

Capacity Planning in Stable Matching

Federico Bobbio, Margarida Carvalho

CIRRELT and DIRO, Université de Montréal, federico.bobbio@umontreal.ca, carvalho@iro.umontreal.ca

Andrea Lodi

Jacobs Technion-Cornell Institute, Cornell Tech, andrea.lodi@cornell.edu

Ignacio Rios

School of Management, The University of Texas at Dallas, ignacio.riosuribe@utdallas.edu

Alfredo Torrico

CDSES, Cornell University, alfredo.torrico@cornell.edu

Motivated by the shortage of seats that the Chilean school choice system is facing, we introduce the problem of jointly increasing school capacities and finding a student-optimal assignment in the expanded market. Due to the theoretical and practical complexity of the problem, we provide a comprehensive set of tools to solve the problem, including different mathematical programming formulations, a cutting plane algorithm, and two heuristics that allow obtaining near-optimal solutions quickly. On the theoretical side, we show the correctness of our formulations, different properties of the objective and feasible region that facilitate computation, and also several properties of the underlying mechanism to find a student-optimal matching under capacity expansions. On the computational side, we use data from the Chilean school choice system to demonstrate the impact of our framework and derive insights that could help alleviate the problem. Our results show that each additional seat can benefit multiple students and that we can effectively target the assignment of previously unassigned students or improve the assignment of several students through improvement chains. Nevertheless, our results show that the marginal effect of each additional seat is decreasing and that simply adding seats is insufficient to ensure every student gets assigned to some school. Finally, we discuss several extensions of our framework, showcasing its flexibility to accommodate different needs.

Key words: Stable Matching, Capacity Planning, School Choice, Integer Programming

1. Introduction

Centralized mechanisms are becoming the standard approach to solve several assignment problems, including the allocation of students to schools, high-school graduates to colleges, residents to hospitals, and refugees to cities. In most of these markets, a desirable property of the assignment is *stability*, which guarantees that no pair of agents has incentive to circumvent the matching. As discussed in (Roth and Sotomayor 1990) and (Roth 2002), finding a stable matching is crucial for the clearinghouse's success, long-term sustainability and also ensures some notion of fairness as it eliminates the so-called *justified-envy*.¹

¹ See Romm et al. (2020) for a discussion of the differences between *stability* and *no justified-envy*.

A common assumption in these markets is that capacities are fixed and known. However, capacities are only a proxy of how many agents can be accommodated, and there might be some flexibility to modify them in many settings. For instance, in some college admissions systems, colleges may increase their capacities to admit all tied students competing for the last seat (Rios et al. 2021). Moreover, in several colleges/universities, the number of seats offered in a given course or program is adjusted based on their popularity among students.² In school choice, school districts may experience overcrowding, where some schools serve more students than their designed capacity.³ In response, school districts often explore alternative strategies to accommodate the excess demand, such as utilizing portable classrooms, implementing multi-track or staggered schedules, or adopting other temporary measures, and use students' preferences as input to make these decisions.

An actual example of the latter is the crisis that the Chilean school choice system—known as “Sistema de Admisión Escolar” (SAE)—is currently facing, where the strain on the public school system has reached critical levels. Specifically, in many regions of the country, the number of students applying to the system largely exceeds the number of seats offered. Consequently, more than 3,000 students were not assigned and were forced out of the public system in the last admissions process (2023-2024), with many of them resorting to home-schooling as they cannot afford private schools. One of the most notable examples of the crisis is at the Pre-K level in the Antofagasta region, where there are roughly 0.9 seats per student applying to the system, and 4 out of 10 students do not get assigned in any of their preferred schools after the main round of the admissions process (see Table 1). As a result, there has been public outrage, with parents, teacher associations, members of Congress, and children advocates calling for an immediate solution.

In response to the crisis, Chile's Ministry of Education (MINEDUC) implemented the “Plan de Fortalecimiento de Matrícula” (MINEDUC 2022). Since 2022, this plan has increased capacity by more than 13,000 seats, with nearly 8,700 added during the latest admission process. Additionally, MINEDUC launched a platform that enables parents to search for schools with available capacity and request a seat if their child was left unassigned, streamlining the waitlists processing during the after-market period. Despite these efforts the problem persists, partially because the additional seats often end up allocated sub-optimally, limiting the plan's effectiveness.

² Examples include the School of Engineering at the University of Chile, where all students who want to study any of its programs must take a shared set of courses in the first two years and then must apply to a specific program (*e.g.*, Civil Engineering, Industrial Engineering, etc.) based on their GPA, without knowing the number of seats available in each of them. Similarly, in many schools that use course allocation systems such as Course Match Budish et al. (2016), over-subscribed courses often increase their capacities, while under-subscribed ones are merged or canceled.

³ According to the results of a nationwide survey National Center for Education Statistics (1999), 22% of schools in the US experienced some degree of overcrowding, and 8% had enrollments that exceeded their capacity by more than 25%.

Table 1 Summary Statistics: Pre-K in Antofagasta

Year	Students	Schools	Seats	Seats per student	Assignment		Enrollment	
					Total	Rate	Total	Rate
2018	4371	75	3551	0.812	2509	0.574	2422	0.554
2019	4455	74	3605	0.809	2435	0.547	2372	0.532
2020	3918	75	3675	0.938	2647	0.676	2560	0.653
2021	3629	76	3628	1.000	2368	0.653	2312	0.637
2022	3952	76	3573	0.904	2417	0.612	2332	0.590
2023	3795	71	3558	0.938	2412	0.636	N/A	N/A

Note. Summary statistics on applicants, schools, seats, and assignments correspond to the main round of the assignment process.

Finding a stable matching and making capacity decisions jointly poses several challenges. First, standard approaches like the Deferred Acceptance algorithm (DA) (Gale and Shapley 1962) require fixed and known capacities, making them unsuitable. Second, the clearinghouse may have multiple objectives beyond finding a student-optimal assignment, and these objectives can introduce trade-offs in allocating additional seats. For instance, one possible goal is to maximize *access*, *i.e.*, to allocate extra seats to achieve a stable matching that maximizes the total number of assigned students. Like most school choice systems, MINEDUC is legally required to guarantee that every student who wishes to attend a public school can do so, making *access* one of their primary goal. However, the clearinghouse may also aim to prioritize *improvement*, *i.e.*, enhancing the assignment of high-priority students. This approach is common in merit-based settings such as college admissions and entry level job-markets, where the optimal placement of candidates is crucial.⁴ Finally, even if the objective is clear, the computation of a student-optimal matching under capacity planning is theoretically hard (Bobbio et al. 2022), in sharp contrast with the standard case (with no capacity decisions) that can be solved in polynomial time.

In this paper, we study the problem of jointly deciding how to allocate a budget of additional seats and finding a student-optimal assignment that takes advantage of the expanded capacities. The goal of our paper is threefold. First, to present a novel framework for finding a student-optimal assignment while allocating extra capacities targeting different objectives, such as access, improvements, or any combination. Second, to develop an optimization toolbox composed of exact algorithms and heuristics that allows us to efficiently obtain optimal or near-optimal solutions. This is of utmost importance, as a flexible and effective set of methods would facilitate policymakers in making capacity decisions. Our last goal is to obtain operational insights that could help MINEDUC navigate the current educational crisis in Chile.

⁴ Note that this trade-off between *access* and *improvement* does not arise in the standard version of the problem, as there is a unique student-optimal stable matching under fixed capacities.

1.1. Contributions

Our work makes several contributions that we now describe in detail.

Model and mechanism analysis. To capture the problem described above, we introduce a stylized model of a many-to-one matching market in which the clearinghouse can make capacity planning decisions while simultaneously finding a student-optimal stable matching, generalizing the standard model by Gale and Shapley (1962). We show that the clearinghouse can prioritize different goals by changing the penalty values of unassigned students. Namely, it can obtain the minimum or the maximum cardinality student-optimal stable matchings and, thus, prioritize improvements and access, respectively. In addition, we study other properties of interest, including agents' incentives and the monotonicity of the mechanism.

Exact solution methods. We first formulate our problem as an integer quadratically constrained program by extending existing approaches. On the one hand, to overcome the challenges imposed by the non-linearity of some constraints, we provide a linear reformulation exploiting the concept of stability. Using simulations, we show that this formulation—which we refer to as *compact formulation with L-constraints*—outperforms other linearizations obtained using standard techniques. On the other hand, to overcome the challenges imposed by the integrality of the decision variables, we introduce a second formulation—which we refer to as *generalized comb formulation*—that relaxes the integrality of the matching variables and recovers it by adding an exponential number of comb constraints. To overcome the problem's dimensionality, we propose an efficient cutting plane algorithm that relies on a novel separation algorithm that identifies the most violated constraint for each school in each iteration. Furthermore, we show that we can speed up the search for violated comb constraints by exploiting (i) new structural results characterizing properties of the objective function and the feasible region and (ii) two related matchings to limit the search for new violated combs. The sum of all these technical enhancements ensures that our cutting-plane method outperforms the benchmarks solved by state-of-the-art mixed-integer programming solvers, both in terms of solving time and solution quality (optimality gap) guarantees.

Heuristic solution methods. As shown in (Bobbio et al. 2022), the problem is NP-hard and cannot be approximated within a $\mathcal{O}(n^{(\frac{1}{6}-\varepsilon)})$ factor, where n is the number of students. Moreover, many real-life instances could not be solved in a reasonable time (even using our cutting plane algorithm), making our exact methods unsuitable when time is sensitive (*e.g.*, to perform policy evaluations and counterfactual analyses that require many simulations). These limitations motivated us to devise two efficient heuristics that quickly provide near-optimal solutions. Our first heuristic adapts the standard *Greedy* algorithm for set functions, sequentially adding in each iteration one extra seat to the school leading to the greatest marginal improvement in the objective function. Our second heuristic, called *LPH*, proceeds in two steps: (i) solve the problem without stability constraints to

find the allocation of extra seats, and (ii) find the student-optimal stable matching conditional on the capacities defined in the first step. Our computational experiments show that both heuristics significantly reduce the time to find a near-optimal solution, with LPH being the fastest and unaffected by the budget size. Hence, LPH could be a good approach to quickly obtain *high-quality* solutions to large-scale instances of the problem in time sensitive settings.

Practical insights and societal impact. To illustrate the benefits of embedding capacity decisions towards addressing the current crisis in Chile, we adapt our framework to solve the problem at the Pre-K level in the Antofagasta region, including all the specific features described by Correa et al. (2022). First, we show that each additional seat can benefit multiple students, but the marginal improvement is decreasing. Moreover, we observe that having a rate of one seat per student is not enough to solve the problem due to the high correlation in preferences and the short length of preference lists. Second, in line with our theoretical results, we find that *access* and *improvement* can be prioritized depending on how unassigned students are penalized in the objective. However, for high budget values, the differences in entry are relatively small and, thus, a low penalty that benefits the maximum number of students may be preferable. Third, we show that we can still obtain near-optimal solutions if schools limit the additional seats they can receive to ensure practical feasibility, avoiding solutions that overcrowd some over-demanded schools.

These positive results have led to a fruitful collaboration with MINEDUC and the NGO responsible for the allocation to test our framework in the field and devise potential solutions to the crisis. Our model can be easily adapted to incorporate all the elements of the Chilean system (*e.g.*, static and dynamic priorities, quotas, secured enrollment, etc.) and address other policy-relevant questions. For instance, it can be used to optimally decrease capacities in school districts experiencing significant drops in enrollment (Tucker 2022). Additionally, our model could be employed to optimally allocate tuition waivers under budget constraints, as seen in Hungary’s college admissions system. Our methodology is also applicable in other markets, such as refugee resettlement (Delacretaz et al. 2016, Andersson and Ehlers 2020, Ahani et al. 2021), where local authorities determine how many refugees they are willing to receive but could increase their capacity given proper incentives. Similarly, in healthcare rationing (Pathak et al. 2020, Aziz and Brandl 2021), policymakers could make additional investments to expand available resources. These examples highlight the importance of jointly optimizing stable assignments and capacity decisions, demonstrating the model’s potential to answer crucial questions across various settings.

1.2. Organization

The remainder of the paper is organized as follows. In Section 2, we provide a literature review. In Section 3, we formalize the stable matching problem with capacity decisions and present our

exact methods. In Section 4, we present our heuristics. In Section 5, we evaluate our framework using Chilean school choice data. In Section 6, we discuss several properties of the mechanism to find a student-optimal stable matching with capacity decisions. Finally, in Section 7, we draw some concluding remarks. All the proofs, examples, extensions and additional discussions can be found in the Appendix.

2. Related Work

Our paper lies at the intersection of three streams of literature. First, it relates to the mathematical programming formulations of the stable matching problem and its extensions. Second, it contributes to the emerging field that integrates matching and capacity decisions. Finally, it connects to the literature on school choice.

Our contribution to these areas is threefold: (i) we extend the standard stable matching problem by incorporating capacity decisions; (ii) we provide a comprehensive set of tools to solve the problem, including both exact methods and heuristics; and (iii) we demonstrate the potential impact of our framework in a school choice setting. Below, we provide a detailed review of these three streams of literature and refer interested readers to (Manlove 2013) for a broader overview.

Mathematical programming formulations. The first mathematical programming formulations of the stable matching problem were studied in (Gusfield and Irving 1989, Vate 1989, Rothblum 1992) and (Roth et al. 1993). Closer to our work, Baïou and Balinski (2000) provided an exponential-size linear programming formulation describing the convex hull of the set of feasible stable matchings and introduced a polynomial-time separation algorithm. Thereafter, this strand of the literature has devised mathematical programming formulations for different extensions of the standard model, including ties in preferences (*i.e.*, when agents are indifferent between two or more options) (Kwanashie and Manlove 2014, Ágoston et al. 2021), regional constraints (Kojima et al. 2018), upper and lower quotas (Ágoston et al. 2016), incomplete lists (Delorme et al. 2019), and dynamic priorities (Rios et al. 2024).

Capacity expansion. Following the preliminary version of our work, several subsequent papers have explored a similar setting. Bobbio et al. (2022) investigated the complexity of the capacity planning problem and its variations, demonstrating that the decision version of the problem is NP-complete and the optimization version inapproximable within an $\mathcal{O}(n^{(\frac{1}{6}-\varepsilon)})$ factor, where n is the number of students. Abe et al. (2022) introduced a heuristic method to solve the capacity planning problem that relies on the Upper Confidence Tree, a Monte Carlo based search method covering the space of capacity expansions. Afacan et al. (2024), in an independent work, also analyzed the problem of allocating additional seats across schools in response to students' preferences. The authors introduced an algorithm that characterizes the set of efficient matchings among those who respect

preferences and priorities and analyzed its incentives' properties. Their work is complementary to ours in several ways. First, they discussed different applications where capacity decisions are made in response to students' preferences. These include some school districts in French-speaking Belgium, where close to 1% of seats are consistently reported after students submit their preferences, and certain college admissions systems, such as in India, where the Ministry of Education plans to increase capacities by up to 50%. Second, their proposed approach can recover any Pareto efficient allocation, including the ones returned by our mechanism. However, they cannot target a specific outcome. Our methodology is flexible enough to enable policymakers to target a particular goal when allocating the extra seats, including access, improvement, or any other objective beyond student optimality. Finally, Afacan et al. (2024) showed that their mechanism is strategy-proof when schools share the same preferences, but it is not in the general case. In this work, we also discuss incentive properties and expand the analysis to study other relevant properties of the assignment mechanism, such as strategy-proofness in the large and monotonicity. Finally, (Kumano et al. 2022) studied and implemented the reallocation of capacities among programs within a restructuring process at the University of Tsukuba. Their capacity allocation constraints could be readily added to our model while maintaining the validity of our methodology.

School choice. Starting with (Abdulkadiroğlu and Sönmez 2003), a large body of literature has studied different elements of the school choice problem, including the use of different mechanisms such as DA (introduced in the seminar work by (Gale and Shapley 1962)), Boston, and Top Trading Cycles (Abdulkadiroğlu et al. 2005, Pathak and Sönmez 2008, Abdulkadiroğlu et al. 2011); the use of different tie-breaking rules (Abdulkadiroğlu et al. 2009, Arnosti 2015, Ashlagi et al. 2019); the handling of multiple and potentially overlapping quotas (Kurata et al. 2017, Sönmez and Yenmez 2019); the addition of affirmative action policies (Ehlers 2010, Hafalir et al. 2013); and the implementation in many school districts and countries (Abdulkadiroğlu et al. 2005, Calsamiglia and Güell 2018, Correa et al. 2022, Allman et al. 2022). Within this literature, the closest papers to ours combine optimizing different objectives with finding a stable assignment. Caro et al. (2004) introduced an integer programming model to make school redistricting decisions. Shi (2016) proposed a convex optimization model to decide the assortment of schools to offer each student to maximize the sum of utilities. Ashlagi and Shi (2016) presented an optimization framework that allowed them to find an assignment pursuing (the combination of) different objectives, such as average and min-max welfare. Bodoh-Creed (2020) presented an optimization model to find the best stable and incentive-compatible match that maximizes any combination of welfare and diversity and prioritizes the allocation of students to their neighborhood schools. Finally, Feigenbaum et al. (2020) introduced a novel mechanism to efficiently reassigned vacant seats after an initial round of a centralized assignment and used data from the NYC high school admissions system to showcase its benefits.

3. Model

We start by introducing the standard stable matching problem and extending it to include capacity decisions in Sections 3.1 and 3.2, respectively. Then, we present two exact mathematical programming formulations that extend those in Baïou and Balinski (2000) to incorporate capacity decisions: (i) a compact formulation (in Section 3.3), and (ii) a comb formulation (in Section 3.4). In the remainder of the paper, we use school choice as motivating example.

3.1. Stable Matching Problem

Let $\mathcal{S} = \{s_1, \dots, s_n\}$ be the set of n students, and let $\mathcal{C} = \{c_1, \dots, c_m\}$ be the set of m schools. To facilitate the exposition, we assume all students belong to the same grade, *e.g.*, Pre-K. On the one side of the market, each school $c \in \mathcal{C} \cup \{\emptyset\}$ ranks the students that applied to it according to a strict order \succ_c . Moreover, we assume that each school $c \in \mathcal{C}$ has a capacity $q_c \in \mathbb{Z}_+$. When clear from the context, we extend the set of schools to include the outside option of being unassigned, which we denote by \emptyset , and we assume that its capacity is sufficiently high (*e.g.*, $q_\emptyset = |\mathcal{S}|$) so every student can potentially be unassigned. Finally, throughout the rest of the paper, we use bold notation to denote vectors/matrices of elements (parameters and variables) and italic notation for one-dimensional ones, *e.g.*, $\mathbf{q} = \{q_c\}_{c \in \mathcal{C} \cup \{\emptyset\}}$.

On the other side of the market, each student $s \in \mathcal{S}$ has a strict preference order \succ_s over the elements in $\mathcal{C} \cup \{\emptyset\}$. Then, we use $c \succ_s c'$ to represent that student s *prefers* school c over school c' and $\emptyset \succ_s c$ to denote that s prefers to be unassigned over attending school c . With a slight abuse of notation, we also refer to \succ_s as student s 's preference list, and we assume that it includes all schools that s *weakly* prefers over being unassigned, including the latter option at the bottom of the list. Furthermore, we use $|\succ_s|$ to represent the number of schools that s prefers over being unassigned and $c' \succeq_s c$ to represent that either $c' \succ_s c$ or that $c' =_s c$ (*i.e.*, s is indifferent between c and c'). Finally, let $r_{s,c}$ be the position of school $c \in \mathcal{C}$ in the preference list of student $s \in \mathcal{S}$ and $r_{s,\emptyset}$ be a parameter that represents a penalty for having student s unassigned.⁵

Let $\mathcal{E} \subseteq \mathcal{S} \times (\mathcal{C} \cup \{\emptyset\})$ be the set of feasible pairs, with $(s, c) \in \mathcal{E}$ meaning that student s includes school c in their preference list. A *matching* is an assignment $\mu \subseteq \mathcal{E}$ such that each student is assigned to one school in $\mathcal{C} \cup \{\emptyset\}$, and each school c receives at most q_c students. We use $\mu(s) \in \mathcal{C} \cup \{\emptyset\}$ to represent the school of student s in the assignment μ , with $\mu(s) = \emptyset$ representing that s is unassigned in μ . Similarly, we use $\mu(c) \subseteq \mathcal{S}$ to represent the set of students assigned to c in μ . We say that a school $c \in \mathcal{C}$ is *fully-subscribed* in matching μ if $|\mu(c)| = q_c$; otherwise, we say that school c is *under-subscribed*. A matching μ is stable if it has no *blocking pairs*, *i.e.*, no pair $(s, c) \in \mathcal{E}$

⁵ For instance, $r_{s,c} = 1$ implies that c is student s 's top preference, *i.e.*, $c \succ_s c'$ for all $c' \in \mathcal{C} \cup \{\emptyset\} \setminus \{c\}$. Also, note that the penalty $r_{s,\emptyset}$ may be different from the position of \emptyset in the preference list of student s .

would prefer to be assigned to each other compared to their current assignment in μ . Formally, we say that (s, c) is a blocking pair if the following two conditions are satisfied: (i) student s prefers school c over $\mu(s) \in \mathcal{C} \cup \{\emptyset\}$, and (ii) $|\mu(c)| < q_c$ or there exists $s' \in \mu(c)$ such that $s \succ_c s'$, i.e., c prefers s over s' .

An instance of the school choice problem can be fully described as $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} \rangle$, where $\succ = \{\succ_s\}_{s \in \mathcal{S}} \cup \{\succ_c\}_{c \in \mathcal{C} \cup \{\emptyset\}}$ and $\mathbf{q} = \{q_c\}_{c \in \mathcal{C} \cup \{\emptyset\}}$. Gale and Shapley (1962) showed that the student-proposing Deferred Acceptance algorithm can find the unique student-optimal stable matching—i.e., the matching that is *weakly* preferred by all students among all stable matchings—in polynomial time for any instance Γ .⁶

3.2. Capacity Expansion

Let $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} \rangle$ be an instance of the school choice problem described in the previous section, and suppose that the clearinghouse aims to allocate a budget $B \in \mathbb{Z}_+$ of additional seats. We define the *capacity expansion problem* as finding a vector of additional seats $\mathbf{t} = \{t_c\}_{c \in \mathcal{C}} \in \mathbb{Z}_+^{\mathcal{C}}$ that complies with the available budget (i.e., $\sum_{c \in \mathcal{C}} t_c \leq B$) and leads to a student-optimal matching in the instance with expanded capacities $\Gamma_{\mathbf{t}} = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} + \mathbf{t} \rangle$.⁷ We formalize this in Problem 1.

PROBLEM 1 (CAPACITY EXPANSION PROBLEM). Given an instance $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} \rangle$ and a budget B , the capacity expansion problem is the following:

$$\min_{\mathbf{t}, \mu} \left\{ \sum_{(s, c) \in \mu} r_{s,c} : \sum_{c \in \mathcal{C}} t_c \leq B, \mu \text{ is a stable matching in instance } \Gamma_{\mathbf{t}}, \mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}} \right\}. \quad (1)$$

Both \mathbf{t} and μ are decision variables in Problem 1. The first constraint guarantees that the former satisfies the budget, while the second constraint ensures that the latter is a stable matching in the expanded instance $\Gamma_{\mathbf{t}}$. The third constraint establishes the integrality of \mathbf{t} . Finally, the objective is to minimize the sum of the preferences of assignment and the penalty of being unassigned. As we show in Lemma 1, this objective function ensures that we obtain a student-optimal matching in the expanded instance. We defer the proof to Appendix A.1.

LEMMA 1. Fix $\mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}}$. Then, minimizing $\sum_{(s, c) \in \mu} r_{s,c}$ over the space of stable matchings μ in $\Gamma_{\mathbf{t}}$ is equivalent to finding the unique student-optimal stable matching in $\Gamma_{\mathbf{t}}$.

Note that an optimal solution of Problem 1 always exists (e.g., $\mathbf{t} = 0$ and μ obtained from using student-proposing DA in the instance $\Gamma_0 = \Gamma$ are a feasible solution) but may not be unique, as we show in Appendix A.2. Moreover, the optimal solution may not use the entire budget. For instance, this would hold if every student is assigned to their top choice in the initial instance Γ .

⁶ DA can be adapted to find the *school-optimal* stable matching, i.e., the unique stable-matching that is weakly preferred by all schools. In Appendix G.1, we formally describe the DA version that finds the student-optimal stable matching.

⁷ Note that Γ_0 corresponds to the original instance Γ with no capacity expansion.

REMARK 1. It is important to highlight that $\succ = \{\succ_s\}_{s \in \mathcal{S}} \cup \{\succ_c\}_{c \in \mathcal{C} \cup \{\emptyset\}}$ is an input of Problem (1); thus, capacity decisions respond to students' preferences and schools' orderings. This assumption matches Chile's current practice and is suitable in settings involving short time horizons, *e.g.*, adding seats in a course/grade/program, merging neighboring schools, or re-organizing classrooms to accommodate different courses based on their popularity. Our model could also accommodate longer-term policies that go beyond the specific preferences of a cohort of students (*e.g.*, building an entirely new school building) if the number of applicants and their preferences are consistent over time. Finally, note that our framework is flexible enough to accommodate other objectives beyond student optimality, such as minimizing implementation (*e.g.*, adding teachers, portable classrooms, etc.) or transportation costs, students' estimated welfare, or any combination of goals.

3.3. Compact Formulation

One natural approach to solve Problem 1 is to generalize the formulation of the stable matching problem studied in (Gusfield and Irving 1989, Vate 1989, Baïou and Balinski 2000), incorporating the decision vector \mathbf{t} . This naïve approach results in the following integer quadratically constrained program:

$$\min_{\mathbf{x}, \mathbf{t}} \sum_{(s,c) \in \mathcal{E}} r_{s,c} \cdot x_{s,c} \quad (2a)$$

$$s.t. \quad (t_c + q_c) \cdot \left(1 - \sum_{c' \succeq_s c} x_{s,c'}\right) \leq \sum_{s' \succ_c s} x_{s',c}, \quad \forall (s,c) \in \mathcal{E}, c \succ_s \emptyset \quad (2b)$$

$$(\mathbf{x}, \mathbf{t}) \in \mathcal{P}_{\mathbb{Z}}, \quad (2c)$$

where the decision variables $x_{s,c}$ indicate whether student s is assigned to school c , and t_c represents the number of additional seats added to school c . The family of constraints in (2b) guarantees that the matching \mathbf{x} is stable in $\Gamma_{\mathbf{t}}$ for any fixed \mathbf{t} . Moreover, $\mathcal{P}_{\mathbb{Z}}$ is the set of integer points of

$$\mathcal{P} = \left\{ (\mathbf{x}, \mathbf{t}) \in [0, 1]^{\mathcal{E}} \times [0, B]^{\mathcal{C}} : \sum_{c:(s,c) \in \mathcal{E}} x_{s,c} = 1 \quad \forall s \in \mathcal{S}, \quad \sum_{s:(s,c) \in \mathcal{E}} x_{s,c} \leq q_c + t_c \quad \forall c \in \mathcal{C}, \quad \sum_{c \in \mathcal{C}} t_c \leq B \right\}, \quad (3)$$

which ensures the feasibility of the assignment under the expanded capacities, *i.e.*, that (i) every student is matched to exactly one school (including \emptyset), (ii) every school receives at most $q_c + t_c$ students, and (iii) the budget is not exceeded.

Formulation (2) may seem attractive due to the polynomial number of variables and constraints. However, directly solving it with an off-the-shelf solver remains challenging because of (i) the non-convexity of the quadratic constraints in (2b) and (ii) the integrality of the decision variables. To address the former, we propose a linear reformulation, obtained by replacing (2b) with

$$t_c + q_c - (B + q_c) \cdot \sum_{c' \succeq_s c} x_{s,c'} \leq \sum_{s' \succ_c s} x_{s',c}, \quad \forall (s,c) \in \mathcal{E}, c \succ_s \emptyset. \quad (2d)$$

As we show in Lemma 2, this reformulation is a valid formulation of Problem 1. In Appendix B, we provide the proof and discuss other more natural linear reformulations obtained via McCormick envelopes. However, preliminary computational results show that the family of constraints in (2d)—henceforth referred to as L-constraints—performs better than those alternatives, so we focus on the implementation of the former in the remainder of the paper.

LEMMA 2. *The compact formulation with L-constraints, obtained by replacing (2b) with (2d) in Formulation (2), is a valid formulation of Problem (1).*

To deal with the challenges imposed by the integrality of the decision variables, one natural approach is to allow variables $\mathbf{x} \in [0, 1]^{S \times C}$ in (2). However, as observed by Baïou and Balinski (2000) in the base case (*i.e.*, with $\mathbf{t} = 0$), the resulting formulation may not be valid, *i.e.*, the optimal solution may be fractional. In the next section, we focus on devising a valid formulation for Problem 1 that relaxes the integrality of the matching variables \mathbf{x} , while keeping that of capacity decisions (*i.e.*, \mathbf{t}).

3.4. Comb Formulation

To overcome the challenges involved with the integrality of \mathbf{x} , Baïou and Balinski (2000) introduced a valid linear programming formulation that relies on an exponential number of inequalities, called *comb constraints*, and also provided a separation algorithm that allows to efficiently solve the base problem (with $\mathbf{t} = 0$) using a cutting plane algorithm. Since these are the building blocks for our exact proposed methods, we provide more background on the comb formulation in Section 3.4.1. Then, in Section 3.4.2, we generalize it to incorporate capacity decisions. Finally, in Section 3.4.3, we discuss our cutting plane method to solve Problem 1.

3.4.1. Background. Baïou and Balinski (2000) analyze the stable matching problem using a graph representation of Γ . In their *admissions graph*, each node corresponds to a pair $(s, c) \in \mathcal{E}$, and there are two types of (directed) arcs: (i) horizontal, which go from a node (s, c) to a node (s', c) whenever $s' \succ_c s$; and (ii) vertical, which go from a node (s, c) to a node (s, c') whenever $c' \succ_s c$.⁸

Let \mathcal{E}_c^+ be the set of pairs $(s, c) \in \mathcal{E}$ such that c prefers at least $q_c - 1$ students over s for $c \in \mathcal{C}$. For $(s, c) \in \mathcal{E}_c^+$, a *shaft* with base (s, c) is denoted by $S(s, c)$ and consists of (s, c) and all pairs (s', c) such that $s' \succ_c s$. A *tooth* $T(s, c)$ with base $(s, c) \in \mathcal{E}$ consists of (s, c) and all pairs (s, c') such that $c' \succ_s c$. Based on these elements, we can now define a comb.

⁸ Arcs that are implied by transitivity are omitted. For example, if $c'' \succ_s c' \succ c$, then the admissions graph only includes the arcs $((s, c), (s, c'))$ and $((s, c'), (s, c''))$.

DEFINITION 1 (COMB). For $(s, c) \in \mathcal{E}_c^+$, a comb with base (s, c) , denoted by $C(s, c)$, consists of the union of $S(s, c)$ and exactly q_c teeth $T(s', c)$ with $(s', c) \in S(s, c)$, including $T(s, c)$. Furthermore, let \mathbf{C}_c be the family of combs for school $c \in \mathcal{C}$.

Note that combs can only be defined for schools that are over-subscribed, *i.e.*, that have at least as many applicants as seats available. Additionally, there may be multiple combs for each base (s, c) since the shaft can have more than q_c elements and, therefore, we can select different subsets of $q_c - 1$ teeth.

EXAMPLE 1. In Figure 1a, we present a subset of the admissions graph of a given instance that includes all students who applied to c , where $q_c = 3$. Then, the set of feasible pairs that can be the base of a comb is $\mathcal{E}_c^+ = \{(s'_3, c), (s'_4, c), (s, c), (s'_5, c)\}$. If we pick (s, c) as the base of the comb, then we have shaft $S(s, c) = \{(s, c), (s'_1, c), (s'_2, c), (s'_3, c), (s'_4, c)\}$. Finally, each comb based on (s, c) must include q_c teeth based in the shaft, one of which must be $T(s, c)$. In this case, there are $\binom{4}{2}$ possible teeth choices (and so is the number of combs based on (s, c)); Figure 1a shows in blue the comb that results from selecting $T(s'_2, c)$ and $T(s'_3, c)$ as the two additional teeth.

In Figure 1b, we clarify the inspiration for the naming of the L-constraints. The terms on the left-hand side of (2d) are highlighted in yellow, while the terms on the right-hand side are in red.

□

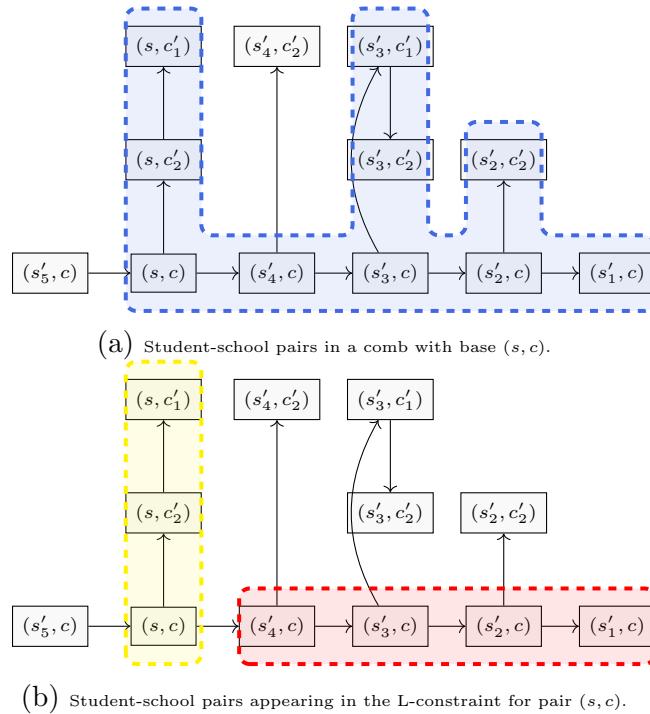


Figure 1 Example of a subset of the admissions graph when school c has capacity 3.

Baiou and Balinski (2000) showed that, if $t = 0$, the family of constraints in (2b) can be replaced by

$$\sum_{(s',c') \in C} x_{(s,c)} \geq q_c, \quad \forall c \in \mathcal{C}, \quad C \in \mathbf{C}_c. \quad (4)$$

To see why these two families of constraints are equivalent, let $(s, c) \in \mathcal{E}$ be such that $x_{s,c} = 0$. The stability constraints in (2b) guarantee that either (i) s is assigned to something more preferred, *i.e.*, $\sum_{c' \succ_s c} x_{s,c'} = 1$, or (ii) there are at least q_c students with higher priority than s currently assigned to c , *i.e.*, $\sum_{s' \succ_c s} x_{s',c} \geq q_c$. If $\sum_{(s',c) \in S(s,c)} x_{s',c} \geq q_c$, then it is direct that (ii) holds by definition of the shaft. Conversely, if $\sum_{(s',c) \in S(s,c)} x_{s',c} < q_c$, (4) and the definition of the combs imply that each pair $(s', c) \in S(s, c)$ for which $x_{s',c} = 0$ is such that $\sum_{c' \succ_{s'} c} x_{s',c'} = 1$ and, thus, (i) holds for (s, c) .

The key result in Baiou and Balinski (2000) is that the polytope $\{\mathbf{x} : (\mathbf{x}, \mathbf{0}) \in \mathcal{P}, \mathbf{x} \text{ satisfies (4)}\}$ that allows for fractional solutions coincides with the convex hull of all stable matchings and, thus, the polytope is integral. Moreover, the authors showed that the constraints in (4) can be separated in polynomial time, which aligns with the result on the equivalence between optimization and separation (Grötschel et al. 1981).

3.4.2. Generalized Comb Formulation. For $k \in \mathbb{Z}_+$ and $c \in \mathcal{C}$, let $\mathcal{E}_c^+(k)$ be the set of pairs $(s, c) \in \mathcal{E}$ such that c prefers at least $q_c + k - 1$ students over s . In Definition 2, we extend Definition 1 to allow for extra capacities.

DEFINITION 2 (GENERALIZED COMBS). For $(s, c) \in \mathcal{E}_c^+(k)$ and $k \in \mathbb{Z}_+$, a *generalized comb* with base (s, c) , denoted by $C_k(s, c)$, consists of the union of the shaft $S(s, c)$ and exactly $q_c + k$ teeth $T(s', c)$ with $(s', c) \in S(s, c)$, including $T(s, c)$. Furthermore, let $\mathbf{C}_c(k)$ be the family of combs for school $c \in \mathcal{C}$ with k extra seats.

Given Definition 2, note that one could naïvely attempt to design a formulation with generalized comb constraints in the form of (4) fixing the right-hand side to $q_c + k$ for each school $c \in \mathcal{C}$, $k \in \mathbb{Z}_+$ and combs in $\mathbf{C}_c(k)$. However, this may lead to infeasible or suboptimal solutions because the student-optimal stable matchings of two expanded instances Γ_t and $\Gamma_{t'}$ for $t \neq t'$ may not coincide. We address this by considering the capacity expansion variables in the right-hand side of the constraints (instead of fixing their value) and appropriately connecting them with the families of generalized combs constraints $\mathbf{C}_c(\cdot)$. This approach aligns with our goal of jointly deciding the extra capacity and the assignment of students to schools. Thus, we model the capacity expansion problem using the following mixed-integer programming formulation:

$$\min_{\mathbf{x}, \mathbf{y}} \quad \sum_{(s,c) \in \mathcal{E}} r_{s,c} \cdot x_{s,c} \quad (5a)$$

$$s.t. \quad \sum_{(s',c') \in C} x_{s',c'} \geq q_c + \sum_{j=1}^k y_c^j, \quad \forall c \in \mathcal{C}, \forall k = 0, \dots, B, \forall C \in \mathbf{C}_c(k), \quad (5b)$$

$$(\mathbf{x}, \mathbf{y}) \in \mathcal{P}^{\text{ext}}, \quad (5c)$$

where \mathcal{P}^{ext} is the set

$$\begin{aligned} \mathcal{P}^{\text{ext}} = \left\{ (\mathbf{x}, \mathbf{y}) \in [0, 1]^{\mathcal{E}} \times \{0, 1\}^{\mathcal{C} \times \{1, \dots, B\}} : \right. & \sum_{c:(s,c) \in \mathcal{E}} x_{s,c} = 1, \quad \forall s \in \mathcal{S}, \\ & \sum_{s:(s,c) \in \mathcal{E}} x_{s,c} \leq q_c + \sum_{k=1}^B y_c^k, \quad \forall c \in \mathcal{C}, \\ & \sum_{c \in \mathcal{C}} \sum_{k=1}^B y_c^k \leq B, \\ & \left. y_c^k \geq y_c^{k+1}, \quad \forall c \in \mathcal{C}, k = 1, \dots, B-1 \right\}. \end{aligned} \quad (6)$$

The decision variable y_c^k indicates whether school c receives at least k extra seats. As a result, the vector \mathbf{y} corresponds to the pseudo-polynomial description (or unary expansion) of an allocation vector $\mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}}$ since $t_c = \sum_{k=1}^B y_c^k$ and, consequently, there is a one-to-one correspondence between the elements of \mathcal{P}^{ext} and \mathcal{P} for integral points \mathbf{t} . In line with (3), the first three families of constraints in (6) ensure that each student is assigned to one school (including \emptyset), each school receives at most their expanded capacity, and the budget is not exceeded, respectively. The last family of constraints ensures the consistency of \mathbf{y} . In Theorem 1, we show the correctness of our generalized comb formulation; we defer the proof to Appendix C.2.

THEOREM 1. *The generalized comb formulation, Formulation (5), is a valid formulation of Problem (1).*

REMARK 2. For each school $c \in \mathcal{C}$, only the generalized comb constraints for $k^* = \max \{k \in \mathbb{Z}_+ : y_c^{k*} = 1\}$ are required to obtain a stable solution; see Lemmas 3 and 4 in Appendix C.1. For instance, if school $c \in \mathcal{C}$ has capacity $q_c + k^*$ in the optimal solution, we only need the constraints for the combs in $\mathbf{C}_c(k^*)$. Note that this reduces considerably the number of comb constraints needed to ensure stability, although it requires knowledge about the optimal expansion \mathbf{y}^* .

3.4.3. Cutting-plane Algorithm. Formulation (5) has exponentially many generalized comb constraints, so it cannot be solved directly with an off-the-shelf solver.⁹ For this reason, in Algorithm 1, we present a cutting plane approach that relies on a separation algorithm to iteratively identify the most violated constraints (if any).¹⁰ Specifically, we start by solving a relaxation of (5)

⁹ Same holds for the comb formulation in Baiou and Balinski (2000).

¹⁰ Note that in Formulation (5), the decision vector \mathbf{y} is binary. Therefore, our separation method works for \mathbf{x} fractional and \mathbf{y} binary. Thus, the fact that the separation runs in polynomial time does not guarantee that the cutting-plane method runs in polynomial time.

involving only a subset \mathcal{J} of the comb constraints (in Step 2); note that \mathcal{J} may be empty. Then, for each school $c \in \mathcal{C}$, we run our separation algorithm that identifies the most violated constraint (if any) and adds it to the set of violated constraints \mathcal{J}' (Steps 5 to 7). If the solution \mathbf{x}^* is not stable, then $\mathcal{J}' \neq \emptyset$ and, thus, we add these constraints to the main program (Step 9) and resolve it, repeating the process. Conversely, if \mathbf{x}^* is stable and no additional comb constraints are added to the main problem (as $\mathcal{J}' = \emptyset$), the solution is stable and optimal (due to Theorem 1). Note that Algorithm 1 terminates in a finite number of steps since the set of comb constraints is finite.

Algorithm 1 Cutting-plane method

Input: An instance $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} \rangle$ and a budget B .

Output: The student-optimal stable matching \mathbf{x}^* and the optimal vector of additional seats \mathbf{t}^* .

- 1: $\mathcal{J} \leftarrow$ subset of initial comb constraints (5b)
 - 2: $(\mathbf{x}^*, \mathbf{y}^*) \leftarrow \operatorname{argmin} \left\{ \sum_{(s,c) \in \mathcal{E}} r_{s,c} \cdot x_{s,c} : (\mathbf{x}, \mathbf{y}) \in \mathcal{P}^{\text{ext}} \cap \mathcal{J} \right\}$
 - 3: $\mathcal{J}' \leftarrow \emptyset$
 - 4: **for** $c \in \mathcal{C}$ **do**
 - 5: $C^* \in \operatorname{argmin} \left\{ \sum_{(s',c') \in C} x_{s',c'}^* : C \in \mathbf{C}_c(t_c^*) \text{ where } t_c^* = \sum_{j=1}^B y_c^{j*} \right\}$
 - 6: **if** $\sum_{(s',c') \in C^*} x_{s',c'}^* < q_c + t_c^*$ **then**
 - 7: $\mathcal{J}' \leftarrow \mathcal{J}' \cup \{C^*\}$
 - 8: **if** $\mathcal{J}' \neq \emptyset$ **then**
 - 9: $\mathcal{J} \leftarrow \mathcal{J} \cup \mathcal{J}'$ and go to Step 2.
 - 10: **else**
 - 11: Terminate and return \mathbf{x}^* and \mathbf{t}^* where $t_c^* = \sum_{j=1}^B y_c^{j*}$ for every $c \in \mathcal{C}$.
-

We can exploit some structural properties of the objective function and the incumbent solution (discussed in Appendix C.3) to speed up the search for violated comb constraints. Specifically, for a given incumbent solution $(\mathbf{x}^*, \mathbf{y}^*)$, we can guide our search using two matchings: (i) the (potentially fractional and non-stable) matching given by \mathbf{x}^* , and (ii) the feasible (but potentially suboptimal) stable matching μ^* obtained using the DA algorithm over the instance $\Gamma_{\mathbf{t}^*}$, where $t_c^* = \sum_{j=1}^B y_c^{j*}$ for all $c \in \mathcal{C}$. As we show in Appendix C.3, we can restrict the search for violated comb constraints to schools $c \in \mathcal{C}$ such that $\sum_{s:(s,c) \in \mathcal{E}} x_{s,c}^* = |\mu^*(c)|$ and $\{s : x_{s,c}^* > 0\} \setminus \mu^*(c) \neq \emptyset$. The first condition guarantees that the school is fully subscribed, while the second ensures that there are students matched (at least fractionally) to school c but for whom their assignment is not stable. Considering schools that satisfy these two conditions allows us to focus on schools that certainly have a blocking pair, effectively reducing the search space and resulting in a efficient method. We formalize our method in Algorithm 4 in Appendix C.3.

REMARK 3. As we will show in Section 5.3.4, we can speed up computation by combining the compact and comb formulations introduced in Sections 3.3 and 3.4, respectively. Specifically, we propose a *hybrid* approach with input parameter γ that is exactly Algorithm 1, except that the main program of Step 2 has \mathcal{J} replaced by L-constraints for all $(s, c) \in \mathcal{E}$ such that school c ranks student s as $q_c + \lfloor \gamma \cdot B + 0.5 \rfloor$ or higher. Then, it separates infeasible solutions through the separation algorithm (Algorithm 4), *i.e.*, it cuts unstable matchings by adding comb constraints. Appendix D.1 thoroughly discusses this approach.

4. Heuristics

Our toolkit for solving Problem 1 includes not only exact formulations but also efficient heuristics that provide near-optimal solutions. In this section, we introduce two natural methods: (i) a greedy approach (Greedy), and (ii) a linear programming-based heuristic (LPH). Both heuristics rely on the computation of a student-optimal stable matching, which can be achieved in polynomial time using the DA algorithm. A description of the DA algorithm can be found in Appendix G.1. With a slight abuse of notation, we use μ_t and $f(t)$ to denote the student-optimal matching and the corresponding value of the objective function that result from running DA in the instance Γ_t , *i.e.*,

$$f(t) := \min_{\mu \subseteq \mathcal{E}} \left\{ \sum_{(s,c) \in \mu} r_{s,c} : \mu \text{ is a stable matching in } \Gamma_t \right\}. \quad (7)$$

Greedy Approach. Under this approach, we exploit the fact that the objective function is component-wise decreasing, *i.e.*, for $t \leq t'$ component-wise, we have $f(t') \leq f(t)$. Greedy iteratively assigns an extra seat to the school with the best marginal improvement. Specifically, Greedy performs at most B sequential iterations—until exhausting the budget, each involving the allocation of one extra seat through two steps: (i) evaluating the objective function for each possible allocation of one extra seat using DA, and (ii) allocating the extra seat to the school leading to the lowest objective value. This is equivalent to look for the highest marginal $f(t) - f(t + \mathbf{1}_c)$ where $\mathbf{1}_c \in \{0, 1\}^C$ denotes the indicator vector whose value is 1 in component $c \in \mathcal{C}$ and 0 otherwise. We formalize this heuristic in Algorithm 2.

LP-based Heuristic. Under this approach, we exploit the fact that the optimal solution of $\min \left\{ \sum_{(s,c) \in \mathcal{E}} r_{s,c} : (\mathbf{x}, \mathbf{t}) \in \mathcal{P} \right\}$, when projected in \mathbf{t} , is integral since \mathcal{P} has integral vertices in terms of \mathbf{t} .¹¹ Hence, LPH starts by solving this linear program. As a result, we obtain an allocation of extra seats \mathbf{t} and an assignment \mathbf{x} that may be fractional and non-stable. Nevertheless, since \mathbf{t} is integral, LPH obtains a student-optimal stable matching using the DA algorithm in the instance with expanded capacities Γ_t . We formalize this heuristic in Algorithm 3. Note that the solution obtained by LPH is equivalent to the solution generated from solving the problem in Step 2 in Algorithm 1 considering $\mathcal{J} = \emptyset$.

¹¹ This can be concluded by noting that \mathcal{P} can be reformulated with flow conservation constraints.

Algorithm 2 Greedy

Input: An instance $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} \rangle$ and a budget B .

Output: A feasible allocation \mathbf{t} and a stable matching $\mu_{\mathbf{t}}$ in the expanded instance $\Gamma_{\mathbf{t}}$.

- 1: Initialize $\mathbf{t} \leftarrow \mathbf{0}$
 - 2: **while** $\sum_{c \in \mathcal{C}} t_c < B$ **do**
 - 3: $c^* \in \operatorname{argmin} \{f(\mathbf{t} + \mathbf{1}_c) : c \in \mathcal{C}\}$, where f is defined as in (7).
 - 4: $\mathbf{t} \leftarrow \mathbf{t} + \mathbf{1}_{c^*}$
 - 5: Compute $\mu_{\mathbf{t}}$ using the DA algorithm over $\Gamma_{\mathbf{t}}$.
-

Algorithm 3 LPH

Input: An instance $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} \rangle$ and a budget B .

Output: A feasible allocation \mathbf{t} and a stable matching $\mu_{\mathbf{t}}$ in the expanded instance $\Gamma_{\mathbf{t}}$.

- 1: Obtain $(\mathbf{x}, \mathbf{t}) \in \operatorname{argmin} \left\{ \sum_{(s,c) \in \mathcal{E}} r_{s,c} \cdot x'_{s,c} : (\mathbf{x}', \mathbf{t}') \in \mathcal{P} \right\}$.
 - 2: Compute the student-optimal matching $\mu_{\mathbf{t}}$ in $\Gamma_{\mathbf{t}}$ using the DA algorithm.
-

5. Application to School Choice in Chile

As discussed in Section 1, the third goal of our work is to illustrate the potential benefits of our proposed methodology and devise some insights that could help guide a solution to the crisis currently affecting the Chilean school choice system. In Section 5.1, we provide additional background on the Chilean school choice system and discuss why (i) it is a good application for our framework and (ii) a general enough setting to showcase the applicability of our approach. Then, in Section 5.2, we discuss the data and simulation setup in detail. Finally, in Section 5.3, we present our simulation results and discuss some key insights.

5.1. Background

Starting with the 2015 School Inclusion Law, every school that receives public funding (*e.g.*, public and voucher schools) must use the centralized assignment system at the core of the “Sistema de Admisión Escolar” (SAE). After a scaling process, the new system reached its full implementation in 2020 and currently serves every year more than half a million students and eight thousand schools nationwide across all levels, from Pre-K to 12th grade. We refer to (Correa et al. 2022) for a detailed description of the Chilean school choice system and the algorithm used to perform the allocation.

The Chilean school choice system is a good application for our methodology for multiple reasons. First, the system’s goal is to find a student-optimal stable assignment and, thus, uses a variant of the student-proposing Deferred Acceptance algorithm. Second, we can easily adapt our framework to incorporate all the features of the Chilean system, including the block application, the dynamic

Table 2 Summary Statistics: Entry Levels in Antofagasta region, Admissions Process 2023-2024

Level	Students	Schools	Seats	Seats per student	Assignment	
					Total	Rate
Pre-K	3795	71	3558	0.938	2412	0.636
K	1443	85	1472	1.020	734	0.509
1st	3818	121	4829	1.265	2924	0.766
7th	1454	111	1938	1.333	1009	0.694
9th	5440	67	4961	0.912	3788	0.696

siblings' priority, etc. Third, the Ministry of Education manages all schools participating in the system and, thus, can ask them to modify their vacancies within a reasonable range of variation. Finally, as discussed earlier, the system is under scrutiny due to the insufficient seats offered in many regions of the country. As a result, MINEDUC introduced a plan in 2022 aiming to increase school capacities and, in most cases, these capacity decisions are made after students submit their preferences and before performing the allocation of the main round, which is the exact timing assumed in our framework. Given the direct applicability of our work, we contacted MINEDUC and the NGO in charge of the implementation of the system, and we have been collaborating since then to devise solutions to mitigate the current crisis.

5.2. Data

We consider data from the 2023-2024 admissions process and, specifically, from students applying to Pre-K in the Antofagasta region during the main round.¹² As discussed in Section 1, we focus on the main round of the admissions process because it involves the highest share of students and seats—only 15% of students in our sample participated in the complementary round, and the number of seats offered was close to 16% of those available in the main round. Also, we restrict ourselves to Antofagasta because it is the region with the highest need of additional seats. We focus on a single level to speed up the computation, and we choose Pre-K because it is one of the largest entry levels with the lowest number of seats per applicant and assignment rate, as shown in Table 2.

For all simulations conducted, we use the actual preferences reported by students during the main round. Additionally, our simulations consider the seat availability reported by each school during the main round, the prioritization established by the priority groups, and the actual lotteries drawn. To simplify the exposition, we omit the siblings' priority scheme used in the actual admission system (for more details, see (Correa et al. 2022, Rios et al. 2024, Bobbio 2024)), and we also do not consider the quotas for students with special needs and those for disadvantaged and high-achieving students. Finally, we perform all our simulations for two possible values of the penalty for

¹² All the data is publicly available and can be downloaded from this datosabiertos.mineduc.cl.

unassigned students, namely, $r_{s,\emptyset} \in \{|\succ_s| + 1, |\mathcal{C}| + 1\}$ for all $s \in \mathcal{S}$. As will be detailed in Section 6, these penalty values allow us to measure the trade-off between *access*, *i.e.*, prioritizing unassigned students by using a hefty penalty ($r_{s,\emptyset} = |\mathcal{C}| + 1$), and *improvement*, *i.e.*, using each additional seat to reach the longest chain of improvements and benefit the largest share of students by using a minor penalty ($r_{s,\emptyset} = |\succ_s| + 1$). Unless otherwise stated, all these simulations were solved considering a MipGap tolerance of $1e^{-4}$ and using the hybrid approach discussed at the end of Section 3.4 with a time limit of ten hours.

5.3. Results

In Section 5.3.1, we study the impact on students' outcomes when varying the budget of the extra capacities, including access and improvement with respect to their initial assignment. Then, in Section 5.3.2, we analyze the effect of imposing bounds on the number of additional seats added to each school. In Section 5.3.3, we provide a sensitivity analysis of the results to changes in students' preferences. Finally, in Section 5.3.4, we compare the computing time of the proposed hybrid approach with other relevant benchmarks, and we analyze these results with respect to those obtained from the two heuristics discussed in Section 4.

5.3.1. Effect of Budget. In Figure 2, we report simulation results showing the impact of the budget ($B \in \{1, 2, 5, 10, 15, 20, 25, 30, 50, 100, 150, 200\}$) on the number of students who (i) *enter* the assignment, *i.e.*, students who are initially unassigned (with $B = 0$) and get assigned to one of their preferences under the expanded capacities; and who (ii) *improve* their assignment, *i.e.*, students who are initially assigned and get assigned to a more preferred school (according to their reported preferences) under the expanded capacities. We also include in the analysis the *overall* number of students who benefit from the extra capacity relative to the baseline (with $B = 0$), which is equal to the sum of the number of students who *enter* and *improve*. In each case, we report the total number of students who benefit (solid lines), and we also include the number of students who benefit per additional seat (dashed lines). Note that *enter* captures the reduction in unassigned students.

First, we observe that the total number of students who benefit (in green) is greater than the number of additional seats and, thus, the overall number of students who benefit per additional seat is above one. However, we also observe that this rate decreases as the budget increases. For instance, when the penalty is low (Figure 2a), five students benefit per additional seat when the budget is low, while this decreases to approximately two when the budget is high. This result is intuitive, as the chains of improvement targeted with a low budget will likely be triggered when the budget is increased and, thus, the additional chains reach fewer students.

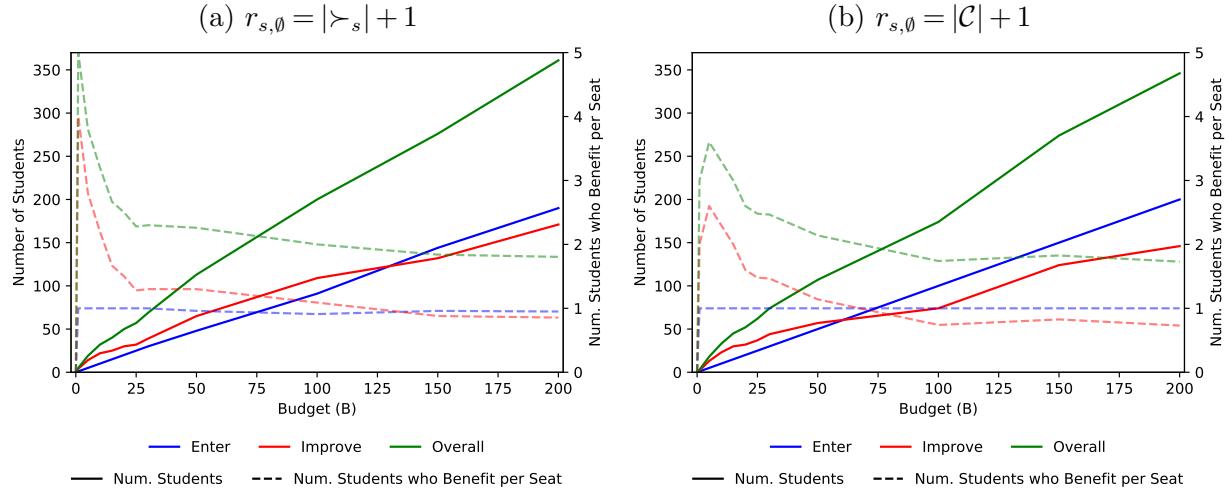


Figure 2 Effect of the Budget on Students' Outcome

Second, comparing Figures 2a and 2b, we observe that the total number of students who benefit per additional seat is greater under a low penalty if the budget is low. As we formally show in Section 6, we know that each chain of improvement ends either in a under-demanded school (after benefiting some students) or in an unassigned student, with the latter chains being the primary target when the penalty is high. Then, since the former are more likely to cover a larger set of students, it is expected that a low penalty results in a greater number of students who benefit per seat. Nevertheless, we observe that the difference becomes negligible as we increase the budget, with both penalty values leading to an improvement of roughly two students per additional seat.

Finally, in line with the previous result, we observe that the number of students who enter the assignment per additional seat (blue dashed line) is equal to one when the penalty is high (Figure 2b), while it is slightly below one when the penalty is low. Since the budget provides an upper bound on the maximum number of students who can enter the assignment, this result suggests that considering a high penalty maximizes entry, consistent with our theoretical results in Section 6.

Overall, the results in this section suggest that each additional seat can improve the allocation of multiple students and that a high penalty for unassigned students can minimize the number of unassigned students. However, these results also show that ensuring a minimum of one seat per student in the system (which could be achieved with a budget $B = 200$) does not guarantee that every student will get assigned, as there were initially 1,383 ($=3,795 - 2,412$ from Table 2) unassigned students. The most likely explanation lies around the high correlation between students' preferences and the small number of schools that students report in their preference list. Hence, MINEDUC should combine capacity planning with other information policies to expand students' consideration sets, encouraging them to apply to a more diverse set of schools.

5.3.2. Effect of Bounds on Expansion. A valid concern raised during our discussions with MINEDUC and the NGO in charge of the allocation is that our approach would assign most extra seats to a few over-demanded schools if preferences are highly correlated. In fact, from Figure 3, we observe that the maximum expansion relative to the original capacity (in red) is quite high for certain budgets (*e.g.*, doubling the capacity of a school when the penalty is low, as shown in Figure 3a for $B = 150$), while the fraction of schools that receive at least one additional seat (in blue) is relatively small (less than 20%, as shown in Figure 3a for $B = 150$). The results for a high penalty (in Figure 3b) are along the same lines, although the maximum relative expansion is more attenuated; this may occur because the optimal solution aims to maximize the number of students who enter by adding at least one seat to more schools (roughly 32% when $B = 150$ in blue). These results confirm MINEDUC’s concern. Additionally, MINEDUC has highlighted that some schools have limited flexibility to accommodate more students, such as those already operating in two shifts—serving different groups of students in the morning and afternoon—or those with space constraints that prevent the installation of modular classrooms.

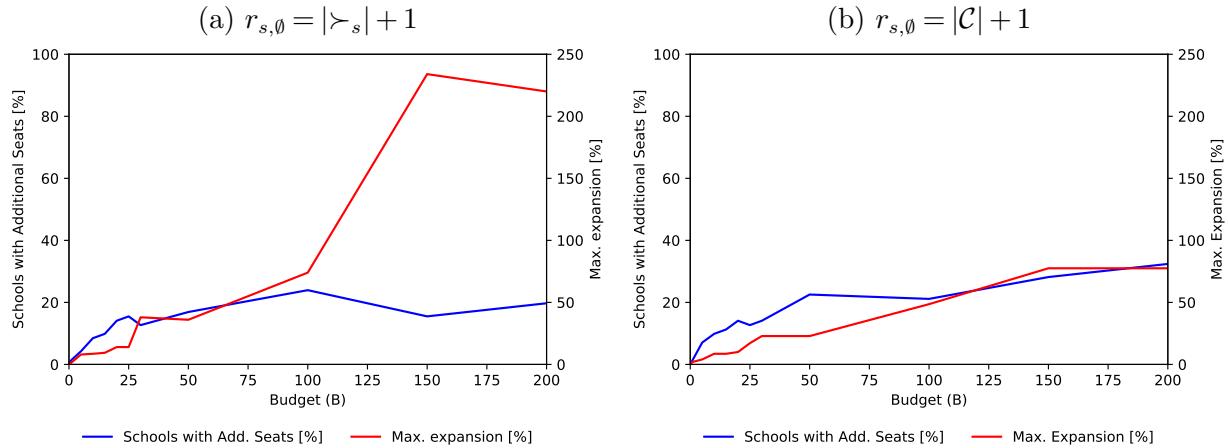


Figure 3 Effect for Schools

To overcome this practical limitation, we can easily adapt our framework by incorporating the new set of constraints $t_c \leq b_c$, for all $c \in \mathcal{C}$, where b_c is the maximum number of additional seats that can be allocated to school c .¹³ Let $\bar{\mathbf{t}}$ and \mathbf{t}^* be the optimal allocations of extra seats with and without these bounds, respectively. In Figure 4, we compare the optimal (unconstrained) solution with that obtained for different bounds ($b_c \in \{1, 2, 5, 10\}$ for all $c \in \mathcal{C}$) focusing on (i) the gap in the objective function (Figure 4a), *i.e.*, $(f(\bar{\mathbf{t}}) - f(\mathbf{t}^*))/f(\bar{\mathbf{t}})$; and (ii) the gap in the number of students

¹³ In Appendix F, we discuss some model extensions to incorporate other relevant aspects from a practical standpoint.

that enter the assignment (Figure 4b), *i.e.*, $(|\mu_{t^*}| - |\mu_{\bar{t}}|) / |\mu_{t^*}|$, where $f(\mathbf{t})$ and μ_t are as defined in Section 4. We include the gap in entry because MINEDUC’s primary concern is reducing the number of unassigned students; thus, maximizing entry is one of the most relevant outcomes. In both cases, the solid (resp. dashed) lines report the results obtained considering a low (resp. high) penalty for unassigned students, *i.e.*, $r_{s,\emptyset} = |\succ_s| + 1$ (resp. $|\mathcal{C}| + 1$) for all $s \in \mathcal{S}$.

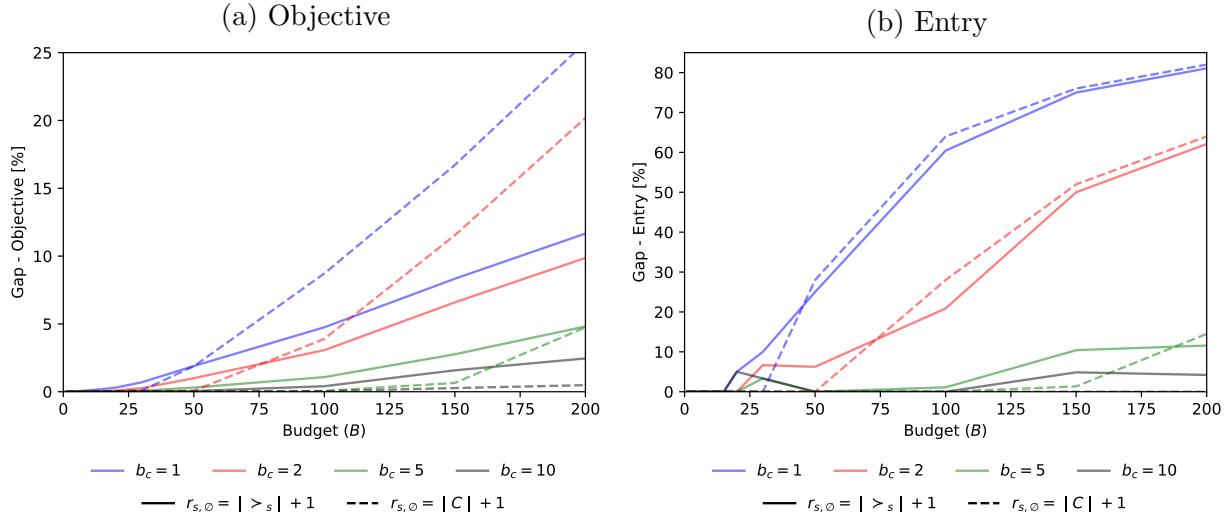


Figure 4 Effect of Bounds on Expansion

On the one hand, from Figure 4a, we observe that the gap in the objective function increases as we increase the budget. However, we also observe that this gap is relatively small when the upper bound is sufficiently high ($b_c \in \{5, 10\}$) and when we prioritize *access* over *improvement* (*i.e.*, $r_{s,\emptyset} = |\mathcal{C}| + 1$). The small gap in the latter case suggests that the problem has many near-optimal solutions and, thus, we can select one that does not over-expand some schools. On the other hand, from Figure 4b, we observe that imposing tight limits (*i.e.*, $b_c \in \{1, 2\}$) can significantly reduce the number of students who enter the assignment, especially when the budget is sizable. Nevertheless, we also observe that considering $b_c = 10$ leads to no reduction in entry when coupled with a high penalty, while the gap in entry is minor (less than 5%) with a low penalty. These results suggest that limiting expansions to ten seats per school results in a high-quality solution without overcrowding schools.

5.3.3. Effect of Preferences. Our methodology relies on students’ reported preferences to optimally decide the allocation of extra seats to maximize the desired objective (which also depends on the penalty values $r_{s,\emptyset}$ ’s). The timing of capacity decisions matches the current practice in Chile, as MINEDUC has previously expanded capacities (i) after students apply to the main round

but before performing the assignment or (ii) after the main round but before students apply to the complementary round. However, in other school districts, policy-makers may want to make capacity decisions before students submit their preferences.

If preferences across years are highly correlated and the number of applicants per year is stable (as it is the case in Chile), one approach would be to use the applicants and their realized preferences from the previous admissions process, and combine this information with the current set of schools and their capacities to decide the allocation of additional seats. In Figure 5, we report the results evaluating this alternative approach. Specifically, for each budget B , we proceed in three steps that lead us to two solutions, $(\tilde{\mathbf{x}}, \tilde{\mathbf{t}})$ and $(\mathbf{x}^*, \mathbf{t}^*)$. First, we obtain the optimal allocation of extra seats $\tilde{\mathbf{t}}$ considering (i) the students, their preferences, and the schools' orderings of the 2022-2023 admission process, and (ii) the school capacities of the 2023-2024 admission process. Second, we compute the student-optimal stable assignment $\tilde{\mathbf{x}}$ considering the instance of the 2023-2024 admission process (with its corresponding sets of students, schools, preferences and orderings) but assuming that capacities are $q_c + \tilde{t}_c$ for each school $c \in \mathcal{C}$. Finally, we directly obtain the optimal allocation of extra seats and assignment $(\mathbf{x}^*, \mathbf{t}^*)$ with the instance of 2023-2024. By comparing the outcomes obtained from $(\tilde{\mathbf{x}}, \tilde{\mathbf{t}})$ and $(\mathbf{x}^*, \mathbf{t}^*)$, we can assess the value of knowing students' actual preferences when making capacity decisions.

In Figure 5, we report the results comparing (i) the gap in the objective function (Figure 5a) and (ii) the gap in entry (Figure 5b) achieved with $(\tilde{\mathbf{x}}, \tilde{\mathbf{t}})$ relative to $(\mathbf{x}^*, \mathbf{t}^*)$. On the one hand, from Figure 5a, we observe that using students' preferences from the previous admissions process to decide the allocation of extra seats leads to a gap in the objective that is (i) increasing in the budget, (ii) higher for high values of the penalty, and (iii) lower than 10% for all budgets and penalties. These results suggest that the distribution of preference of assignment (which mainly affects the objective) is not significantly affected by whether we use the current or previous preferences to decide the allocation of extra seats.

On the other hand, from Figure 5b, we observe that the gap in entry is small when the penalty is low, taking either positive or negative values depending on the budget. In contrast, when the penalty is high, we observe that the gap in entry is always positive and can be relatively high (up to 30%) for some budget values. One possible explanation is that when the penalty is high, the optimal solution is tailored to ensure that students with particular circumstances (*e.g.*, applying to rural schools, with short preference lists, etc.) get assigned, so any differences in these specific cases (from one year to the other) largely affect the effectiveness in entry. This is not the case when the penalty is low since the distribution of preferences, which is more stable across years compared to the particular cases mentioned above, is the primary driver of the allocation of additional seats.

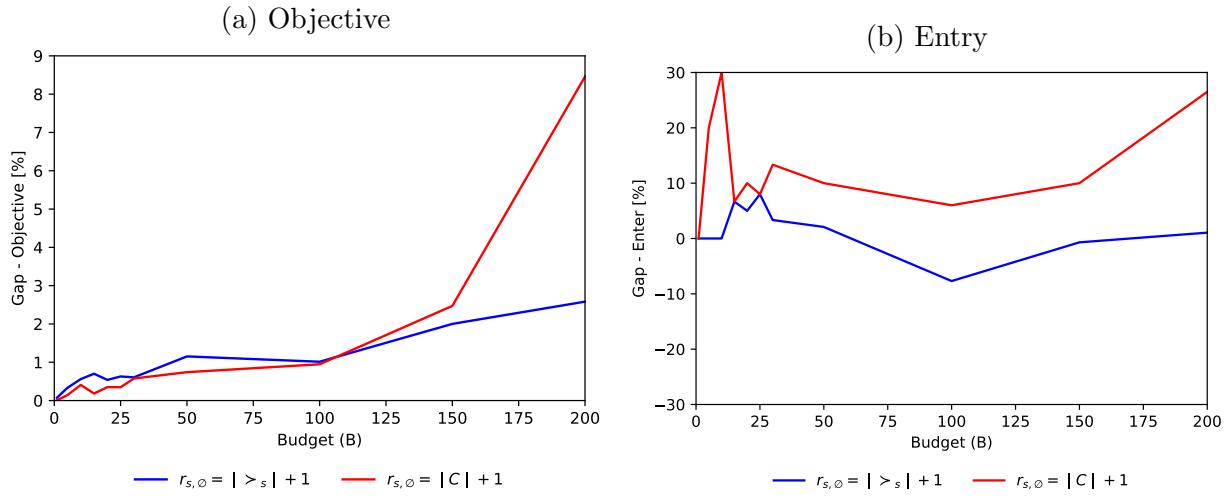


Figure 5 Effect of Using Previous Preferences

5.3.4. Methods’ Computational Performance. One of the primary goals of the exact and heuristic methods introduced in Sections 3 and 4 is to speed up computation relative to the quadratic formulation described in (2), enabling us to handle realistically-sized instances. To evaluate their effectiveness and understand the drivers of improvement, we compare the quality of the solutions (gap) and the computational time required by each approach to solve the Antofagasta instance described previously for different budget values. Specifically, we compare these two elements when solving (i) the quadratic formulation (*Quad*) and (ii) the compact formulation with L-constraints (*Lin*) with an off-the-shelf solver, when running (iii) Algorithm 1 for the generalized comb formulation (*Comb*), (iv) the hybrid approach (*Hybrid*), and (v) our two heuristics, *i.e.*, *Greedy* and *LPH*. For (iii) and (iv), we use the initialization and acceleration tools for Algorithm 1 discussed in Appendix D.2.

In Table 3, we report the running time and the gap achieved (in brackets) by each method. These results were obtained on an Intel(R) Xeon(R) Gold 6226 CPU clocked at 2.70GHz, using Gurobi 11.0.0 to solve optimization problems with code written in Julia 1.10.2. For each instance, the run of each method was restricted to a single thread and, for the exact methods (i) to (iv), a time limit of one hour and a MipGap tolerance of 10^{-4} . To facilitate the comparison across methods, we compute the gap with respect to the best solution obtained with *Hybrid* within a time limit of ten hours and MipGap tolerance of 10^{-4} (*i.e.*, the solutions used in the previous sections).

First, note that we cannot solve the problem using *Quad* for high budget values. Second, we observe that *Lin* helps speed up computation relative to *Quad* but fails to solve the instance for greater budgets and even to output a feasible solution on those cases. Third, we observe that the proposed methods involving our cutting plane method (*i.e.*, *Comb* and *Hybrid*) find a feasible

Table 3 Methods Comparison

B	$r_{s,\emptyset} = \succ_s + 1$						$r_{s,\emptyset} = \mathcal{C} + 1$					
	Quad	Lin	Comb	Hybrid	Greedy	LPH	Quad	Lin	Comb	Hybrid	Greedy	LPH
0	24.916 [0.000]	12.294 [0.000]	6.478 [0.000]	5.741 [0.000]	0.008 [0.000]	0.730 [0.000]	25.362 [0.000]	10.256 [0.000]	4.368 [0.000]	5.158 [0.000]	0.007 [0.000]	0.482 [0.000]
1	56.837 [0.000]	18.885 [0.000]	63.156 [0.000]	10.705 [0.000]	31.000 [0.000]	0.428 [0.000]	62.163 [0.000]	16.747 [0.000]	40.624 [0.000]	9.126 [0.000]	33.774 [0.000]	0.441 [0.000]
2	38.474 [0.000]	21.602 [0.000]	94.173 [0.000]	21.774 [0.000]	63.451 [0.000]	1.447 [0.001]	40.415 [0.000]	16.880 [0.000]	80.941 [0.000]	16.788 [0.000]	78.872 [0.000]	0.466 [0.000]
5	346.913 [0.000]	39.580 [0.000]	1499.293 [0.000]	23.489 [0.000]	124.519 [0.000]	0.459 [0.002]	206.549 [0.000]	37.673 [0.000]	916.264 [0.000]	17.103 [0.000]	134.592 [0.000]	0.446 [0.000]
10	2549.262 [0.000]	121.820 [0.000]	tl	65.105 [0.000]	233.805 [0.001]	0.457 [0.003]	tl	61.807 [0.000]	tl	31.775 [0.000]	222.407 [0.000]	0.498 [0.001]
25	tl	tl	tl	2118.371 [0.002]	611.750 [0.000]	0.994 [0.003]	tl	226.337 [0.000]	tl	201.855 [0.000]	592.394 [0.000]	1.750 [0.003]
50	tl	tl	tl	tl	1114.221 [0.005]	0.718 [0.008]	tl	tl	tl	862.415 [0.001]	1404.633 [0.001]	0.624 [0.001]
100	tl	tl	tl	tl	tl	3.357 [0.016]	tl	tl	tl	tl	2.984 [0.001]	
150	tl	tl	tl	tl	tl	2.895 [0.023]	tl	tl	tl	tl	2.101 [0.002]	
200	tl	tl	tl	tl	2906.369 [0.014]	3.373 [0.029]	tl	tl	tl	2642.118 [0.010]	3.720 [0.004]	
	-	-	-	-	-	-	-	-	-	[0.010]	[0.000]	[0.016]

Note. For each method and budget value, we report the time needed to solve the problem (within the tolerance for exact methods) and the gap to the best objective value found in brackets. We use “tl” to denote instances not solved within the time limit and “-” to represent instances where a method did not find a feasible solution.

solution within the time limit for all the budgets considered. This is not surprising, since when the separation algorithm is called, a student-optimal stable matching for the current seat allocation is computed. Fourth, comparing *Comb* and *Hybrid*, we find that combining the two formulations helps reduce computing times and achieve lower gaps. Finally, as expected, we observe that *LPH* is substantially faster than all the other methods considered, and its running time is unaffected by the budget, unlike all the other methods. These results suggest that our proposed methods are flexible enough to accommodate different needs. On the one hand, if time is not a concern and the main focus is to obtain a low optimality gap—which is the case when solving the allocation as the timeline of the process is such that there is almost a month between the deadline to submit preferences and the publication of the assignment results—, then *Hybrid* may be a good approach. On the other hand, if speed is required—for instance, to perform simulations on different counterfactual policies—then *LPH* becomes handy. It should be noted that, unlike the *Comb* or *Hybrid*, which can output optimality gaps (*i.e.*, a measure of how far the method is from proving optimality), heuristics offer no guarantee of solution performance.

To further compare our heuristics, we report (i) the gap in the objective (in Figure 6a) and (ii) the gap in entry (in Figure 6b) obtained relative to the best solution found. On the one hand, we observe that the gap in the objective is relatively small, remaining below 3% for both heuristics and

all budget values and penalty levels. Moreover, we observe that *Greedy* outperforms *LPH* when (i) the penalty is high and (ii) for low budget values when the penalty is low. On the other hand, we observe that both methods achieve small gaps in entry, with *LPH* leading to the best results when the penalty is low. These results, combined with the fact that *LPH* swiftly returns a near-optimal solution, suggest that this heuristic can be a practical approach for large instances (involving high budgets) and time-sensitive tasks, especially when the goal is to minimize the number of unassigned students. Furthermore, as pointed out by MINEDUC, *LPH* has the advantage that it does not rely on students' random tie-breakers, so it can be used to make capacity decisions in cases where it may require time to contact schools to request additional seats and, thus, delaying capacity decisions may not be feasible.

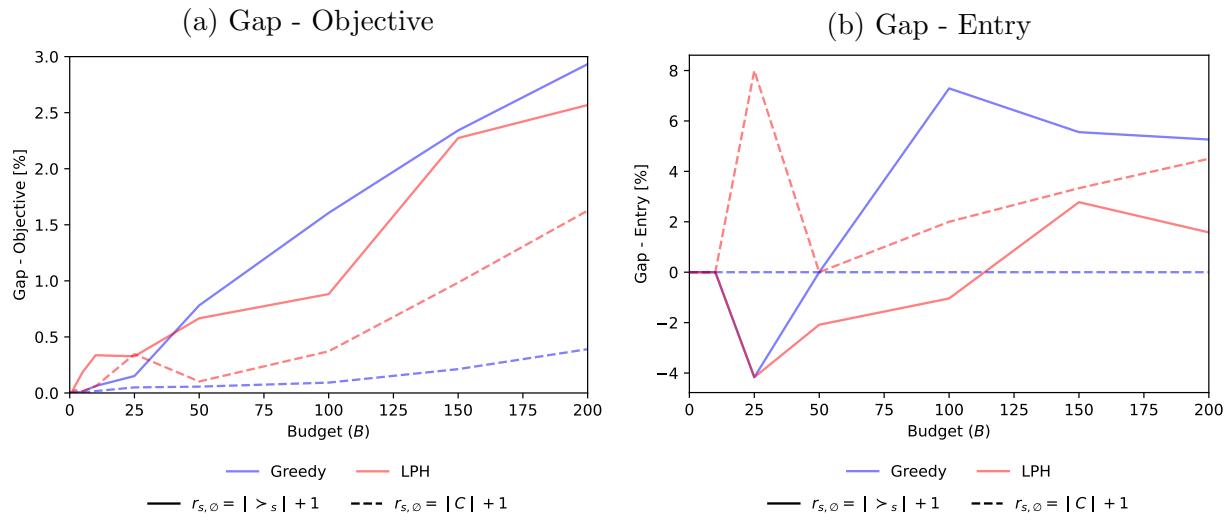


Figure 6 Heuristics

6. Properties of the Mechanism

Now that we have devised a framework to solve Problem (1), we briefly discuss some properties of the mechanism¹⁴ and the underlying optimal solutions. We provide a thorough discussion of these properties in Appendix E.

Cardinality. In the standard setting with no capacity decisions, we know from the Rural Hospital theorem (Roth 1986) that the set of students assigned in any stable matching is the same. This result no longer holds when we add capacity decisions, as the cardinality of the matching largely depends on the penalty values $r_{s,\emptyset}$ of unassigned students. In Appendix E.1, we show that if these values are sufficiently small, then there exists an optimal solution of Problem 1 whose allocation of

¹⁴ Mechanism in this case stands for the optimal method that solves Problem (1).

capacities yields a student-optimal stable matching of minimum cardinality among all the possible student-optimal stable matchings (Theorem 4).¹⁵ In contrast, if the penalty values are sufficiently high, we show that the optimal solution of the problem corresponds to a student-optimal stable matching of maximum cardinality (Theorem 5).

Note that this result is not surprising in hindsight, as the objective function in Problem (1) is the weighted sum of the students' preference of assignment and the value of unassigned students. Nevertheless, Theorems 4 and 5 are valuable from a policy standpoint, as they provide policymakers a tool to obtain an entire spectrum of stable assignments controlled by capacity planning where two extreme solutions stand out: (i) the solution that maximizes the number of assigned students (access), and (ii) the solution that allocates the extra seats to benefit the preferences of the students in the initial assignment (improvement). The former is a common goal in school choice settings, where the clearinghouse must guarantee a spot to each student that applies to the system, while the latter is common in college admissions, where merit plays a more critical role. Independent of the goal (or any intermediate point), our framework allows policymakers to achieve it by simply modifying the models' parameters resulting in a solution approach that is flexible and easy to communicate and interpret.

Incentives. A property that is commonly sought after in any mechanism is strategy-proofness, *i.e.*, that students have no incentive to misreport their preferences in order to improve their allocation. Roth (1982) and Dubins and Freedman (1981) show that the student-proposing version of DA is strategy-proof for students in the case with no capacity decisions. Unfortunately, this is not the case when students know that there exists a budget of extra seats to be allocated, as we formally show in Appendix E.2 (see Proposition 1). Nevertheless, we also show that our mechanism is strategy-proof in the large (Azevedo and Budish 2018), which guarantees that it is approximately optimal for students to report their true preferences for any i.i.d. distribution of students' reports. As Azevedo and Budish (2018) argue, this is a more appropriate notion of manipulability in large markets, as students are generally unaware of other students' realized preferences and priorities. Thus, the lack of strategy-proofness is not a major concern in our setting.¹⁶

Monotonicity. Another commonly desired property in any mechanism is student-monotonicity, which guarantees that any improvement in students' priorities (in school choice) or scores (in college admissions) cannot harm their assignment. Balinski and Sönmez (1999) and Baïou and Balinski (2004) show that this property holds for the student-proposing DA algorithm in the

¹⁵ Interestingly, the minimum cardinality student-optimal stable matching is not necessarily the most preferred by the set of students initially assigned when $B = 0$ (see Example E.1.1 in Appendix E).

¹⁶ Note that Afacan et al. (2024) independently show that their mechanism is also manipulable, and in the special case of schools having the same preference list, they provide a mechanism that is efficient and strategy-proof.

standard setting. However, in Appendix E.3, we show that students can be harmed when adding extra seats if their rank improves in a given school. Nevertheless, we can adapt our model to incorporate additional constraints that would rule out non-monotone allocations if this is a concern for policymakers.

7. Conclusions

Motivated by the urgent needs of the Chilean school choice system, where insufficient capacity has led to a significant numbers of students being unassigned, we study the joint problem of capacity planning and stable matching in many-to-one matching markets. Specifically, we extend the standard stable matching model by integrating capacity decisions, allowing policymakers to prioritize different objectives, such as maximizing access or enhancing assignments for high-priority students, by modifying the penalty incurred for having unassigned students. We show that the problem can be modeled using a quadratically-constrained integer program. To overcome the challenges imposed by the non-linearity of constraints and by the intergrality of the matching variables, we provide two alternative formulations that exploit the properties of the problem, enabling us to design an efficient cutting plane algorithm. This approach relies on a novel separation algorithm aimed at enhancing the one introduced by Baïou and Balinski (2000). Furthermore, we develop two efficient heuristics that enable us to find near-optimal solutions, with *LPH* being the most suitable for for time-sensitive tasks and when information about lotteries may not be available.

Finally, we apply our framework to real-world data from the Pre-K level in the Antofagasta region of Chile and identify several key insights. First, we find that each additional seat benefits multiple students, but the marginal improvement diminishes as we increase the number of seats added to the system. Second, we observe that the penalty for unassigned students impacts the allocation, so policymakers can use it to prioritize different goals. However, we observe that the differences in access depending on the penalty decrease for greater budgets. Finally, we show that our framework is flexible enough to accommodate different needs, including school-specific budget constraints, the lack of information about school lotteries or students' current preferences, and other extensions. Building on our positive results, we are collaborating with Chile's Ministry of Education (MINEDUC) and the NGO in charge of performing the allocation to test our framework in the field and develop solutions to the ongoing crisis.

Overall, our results showcase how to extend the classic stable matching problem to incorporate capacity decisions and how to solve the problem effectively. Our methodology is flexible to accommodate different settings, such as capacity reductions, allocations of tuition waivers, quotas, secured enrollment, and arbitrary constraints on the extra seats per school (see Appendix F for more details). Moreover, we can adapt our framework to other stable matching settings, such as the

allocation of budgets to accommodate refugees, the assignment of scholarships or tuition waivers in college admissions, and the rationing of scarce medical resources. All of these are exciting new areas of research in which our results can be used.

Acknowledgments

This work was funded by FRQ-IVADO Research Chair in Data Science for Combinatorial Game Theory, and the NSERC grant 2019-04557.

References

- Abdulkadiroğlu A, Che YK, Yasuda Y (2011) Resolving conflicting preferences in school choice: The “boston mechanism” reconsidered. *American Economic Review* 101(1):399–410.
- Abdulkadiroğlu A, Pathak PA, Roth AE (2009) Strategy-proofness versus efficiency in matching with indifferences: Redesigning the NYC high school match. *American Economic Review* 99(5):1954–1978.
- Abdulkadiroğlu A, Pathak PA, Roth AE, Sönmez T (2005) The boston public school match. *American Economic Review* 95(2):368–371.
- Abdulkadiroğlu A, Sönmez T (2003) School choice: A mechanism design approach. *American Economic Review* 93(3):729–747.
- Abe K, Komiya J, Iwasaki A (2022) Anytime capacity expansion in medical residency match by monte carlo tree search. *arXiv preprint arXiv:2202.06570*.
- Afacan MO, Dur U, Van der Linden M (2024) Capacity design in school choice. *Games and Economic Behavior*.
- Ágoston KC, Biró P, Kováts E, Jank Z (2021) College admissions with ties and common quotas: Integer programming approach. *European Journal of Operational Research* ISSN 0377-2217.
- Ágoston KC, Biró P, McBride I (2016) Integer programming methods for special college admissions problems. *Journal of Combinatorial Optimization* 32(4):1371–1399.
- Ahani N, Andersson T, Martinello A, Teytelboym A, Trapp AC (2021) Placement optimization in refugee resettlement. *Operations Research* 69(5):1468–1486.
- Allman M, Ashlagi I, Lo I, Love J, Mentzer K, Ruiz-Setz L, O'Connell H (2022) Designing school choice for diversity in the san francisco unified school district. *Proceedings of the 23rd ACM Conference on Economics and Computation* (ACM).
- Andersson T, Ehlers L (2020) Assigning refugees to landlords in sweden: Efficient, stable, and maximum matchings. *The Scandinavian Journal of Economics* 122(3):937–965.
- Arnold N (2015) Short lists in centralized clearinghouses. *Proceedings of the Sixteenth ACM Conference on Economics and Computation* (ACM).
- Ashlagi I, Nikzad A, Romm A (2019) Assigning more students to their top choices: A comparison of tie-breaking rules. *Games and Economic Behavior* 115:167–187.

- Ashlagi I, Shi P (2016) Optimal allocation without money: An engineering approach. *Management Science* 62(4):1078–1097.
- Azevedo EA, Budish E (2018) Strategy-proofness in the large. *The Review of Economics Studies* 86(1):81–116.
- Aziz H, Brandl F (2021) Efficient, fair, and incentive-compatible healthcare rationing, arXiv preprint arXiv:2102.04384.
- Baïou M, Balinski M (2000) The stable admissions polytope. *Mathematical programming* 87(3):427–439.
- Baïou M, Balinski M (2004) Student admissions and faculty recruitment. *Theoretical Computer Science* 322(2):245–265.
- Balinski M, Sönmez T (1999) A tale of two mechanisms: student placement. *Journal of Economic theory* 84(1):73–94.
- Bobbio F (2024) *Dynamic capacities and priorities in stable matching*. Phd thesis, Université de Montréal.
- Bobbio F, Carvalho M, Lodi A, Torrico A (2022) Capacity variation in the many-to-one stable matching problem .
- Bodoh-Creed AL (2020) Optimizing for distributional goals in school choice problems. *Management Science* 66(8):3657–3676.
- Budish E, Cachon G, Kessler J, Othman A (2016) Course match: A largescale implementation of approximate competitive equilibrium from equal incomes for combinatorial allocation. *Operations Research* 65(2):314–336.
- Calsamiglia C, Güell M (2018) Priorities in school choice: The case of the boston mechanism in barcelona. *Journal of Public Economics* 163:20–36.
- Caro F, Shirabe T, Guignard M, Weintraub A (2004) School redistricting: embedding GIS tools with integer programming. *Journal of the Operational Research Society* 55(8):836–849.
- Correa J, Epstein N, Epstein R, Escobar J, Rios I, Aramayo N, Bahamondes B, Bonet C, Castillo M, Cristi A, Epstein B, Subiabre F (2022) School choice in chile. *Operations Research* 70(2):1066–1087.
- Delacretaz D, Kominers SD, Teytelboym A (2016) Refugee resettlement.
- Delorme M, García S, Gondzio J, Kalcsics J, Manlove D, Pettersson W (2019) Mathematical models for stable matching problems with ties and incomplete lists. *European Journal of Operational Research* 277(2):426–441.
- Dubins LE, Freedman DA (1981) Machiavelli and the gale-shapley algorithm. *The American Mathematical Monthly* 88(7):485–494.
- Ehlers L (2010) School choice with control, cahiers de recherche 13-2010 CIREQ.
- Feigenbaum I, Kanoria Y, Lo I, Sethuraman J (2020) Dynamic matching in school choice: Efficient seat reassignment after late cancellations. *Management Science* 66(11):5341–5361.

- Gale D, Shapley LS (1962) College admissions and the stability of marriage. *The American Mathematical Monthly* 69(1):9–15.
- Gale D, Sotomayor M (1985) Some remarks on the stable matching problem. *Discrete Applied Mathematics* 11(3):223–232.
- Grötschel, M L Lovász, Schrijver A (1981) The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica* 1(1):169–197.
- Gusfield D, Irving RW (1989) *The stable marriage problem: structure and algorithms* (MIT press).
- Hafalir IE, Yenmez MB, Yildirim MA (2013) Effective affirmative action in school choice. *Theoretical Economics* 8(2):325–363.
- Kesten O, Ünver MU (2015) A theory of school-choice lotteries. *Theoretical Economics* 10(2):543–595.
- Kojima F, Tamura A, Yokoo M (2018) Designing matching mechanisms under constraints: An approach from discrete convex analysis. *Journal of Economic Theory* 176:803–833.
- Kumano T, Kurino M, et al. (2022) Quota adjustment process. Technical report, Institute for Economics Studies, Keio University.
- Kurata R, Hamada N, Iwasaki A, Yokoo M (2017) Controlled school choice with soft bounds and overlapping types. *Journal of Artificial Intelligence Research* 58(1):153–184.
- Kwanashie A, Manlove DF (2014) An integer programming approach to the hospitals/residents problem with ties. *Operations research proceedings 2013*, 263–269 (Springer).
- Manlove D (2013) *Algorithmics of matching under preferences*, volume 2 (World Scientific).
- McCormick GP (1976) Computability of global solutions to factorable nonconvex programs: Part I - convex underestimating problems. *Mathematical Programming* 10:147–175.
- MINEDUC (2022) Plan de Fortalecimiento de Matrícula'. <https://www.mineduc.cl/plan-de-fortalecimiento-de-matricula/>, [Online; last accessed 7-June-2024].
- National Center for Education Statistics (1999) Condition of america's public school facilities: 1999. Access here.
- Pathak PA, Sönmez T (2008) Leveling the playing field: Sincere and sophisticated players in the boston mechanism. *American Economic Review* 98(4):1636–1652.
- Pathak PA, Sönmez T, Ünver MU, Yenmez MB (2020) Fair allocation of vaccines, ventilators and antiviral treatments: Leaving no ethical value behind health care rationing.
- Rios I, Bobbio F, Carvalho M, Torrico A (2024) Stable matching with contingent priorities.
- Rios I, Larroucau T, Parra G, Cominetti R (2021) Improving the chilean college admissions system. *Operations Research* 69(4):1186–1205.
- Romm A, Roth AE, Shorrer RI (2020) Stability vs. no justified envy. *No Justified Envy (March 6, 2020)*.

- Roth AE (1982) The economics of matching: Stability and incentives. *Mathematics of operations research* 7(4):617–628.
- Roth AE (1984) The evolution of the labor market for medical interns and residents: a case study in game theory. *Journal of political Economy* 92(6):991–1016.
- Roth AE (1986) On the allocation of residents to rural hospitals: a general property of two-sided matching markets. *Econometrica: Journal of the Econometric Society* 425–427.
- Roth AE (2002) The economist as engineer: Game theory, experimentation, and computation as tools for design economics. *Econometrica* 70(4):1341–1378.
- Roth AE, Rothblum UG, Vande Vate JH (1993) Stable matchings, optimal assignments, and linear programming. *Mathematics of operations research* 18(4):803–828.
- Roth AE, Sotomayor MAO (1990) *Two-sided matching: A study in game-theoretic modeling and analysis* (Cambridge Univ. Press, Cambridge, MA).
- Rothblum UG (1992) Characterization of stable matchings as extreme points of a polytope. *Mathematical Programming* 54(1):57–67.
- Schrijver A, et al. (2003) *Combinatorial optimization: polyhedra and efficiency*, volume 24 (Springer).
- Shi P (2016) Assortment planning in school choice.
- Soma T, Yoshida Y (2015) A generalization of submodular cover via the diminishing return property on the integer lattice. *Advances in neural information processing systems* 28.
- Sönmez T, Yenmez M (2019) Affirmative action with overlapping reserves.
- Tucker J (2022) Sfusd enrollment plummets this year, doubling peak pandemic declines, new data shows. Access here.
- Vate JHV (1989) Linear programming brings marital bliss. *Operations Research Letters* 8(3):147–153.

Appendix A: Appendix to Section 3.2

A.1. Proof of Lemma 1

We prove the theorem assuming that Problem 1 has no extra capacities, *i.e.*, $B = 0$. Indeed, given \mathbf{t} , the expanded instance $\Gamma_{\mathbf{t}}$ can be seen as an instance of the school choice problem with $B = 0$. Note that, if $B = 0$, an optimal solution of Problem 1 is of the form $(\mathbf{0}, \mu)$.

Next, we prove that a student-optimal stable matching is a stable matching that minimizes the average rank of the students, and vice-versa.

If: In a student-optimal stable matching, each student is assigned to the best school they could achieve in any stable matching (Gale and Shapley 1962). Thus, each unassigned student is unassigned in every stable matching. Moreover, by the Rural Hospital Theorem (Roth 1982, 1984, Gale and Sotomayor 1985, Roth 1986), the same students are assigned in all stable matchings. Suppose that μ is the student-optimal stable matching but it is not optimal to Problem 1. Let μ' be an optimal solution to Problem 1 (recall that \mathbf{t} is

fixed). Hence, the following inequality holds: $\sum_{(s,c) \in \mu} r_{s,c} > \sum_{(s,c) \in \mu'} r_{s,c}$. This means that there is at least one student s' who prefers the stable matching μ' to the stable matching μ , which is a contradiction.

Only if: Let μ be a (stable) matching corresponding to an optimal solution of Problem 1 (with t is fixed). As before, by the Rural Hospital Theorem, we observe that the set of students unassigned in μ is the same set of students unassigned in every stable matching. Hence, the set of assigned students in μ , is the same set for every stable matching. Let us suppose, again by contradiction, that μ is not a student-optimal stable matching. Let μ' be a student-optimal stable matching. Denote by S' the set of students whose assignment to schools differs in the two matchings. By construction, the objective value of Problem 1 for $\mathcal{S} \setminus S'$ is the same in both μ and μ' :

$$\sum_{(s,c) \in \mu: s \in \mathcal{S} \setminus S'} r_{s,c} = \sum_{(s,c) \in \mu': s \in \mathcal{S} \setminus S'} r_{s,c}.$$

Furthermore, S' is the disjoint union of the following two sets of students: S'_1 , the set of assigned students who prefer their school in μ' , and S'_2 , the set of assigned students who prefer their school in μ . By Gale and Shapley (1962), μ' is a stable matching in which all students are assigned the best school they could achieve in any stable matching. Therefore, the set S'_2 is empty. Hence, $S' = S'_1$, and by hypothesis

$$\sum_{(s,c) \in \mu: s \in S'_1} r_{s,c} < \sum_{(s,c) \in \mu': s \in S'_1} r_{s,c}.$$

However, from the definition of S'_1 , $r_{s,\mu'(s)} \leq r_{s,\mu(s)}$ for every $s \in S'_1$, which leads to a contradiction. \square

A.2. Multiple Optimal Solutions

Consider an instance with three schools $\mathcal{C} = \{c_1, c_2, c_3\}$, four students $\mathcal{S} = \{s_1, s_2, s_3, s_4\}$, and capacities $q_{c_1} = q_{c_2} = 1$, $q_{c_3} = 2$. In addition, consider preferences given by:

$$c : s_1 \succ s_2 \succ s_3 \succ s_4, \forall c \in \mathcal{C}$$

$$s_1 : c_1 \succ c_2 \succ c_3$$

$$s_2 : c_2 \succ c_1 \succ c_3$$

$$s_3 : c_1 \succ c_3 \succ c_2$$

$$s_4 : c_2 \succ c_3 \succ c_1.$$

Notice that, with no capacity expansion, the student-optimal stable matching is

$$\mu^* = \{(s_1, c_1), (s_2, c_2), (s_3, c_3), (s_4, c_3)\},$$

which leads to a value of 6. On the other hand, if we have a budget $B = 1$, note that we can allocate it to either c_1 and obtain the matching $\mu' = \{(s_1, c_1), (s_2, c_2), (s_3, c_1), (s_4, c_3)\}$, or to school c_2 and obtain the matching $\mu'' = \{(s_1, c_1), (s_2, c_2), (s_3, c_3), (s_4, c_2)\}$. In both cases, one student moves from their second choice to their top choice, and thus in both cases the sum of preferences of assignment is 5. Hence, we conclude that this problem has more than one optimal solution.

Appendix B: Appendix to Section 3.3

B.1. Proof of Lemma 2

The lemma follows by noting that since $\mathbf{x} \in \mathcal{P}_{\mathbb{Z}}$, then $\sum_{c' \succeq_s c} x_{s,c'}$ is either 0 or 1. When $\sum_{c' \succeq_s c} x_{s,c'} = 0$, the L-constraint becomes $\sum_{s' \succ_c s} x_{s',c} \geq q_c + t_c$, which coincides with (2b). When $\sum_{c' \succeq_s c} x_{s,c'} = 1$, the L-constraint is $\sum_{s' \succ_c s} x_{s',c} \geq t_c - B$. Given that $t_c - B \leq 0$, this constraint is redundant because $\sum_{s' \succ_c s} x_{s',c}$ is always non-negative. Constraint (2b) is also redundant under this case. \square

B.2. Background on McCormick Envelopes

In this section we describe the McCormick convex envelope used to obtain a linear relaxation for bi-linear terms (McCormick 1976); if one of the terms is binary, the linearization provides an equivalent formulation. Consider a bi-linear term of the form $x_i \cdot x_j$ with the following bounds for the variables x_i and x_j : $l_i \leq x_i \leq u_i$ and $l_j \leq x_j \leq u_j$. Let us define $y = x_i \cdot x_j$, $m_i = (x_i - l_i)$, $m_j = (x_j - l_j)$, $n_i = (u_i - x_i)$ and $n_j = (u_j - x_j)$. Note that $m_i \cdot m_j \geq 0$, from which we derive the under-estimator $y \geq x_i \cdot l_j + x_j \cdot l_i - l_i \cdot l_j$. Similarly, it holds that $n_i \cdot n_j \geq 0$, from which we derive the under-estimator $y \geq x_i \cdot u_j + x_j \cdot u_i - u_i \cdot u_j$. Analogously, over-estimators of y can be defined. Make $o_i = (u_i - x_i)$, $o_j = (x_j - l_j)$, $p_i = (x_i - l_i)$ and $p_j = (u_j - x_j)$. From $o_i \cdot o_j \geq 0$ we obtain the over-estimator $y \leq x_i \cdot l_j + x_j \cdot u_i - u_i \cdot l_j$, and from $p_i \cdot p_j \geq 0$ we obtain the over-estimator $y \leq x_j \cdot l_i + x_i \cdot u_j - u_j \cdot l_i$. The four inequalities provided by the over and under estimators of y , define the McCormick convex (relaxation) envelope of $x_i \cdot x_j$.

B.3. McCormick Linearizations of Formulation (2)

Note that the quadratic term $t_c \cdot \sum_{c' \succeq_s} x_{s,c'}$ in constraint (2b) can be linearized in at least two ways. Specifically, we call

- *Aggregated Linearization*, when for each $(s, c) \in \mathcal{E}$, we define $\alpha_{s,c} := t_c \cdot \sum_{c' \succeq_s} x_{s,c'}$;
- *Non-Aggregated Linearization*, when for each $(s, c) \in \mathcal{E}$ and $c' \succeq_s c$, we define $\beta_{s,c,c'} := t_c \cdot x_{s,c'}$.

The mixed-integer programming formulation of the McCormick envelope for the aggregated linearization reads as

$$\min_{\mathbf{x}, \mathbf{t}, \alpha} \sum_{(s,c) \in \mathcal{E}} r_{s,c} \cdot x_{s,c} \quad (8a)$$

$$s.t. \quad t_c - \alpha_{s,c} + q_c \cdot \left(1 - \sum_{c' \succeq_s c} x_{s,c'}\right) \leq \sum_{s' \succ_s c} x_{s',c}, \quad \forall (s,c) \in \mathcal{E}, c \succ_s \emptyset \quad (8b)$$

$$-\alpha_{s,c} + t_c + B \cdot \sum_{c' \succeq_s c} x_{s,c'} \leq B, \quad \forall (s,c) \in \mathcal{E}, c \succ_s \emptyset \quad (8c)$$

$$\alpha_{s,c} \leq t_c, \quad \forall (s,c) \in \mathcal{E}, c \succ_s \emptyset \quad (8d)$$

$$\alpha_{s,c} \leq B \cdot \sum_{c' \succeq_s c} x_{s,c'}, \quad \forall (s,c) \in \mathcal{E}, c \succ_s \emptyset \quad (8e)$$

$$(\mathbf{x}, \mathbf{t}) \in \mathcal{P}_{\mathbb{Z}}, \alpha \geq 0. \quad (8f)$$

Next, we prove that the set of feasible solutions for the original problem coincides with the set of feasible solutions obtained from the linearization.

COROLLARY 1. *The projection of the feasible region in Formulation (8) in the variables \mathbf{t} and \mathbf{x} coincides with the region given by Formulation (2).*

We now discuss the mixed-integer programming formulation of the McCormick envelope for the non-aggregated linearization. Formally, we have the following

$$\min_{\mathbf{x}, \mathbf{t}, \beta} \sum_{(s,c) \in \mathcal{E}} r_{s,c} \cdot x_{s,c} \quad (9a)$$

$$s.t. \quad t_c - \sum_{c' \succeq_s c} \beta_{s,c,c'} + q_c \cdot \left(1 - \sum_{c' \succeq_s c} x_{s,c'}\right) \leq \sum_{s' \succ_s c} x_{s',c}, \quad \forall (s,c) \in \mathcal{E}, c \succ_s \emptyset \quad (9b)$$

$$-\beta_{s,c,c'} + t_c + B \cdot x_{s,c'} \leq B, \quad \forall (s, c) \in \mathcal{E}, \forall c' \succeq_s c \succ_s \emptyset, \quad (9c)$$

$$\beta_{s,c,c'} \leq t_c, \quad \forall (s, c) \in \mathcal{E}, \forall c' \succeq_s c \succ_s \emptyset, \quad (9d)$$

$$\beta_{s,c,c'} \leq B \cdot x_{s,c'}, \quad \forall (s, c) \in \mathcal{E}, \forall c' \succeq_s c \succ_s \emptyset, \quad (9e)$$

$$(\mathbf{x}, \mathbf{t}) \in \mathcal{P}_{\mathbb{Z}}, \beta \geq 0. \quad (9f)$$

Constraints (9c), (9d), (9e) and the non-negativity constraints for $\beta_{s,c,c'}$ form the McCormick envelope. Similar to the case of Corollary 1, we know that this is an exact formulation since $x_{s,c'} \in \{0, 1\}$.

COROLLARY 2. *The projection of the feasible region given by Formulation (9) in the variables \mathbf{t} and \mathbf{x} coincides with the region given by Formulation (2).*

Therefore, Formulation (8) and (9) yield the same set of feasible solutions. Interestingly, the feasible region of the relaxed aggregated linearization (*i.e.*, $(\mathbf{x}, \mathbf{t}) \in \mathcal{P}_{\mathbb{Z}}$ in Formulation (8) is changed to $(\mathbf{x}, \mathbf{t}) \in \mathcal{P}$) is contained in the feasible region of the relaxed non-aggregated linearization.

THEOREM 2. *The feasible region of the relaxed aggregated linearization model is contained in the feasible region of the relaxed non-aggregated linearization model.*

Proof. The constraints that do not involve the linearization terms $\alpha_{s,c}$ or $\beta_{s,c,c'}$ are trivially satisfied by a feasible solution in both formulations. Therefore, we will restrict our analysis to the remaining constraints. Let $(\mathbf{x}, \mathbf{t}, \boldsymbol{\alpha})$ be a feasible solution of the relaxed aggregated linearized program. It is easy to verify that by defining $\bar{\beta}_{s,c,c'} = \alpha_{s,c} \cdot x_{s,c'}$ for every $s \in \mathcal{S}$, $c \in \mathcal{C}$ and $c' \succeq_s c$, the constraints of the relaxed non-aggregated linearization are all met. \square

Theorem 2 implies that the optimal value of the relaxed aggregated linearized model is greater than or equal to the optimal value of the relaxed non-aggregated linearized model.

Appendix C: Appendix to Section 3.4

C.1. Results on the Generalized Comb Constraints

In the following lemma, we show that if a capacity variable is set to 1, then all the comb constraints determined by supersets are satisfied.

LEMMA 3. *Let $\bar{\mathbf{x}} \in \{0, 1\}^{\mathcal{E}}$ be a stable matching for some vector of extra seats $\bar{\mathbf{y}} \in \{0, 1\}^{C \times \{1, \dots, B\}}$. If $\bar{y}_c^k = 1$ then all constraints (5b) for the combs in $\mathbf{C}_c(k + \alpha)$ with $\alpha > 0$ are satisfied.*

Proof. When $\bar{y}_c^k = 1$, the right-hand-side of constraints (5b) for combs in $\mathbf{C}_c(k + \alpha)$ is $q_c + k$ since $y_c^k = 1$ implies that $\bar{y}_c^\ell = 1$ for all $\ell = 1, \dots, k - 1$. Thus, we need to show that the left-hand-side of these constraints is guaranteed to be greater than or equal to $q_c + k$. By the definition of comb, for all $C \in \mathbf{C}_c(k + \alpha)$ there is $\hat{C} \in \mathbf{C}_c(k)$ such that $\hat{C} \subseteq C$. Hence,

$$\sum_{(s', c') \in C} \bar{x}_{s', c'} \geq \sum_{(s', c') \in \hat{C}} \bar{x}_{s', c'} \geq q_c + k,$$

where the second inequality follows from the hypothesis that $\bar{\mathbf{x}}$ is stable for $\bar{\mathbf{y}}$. \square

We now prove the analogous of the previous lemma, but for comb constraints defined by subsets.

LEMMA 4. Let $\bar{\mathbf{x}} \in \{0, 1\}^{\mathcal{E}}$ be a stable matching for some vector of extra seats $\bar{\mathbf{y}} \in \{0, 1\}^{\mathcal{C} \times \{1, \dots, B\}}$. If $\bar{y}_c^k = 1$, then all constraints (5b) for combs in $\mathbf{C}_c(k - \alpha)$ with $\alpha < k$ are satisfied.

Proof. When $\bar{y}_c^k = 1$, the right-hand-side of constraint (5b) for each comb $C \in \mathbf{C}_c(k - \alpha)$ is $q_c + k - \alpha$ because $\bar{y}_c^k = 1$ implies that $\bar{y}_c^\ell = 1$ for all $\ell = 1, \dots, k - 1$. Thus, we need to show that the left-hand-side of this constraint is guaranteed to be greater than or equal to $q_c + k - \alpha$. Let (s, c) be the base of the comb C .

First, we show the result when there is $\hat{C} \in \mathbf{C}_c(k)$ such that $C \subseteq \hat{C}$. By the definition of generalized comb (see Subsection 3.4.2), C contains the shaft $S(s, c)$. Consequently, since \hat{C} contains C , if (s', c) is the base of \hat{C} , then $S(s, c) \subseteq S(s', c)$.

Denote by T_{s_r} for $r = 1, \dots, q_c + k$ be the teeth of comb \hat{C} which (obviously) contains the teeth of comb C . Therefore,

$$\hat{C} = \bigcup_{r=1}^{q_c+k} T_{s_r} \cup S(s', c).$$

Let $I \subseteq \{1, \dots, q_c + k\}$ be the set of indices corresponding to the teeth of C ; thence, $|I| = q_c + k - \alpha$. There are two cases:

- If there is $r \in I$ such that

$$\sum_{(s_r, j) \in T_{s_r}} \bar{x}_{s_r, j} = 0,$$

then student s_r is matched with a school that they prefer less than school c . Under the stable matching $\bar{\mathbf{x}}$, this is only possible if school c is matched with exactly $q_c + k$ students that school c prefers more than student s_r :

$$\sum_{(i, c) \in S(s, c)} \bar{x}_{i, c} = q_c + k.$$

Since $S(s, c) \subseteq C$, then

$$\sum_{(i, j) \in C} \bar{x}_{i, j} \geq q_c + k \geq q_c.$$

- If $\sum_{(s_r, j) \in T_{s_r}} \bar{x}_{s_r, j} = 1$ for all $r \in I$, then

$$\sum_{r \in I} \sum_{(s_r, j) \in T_{s_r}} \bar{x}_{s_r, j} = |I| = q_c + k - \alpha \geq q_c.$$

Since $\bigcup_{r \in I} T_{s_r} \subseteq C$ the results holds.

Second, we prove the lemma when no comb in $\mathbf{C}_c(k)$ contains C . Then, such situation can only occur if it is not possible to find a shaft $S(s', c)$ where $s' = s$ or $s \succ_c s'$. This means that the number of seats $q_c + k$ is greater than the number of students that the school ranks. Consequently, the set $\mathbf{C}_c(k)$ is empty and there is nothing to prove. \square

C.2. Proof of Theorem 1

The statement follows directly from Baïou and Balinski (2000): When \mathbf{y} is fixed, the comb inequalities for the associated capacities provide the convex hull of stable matchings for Γ_t where $t_c = \sum_{k=1}^B y_c^k$ for all $c \in \mathcal{C}$. Thus, as long as the number of seats for each school is integer, constraints (5b) contain the comb constraints from Baïou and Balinski (2000) for Γ_t . \square

C.3. Our Separation Algorithm and Structural Results

Before introducing our separation method, formalized in Algorithm 4, we present key structural results that reduce significantly the running time. Given an optimal solution $(\mathbf{x}^*, \mathbf{y}^*)$ obtained from Step 2 in Algorithm 1, let \mathbf{t}^* be the projection of \mathbf{y}^* in the original problem, i.e., $t_c^* = \sum_{j=1}^B y_c^{j*}$ for all $c \in C$. In addition, let $\mu_{\mathbf{t}^*}^*$ be the student-optimal stable matching for instance $\Gamma_{\mathbf{t}^*}$.¹⁷ To simplify, we assume that \mathbf{y}^* (and thus \mathbf{t}^*) is fixed, and we use μ^* to represent $\mu_{\mathbf{t}^*}^*$. Throughout this section, the role of μ^* will be to guide the search of violated comb constraints.

Given a school c , let $\text{exceeding}(c) = \{s \in \mathcal{S} : x_{s,c}^* > 0 \text{ and } \mu_{s,c}^* = 0\}$ be the set of exceeding students, i.e., the set of students assigned (possibly fractionally) to school c in \mathbf{x}^* that are not assigned to c in μ^* , and let

$$\text{block}(\mathbf{x}^*) = \left\{ c \in \mathcal{C} : \sum_{s:(s,c) \in \mathcal{E}} x_{s,c}^* = \sum_{s:(s,c) \in \mathcal{E}} \mu_{s,c}^* = q_c + t_c^*, \text{exceeding}(c) \neq \emptyset \right\}$$

be the set of fully-subscribed schools in both \mathbf{x}^* and μ^* that have a non-empty set of exceeding students. Finally, a student-school pair (s, c) is called a fractional blocking pair for \mathbf{x}^* if the following two conditions hold: (i) there is a school c' such that $x_{s,c'}^* > 0$ and $c \succ_s c'$, and (ii) c is not fully-subscribed or there is a student s' such that $x_{s',c}^* > 0$ and $s \succ_c s'$.¹⁸

Given that the family of comb constraints is exponentially large, it is crucial to reduce the scope of our search. Our main structural result, formalized in Lemma 7, accomplishes this by further restricting the search of violated combs only among those schools that are in $\text{block}(\mathbf{x}^*)$. To prove Lemma 7, we first show Lemma 5, which states that while performing Algorithm 1, if a school is part of a blocking pair then it must be fully subscribed.

LEMMA 5. *Let $(\mathbf{x}^*, \mathbf{y}^*)$ be the optimal solution obtained at Step 2 in some iteration of Algorithm 1, and suppose that there is a fractional blocking pair (s, c) . Then, c is fully-subscribed in \mathbf{x}^* .*

Proof of Lemma 5. We prove that c is fully-subscribed in \mathbf{x}^* by contradiction. Let (s, c) be a fractional blocking pair for \mathbf{x}^* , where c is an under-subscribed school in \mathbf{x}^* . Let s be such that $x_{s,c}^* > 0$ and $c \succ_s c'$. So s prefers c over c' . Moreover, c has some available capacity which would allow to increase $x_{s,c}^*$ by decreasing $x_{s,c'}^*$. Clearly, such modification would decrease the objective value of the main program. This contradicts the optimality of \mathbf{x}^* . \square

In Appendix C.4.1, we present an example that shows that the set of fully-subscribed schools is not necessarily the same for \mathbf{x}^* and μ^* , and, thus, illustrates the potential of using μ^* . In our next, result we provide a sufficient condition for stopping the separation algorithm for a given allocation \mathbf{t}^* . Specifically, we show that we only need to check for violated constraints among the fully-subscribed schools, thus reducing significantly the number of schools that we need to check at every iteration of the separation algorithm.

¹⁷ We can obtain this by applying DA on the expanded instance $\Gamma_{\mathbf{t}^*}$.

¹⁸ Kesten and Ünver (2015) introduce the notion of ex-ante justified envy, which, in the case of a fully-subscribed school, is equivalent to our definition of fractional stability. Kesten and Ünver (2015) define ex-ante justified envy of student s towards student s' if both $x_{s,c'}^*, x_{s',c}^* > 0$ with $c \succ_s c'$ and $s \succ_c s'$. Note that the definition of fractional blocking pair implies that $x_{s,c}^* < 1$. Moreover, a blocking pair is also a fractional blocking pair.

LEMMA 6. *Let $(\mathbf{x}^*, \mathbf{y}^*)$ be the optimal solution obtained at Step 2 in some iteration of Algorithm 1 and μ^* the student-optimal stable matching in instance $\Gamma_{\mathbf{t}^*}$, where \mathbf{t}^* is the allocation defined by \mathbf{y}^* . If \mathcal{J} contains all the comb constraints of every fully-subscribed school in μ^* , then both matchings \mathbf{x}^* and μ^* coincide, i.e., \mathbf{x}^* is the characteristic vector of μ^* .*

Proof of Lemma 6. Let \mathcal{C}' be the set of fully-subscribed schools in μ^* and \mathcal{S}' be the set of students matched to the schools \mathcal{C}' in μ^* . We denote by \mathcal{C}'' and \mathcal{S}'' the complementary sets of \mathcal{C}' and \mathcal{S}' respectively. Let us analyze the matches in \mathcal{S}' . Some facts:

- Note that for every student $s \in \mathcal{S}'$ we have $\sum_{c \succeq_s \mu^*(s)} x_{s,c}^* = 1$, i.e., student s must be matched in \mathbf{x}^* to some school that she prefers at least as much as $\mu^*(s) \in \mathcal{C}'$. This is because \mathcal{J} contains all the comb constraints of the schools that are fully subscribed in the unique student-optimal stable matching μ^* . Since \mathbf{x}^* minimizes the rank of the schools that students are assigned to, then some fully subscribed schools may be part of a blocking pair, which can be addressed by introducing a comb constraint, however, all the comb constraints of the fully subscribed schools are already in \mathcal{J} .
- Every student $s \in \mathcal{S}'$ must be matched in \mathbf{x}^* to a school that is in \mathcal{C}' , i.e., $\sum_{c \in \mathcal{C}'} x_{s,c}^* = 1$. This is because, otherwise, s would create a blocking pair in μ^* with a under-subscribed school in \mathcal{C}'' due to our first argument. As a consequence, we cannot have $x_{s,c}^* > 0$ for a student $s \in \mathcal{S}''$ and a school $c \in \mathcal{C}'$. This is because we proved above that all students in \mathcal{S}' are matched in \mathcal{C}' and schools in \mathcal{C}' are fully-subscribed.

All the above imply that we can consider \mathcal{S}' and \mathcal{C}' as a sub-instance of the problem and the integrality of \mathbf{x}^* is given by the result proved by Baïou and Balinski (2000). Because of optimality, μ^* and \mathbf{x}^* must coincide in \mathcal{S}' and \mathcal{C}' .

Let us now analyze the assignments of the students in \mathcal{S}'' . First, note that the integrality of \mathbf{x}^* in the sub-instance defined by \mathcal{S}'' and \mathcal{C}'' is given by the integrality of the matching polytope without stability constraints, i.e., \mathcal{P} (restricted to \mathcal{S}'' and \mathcal{C}'' with \mathbf{t}^* fixed) has integral vertices (see e.g. (Schrijver et al. 2003)). Now, we will show that the assignments coincide in μ^* and \mathbf{x}^* . By contradiction, assume that the students in \mathcal{S}'' are matched differently in \mathbf{x}^* and in μ^* . First, if there is a student $s \in \mathcal{S}''$ that prefers a school $c \in \mathcal{C}''$ over $\mu^*(s)$, then (s, c) would be a blocking pair in μ^* because c is under-subscribed in μ^* , which is not possible. Therefore, every student $s \in \mathcal{S}''$ weakly prefers $\mu^*(s)$ over their assignment in \mathbf{x}^* , and, by the hypothesis in our contradiction argument, there is at least one student s that strictly prefers $\mu^*(s)$ over their match in \mathbf{x}^* (i.e., the two schools are different). Given that \mathcal{S}'' and \mathcal{C}'' form a separate sub-instance, we can rearrange the matching \mathbf{x}^* of every student in \mathcal{S}'' by matching the student to her matching in μ^* . This new matching would have a lower objective than the one of \mathbf{x}^* , which would violate its optimality. Therefore, both μ^* and \mathbf{x}^* also coincide in \mathcal{S}'' and \mathcal{C}'' . \square

Our next structural result, formalized in Lemma 7, further reduces the search by restricting the search of violated combs only among those schools that are in $\text{block}(\mathbf{x}^*)$.

LEMMA 7. *If \mathbf{x}^* has a fractional blocking pair, then there is at least one student-school pair (s, c) , where c is in $\text{block}(\mathbf{x}^*)$ and s is preferred by c over the least preferred student in $\text{exceeding}(c)$, such that the value of the tooth $T(s, c)$ is smaller than 1, i.e., $\sum_{(s, c') \in T(s, c)} x_{s,c'}^* < 1$.*

Proof of Lemma 7. Suppose that the statement is false. Then, for every school c in $\text{block}(\mathbf{x}^*)$, and for every student s more preferred by c to the least preferred student in $\text{exceeding}(c)$, the pair (s, c) has the tooth evaluated to 1. By Lemma 5, we know that the fractional blocking pairs must all involve a school that is fully-subscribed in \mathbf{x}^* . Among these schools, it is sufficient to focus our attention on the schools $c \in C$ such that there exists $s \in S$ with $x_{s,c}^* > 0$ and student s is not matched to c in μ^* . Therefore, for these schools the set of exceeding students is non-empty.

We now observe that, by hypothesis, every school c in $\text{block}(\mathbf{x}^*)$, has the property that all the teeth are evaluated to 1 until the least preferred exceeding student. This implies that all the combs in $\mathbf{C}_c(\mathbf{t}_c^*)$ are not violated since such combs are composed by $q_c + t_c^*$ teeth. Hence, \mathbf{x}^* is a stable matching with a better objective for the students than the student-optimal stable matching. \square

We now describe our separation method, which is illustrated in Algorithm 4. Algorithm 4 begins by initializing the set of violated combs \mathcal{J}' equal to \emptyset . Based on Lemma 7, we focus only on the schools in $\text{block}(\mathbf{x}^*)$. At Step 2, we iterate to find the most violated (*i.e.*, least valued) comb of every school in $\text{block}(\mathbf{x}^*)$. To do so, we first initialize as empty the list of teeth \mathcal{T} (Step 3). The list \mathcal{T} will be updated to store the set of students whose teeth are part of the least valued comb. At Step 4, we look for the least preferred exceeding student \underline{s} in school $c \in \text{block}(\mathbf{x}^*)$, and, finally, we introduce the preference list of school c that terminates with student \underline{s} . As previously mentioned, the key idea is to recursively update \mathcal{T} so that it contains the students whose teeth form the least valued comb. To accomplish this, we iterate over the set of students following the preference list \succ_c , going up to \underline{s} .¹⁹ At Step 6 we select the student s' and we find the value $v_{s',c}$ of $T^-(s', c) := T(s', c) \setminus \{(s', c)\}$ in \mathbf{x}^* (Step 7). If the list \mathcal{T} does not contain a sufficient number of students to build a comb (*i.e.*, $q_c + t_c^*$), we introduce s' in the list \mathcal{T} while respecting a descending ordering of the elements of \mathcal{T} according to $v_{s',c}$ (Step 8). Moreover, if we have obtained a set \mathcal{T} of cardinality equal to the capacity of school c then, we create the first comb C at Step 11. Once we have obtained the first comb, at Step 13 we define s^* as the student in \mathcal{T} with the highest value $T^-(s^*, c)$. At Step 14, we compare the value of $T^-(s', c)$ with the value of $T^-(s^*, c)$. If the former value is smaller, then it is worth pursuing the search for a comb valued less than C with a tooth based in (s', c) ; we build such a comb C' at Step 15. If the value of C' is smaller than the value of C , then we update \mathcal{T} to include s' at the place of s^* (Step 17 and Step 18) and we update C as C' (Step 19). At the end of the inner **for** cycle, we obtain the least valued comb C of school c . If C has a value in \mathbf{x}^* smaller than $q_c + t_c$, then the stability condition is violated for school c . Hence, at Step 21, we add the violated comb C to the set \mathcal{J}' . In Appendix C.4.2, we exemplify the application of Algorithm 4.

Note that Algorithm 4 resembles the one introduced in Baïou and Balinski (2000).²⁰ However, there is one key difference: We begin the search of the most violated comb at the head of the preference list of the school

¹⁹ Note that, since school c is fully-subscribed, we know that there are $q_c + t_c^*$ students $s' \succ_c \underline{s}$ assigned to c , and thus the comb constraint is always satisfied.

²⁰ The separation by Baïou and Balinski (2000) runs in $\mathcal{O}(m \cdot n^2)$, where m is the number of schools and n is the number of students, and is valid to separate fractional solutions when capacities are fixed (*i.e.*, \mathbf{y}^* is binary).

Algorithm 4 Separation method

Input: An instance $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} + \mathbf{t}^* \rangle$ and a (fractional) matching \mathbf{x}^* .

Output: A non-empty set \mathcal{J}' of constraints (5b) violated by $(\mathbf{x}^*, \mathbf{y}^*)$, if it exists.

```

1:  $\mathcal{J}' \leftarrow \emptyset$                                  $\triangleright$  (empty set of constraints)
2: for  $c \in \text{block}(\mathbf{x}^*)$  do
3:    $\mathcal{T} \leftarrow \emptyset$                                  $\triangleright$  (empty list of teeths)
4:    $\underline{s} \leftarrow \text{least preferred student in } c \text{ such that } x_{s,c}^* > 0$        $\triangleright$  (last student in excess)
5:    $\succ_c [\underline{s}] \leftarrow \text{preference list of } c \text{ until } \underline{s}$ 
6:   for  $s' \in \succ_c [\underline{s}]$  do                          $\triangleright$  (in order following  $\succ_c$ )
7:      $v_{s',c} \leftarrow \sum_{c' \succ_c} x_{s',c'}^*$             $\triangleright$  (value of  $T^-(s', c)$ )
8:     if  $|\mathcal{T}| < q_c + t_c^*$  then
9:        $\mathcal{T} \leftarrow \{s'\}$                                  $\triangleright$  ( $s'$  enters  $\mathcal{T}$  in descending order according to  $v_{s,c}$ )
10:    if  $|\mathcal{T}| = q_c + t_c^*$  then
11:       $C \leftarrow \{(s, c) : s \in \mathcal{T}\} \cup \bigcup_{s \in \mathcal{T}} T^-(s, c)$      $\triangleright$  (initial comb, made of shaft and selected teeths)
12:    else
13:       $s^* \leftarrow \text{first student in } \mathcal{T}$            $\triangleright$  (student  $s \in \mathcal{T}$  such that  $T^-(s, c)$  has the highest value)
14:      if  $v_{s',c} < v_{s^*,c}$  then
15:         $C' \leftarrow S^{t_c^*}(s', c) \cup \bigcup_{s \in \mathcal{T} \setminus \{s^*\}} T^-(s, c) \cup T^-(s', c)$ 
16:        if  $\sum_{(s,h) \in C'} x_{s,h}^* < \sum_{(s,h) \in C} x_{s,h}^*$  then
17:           $\mathcal{T} \setminus \{s^*\}$                                  $\triangleright$  ( $s^*$  is removed from  $\mathcal{T}$ )
18:           $\mathcal{T} \leftarrow \{s'\}$                                  $\triangleright$  ( $s'$  enters  $\mathcal{T}$  in descending order according to  $v_{s,c}$ )
19:           $C \leftarrow C'$ 
20:        if  $\sum_{(s,h) \in C} x_{s,h}^* < q_c + t_c^*$  then
21:           $\mathcal{J}' \leftarrow \mathcal{J}' \cup \{C\}$ 
22: return  $\mathcal{J}'$ 

```

rather than at the tail (as it is done in Baïou and Balinski (2000)), which guarantees that our method finds the most violated comb constraint.²¹

THEOREM 3. *Algorithm 4 finds the combs solving Step 5 in Algorithm 1 for every school in $\mathcal{O}(m \cdot n \cdot \bar{q})$ time, where n is the number of students, m is the number of schools and $\bar{q} = \max_{c \in \mathcal{C}} \{q_c + t_c^*\}$. Moreover, Algorithm 4 returns a set of combs such that the corresponding constraints (5b) are violated by $(\mathbf{x}^*, \mathbf{y}^*)$, if \mathbf{x}^* is not stable for $\Gamma_{\mathbf{t}^*}$.*

²¹In Appendix G.2 we show that the separation algorithm by Baïou and Balinski (2000) may not find the most violated constraint.

REMARK 4. Algorithm 4 can be easily adapted to separate the comb constraints with any linear objective function. Indeed, it is sufficient to iterate over the whole set of schools at Step 2, remove Steps 4 and 5, and iterate over the whole preference list \succ_c at Step 6. However, this generalization deteriorates the running-time of the separation algorithm.

Proof of Theorem 3. The proof is based on the validity of the following statement, that we prove by induction: For every school $c \in \text{block}(\mathbf{x}^*)$, Steps 3-19 of Algorithm 4 provide the comb solving the problem in Step 5 in Algorithm 1 in $\mathcal{O}(n \cdot \bar{q})$. First, let us observe that the search for the least valued comb in $\mathbf{C}_c(t_c^*)$ terminates in a finite number of steps because the preference list \succ_c is finite. On the other hand, at the beginning of every cycle in Step 6, we select a student s' and by the end of the $(q_c + t_c^*)$ -th iteration, we claim that the comb C is the smallest valued comb in $\mathbf{C}_c(t_c^*)$. If we prove the main statement presented above integrated with the two previous observations, we prove that the algorithm solves Step 5 in Algorithm 1 for c . Note that at the first $q_c + t_c^* - 1$ steps, a comb cannot be defined; indeed, a comb needs $q_c + t_c - 1$ teeth and a base with its tooth. The proof follows an inductive argument:

Base: At the $(q_c + t_c^*)$ -th iteration, we create comb C at Step 11. The base of comb C is the student s' that is ranked $q_c + t_c^*$ by c , and the bases of the other $q_c + t_c^* - 1$ teeth are the students preferred more than s' . Since this comb is the only possible that we can create with a base preferred more or equal than s' , it is also the least valued comb possible.

Inductive step: The inductive hypothesis states that at the end of iteration $\ell \geq q_c + t_c^*$, after selecting student s' , the selected comb C is the comb with the lowest value among all combs with a base student preferred equal to or more than s' . We want to prove that at the end of step $\ell + 1$, the algorithm selects the comb with lowest value among all combs with a base student ranked less or equal than $(\ell + 1)$ -th one. Let s' be the student ranked $\ell + 1$ and let s^* be the student in \mathcal{T} with the highest valued T^- . We need to verify if adding $T^-(s', c)$ to \mathcal{T} may produce a comb of smaller value than the one recorded. If $T^-(s', c) \geq T^-(s^*, c)$, then adding s' in \mathcal{T} would produce a comb with equal or higher value. Otherwise, $T^-(s', c) < T^-(s^*, c)$, and adding s' in \mathcal{T} may provide a better comb. If $T^-(s', c) < T^-(s^*, c)$, then we create a new comb C' with base in (s', c) and teeth in $\mathcal{T} \setminus \{s^*\}$. Any other choice of teeth is sub-optimal for the base (s', c) since they are the least valued $q_c + t_c^*$ teeth among those in the first $\ell + 1$ positions. If the new comb C' has value smaller than the comb C , then we have found the best comb among those with base ranked at most $\ell + 1$.

To establish the correctness of Algorithm 4, it remains to show that it finds the violated combs (if any) that cut-off the input solution. First, the algorithm terminates in a finite number of steps because $|\mathcal{C}| = m$ and $|\succ_c| \leq |\mathcal{S}| = n$. Second, by the previous inductive argument, we have found the least valued comb for each school.

Running-time analysis. The inner **for** loop (Step 6) takes at most n rounds and the most expensive computation within it is placing s' in \mathcal{T} to maintain the list \mathcal{T} in descending order according to the values $v_{s,c}$. Note that this task can be performed by comparing s' with at most each element in \mathcal{T} , i.e., in a number of operations equal to $|\mathcal{T}| = q_c + t_c^*$. Hence, the number of elementary operations for finding the most violated comb of school c is at most $\mathcal{O}(n \cdot (q_c + t_c^*))$. Hence, Algorithm 4 can take at most $\mathcal{O}(m \cdot n \cdot \bar{q})$ time, where $\bar{q} = \max_{c \in \text{block}(\mathbf{x}^*)} \{q_c + t_c^*\}$ and $m = |\mathcal{C}|$. \square

C.4. Missing Examples in Appendix C

C.4.1. Difference in the Sets of Fully-subscribed Schools. Let $\mathcal{S} = \{s_1, s_2, s_3\}$ be the set of students and let $\mathcal{C} = \{c_1, c_2, c_3, c_4\}$ be set of schools; every school has capacity 1. The preferences of the students are $s_1 : c_1 \succ c_4 \succ c_3 \succ c_2$; $s_2 : c_1 \succ c_2 \succ c_3 \succ c_4$; $s_3 : c_4 \succ c_1 \succ c_2 \succ c_3$, and the preferences of the schools are $c_1 : s_2 \succ s_1 \succ s_3$; $c_2 : s_2 \succ s_1 \succ s_3$; $c_3 : s_1 \succ s_2 \succ s_3$; $c_4 : s_3 \succ s_1 \succ s_2$. The optimal matching with no stability constraints (*i.e.*, $\mathcal{J} = \emptyset$) and budget $B = 0$, is $x_{s_1, c_1}^* = 1$, $x_{s_2, c_2}^* = 1$, $x_{s_3, c_4}^* = 1$ and all other entries equal to zero. On the other side, the student-optimal stable matching is $\mu^* = \{(s_2, c_1), (s_1, c_3), (s_3, c_4)\}$. Note that the set of fully-subscribed schools is mutually not-inclusive.

C.4.2. Separation Algorithm: Example. Let $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} \rangle$ be the instance of the school choice problem with $\mathcal{S} = \{s_1, s_2, s_3, s_4, s_5\}$ as the set of students and $\mathcal{C} = \{c_1, c_2, c_3, c_4, c_5, c_6\}$ as the set of schools; every school has capacity 1. The preferences of the students are $s_1 : c_1 \succ c_4 \succ c_3 \succ c_2 \succ \dots$; $s_2 : c_1 \succ c_2 \succ c_3 \succ c_4 \succ \dots$; $s_3 : c_4 \succ c_1 \succ c_5 \succ c_6 \succ \dots$; $s_4 : c_5 \succ c_1 \succ c_4 \succ c_6 \succ \dots$; $s_5 : c_4 \succ c_5 \succ c_6 \succ c_1 \succ \dots$, and the preferences of the schools are $c_1 : s_2 \succ s_1 \succ s_3 \succ \dots$; $c_2 : s_2 \succ s_1 \succ s_3 \succ \dots$; $c_3 : s_1 \succ s_2 \succ s_3 \succ \dots$; $c_4 : s_3 \succ s_4 \succ s_5 \succ \dots$; $c_5 : s_4 \succ s_3 \succ s_5 \succ \dots$; $c_6 : s_5 \succ \dots$. The “ \dots ” at the end of a preference list represent any possible strict ranking of the remaining agents on the other side of the bipartition. The optimal solution of the main program with no stability constraints (*i.e.*, $\mathcal{J} = \emptyset$) and budget $B = 1$, is $(\mathbf{x}^*, \mathbf{t}^*)$ with $x_{s_1, c_1}^* = 1$, $x_{s_2, c_2}^* = 1$, $x_{s_3, c_4}^* = 1$, $x_{s_5, c_4}^* = 1$, $x_{s_4, c_5}^* = 1$, $t_4^* = 1$ and all other entries equal to zero. We take the instance $\Gamma_{\mathbf{t}^*}$ (with the expanded capacities in accordance to \mathbf{t}^*) and the matching \mathbf{x}^* as the input for Algorithm 4. At Step 1 we initialize the set of violated comb constraints \mathcal{J}' to \emptyset . In order to proceed, we need to compute the set $\text{block}(\mathbf{x}^*)$. First, note that the student-optimal stable matching of $\Gamma_{\mathbf{t}^*}$ is $\mu = \{(s_2, c_1), (s_1, c_3), (s_3, c_4), (s_5, c_4), (s_4, c_5)\}$. Therefore, the set of schools that are fully subscribed in both μ and \mathbf{x}^* is $\{c_1, c_4, c_5\}$. The only school that is fully subscribed in both μ and \mathbf{x}^* that has an exceeding student is c_1 . Hence, at Step 2, we select c_1 . We initialize the empty list \mathcal{T} , and we select the least preferred student enrolled in c_1 , which is $\underline{s} = s_1$. Note that $\succ_{c_1}[s_1] = [s_2, s_1]$. The inner loop selects s_2 as the most preferred student. At Step 7 we compute the value of $T^-(s_2, c_1)$, which is 0 (we set to 0 the value of an empty T^-). Then, at Step 9, we include s_2 in the teeth list, *i.e.*, $\mathcal{T} = \{s_2\}$, which is composed of only one student because 1 is the capacity of school c_1 in $\Gamma_{\mathbf{t}^*}$. Therefore, we can find the comb C with base (s_2, c_1) at Step 11. At the next iteration, we have that $s' = s_1$ and we find that s^* is necessarily s_2 . The initial comb C built at the previous iteration is composed only by the tooth of (s_2, c_1) , *i.e.*, $C = T(s_2, c_1) = \{(s_2, c_1)\}$. The value of C in \mathbf{x}^* is 0. Since the value of $T^-(s_1, c_1)$ is 0, the condition at Step 14 is false, meaning that it is not worth pursuing a comb built with basis (s_1, c_1) . Therefore, the algorithm jumps to Step 20, where it finds that the condition is satisfied since the value of C is 0 and the capacity of c_1 is 1. Hence, at Step 21, we add C to the set of cuts to be added to the main program. Note that at this point Algorithm 4 terminates and returns to the main program the set of cuts \mathcal{J}' containing only the comb based in (s_2, c_1) . Interestingly, the next optimal solution of the main program is the optimal solution of the problem.

Appendix D: Appendix to Section 5

D.1. Hybrid Approach

Our hybrid approach combines in a mathematical program the compact formulation with L-constraints and the generalized comb formulation. The idea is that for each $(s, c) \in \mathcal{E}$, we must guarantee that it is not a blocking pair, and this can be done either through comb or L-constraints. To be more precise, in Algorithm 1, we can enforce stability by introducing some comb constraints and L-constraints in the main program (Step 1), and then by separating comb constraints (Steps 5 and 7) at later iterations. We consider the following family of hybrid approaches parameterized by $\gamma \in [0, 1]$, whereby for each school $c \in \mathcal{C}$:

- (i) we separate only the comb constraints (5b) for those combs with base (s, c) where s has a priority in c of at least $q_c + \lfloor \gamma \cdot B + 0.5 \rfloor$; and
- (ii) we add the L-constraint (2d) for (s, c) for every student s with a priority less than $q_c + \lfloor \gamma \cdot B + 0.5 \rfloor$.

Note that when $q_c + B < |\{s \in \mathcal{S} : c \succ_s \emptyset\}|$, i.e., the budget is not enough to cover for all students applying to c , then some pairs (s, c) will have L-constraints even with $\gamma = 1$.

In Figure 7, we compare the computational time needed to solve the instance for the hybrid formulation considering $\gamma \in \{0, 0.25, 0.5\}$ for $B \in \{10, 25\}$. On the one hand, we observe that $\gamma = 0.25$ leads to the best results, although the differences are relatively small when $B = 10$. On the other hand, when $B = 25$, we observe large differences when the penalty is low, with $\gamma = 0$ leading to the shortest computational time. Given these results, and to simplify the exposition and implementation, we decided to use $\gamma = 0$ to obtain the results reported in Section 5.3.

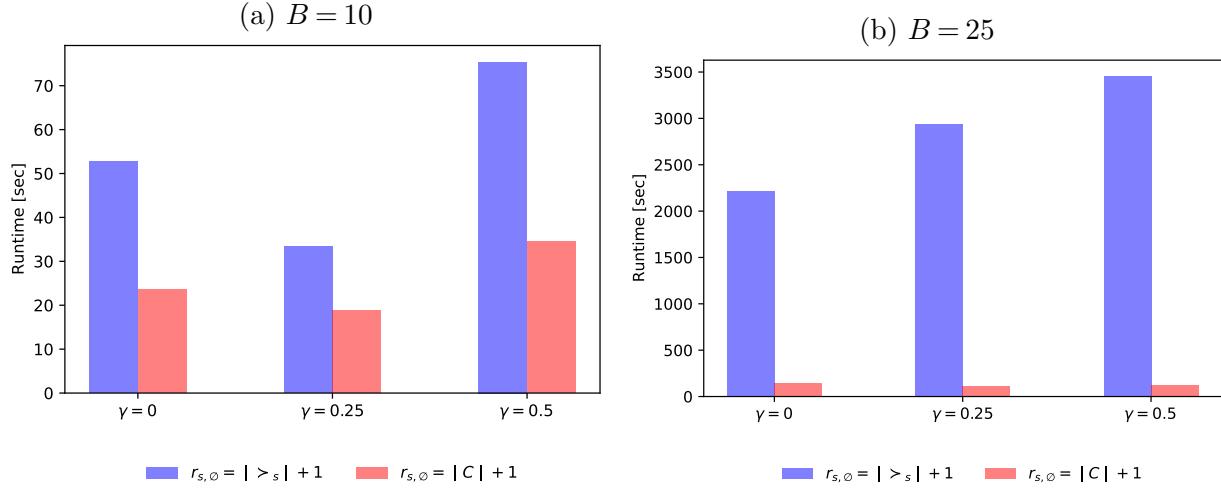


Figure 7 Sensitivity to γ

D.2. Initialization Details and Acceleration of the Cutting-plane Method

Many natural accelerations techniques can be added to the cutting plane method described in Algorithm 1 as well as its modified version, the hybrid approach.

- We implement the cutting-plane method as a branch-and-cut, incorporating the constraints generated by the separation algorithm as lazy constraints in Gurobi.
- We use the solution obtained from using *LPH* as a warm-start. This allows the branch-and-cut to have immediately an upper bound, which may enable to prune earlier certain branches.
- In Step 1, we add no comb constraints, *i.e.*, $\mathcal{J} = \emptyset$. However, we add several valid constraints to the initial main program (Step 2) that we derive from the student-optimal stable matching μ_0 , obtained with no additional budget. First, we know that each student s must get an assignment they weakly prefer over $\mu_0(s)$ when $B > 0$. Hence, we introduce matching variables only among those pairs (s, c) where c is a school at least as preferred as the school to which student s is matched in μ_0 . Second, the first equality constraints in (6) were implemented as type 1 SOS constraints in Gurobi and the decision vector \mathbf{x} forced to be binary. Although the latter is not necessary for the correctness of the method (in the worst-case, all comb constraints are added), we observed that Gurobi was able to leverage on this element and obtain faster execution times. Third, we set $x_{s,c} = 1$ for all $(s, c) \in \mu_0$ such that $r_{s,c} = 1$. Finally, we know that any school c such that $|\mu(c)| < q_c$ should receive no additional seats when $B > 0$. Hence, we set all $y_c^k = 0$ for all such schools. For the hybrid approach, additional constraints are added to the main program (see Appendix D.1).

Appendix E: Appendix to Section 6

E.1. Cardinality

A crucial aspect of the solutions of Formulation 1 is that they largely depend on the value of $r_{s,\emptyset}$, *i.e.*, the *penalty* for having unassigned students. We emphasize that $r_{s,\emptyset}$ does not indicate the position of \emptyset in the ranking of s over schools, but a penalty value for being unassigned. One could expect that for greater penalty values, the optimal solution will prioritize *access* by matching initially unassigned students and increasing the cardinality of the match. In contrast, for smaller penalty values, the focus will be on the *improvement* of previously assigned students by prioritizing *chains of improvement* that result in multiple students obtaining a better assignment than the initial matching with no extra capacities. In Theorems 4 and 5 we formalize this intuition.

THEOREM 4. *There is a sufficiently small and finite $r_{s,\emptyset}$ for all $s \in \mathcal{S}$ such that the optimal solution of Formulation (1) returns a minimum cardinality student-optimal stable matching.*

Proof. Let μ^0 be the student-optimal assignment when there is no budget, and let $M(\mu) = \{s \in \mathcal{S} : \mu(s) \in \mathcal{C}\}$ be the set of students assigned in match μ . In addition, let $r_{s,\emptyset} = r_\emptyset = |\mathcal{S}| \cdot (1 - \xi)$, where $\xi = \max_{s \in \mathcal{S}} \{|\succ_s|\}$ is the maximum length of a list of preferences among all students.

Let μ^* be the optimal allocation implied by the optimal solution of the problem $(\mathbf{x}^*, \mathbf{t}^*)$ considering a budget B and the aforementioned penalties. To find a contradiction, suppose that there exists an alternative match μ' that uses the entire budget B and that satisfies $|M(\mu^*)| > |M(\mu')|$. Without loss of generality, suppose that $|M(\mu^*)| = |M(\mu')| + 1$. We know that $M(\mu^0) \subseteq M(\mu^*)$ and $M(\mu^0) \subseteq M(\mu')$, since no student who was initially assigned can result unassigned when capacities are expanded. Then, the difference in objective function between μ^* and μ' can be written as:

$$\begin{aligned}
 \Delta &= \sum_{(s,c) \in \mu^*} r_{s,c} - \sum_{(s,c) \in \mu'} r_{s,c} = \sum_{s \in (M(\mu^*) \cap M(\mu'))} r_{s,\mu^*(s)} - r_{s,\mu'(s)} \\
 &\quad + \sum_{s \in M(\mu^*) \setminus M(\mu')} r_{s,\mu^*(s)} - r_{s,\emptyset} + \sum_{s \in M(\mu') \setminus M(\mu^*)} r_{s,\emptyset} - r_{s,\mu'(s)} \\
 &\quad + \sum_{s \in S \setminus (M(\mu^*) \cup M(\mu'))} r_{s,\emptyset} - r_{s,\emptyset} \\
 &= \sum_{s \in (M(\mu^*) \cap M(\mu'))} r_{s,\mu^*(s)} - r_{s,\mu'(s)} \\
 &\quad + \sum_{s \in M(\mu^*) \setminus M(\mu')} r_{s,\mu^*(s)} - r_{s,\emptyset} + \sum_{s \in M(\mu') \setminus M(\mu^*)} r_{s,\emptyset} - r_{s,\mu'(s)} \\
 &\geq \sum_{s \in (M(\mu^*) \cap M(\mu'))} (1 - |\succ|_s) \\
 &\quad + \sum_{s \in M(\mu^*) \setminus M(\mu')} 1 - r_{s,\emptyset} + \sum_{s \in M(\mu') \setminus M(\mu^*)} r_{s,\emptyset} - |\succ_s| \\
 &\geq |(M(\mu^*) \cap M(\mu'))| \cdot (1 - \xi) \\
 &\quad + |M(\mu^*) \setminus M(\mu')| \cdot (1 - r_\emptyset) + |M(\mu') \setminus M(\mu^*)| \cdot (r_\emptyset - \xi) \\
 &= |(M(\mu^*) \cap M(\mu'))| \cdot (1 - \xi) + (|M(\mu') \setminus M(\mu^*)| + 1) \cdot (1 - r_\emptyset) \\
 &\quad + |M(\mu') \setminus M(\mu^*)| \cdot (r_\emptyset - \xi) \\
 &\geq |(M(\mu^*) \cap M(\mu'))| \cdot (1 - \xi) + (|M(\mu') \setminus M(\mu^*)|) \cdot (1 - \xi) - r_\emptyset \\
 &= (|M(\mu^*) \cap M(\mu')| + |M(\mu') \setminus M(\mu^*)|) \cdot (1 - \xi) - r_\emptyset \\
 &= (|M(\mu^*) \cap M(\mu')| + |M(\mu') \setminus M(\mu^*)|) \cdot (1 - \xi) - |\mathcal{S}| \cdot (1 - \xi) \\
 &= (|M(\mu^*) \cap M(\mu')| + |M(\mu') \setminus M(\mu^*)| - |\mathcal{S}|) \cdot (1 - \xi) \\
 &\geq 0,
 \end{aligned} \tag{10}$$

since both terms are negative as $|M(\mu^*) \cap M(\mu')| + |M(\mu') \setminus M(\mu^*)| \leq |S|$ and $1 \leq \xi$. Hence, we obtain that $\Delta \geq 0$, which implies that the objective function evaluated at μ' is strictly less than the objective function evaluated at μ^* , which contradicts the optimality of μ^* . Finally, note that this derivation holds for any set of penalties such that $r_{s,\emptyset} \leq r_\emptyset$, and it also holds for the case when the clearinghouse can decide not to allocate all the budget (as long as this condition applies for both μ^* and μ'), so we conclude. \square

Note that the penalty used in the proof is negative. A negative penalty can be interpreted as the policy-maker willingness to allocate extra capacities to improve the assignment of students already in the system, for example, when merit scholarships are awarded to students who already have a secured position.

By the stability constraints, every student that is initially assigned (i.e., when $B = 0$) should be weakly better off when capacities are expanded. Hence, one may think that the minimum cardinality student-optimal stable matching is the one that is optimal for this subset of students. However, as we show in the example in Appendix E.1.1 below, this is not the case.

On the other hand, in Theorem 5, we show that if the penalty values are sufficiently high, then the optimal solution will prioritize *access* by obtaining a student-optimal stable matching of maximum cardinality.

THEOREM 5. *There is a sufficiently high and finite $r_{s,\emptyset}$ for all $s \in \mathcal{S}$ such that the optimal solution of Formulation (1) returns a maximum cardinality student-optimal stable-matching.*

Proof. Let μ^* be the stable-matching corresponding to the solution $(\mathbf{x}^*, \mathbf{t}^*)$. To find a contradiction, suppose there exists another stable matching μ' that has a higher cardinality, i.e., $|s \in \mathcal{S} : \mu'(s) = \emptyset| < |s \in \mathcal{S} : \mu^*(s) = \emptyset|$; we also assume that $r_{s,\emptyset} = \bar{r}$ for all $s \in \mathcal{S}$, where $\bar{r} > \sum_{s \in \mathcal{S}} |\succ_s|$. By optimality of $(\mathbf{x}^*, \mathbf{t}^*)$, we know that

$$\sum_{s \in \mathcal{S}} r_{s,\mu^*(s)} < \sum_{s \in \mathcal{S}} r_{s,\mu'(s)}.$$

On the other hand, we know that

$$\begin{aligned} \sum_{s \in \mathcal{S}} r_{s,\mu^*(s)} - \sum_{s \in \mathcal{S}} r_{s,\mu'(s)} &= \sum_{s \in \mathcal{S} : \mu^*(s) \in \mathcal{C}} r_{s,\mu^*(s)} - \sum_{s \in \mathcal{S} : \mu'(s) \in \mathcal{C}} r_{s,\mu'(s)} + \sum_{s \in \mathcal{S} : \mu^*(s) = \emptyset} r_{s,\emptyset} - \sum_{s \in \mathcal{S} : \mu'(s) = \emptyset} r_{s,\emptyset} \\ &= \sum_{s \in \mathcal{S} : \mu^*(s) \in \mathcal{C}} r_{s,\mu^*(s)} - \sum_{s \in \mathcal{S} : \mu'(s) \in \mathcal{C}} r_{s,\mu'(s)} \\ &\quad + \bar{r} \cdot [|s \in \mathcal{S} : \mu^*(s) = \emptyset| - |s \in \mathcal{S} : \mu'(s) = \emptyset|] \\ &> - \sum_{s \in \mathcal{S}} |\succ_s| + \bar{r} \cdot [|s \in \mathcal{S} : \mu^*(s) = \emptyset| - |s \in \mathcal{S} : \mu'(s) = \emptyset|] \\ &> 0. \end{aligned} \tag{11}$$

The first equality follows from $r_{s,\emptyset} = \bar{r}$ for all $s \in \mathcal{S}$. The first inequality follows from the fact that, given a student s that is assigned, the maximum improvement is to move from their last preference, $|\succ_s|$, to their top preference, and therefore $r_{s,\mu^*(s)} - r_{s,\mu'(s)} > 1 - |\succ_s| > -|\succ_s|$. Then, we have that $\sum_{s \in \mathcal{S} : \mu^*(s) \in \mathcal{C}} r_{s,\mu^*(s)} - \sum_{s \in \mathcal{S} : \mu'(s) \in \mathcal{C}} r_{s,\mu'(s)} \geq -\sum_{s \in \mathcal{S}} |\succ_s|$. Finally, the last inequality follows from the fact that

$$|s \in \mathcal{S} : \mu^*(s) = \emptyset| - |s \in \mathcal{S} : \mu'(s) = \emptyset| \geq 1$$

and that \bar{r} is arbitrarily high. As a result, we obtain that

$$\sum_{s \in \mathcal{S}} r_{s,\mu^*(s)} - \sum_{s \in \mathcal{S}} r_{s,\mu'(s)} > 0,$$

which contradicts the optimality of $(\mathbf{x}^*, \mathbf{t}^*)$. \square

E.1.1. Example: Minimum Cardinality Suboptimality. Suppose there are $n + 2$ students and n schools, each with capacity $q_c = 1$, and a total budget $B = 2$. For each student s_k with $k \in \{4, \dots, n\}$, we assume that their preferences are given by $c_k \succ_{s_k} \emptyset \succ_{s_k} \dots$, and we assume that the priorities at school c_k are $s_k \succ_{c_k} \emptyset \succ_{c_k} \dots$, where the $\succ \dots$ represent an arbitrary ordering of the missing agents after \emptyset . In addition, we assume that the preferences and priorities of the other agents are:

$$\begin{array}{ll} s_1 : c_1 \succ \emptyset \succ \dots & c_1 : s_1 \succ s'_2 \succ s_2 \succ \emptyset \succ \dots \\ s_2 : c_1 \succ c_3 \succ c_2 \succ \emptyset \succ \dots & c_2 : s_2 \succ s'_2 \succ s'_3 \succ \emptyset \succ \dots \\ s_3 : c_3 \succ c_2 \succ \emptyset \succ \dots & c_3 : s_3 \succ s'_3 \succ s_2 \succ \emptyset \succ \dots \\ s'_2 : c_n \succ \dots \succ c_4 \succ c_1 \succ \emptyset \succ \dots & \\ s'_3 : c_3 \succ \emptyset \succ \dots & \end{array}$$

In this case, the only option is to allocate the budget between schools c_1 and c_3 , since allocating the budget to the other schools (c_2 or c_k for $k \geq 4$) would have no effect. If $B = 0$, then the student-optimal stable matching is $\mu^0 = \{(s_1, c_1), (s_2, c_2), (s_3, c_3), (s'_2, \emptyset), (s'_3, \emptyset)\} \cup \{(s_k, c_k)\}_{k \geq 4}$. Note that the optimal allocation for the set of students initially matched (when $B = 0$) is to allocate the two extra seats to school c_1 .

- If $t_{c_1} = 2$, then both students s_2 and s'_2 get assigned to c_1 , and thus the change among initially assigned students is -2 (as student s_2 moves from their third to their top preference).
- If $t_{c_1} = t_{c_3} = 1$, then student s'_2 and s_2 get assigned to c_1 and c_3 , respectively. Hence, the change among initially assigned students is -1 (as student s_2 moves from their third to their second preference).
- If $t_{c_3} = 2$, then s_2 and s'_3 get assigned to c_3 . Hence, the change among initially assigned students is -1 (as student s_2 moves from their third to their second preference).

Finally, if $r_{s,\emptyset} = 0$ for every student s , it is easy to see that an optimal solution is to assign both additional seats to school c_3 , as the change in the objective function would be 0; in the other two cases analysed above (i.e., at least one extra capacity is allocated to c_1), the change in the objective function would be at least $n - 2$ as student s'_2 would be assigned to c_1 (c_1 is the $n - 2$ -th preferred school of student s'_2). Hence, the optimal assignment is not the one that benefits the most those students initially assigned in μ^0 . Note that another optimal solution is to simply not assign any of the additional seats, but this case is not interesting as it would lead to no gains from the budget.

E.2. Incentives

It is direct from (Roth and Sotomayor 1990) that our mechanism is not strategy-proof for schools, as we know that it is not in the base case with $B = 0$. In Proposition 1, we show that if the budget is positive and students know about it, then the mechanism that assigns students to schools and jointly allocates extra capacities is not strategy-proof for students.

PROPOSITION 1. *The mechanism is not strategy-proof for students.*

Proof. Consider an instance with five students and five schools, each with capacity one. In addition, suppose that $B = 1$ and that the preferences and priorities are:

$$\begin{array}{ll} s_1 : c_1 \succ \dots & c_1 : s_1 \succ s_3 \succ \dots \\ s_2 : c_2 \succ \dots & c_2 : s_2 \succ s_3 \succ \dots \\ s_3 : c_1 \succ c_2 \succ c_3 \succ \dots & c_3 : s_3 \succ \dots \\ s'_1 : c'_1 \succ \dots & c'_1 : s'_1 \succ s'_2 \succ \dots \\ s'_2 : c'_1 \succ c'_2 \succ \dots & c'_2 : s'_2 \succ \dots \end{array}$$

where the “...” represent an arbitrary completion of the preferences. If agents are truthful, the optimal allocation is to assign the extra seat to school c_1 , which will admit student s_3 ; thus, the final matching would be $\{(s_1, c_1), (s_2, c_2), (s_3, c_1), (s'_1, c'_1), (s'_2, c'_2)\}$. If student s'_2 misreports her preferences by reporting

$$s'_2 : c'_1 \succ c_1 \succ c_2 \succ c'_2 \succ c_3,$$

the extra seat would be allocated to c'_1 and s'_2 would get her favorite school. \square

Despite this negative result, in the next proposition we show that our mechanism is strategy-proof in the large.

PROPOSITION 2. *The mechanism is strategy-proof in the large.*

Proof. In an extension of their Theorem 1, Azevedo and Budish (2018) show that a sufficient condition for a semi-anonymous mechanism to be strategy-proof in the large is *envy-freeness but for ties* (EF-TB), which requires that no student envies another student with a strictly worse lottery number. Hence, it is enough to show that our mechanism satisfies these two properties, i.e., semi-anonymity and EF-TB, for showing that our mechanism is strategy-proof in the large. Next, we provide definitions for semi-anonymity and for EF-TB, and prove for each property that our mechanism satisfies it.

Semi-anonymity. As defined in Azevedo and Budish (2018), a mechanism is semi-anonymous if the set of students can be partitioned in a set of groups G . Within each group $g \in G$, each student belongs to a type t ; we denote by T_g the finite set of types that limits the actions of the students to the space A_g . In our school choice setting, the groups are the set of students belonging to the same priority group (e.g., students with siblings, students with parents working at the school, etc.), the types are defined by the students' preferences \succ_s , and the actions are the lists of preferences that students can submit. Then, two students s and s' that belong to the same group g and share the same type $t \in T_g$ have exactly the same preferences and priorities and differ only their specific position in the schools' lists, which can be captured through their lottery numbers $l_s, l_{s'} \in [0, 1]$.²² Note that G is finite because the number of priority groups is finite in most applications.²³ Moreover, since the number of schools is finite, we know that the number of possible preference lists \succ_s is finite and, thus, the number of types within each group is finite. Hence, we conclude that our mechanism is semi-anonymous.

EF-TB. Given a market with n students, a direct mechanism is a function $\Phi^n : T^n \rightarrow \Delta(\mathcal{C} \cup \{\emptyset\})^n$ that receives a vector of types T (the application list of each student) and returns a (potentially randomized) feasible allocation. In addition, let $u_t(\tilde{c})$ be the utility that a student with type $t \in T_g, g \in G$ gets from the lottery over assignments $\tilde{c} \in \Delta(\mathcal{C} \cup \{\emptyset\})$ (note that, by assumption, two students belonging to the same type have exactly the same preferences and, thus, get the same utility in each school $c \in \mathcal{C} \cup \{\emptyset\}$). Then, a semi-anonymous mechanism is envy-free but for tie-breaking if for each n there exists a function $x^n : (T \times [0, 1])^n \rightarrow \Delta(\mathcal{C} \cup \{\emptyset\})^n$ such that

$$\Phi^n(\mathbf{t}) = \int_{\mathbf{l} \in [0, 1]^n} x^n(\mathbf{t}, \mathbf{l}), d\mathbf{l}$$

and, for all i, j, n, \mathbf{t} and \mathbf{l} with $l_i \geq l_j$, and if t_i and t_j belong to the same type, then

$$u_{t_i}[x_i^n(\mathbf{t}, \mathbf{l})] \geq u_{t_i}[x_j^n(\mathbf{t}, \mathbf{l})].$$

²²In other words, any ordering of students \succ_c in a given school c can be captured by $s \succ_c s' \Leftrightarrow l_s > l_{s'}$ for any two students s, s' belonging to the same group $g \in G$. Note that, for simplicity, we are implicitly assuming that ties within a group are broken using a single tie-breaker; this argument can be extended to multiple tie-breaking.

²³In the Chilean school choice setting, there are 5 priority groups: Students with siblings, students with working parents at the school, students who are former students, regular students and disadvantaged students.

In words, to show that our mechanism is EF-TB we need to show that whenever two students belong to the same group and one of them has a higher lottery, then the assignment of the latter cannot be worse than that of the former. This follows directly from the stability constraints, since for any budget allocation, we know that the resulting assignment must be stable. As a result, for any budget allocation, we know that given two students s, s' that belong to the same type, the resulting assignment μ satisfies $\mu(s) \succ_s \mu(s')$ if $s \succ_c s'$ for all $c \in \mathcal{C}$. Then, it is direct that $u_{t_s}[x_s^n(\mathbf{t}, \mathbf{l})] \geq u_{t_s}[x_{s'}^n(\mathbf{t}, \mathbf{l})]$, for whatever function x that captures our mechanism. Thus, we conclude that our mechanism is semi-anonymous and EF-TB, and therefore it is strategy-proof in the large. \square

E.3. Monotonicity

Suppose there are n students and n schools, each with capacity $q_c = 1$, and a total budget $B = 1$. For each student s_k with $k \in \{5, \dots, n\}$, we assume that their preferences are given by $c_{k-1} \succ_{s_k} c_k \succ_{s_k} \emptyset$, and we assume that the priorities at school c_k are $s_4 \succ_{c_k} \dots \succ_{c_k} s_n \succ_{c_k} \emptyset$ for all $k \geq 4$. In addition, we assume that the preferences and priorities of the other agents are:

$$\begin{array}{ll} s_1 : c_1 \succ c_3 \succ \emptyset & c_1 : s_1 \succ s_3 \succ \emptyset \\ s_2 : c_2 \succ c_3 \succ \emptyset & c_2 : s_2 \succ s_3 \succ \emptyset \\ s_3 : c_1 \succ c_2 \succ \emptyset & c_3 : s_1 \succ s_2 \succ \emptyset \\ s_4 : c_4 \succ \emptyset. & \end{array}$$

It is easy to see that (when $B = 0$) $\mu^0 = \{(s_1, c_1), (s_2, c_2), (s_3, \emptyset), (s_4, c_4), \dots, (s_n, c_n)\}$. In addition, if the penalty for having student s_3 is sufficiently high (specifically, $r_{s_3, \emptyset} > n - 5$), then the optimal budget allocation is to add one seat to school c_1 , so that the resulting match is $\mu^1 = \{(s_1, c_1), (s_2, c_2), (s_3, c_1), (s_4, c_4), \dots, (s_n, c_n)\}$. As a result, student s_3 is now assigned to their top preference.

Next, suppose that student s_3 improves her lottery/score in school c_2 , so that now c_2 's priorities are $c_2 : s_3 \succ s_2 \succ \emptyset$. As a result, the initial assignment is $\mu^{00} = \{(s_1, c_1), (s_2, c_3), (s_3, c_2), (s_4, c_4), \dots, (s_n, c_n)\}$. In addition, note that (omitted allocations are clearly dominated):

- If $t_{c_1} = 1$, then both s_2 and s_3 improve their assignment, and thus the change in the objective function is -2 (both students move from their second to their top preference).
- If $t_{c_2} = 1$, then only s_2 improves their assignment, and thus the change in the objective function is -1 .
- If $t_{c_4} = 1$, then a chain of improvements going from student s_5 to s_n starts, with each of these students getting assigned to their top preference. As a result, the change in the objective function is $-(n - 4)$.

As a result, if $n > 6$, it would be optimal to assign the extra seat to school c_4 , and the resulting assignment would be $\mu^{11} = \{(s_1, c_1), (s_2, c_3), (s_3, c_2), (s_4, c_4), (s_5, c_4), \dots, (s_n, c_{n-1})\}$. Note that student s_3 is worse off in μ^{11} than in μ^1 , since she is now assigned to her second preference compared to the top one when her priority in school c_2 was lower.

E.4. Complexity

In Bobbio et al. (2022), the authors analyze the complexity of Problem (1) and prove that it is NP-hard even when the preference lists of the students are complete, i.e., when students apply to all the schools. In this context, a possible approach is to design an approximation algorithm for this problem. Let us recall the definition of $f(\mathbf{t})$ for a fixed $\mathbf{t} \in \mathbb{Z}_+^c$:

$$f(\mathbf{t}) := \min_{\mu} \left\{ \sum_{(s,c) \in \mu} r_{s,c} : \mu \text{ is a stable matching in instance } \Gamma_{\mathbf{t}} \right\} \quad (13)$$

which can be evaluated in polynomial time by using the DA algorithm on instance $\Gamma_{\mathbf{t}}$. Therefore, one might be tempted to show that f is a lattice submodular function due to the existence of known approximation algorithms (see, e.g., (Soma and Yoshida 2015)). As we show in the following result, f is neither lattice submodular nor supermodular.

PROPOSITION 3. *The function $f(\mathbf{t})$ defined in Expression (13) is neither lattice submodular nor supermodular.*

Proof. Let us recall the definition of lattice submodularity. A function $f: \mathbb{Z}_+^c \rightarrow \mathbb{R}_+$ is said to be lattice submodular if $f(\mathbf{t} \vee \mathbf{t}') + f(\mathbf{t} \wedge \mathbf{t}') \leq f(\mathbf{t}) + f(\mathbf{t}')$ for any $\mathbf{t}, \mathbf{t}' \in \mathbb{Z}_+^c$, where $\mathbf{t} \vee \mathbf{t} := \max\{\mathbf{t}, \mathbf{t}'\}$ and $\mathbf{t} \wedge \mathbf{t}' := \min\{\mathbf{t}, \mathbf{t}'\}$ component-wise. Function f is lattice supermodular if, and only if, $-f$ is lattice submodular. First, we show that f is not lattice submodular. Consider a set $\mathcal{S} = \{s_1, s_2, s_3, s_4\}$ of students and a set $\mathcal{C} = \{c_1, c_2, c_3, c_4, c_5\}$ of schools. The preference lists are as follows

$$\begin{array}{ll} s_1 : c_1 \succ c_2 \succ \dots & c_1 : s_1 \succ s_2 \succ \dots \\ s_2 : c_2 \succ c_3 \succ \dots & c_2 : s_2 \succ s_3 \succ \dots \\ s_3 : c_2 \succ c_5 \succ c_4 \succ \dots & c_3 : s_2 \succ s_1 \succ \dots \\ s_4 : c_5 \succ \dots & c_4 : s_3 \succ s_4 \succ \dots \\ & c_5 : s_4 \succ s_3 \succ \dots \end{array}$$

School c_1 has capacity 0 and the other schools have capacity 1. We choose the following two allocations: $\mathbf{t} = (1, 0, 0, 0, 0)$ and $\mathbf{t}' = (0, 1, 0, 0, 0)$. Therefore, we obtain

$$f(\mathbf{t} \vee \mathbf{t}') + f(\mathbf{t} \wedge \mathbf{t}') = f(1, 1, 0, 0, 0) + f(0, 0, 0, 0, 0) = 4 > 3 = 2 + 1 = f(\mathbf{t}) + f(\mathbf{t}').$$

Second, we show that f is not lattice supermodular. Consider a set $\mathcal{S} = \{s_1, s_2, s_3\}$ of students and a set $\mathcal{C} = \{c_1, c_2, c_3, c_4, c_5\}$ of schools. The preference lists are as follows

$$\begin{array}{lll} s_1 : c_1 \succ c_3 \succ \dots & c_h : s_1 \succ s_2 \succ \dots & \text{for all } h \in \{1, 2, 3\}. \\ s_2 : c_2 \succ c_4 \succ \dots & & \\ s_3 : c_3 \succ c_4 \succ c_5 \succ \dots & & \\ s_4 : c_5 \succ \dots & & \end{array}$$

Schools c_1 and c_2 have capacity 0 and the other schools have capacity 1. We choose the following two allocations: $\mathbf{t} = (1, 0, 0, 0, 0)$ and $\mathbf{t}' = (0, 1, 0, 0, 0)$. Therefore, we obtain

$$f(\mathbf{t} \vee \mathbf{t}') + f(\mathbf{t} \wedge \mathbf{t}') = f(1, 1, 0, 0, 0) + f(0, 0, 0, 0, 0) = 4 < 5 = 3 + 2 = f(\mathbf{t}) + f(\mathbf{t}'). \quad \square$$

Appendix F: Model Extensions and Further Insights

F.1. Extensions

Our model can be easily extended to capture several relevant variants of the problem. In what follows we name some direct extensions:

- Weighted budget constraints: If there is a unit-cost p_c of increasing the capacity of school c , we can add an additional budget constraint of the form

$$\sum_{c \in C} t_c \cdot p_c \leq B',$$

keeping all the other elements of the model unchanged. This extension could be used to allocate tuition waivers or other sort of scholarships that are school dependent.

- Different levels of granularity: Schools may not be free to expand their capacities by any value in $\{1, \dots, B\}$. This limitation can be easily incorporated into any of our formulations. Let us exemplify this for our Formulation (5). Specifically, recall that $y_c^k = 1$ if the capacity of school c is expanded by at least k seats, and $y_c^k = 0$ otherwise. Then, we know that $\sum_{k=1}^B y_c^k = t_c$, so we must add the precedence constraints $y_c^k \geq y_c^{k+1}$ for each $c \in C$ and $k = 1, \dots, B - 1$. If the capacity of school c can only be expanded by values in a subset $\mathcal{B}' \subseteq \{1, \dots, B\}$, we can enforce this by adding the constraints $y_c^k = y_c^{k+1}$ for all $k \in \{1, \dots, B - 1\} \setminus \mathcal{B}'$ and $y_c^B = 0$ if $B \notin \mathcal{B}'$. This could also be captured using knapsack constraints.

- Adding secured enrollment: The Chilean system guarantees that students that are currently enrolled and apply to switch will be assigned to their current school if they are not assigned to a more preferred one. This can be easily captured in our setting. Let M be the set of students aiming to switch school. First, for each student $s \in M$, \succ_s only contains schools that s strictly prefers over the currently assigned school. Second, we add to the objective function the term

$$\sum_{s \in M} (|\succ_s| + 1) \left(1 - \sum_{s:(s,c) \in \mathcal{E}} x_{s,c} \right),$$

which corresponds to accounting in the objective function for the current matching of students in M who are unable to switch to a more preferred school.

- Quota assignment: Many school choice systems have different quotas to serve under-represented students or special groups. For instance, in Chile there are quotas for low-income students (15% of total seats), for students with disabilities or special needs, and for students with high-academic performance. Moreover, some of these quotas may overlap, i.e., some students may be eligible for multiple quotas, and in most cases students count in only one of them. The number of seats available for each quota are pre-defined by each school, and schools have some freedom to define these quotas. Hence, our problem could be adapted to help schools define what is the best allocation of seats to quotas in order to improve students' welfare.

Note that all these extensions can be easily accommodated in methods involving optimization, i.e., our exact methods and LPH, since these involve either (i) modifying the input (as in the case of incorporating quotas or secured enrollment) or (ii) adding constraints (as in the case of a monetary budget or different levels of granularity). However, note that Greedy may not be able to accommodate all of them, specially those requiring the addition of constraints. For instance, it is not direct how to adapt our greedy heuristic to incorporate monetary budget constraints since the school leading to the highest marginal improvement may not be cost-effective. Hence, this provides an additional argument in favor of LPH relative to Greedy.

F.2. Further Insights: An Application to the Chilean College Admission System

Since early 2023, we have collaborated with the Ministry of Education of Chile (MINEDUC) and the “Sistema Único de Admisión” (SUA; the Chilean college board) in several applications of our framework. Specifically, we have been working on two main projects: (i) evaluating which schools should be overcrowded in order to downsize under-demanded schools, and (ii) evaluating minimum requirements regarding seats to offer to under-represented groups. The former problem can be addressed by adapting what we discussed in the previous section, so we focus on the latter.

The centralized part of the Chilean college admissions system includes an affirmative action policy (called “Programa PACE”), which consists of reserved seats for under-represented students (over 10,000 students each year) and special funds for the institutions where these students enroll. To be eligible to participate in this affirmative action policy (and potentially receive these funds), education institutions must commit to reserving a specified number of seats following specific guidelines defined by MINEDUC, e.g., a minimum number of reserved seats per program and a minimum total number of reserved seats across all their programs. Conditional on satisfying these requirements, universities can independently decide how many seats to reserve. Many universities meet these requirements by fulfilling the minimum requirement (of one reserved seat) per program and then devoting the additional required seats to under-demanded programs. As a result, only 18% of the reserved seats were used in the admissions process of 2022-2023.

To tackle this issue, we have been collaborating with MINEDUC to evaluate the effects of potential changes to these requirements. Since we do not know what the objective function of each university is (and, thus, how they allocate these reserved seats), MINEDUC asked us to evaluate different combinations of requirements (e.g., a minimum number of seats for the most popular programs, changes to the way to define the total number of reserved seats to offer, etc.) and objective functions (e.g., minimize the preferences of assignment, maximize the utilization of the reserved seats, maximize the cutoffs, maximize the average scores in Math and Verbal of the admitted students, among others) to obtain a wide range of possible outcomes that could result from each set of requirements.

For each of these combinations (of requirements, objectives, and other specific parameters), we adapted the framework described in this work and performed simulations considering the data from the admissions process of 2022-2023. Using current data to evaluate the implementation of these requirements in future years is without major loss, since students’ preferences are relatively stable over time and, thus, the reported preferences of one year are the best predictor of those in the coming years. Finally, note that having a methodology to solve the problem relatively fast was instrumental in performing this analysis, as it required hundreds of simulations in a fairly large instance.

This application showcases the flexibility of our methodology to answer different questions and stresses the need to solve these problems in a reasonable time.

Appendix G: Additional Background

G.1. The Deferred Acceptance Algorithm

In this section, we recall the Deferred Acceptance algorithm in Algorithm 5 which was originally introduced by Gale and Shapley (1962).

Algorithm 5 DA Algorithm

Input: An instance $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} \rangle$.

Output: Student-optimal matching.

- 1: Each student starts by applying to her most preferred school. Schools temporarily accept the most preferred applications and reject the less preferred applications which exceed their capacity.
 - 2: Each student s who has been rejected, proposes to her most preferred school to which she has not applied yet; if she has proposed to all schools, then she does not apply. If the capacity of the school is not met, then her application is temporarily accepted. Otherwise, if the school prefers her application to one of a student s' who was temporarily enrolled, s is temporarily accepted and s' is rejected. Vice-versa, if the school prefers all the students temporarily enrolled to s , then s is rejected.
 - 3: If all students are enrolled or have applied to all the schools they rank, return the current matching. Otherwise, go to Step 2.
-

G.2. Separation Algorithm of (Baïou and Balinski 2000) and Counterexample

In this counterexample, we show that Algorithm 6, the separation algorithm provided in Baïou and Balinski (2000), does not find the most violated comb constraint as claimed in their Theorem 5.

Let $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} \rangle$ be the instance of the school choice problem with $\mathcal{S} = \{s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8\}$ as the set of students and $\mathcal{C} = \{c_1, c_2, c_3, c_4, c_5, c_6\}$ as the set of schools; schools c_1, c_6 have capacity 2, while the others have capacity 1.

The preferences and priorities are given by:

$$\begin{array}{ll}
 s_1, s_2 : c_6 \succ c_1 \succ c_2 \succ c_3 \succ c_4 \succ c_5 & c_1 : s_1 \succ s_2 \succ s_3 \succ s_4 \succ s_5 \succ s_6 \succ s_7 \succ s_8 \\
 s_3, s_4 : c_6 \succ c_2 \succ c_3 \succ c_4 \succ c_5 \succ c_1 & c_2 : s_5 \succ s_6 \succ s_7 \succ s_8 \succ s_1 \succ s_2 \succ s_3 \succ s_4 \\
 s_5 : c_2 \succ c_3 \succ c_4 \succ c_5 \succ c_1 \succ c_6 & c_3 : s_6 \succ s_5 \succ s_7 \succ s_8 \succ s_1 \succ s_2 \succ s_3 \succ s_4 \\
 s_6 : c_3 \succ c_2 \succ c_4 \succ c_5 \succ c_1 \succ c_6 & c_4 : s_7 \succ s_6 \succ s_5 \succ s_8 \succ s_1 \succ s_2 \succ s_3 \succ s_4 \\
 s_7 : c_4 \succ c_2 \succ c_3 \succ c_5 \succ c_1 \succ c_6 & c_5 : s_7 \succ s_6 \succ s_5 \succ s_8 \succ s_1 \succ s_2 \succ s_3 \succ s_4 \\
 s_8 : c_5 \succ c_2 \succ c_3 \succ c_4 \succ c_1 \succ c_6 & c_6 : s_1 \succ s_2 \succ s_3 \succ s_4 \succ s_5 \succ s_6 \succ s_7 \succ s_8
 \end{array}$$

Let us consider the following optimal matching \mathbf{x}^* of the main program with no stability constraints (i.e., $\mathcal{J} = \emptyset$) and budget $B = 0$: $x_{s_1, c_1}^* = 1, x_{s_2, c_1}^* = 1, x_{s_3, c_6}^* = 1, x_{s_4, c_6}^* = 1, x_{s_5, c_2}^* = 1, x_{s_6, c_3}^* = 1, x_{s_7, c_4}^* = 1, x_{s_8, c_5}^* = 1$.

If we use the separation algorithm provided by Baïou and Balinski (2000), we search for a violated comb from school c_1 to c_5 and we find none. Finally, we check for violated combs in school c_6 , and we find the comb $C(s_3, c_6)$ with teeth $T(s_3, c_6)$ and $T(s_2, c_6)$. Note that comb $C(s_3, c_6)$ has value 1 in \mathbf{x}^* . However, if we use our Algorithm 4, we have that the only school in $\text{block}(\mathbf{x}^*)$ is c_6 . We find the comb $C(s_2, c_6)$ with teeth

Algorithm 6 Separation algorithm in Baïou and Balinski (2000)

Input: An instance $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} \rangle$, and a matching \mathbf{x} . Schools are labeled as $c_1, \dots, c_{|\mathcal{C}|}$.

Output: A violated comb.

```

1:  $i \leftarrow 1$                                      ▷ Index of the school
2:  $\mathcal{Q} \leftarrow \emptyset$                          ▷ The students that will form the comb
3: Assign  $q_i - 1$  elements of  $\mathcal{S}$  to  $\mathcal{Q}$  such that  $\sum_{(s,c_i) \in T^-(s',c_i)} x_{s,c_i} \leq \sum_{(s,c_i) \in T^-(s'',c_i)} x_{s,c_i}$ , for every
    $s' \in \mathcal{Q}$ ,  $s'' \in \mathcal{S} \setminus \mathcal{Q}$ , where  $T^-(\cdot)$  indicates the tooth without its base.
4:  $U \leftarrow \mathcal{S} \setminus \mathcal{Q}$ , and let  $s$  be  $c_i$ 's least preferred applicant
5: If the rank of  $s$  in  $c_i$ 's preference list is less than  $q_i$ , go to Step 17
6: if  $s \in \mathcal{Q}$  then
7:    $\bar{s} = \operatorname{argmin} \left\{ \sum_{(s,c_i) \in T^-(s',c_i)} x_{s,c_i} : s' \in U \right\}$ 
8:    $U \leftarrow U \setminus \{\bar{s}\}$ 
9:    $\mathcal{Q} \leftarrow (\mathcal{Q} \setminus \{s\}) \cup \{\bar{s}\}$ 
10: else
11:    $U \leftarrow U \setminus \{s\}$ 
12:  $C \leftarrow S(s, c_i) \cup T(s, c_i) \cup \bigcup_{s' \in \mathcal{Q}} T(s', c_i)$ 
13: if  $\sum_{(s',c') \in C} x_{s',c'}^* < q_i$  then
14:   Return  $C$ 
15: else
16:   Replace  $s$  by its immediate successor in  $c_i$ 's preference list and go to Step 5
17:  $i \leftarrow i + 1$ 
18: if  $i \leq |\mathcal{C}|$  then
19:   Go to Step 2
20: else
21:   Return  $\emptyset$ 

```

$T(s_2, c_6)$ and $T(s_1, c_6)$. Note that $C(s_2, c_6)$ has value 0 in \mathbf{x}^* . Therefore, if we use the separation algorithm of Baïou and Balinski (2000), we find a comb which is not the most violated one.