	401.80	462.08
$\bar{r}_{S,COSM}$	63.52	63.84	57.12
$\bar{r}_{S,SOSM}$	63.33	63.82	57.04
(3) *0.99 slots per applicant*			
slots	50385	50469	52138
slots	50385	50469	52138
filled slots	50385	50469	52138
differently allocated applicants	23	10	2
$\bar{r}_{C,COSM}$	267.03	275.46	309.46
$\bar{r}_{C,SOSM}$	267.14	275.51	309.47
$\bar{r}_{S,COSM}$	91.96	89.45	81.95
$\bar{r}_{S,SOSM}$	91.92	89.43	81.95
(4) *0.90 slots per applicant*			
slots	45804	45881	47398
slots	45804	45881	47398
filled slots	45804	45881	47398
differently allocated applicants	0	0	0
$\bar{r}_{C,COSM}$	134.01	136.36	154.65
$\bar{r}_{C,SOSM}$	134.01	136.36	154.65
$\bar{r}_{S,COSM}$	142.81	140.24	129.08
$\bar{r}_{S,SOSM}$	142.81	140.24	129.08
(5) *Original unbalancedness*			
slots	16655	16425	15210
filled slots	16655	16425	15210
differently allocated applicants	0	0	0
$\bar{r}_{C,COSM}$	23.48	23.65	23.24
$\bar{r}_{C,SOSM}$	23.48	23.65	23.24
$\bar{r}_{S,COSM}$	206.45	202.85	188.62
$\bar{r}_{S,SOSM}$	206.45	202.85	188.62

0.999 slots per applicant stands for removing 0.1%, *0.99 slots per applicant* for removing 1%, and *0.90 slots per applicant* for removing 10% of slots from program capacities. In *Original unbalancedness*, the number of slots per applicant is approximately one third varying for each year markets. A notable relative change occurs after the length of ROLs is smaller than 50. However, the most radical change occurs after we restrict the length from 10 to the maximal length of 4 of the original application process[4]. The average rank of applicants matched to programs increases significantly from the baseline of approximately 600 to 22 000 when the length of the ROLs is restricted to four. Thus, programs get worse matches in terms of average ranks when the ROL lengths are restricted heavily.

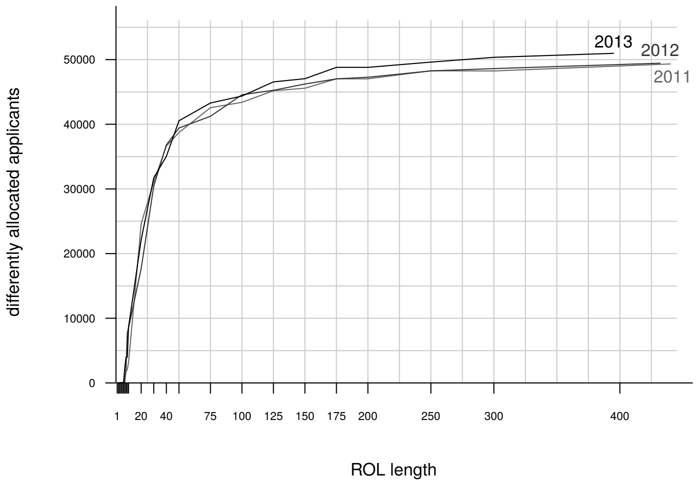

Figure 1: Number of differently allocated applicants under COSM and SOSM as a function of ROL length for each year

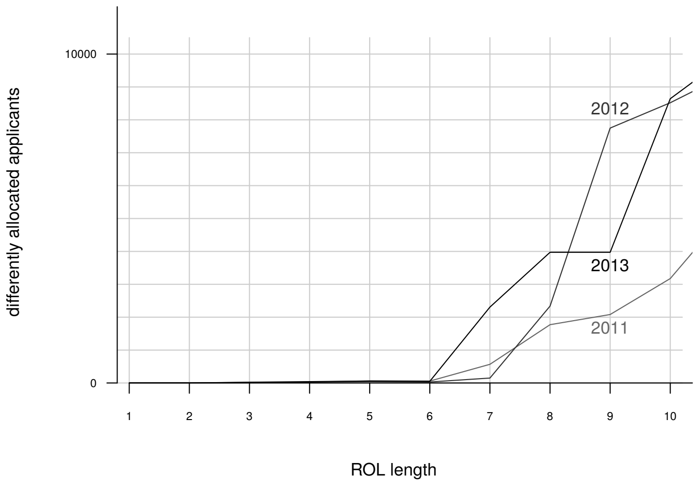

Figure 2: Number of differently allocated applicants under COSM and SOSM with ROL lengths of one to ten for each year

In summary, the empirical correlation of program priorities reduces the size

[4]Additional simulation results of various ROL length changes are found in Appendix A Table 9.

of the core substantially, even if the core is not a singleton in any of the three years. It further improves applicant outcomes at the cost of program outcomes, especially in the college-optimal matching. Additionally, the unbalancedness of the market dramatically reduces the size of the core. Average ranks for programs tend to decrease while average ranks for applicants tend to increase as the unbalancedness becomes more severe. Finally, limiting the length of ROLs reduces the size of the core dramatically.

7 Discussion

In the literature, it is sometimes suggested that it is beyond question that schools are there for the students, and not the other way around. If so, it is hard to see that COSM would ever be preferable to SOSM. It can however also be argued that school preferences at times reflect social preferences on who should be admitted where, especially when school preferences are based on government laws and regulations. Under such circumstances, the policy maker faces a real trade-off in weighing the student's wishes against those of society.

Under idealized circumstances, with independently drawn preferences, a balanced market, and long ROLs, the difference between SOSM and COSM matchings can be large. This suggests that the choice between SOSM and COSM matchings can have policy relevance. In this paper, we however find that the differences between COSM and SOSM matchings are negligible in Finnish polytechnic applications. We mainly contribute this result to the unbalancedness of the market (see Ashlagi et al., 2017), but also to the low number of programs Finnish applicants each apply to, as well as to the correlation between program preferences.

Correlated preferences, unbalanced markets and short ROLs are ubiquitous in real application systems, and conditional on submitted preferences the choice between the COSM and SOSM matchings themselves is likely to be trivial under most circumstances. The policy maker should thus concentrate on other design aspects of the application system. SOSM may for example be preferred to COSM in order to discourage applicants from trying to strategize.

The similarity of SOSM and COSM matchings does however not imply that a trade-off between student and school preferences does not exist. Even though Finnish polytechnics jointly accept less than a third of applicants each year, we find that the average quality of accepted applicants in terms of matriculation grades is not much better than of the median applicant. At the same time, a majority of accepted applicants get accepted to their most preferred program, even when we randomize applicant preferences. Programs' low match quality is a direct consequence of the low number of programs applicants each apply to. Policies such as giving extra points for the first listed choice, or requiring applicants to take separate entry exams for the programs they wish to apply to, have the effect of further restricting the effective ROL length, and may thus have the side effect of reducing programs' match quality while redistributing slots between students in a manner that may or may not be intended. We argue that the policy maker would do well to take the trade-off between student and school preferences more explicitly into account when designing the application process.

References

Akahoshi, T. (2014) Singleton core in many-to-one matching problems. Math Soc Sci 72:7–13.

Ashlagi, I., Kanoria, Y., Leshno, J. (2017) Unbalanced Random Matching Markets: The Stark Effect of Competition. J Polit Econ 125:69–98.

Clark, S., (2006) The Uniqueness of Stable Matchings. Contrib Theor Econ 6:1–28.

Colenbrander, A. (1996a). Examining the NRMP algorithm. Acad Med 71 (4):309–312.

Colenbrander, A. (1996b) Match algorithms revisited. Acad Med 71 (5):414–416.

Eeckhout, J. (2000) On the uniqueness of Stable Marriage Matchings. Econ Lett 69:1–8.

Gale, D., Shapley, L. (1962) College Admissions and the Stability of Marriage. Am Math Mon 69:9–15.

Holzman, R., Samet, D. (2014) Matching of like rank and the size of the core in the marriage problem. Games Econ Behav 88:277–285.

Irving, R., Leather, P. (1986) The Complexity of Counting Stable Marriages. SIAM J Comput 15 (3):655–667.

Kojima, F., Pathak, P., (2009) Incentives and Stability in Large Two-Sided Matching Markets. Am Econ Rev 99 (3):608–627.

Peranson, E., Randlett, R. (1995) The NRMP matching algorithm revisited: theory versus practice. National Resident Matching Program. Acad Med 70 (6):477–484.

Pittel, B. (1989) The average number of stable matchings. SIAM J Discrete Math 2 (4):530 – 549.

Roth, A., Peranson, E. (1999) The Redesign of the Matching Market of American Physicians: Some Engineering Aspects of Economic Design. Am Econ Rev 89 (4):748–780.

Roth, A., Sotomayor, M (1992) Two-Sided Matching: A Study in Game-Theoretic Modeling and Analysis, Cambridge University Press, Cambridge

Salonen, H., Salonen, M. (2018) Mutually best matches, Math Soc Sci 91:42-50.

Shapley, L., Scarf, H. (1974) On cores and indivisibility, J Math Econ 1:23–37.

A Algorithms and Tables

In order to facilitate replication of our results, we describe the matching algorithms used in the paper. The descriptions are modified such that they follow more closely the actual simulation code used rather than the common theoretical presentation type of describing the algorithms.

> *College-optimal stable mechanism (COSM)*
>
> Each program has a quota of q_j slots.
>
> Step 1: each program proposes to its q_j most preferred applicants. Applicants each tentatively accept their most preferred proposal. Other proposals are rejected.
>
> Step k: each program proposes to a number of their most preferred remaining applicants equal to the number of rejections it has received during the previous round. If the program has less remaining applicants to propose to than its number of rejections, it proposes to all remaining applicants. Applicants each tentatively accept their most preferred proposal. Other proposals are rejected.
>
> The algorithm ends when no more proposals are rejected, and every applicant is permanently accepts his last assignment.

Though this algorithm runs relatively efficiently in terms of computing time since programs can each propose to up to q_j applicants in each round, for large numbers of applicants applying to a large number of programs, the algorithm still spends unnecessary time chasing small numbers of rejection chains through the full data set. We have found that adapting the algorithm to iteratively run on subsets of the full preference tables can yield performance improvements of orders of magnitude for large data sets, intuitively because it allows the computer to resolve more rejection chains simultaneously in each step.

> *College-optimal stable mechanism (COSM) for large data sets*
>
> Step 1.0: consider only each program's $\alpha \cdot q_j$ most preferred applicants, with $\alpha \geq 1$. If the number of applicants is smaller than $\alpha \cdot q_j$ for any program, include all applicants for that program.
>
> Step 1.1: each program proposes to its q_j most preferred applicants. Applicants each tentatively accept their most preferred proposal. Other proposals are rejected.
>
> Step 1.ℓ: each program proposes to a number of their most preferred remaining applicants equal to the number of rejections it received during the previous round. If the program has less remaining applicants to propose to than the number of rejections, it proposes to the remaining applicants. Applicants each tentatively accept their most preferred proposal. Other proposals are rejected.
>
> Step 1 ends when no more proposals are rejected within this subset of preferences.
>
> Step k.0: From the full preference tables, consider only each program's $\alpha \cdot q_j$ most preferred applicants that have not rejected that

program yet. If the number of such applicants is smaller than $\alpha \cdot q_j$ for any program, include all remaining applicants for that program.

Step k.1: each program proposes to a number applicants equal to its current number of unfilled slots. Applicants each tentatively accept their most preferred proposal. Other proposals are rejected.

Step k.ℓ: each program proposes to a number of their most preferred remaining applicants equal to the number of rejections it received during the previous round. If the program has less remaining applicants to propose to than the number of rejections, it proposes to the remaining applicants. Applicants each tentatively accept their most preferred proposal. Other proposals are rejected.

Step k ends when no more proposals are rejected within its subset of preferences.

The algorithm ends when all slots have been tentatively filled or all possible proposals have been made.

The optimal α is a trade-off between the computing time used to select a new subset of preferences in steps k.0, and the additional time that is needed for steps k.ℓ when run on a larger subset. A larger α reduces the number of i steps needed, but increases the number of j steps within each i step. In settings where we simulate outcomes where every single applicant applies to every single program, and the number of applications to each program is substantially larger than its quota, we find that our algorithm runs quickest with an α of about 1.6.

Student-optimal stable mechanism (SOSM)

Step 1: each applicant proposes to his first choice. Each program tentatively accepts a number of applicants equal to its quota. Other proposals are marked as rejected.

Step k: each rejected applicant proposes to his first remaining choice. Each program tentatively accepts a total number of applicants equal to its quota. Other proposals are marked as rejected.

The algorithm ends when no more proposals are rejected, and every applicant is permanently assigned to his last tentative assignment.

SOSM is less problematic to run on large data sets with small to moderate numbers of programs since a number of proposals up to the number of applicants can be processed in each step, and the total number of steps is reduced by the more limited total number of programs. Even if it would in theory be possible to iteratively subset preferences in the same way as as with COSM, we have not attempted to do so in practice.

We do adapt a top trading cycles algorithm to run faster by letting it look for cycles on the program level rather than on the individual level.

Top trading cycles after COSM (COSM + Trade)

Tentatively assign each applicant to his COSM matching. Make the assignment of applicants who have received no slot permanent since no other applicants will want to trade with them. Consider only the remaining, tentatively assigned applicants. Also create an arbitrary random variable ε which contains a unique value for each applicant.

Cycle step 1: within the programs that applicants have been tentatively (but not permanently) assigned to at this point, let all applicants point to the program they prefer most. If there are any applicants pointing to the program they have already been tentatively assigned to, permanently assign these applicants to those programs and continue with the cleanup step. Otherwise continue with the next cycle step.

Cycle step 2: aggregate pointers to the program level, and find out if there are any cycles of length 2 on the program level, i.e. for all combinations of programs c_j and c_{-j}, how many applicants are tentatively assigned to program c_{-j} but prefer c_j and vice versa. For each combination of programs for which both numbers are positive, take the minimum of the two numbers of potential switchers $n_{switchers}$. Then permanently assign the $n_{switchers}$ potential switchers who have the lowest value of ε in either program to the program she would like to switch to. If any cycle was found, continue to the cleanup step. Otherwise continue with the next cycle step.

Cycle step k: find out if there are any cycles of length k on the program level. Within each cycle, take the minimum of switchers that want to switch from one program to the next, and permanently assign that number of applicants with the lowest values of ε to their preferred program. If any cycle was found, continue to the cleanup step. Otherwise continue with the next cycle step.

Cleanup step: remove permanently assigned applicants from the applicants under consideration. If all applicants are permanently assigned, the algorithm ends. If not, return to cycle step 1.

Table 8 shows the effects of the same specifications from Tables 4 and 5, but for random ROLs for applicants and grade-based priorities for programs with the average ranking ranging from one to the total number of applicants $|S|$. The core is a singleton in each case.

Table 9 shows the simulations when the length of ROLs differs from the maximum. The preferences on both sides of the market are completely uncorrelated and the market is balanced. As we saw from Figures 1 and 2, the core converges to a singleton when the ROL restriction comes closer to the real world case. Note that the average rankings for programs are surprisingly high; when the ROL lengths are of original length for each applicant, programs on average get matched to applicants it ranks as good as random applicants.

Table 8: Robustness checks of Tables 4 and 5

	2011	2012	2013
(1) *Baseline: grade-based program priorities*			
applicants	50894	50979	52665
slots	16655	16425	15210
filled slots	16494	16291	15122
differently allocated applicants	0	0	0
$\bar{r}_{C,COSM}$	9215.94	9347.31	8650.62
$\bar{r}_{C,SOSM}$	9215.94	9347.31	8650.62
$\bar{r}_{S,COSM}$	1.49	1.49	1.47
$\bar{r}_{S,SOSM}$	1.49	1.49	1.47
(2) *Random program priorities*			
slots	16655	16425	15210
filled slots	16496	16291	15123
differently allocated applicants	0	0	0
$\bar{r}_{C,COSM}$	5924.58	5984.92	5569.14
$\bar{r}_{C,SOSM}$	5924.58	5984.92	5569.14
$\bar{r}_{C,COSM(GPA)}$	N/A	N/A	N/A
$\bar{r}_{C,SOSM(GPA)}$	N/A	N/A	N/A
$\bar{r}_{S,COSM}$	1.97	1.96	1.97
$\bar{r}_{S,SOSM}$	1.97	1.96	1.97
(3) *One slot per applicant*			
slots	50894	50979	52665
filled slots	42777	42697	44107
differently allocated applicants	0	0	0
$\bar{r}_{C,COSM}$	21943.82	21967.34	22660.95
$\bar{r}_{C,SOSM}$	21943.82	21967.34	22660.95
$\bar{r}_{S,COSM}$	1.32	1.32	1.31
$\bar{r}_{S,SOSM}$	1.32	1.32	1.31
(5) *Full length ROL*			
slots	16655	16425	15210
filled slots	16655	16425	15210
differently allocated applicants	0	0	0
$\bar{r}_{C,COSM}$	5643.28	5521.54	5051.09
$\bar{r}_{C,SOSM}$	5643.28	5521.54	5051.09
$\bar{r}_{S,COSM}$	13.36	14.12	13.39
$\bar{r}_{S,SOSM}$	13.36	14.12	13.39

Core size for random applicant preferences and grade-based priorities for the programs with specifications similar to those in Table 4. Average ranks are measured from one to to the total number of applicants

Table 9: Core size and varying ROL lengths

	2011	2012	2013
(1) *Full length*			
applicants	50894	50979	52665
slots	50894	50979	52665
filled slots	50894	50979	52665
differently allocated applicants	49328	49444	50967
$\bar{r}_{C,COSM}$	602.71	614.00	685.49
$\bar{r}_{C,SOSM}$	20104.28	20037.30	20723.16
$\bar{r}_{S,COSM}$	42.37	41.61	38.58
$\bar{r}_{S,SOSM}$	1.28	1.28	1.28
(2) *ROL length: 200*			
slots	50894	50979	52665
filled slots	50894	50979	52665
differently allocated applicants	47026	47270	48798
$\bar{r}_{C,COSM}$	1529.72	1477.56	1549.07
$\bar{r}_{C,SOSM}$	20194.22	20193.04	21047.96
$\bar{r}_{S,COSM}$	16.89	17.28	17.10
$\bar{r}_{S,SOSM}$	1.28	1.28	1.26
(3) *ROL length: 100*			
slots	50894	50979	52665
filled slots	50894	50979	52665
differently allocated applicants	43395	44558	44363
$\bar{r}_{C,COSM}$	2946.64	2529.92	3326.33
$\bar{r}_{C,SOSM}$	20194.22	20193.04	21047.96
$\bar{r}_{S,COSM}$	8.76	10.15	7.90
$\bar{r}_{S,SOSM}$	1.28	1.28	1.26
(4) *ROL length: 50*			
slots	50894	50979	52665
filled slots	50894	50979	52665
differently allocated applicants	38726	39412	40541
$\bar{r}_{C,COSM}$	4839.49	4561.87	4861.77
$\bar{r}_{C,SOSM}$	20194.22	20193.04	21047.96
$\bar{r}_{S,COSM}$	5.32	5.63	5.42
$\bar{r}_{S,SOSM}$	1.28	1.28	1.26
(5) *ROL length: 10*			
slots	50894	50979	52665
filled slots	50894	50979	52665
differently allocated applicants	3175	8520	8643
$\bar{r}_{C,COSM}$	18927.84	16810.61	17608.57
$\bar{r}_{C,SOSM}$	20194.22	20193.04	21047.96
$\bar{r}_{S,COSM}$	1.36	1.53	1.51
$\bar{r}_{S,SOSM}$	1.28	1.28	1.26
(6) *ROL length: 4*			
slots	50894	50979	52665
filled slots	50856	50942	52633
differently allocated applicants	2	35	5
$\bar{r}_{C,COSM}$	21718.15	21566.17	22335.27
$\bar{r}_{C,SOSM}$	21719.27	21579.26	22336.98
$\bar{r}_{S,COSM}$	1.18	1.19	1.18
$\bar{r}_{S,SOSM}$	1.18	1.19	1.18
(7) *ROL length: Original length*			
slots	50894	50979	52665
filled slots	50061	50242	51867
differently allocated applicants	0	0	0
$\bar{r}_{C,COSM}$	23765.64	24014.70	24784.81
$\bar{r}_{C,SOSM}$	23765.64	24014.70	24784.81
$\bar{r}_{S,COSM}$	1.06	1.06	1.05
$\bar{r}_{S,SOSM}$	1.06	1.06	1.05

Original length stands for the actual ROL length of each applicant