Université de Montréal

Dynamic Capacities and Priorities in Stable Matching

par

Federico Bobbio

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Thèse présentée en vue de l'obtention du grade de
Philosophiæ Doctor (Ph.D.)
en Informatique

January 12, 2024

Université de Montréal

Faculté des arts et des sciences

Cette thèse intitulée

Dynamic Capacities and Priorities in Stable Matching

présentée par

Federico Bobbio

a été évaluée par un jury composé des personnes suivantes :

Fabian Bastin
(président-rapporteur)

Margarida Carvalho
(directrice de recherche)

Geňa Hahn
(membre du jury)

David Manlove
(examinateur externe)

Matilde Lalín
(représentant du doyen de la FESP)

Résumé

Cette thèse aborde les facettes dynamiques des principes fondamentaux du problème de l'appariement stable plusieurs-à-un. Nous menons notre étude dans le contexte du choix de l'école et de l'appariement entre les hôpitaux et les résidents.

Dans la première étude, en utilisant le modèle résident-hôpital, nous étudions la complexité de calcul de l'optimisation des variations de capacité des hôpitaux afin de maximiser les résultats pour les résidents, tout en respectant les contraintes de stabilité et de budget. Nos résultats révèlent que le problème de décision est NP-complet et que le problème d'optimisation est inapproximable, même dans le cas de préférences strictes et d'allocations de capacités disjointes. Ces résultats posent des défis importants aux décideurs qui cherchent des solutions efficaces aux problèmes urgents du monde réel.

Dans la seconde étude, en utilisant le modèle du choix de l'école, nous explorons l'optimisation conjointe de l'augmentation des capacités scolaires et de la réalisation d'appariements stables optimaux pour les étudiants au sein d'un marché élargi. Nous concevons une formulation innovante de programmation mathématique qui modélise la stabilité et l'expansion des capacités, et nous développons une méthode efficace de plan de coupe pour la résoudre. Des données réelles issues du système chilien de choix d'école valident l'impact potentiel de la planification de la capacité dans des conditions de stabilité.

Dans la troisième étude, nous nous penchons sur la stabilité de l'appariement dans le cadre de priorités dynamiques, en nous concentrant principalement sur le choix de l'école. Nous introduisons un modèle qui tient compte des priorités des frères et sœurs, ce qui nécessite de nouveaux concepts de stabilité. Notre recherche identifie des scénarios où des appariements stables existent, accompagnés de mécanismes en temps polynomial pour leur découverte. Cependant, dans certains cas, nous prouvons également que la recherche d'un appariement stable de cardinalité maximale est NP-difficile sous des priorités dynamiques, ce qui met en lumière les défis liés à ces problèmes d'appariement.

Collectivement, cette recherche contribue à une meilleure compréhension des capacités et des priorités dynamiques dans les scénarios d'appariement stable et ouvre de nouvelles questions et de nouvelles voies pour relever les défis d'allocation complexes dans le monde réel.

Mots Clés : Appariement stable, expansion des capacités, programmation en nombres entiers, préférences dynamiques, choix de l'école, complexité de calcul, familles

Abstract

This research addresses the dynamic facets in the fundamentals of the many-to-one stable matching problem. We conduct our study in the context of school choice and hospital-resident matching.

In the first study, using the resident-hospital model, we investigate the computational complexity of optimizing hospital capacity variations to maximize resident outcomes, while respecting stability and budget constraints. Our findings reveal the NP-completeness of the decision problem and the inapproximability of the optimization problem, even under strict preferences and disjoint capacity allocations. These results pose significant challenges for policymakers seeking efficient solutions to pressing real-world issues.

In the second study, using the school choice model, we explore the joint optimization of increasing school capacities and achieving student-optimal stable matchings within an expanded market. We devise an innovative mathematical programming formulation that models stability and capacity expansion, and we develop an effective cutting-plane method to solve it. Real-world data from the Chilean school choice system validates the potential impact of capacity planning under stability conditions.

In the third study, we delve into stable matching under dynamic priorities, primarily focusing on school choice. We introduce a model that accounts for sibling priorities, necessitating novel stability concepts. Our research identifies scenarios where stable matchings exist, accompanied by polynomial-time mechanisms for their discovery. However, in some cases, we also prove the NP-hardness of finding a maximum cardinality stable matching under dynamic priorities, shedding light on challenges related to these matching problems.

Collectively, this research contributes to a deeper understanding of dynamic capacities and priorities within stable matching scenarios and opens new questions and new avenues for tackling complex allocation challenges in real-world settings.

Keywords : Stable matching, capacity expansion, integer programming, dynamic preferences, school choice, computational complexity, families

Contents

List of tables

List of figures

List of Acronyms & Abbreviations

DA algorithm	Deferred Acceptance algorithm
HR problem	Hospital-resident problem
HRI	Hospital-resident with incomplete preference lists
HRT	Hospital-resident with ties
HRTI	Hospital-resident with ties and incomplete preference lists
SC problem	School Choice problem
SCI	School Choice with incomplete preference lists
SCT	School Choice with ties
SCTI	School Choice with ties and incomplete preference lists

To my grandmothers Anna and Rosa,
who passed on to me the value of education.

To all who yearn to learn but lack the opportunity.

Acknowledgements

Margarida, rightfully, is the first word of these acknowledgments. Being your first Ph.D. student has been an immense honour. I am deeply grateful for your guidance and support as I navigated new academic territories. Your profound humanity enriched our every interaction, aiding my growth both personally and academically. You have always been there, thank you. Your dedication to establishing a solid scientific foundation for the betterment of society continues to inspire me.

Thank you from the bottom of my heart, Andrea. Thank you for building this great group of people, doing the Ph.D. was an amazing experience. The trust you gave me allowed me to pursue my goals with joy. I treasure your invaluable advice and it gives me joy to remember the chats, short or long. Your scientific vision, enthusiasm and curiosity inspire me to tread new paths.

I've encountered only a rare few who possess the depth and strength of determination that Alfredo does. Thank you Alf for your friendship, you are an older brother to me. Your gaze looks right at it; you are the right person at the right moment. This is when wisdom and truth come together. I am immensely grateful to you.

When I think of Ignacio, I think of the wind. Your sharpness and speed are an inspiration to always fly higher. Your true kindness is a rare gem that reflects smiles and wisdom far away. I am incredibly grateful for having met you.

Those who had the fortune of being raised by the shores of the most beautiful sea know the warmth of its waves, the perfect temperature of the water and the calm dunes filled with natural perfumes. Figs, palms, olives, oranges, lemons, the dust of the marble columns from a golden era, and the sound of the cicadas after lunch. When you arrive at the Chair, you find again this oasis in the white cold winter desert. There, you will be welcomed by a wise and noble man, who will offer you tea and share stories from countries far away. The warmth of his heart will make you forget the stinging harsh cold of the world. Yet, he will be honest, he will tell you the truth and he will remind you that sometimes this world is not as it should be; but here we are, you say, and you remind us that we must do the efforts of living a righteous life. Thank you for all the technical and scientific support, Khalid.

Life cannot be described, but it can be seen and admired. If you see Mehdi you perceive the vibrant energy of a life that is always active and creative. Thank you for always being there for all of us, your efforts multiplicate value in the lives of countably many people.

Céline, Nathalie et Lucie sont le cœur de l'université, elles sont des mères, des amies, des professionnelles talentueuses. Je suis profondément redevable de leur compassion et de leur soutien, elles créent un jardin d'humanité et laissent les étudiants prospérer. Je suis sincèrement reconnaissante à toute la communauté du CIRRELT, du DIRO, de EAAMO, et du GERAD pour l'incroyable soutien humain et scientifique qu'elles m'ont apporté. Thank you Nesly for the joyful chats, and for being a careful guardian of knowledge.

Necessito di un nuovo idioma, mentre il flusso della vita prosegue senza sosta. Intreccio questo filo dorato e luminoso che mi hai passato, tempo dopo tempo. Ed io sono qui, e devo andare avanti nel lutto. Ondula nel vento il drappo della saggezza, sferzato dalla tempesta che si leva. Grazie infinitamente Daisaku e Kaneko Ikeda per i vostri incoraggiamenti, che sono onde che battono ripetutamente, perpetuamente. Merci beaucoup Véronique pour tout le temps que tu m'as consacré, en écoutant mes plaintes, la sauce a' spaghetti, l'art et les discussions encourageantes. Merci Denise pour m'avoir accueillie à Montréal, pour les discussions extraordinaires et les activités. Craig, tu es un exemple vivant de force et de détermination, merci pour ton leadership. Grazie mille Rosanna per tutti i tuoi sforzi incessanti, merci Chantal pour ta gentillesse. Louise, tu es une force de la nature et ta joie est contagieuse ! Martine, tu es inarrêtable, ta générosité atteint tous les coins du cosmos. Cher Alberto, tu connais la clé de la joie, ta révolution est étonnante. Philippe, tu es solide comme une montagne, je ne suis que la millionième personne à te remercier pour l'extraordinaire soutien reçu. Robin tu es cool et continu comme une rivière fraîche des Alpes. Gaurav I will never forget the amazing accomplishments we achieved during the pandemic, thank you for your pure seeking spirit. Giada sei una bomba, grazie di cuore per tutto il tuo impegno. Thank you Crystal, your smile and kindness were always a huge encouragement to me. Antonella&Antonio: La gioia che donate al mondo è un gioiello prezioso. Giova, è stato incredibile ritrovarci dall'altra parte del mondo, il nostro legame di amicizia è eterno. Su, petit lion, les portes de l'avenir te sont ouvertes.

Grazie mamma e papà per tutto il vostro sostegno, le chiacchierate notturne, e il vostro impegno per la costruizione di un mondo migliore: Siete i miei eroi. Grazie ai miei fratelli che mi sono sempre fonte di ispirazione, grazie alla mia dolcissima e fortissima sorella, grazie ai miei nipoti per essere i più cool del mondo. Ho la rara fortuna di esser cresciuto in una splendida ed armoniosa famiglia; quando vedo la neve penso a te, zio Roberto, e quando vedo le sorgenti d'acqua fresca penso a te, zia Bianca. Thank you, mama Sitoy, papa Ernie and the Roosevelt Family, I am always touched by your compassion and profound humanity.

It is surprisingly difficult to materialize in words the emotions that flow when I think of the friends and good people I met in Montreal. The following list of acknowledgments is randomly ordered and incomplete. Claudio Nazionale hai un cuore d'oro, grazie per le chiacchierate profonde, i momenti esilaranti, e per esserci sempre stato; ti sono immensamente grato. Ale, la tua saggezza e la tua allegria sono come un camino caldo che riscalda in una gelida serata in riva al lago. Matte sei come un fratello, un amico, un cugino capoccione con il quale discuti fino allo sfinimento, grazie di cuore per essere una persona solida come la roccia e consistente nelle tue intenzioni e azioni, sei sempre fonte di ispirazione e di fiducia; ti sono immensamente grato. Giuseppino Youssouf, you are the lion of the desert; your generosity, ingenuity, and kindness will be sung of for eras to come; I am honored by your friendship. Marchino sei arrivato con un sorriso e te ne sei andato lasciandoci col broncio, Mario vive! Teodorino da Firenze, I had so much fun navigating with you the oceans of words and boarding together ships full of artistic treasures: I am looking forward to our next adventure! Toinou your sweetness, pure heart and joy are a paradigm to me, again and again we will keep walking together. Eccolo qui il dragone preferito: [ci entra?] Soldatino, King D'Artagnan, sei tutti e tre insieme! Oh, il lower bound (molto conservativo) è di 400; a Gabbriè, sempre più in alto. Claudine, Michele and Clara, thank you so much for making MTL the best place on the planet. Pietro che gioia ritrovare un pezzo di sole siciliano qui in laboratorio (Catania non conta), la tua gentilezza mi ricorda le nostre radici. Cher Didier, je pense qu'en ton honneur, je devrais écrire ce message dans une quatrième langue; merci pour tous ces moments et aventures incroyables passés ensemble, et le meilleur reste à venir! Hey cool smart chap Keith, thank you for stopping me the other day in the metro, we truly had an amazing time together: Deep discussions and a lot of life. Linda ci hai sempre azzeccato, sei libera e ti ammiro esplorare nuovi mondi. Cher Antoine et Flo, j'ai eu beaucoup de plaisir à passer du temps ensemble, vous êtes spéciaux dans mon cœur; passer la pandémie à cuisiner avec vous était spécial. Thank you for the amazing time together talking about everything, and the dinners and the laughters, you are a great friend Andrè. Belle et rebelle et forte, c'est toi et notre amitié, ma chère Charlotte. Cèrine when I think of you I think of your smile, your good humour and your laughter, thank you for the magic moments. Hey bello Helios, thank you for welcoming me in MTL and for your friendship! Thank you Sophie for always being there, I will always remember the wonderful dinners together; thank you immensely. Isa, I am so glad I met you, everywhere you go you bring joy and smiles. Grazie Miko per il tempo passato insieme e le chiacchierate sulla musica. VR, la tua determinazione e la tua ironia sono contagiose, grazie per tutti i tuoi consigli. A Clà, finalmente se semo ribbeccati, aribbeccamose presto insieme a Dan da Dolci Desideri sorè! Dear Leandro, thank you for the amazing chats and all the nice moments (lol) with the next person in the list! Hey Jiaqi, it was very nice sharing the office with you, your consistency and talent have always been an inspiration to me. Dear Rose, it was amazing meeting you and becoming your friend! Alex and Sophie, thank you for taking care of my garden

in the house and in our friendship, you are great growers! Marius you are like a hurricane, we rock! Feng, you are such a wonderful human being, I am so grateful our paths are intertwined; thank you immensely. Lèa, when I say this name I see your smile and feel your hug, thank you for everything. Justin, I treasure with joy our beautiful chats about math, life and food, thank you dear friend! Mathieu, thank you for being such a good friend, you always remind me how fun it is to do research! Principe della Nebbia, grazie per le belle chiacchierate! Eli, Chris, Alex, Gonzalo, Maxime, and Cal you have always been examples of great researchers and beings, I am grateful for having met you. Aurelien and Iva thank you for your sweet gifts and your friendship. Nico and Carlie, your presence was a lasting brand. Carlos, thank you for all these years spent together: the music, the joy, the art; thank you for your affection. Hey Rémy, thank you for the *bitter nights*, the amazing sincere chats, and all the deep insights you shared with me. Dear Hani, thank you for your friendship and for always being supportive, your frankness and constructive rebelliousness are an inspiration. Chris and Fiona it was so nice having you here in MTL. Thibaut thank you for always being positive and kind. Yle sei la mejo der Colosseo. Brochnnee, looking forward to see you flying higher and higher! Flore, Warley, William, Justine, Carl, Caroline, how much fun we had, thank you for being so brilliant! Marylou, Maria, Nagisa, Ismail what a wonderful fortune we share!

Introduction

Education serves as a vital pillar in the development of society. Despite a notable rise in literacy rates globally, significant challenges remain, as highlighted by recent studies showing that around 14% of the world's population still endures the hardships of illiteracy [**130**]. Moreover, policymakers are not only tasked with the expansion of the education system but also with ensuring that this expansion is conducted fairly, accommodating diverse societal needs and expectations. Modern mathematical tools have become integral in ensuring equitable school admissions.

While educational access has expanded, the quality of education often remains a secondary concern. This problem is starkly evident in countries within the Organisation for Economic Co-operation and Development, where studies indicate that median literacy levels are alarmingly near the threshold of functional illiteracy [**64**]. This low literacy level has profound societal implications, as it is strongly linked with higher incidences of crime and poverty [**103**]. Addressing this issue is crucial, as improving the quality of education can lead to broader societal benefits, including reduced crime rates and enhanced economic opportunities.

The dilemma facing contemporary policymakers is how to effectively balance the distribution of resources between expanding educational infrastructure and enhancing the quality of education. This includes not only improving the baseline standard of education but also channeling resources towards supporting and nurturing talent and academic excellence. Policymakers must carefully consider the implications of each decision, weighing the benefits of educational access against the need for high-quality instruction. In this complex scenario, mathematical models are invaluable for providing a framework to make informed, equitable decisions, guiding policy towards outcomes that benefit the larger community.

Research Questions

Education can be improved from both a quantitative and qualitative perspective. In this thesis, we develop analytical and computational tools that policymakers can use to improve the education system.

Capacity Expansion

From a quantitative point of view, in 2015, the world average number of years that a student dedicated to education was estimated at 9.2 [**76**], which is a number heavily dependent on the amount of resources that every society dedicates to education. In this regard, it is estimated that in 2020, the average percentage of the world budget dedicated to education was 4.3% [**66**]. The main portion of the budget is often allocated to cover recurring expenses such as salaries, bills, and facilities maintenance. As a consequence, only a small portion of the original budget can be allocated to extend the pool of available seats for students, which can be done by increasing the capacities of schools' programs. Given the constrained budget on education, it is pivotal to find the most efficient way of expanding the number of positions available to students so that they could continue studying in their adolescence or later in life. The following question arises naturally.

Given a budget, how should we allocate it to expand the education system to benefit the students the most? (1)

Formalizing this question in mathematical language requires the introduction of a model in which all the elements in the question are present as fundamentals, variables or parameters of the model. A classical approach is to provide a model as the one proposed in the seminal paper by Gale and Shapley [**67**]. This model is built on a bipartite graph with two disjoint sets of vertices and a set of edges that connect the vertices in the two sets. One set of vertices is interpreted as the set of students, and the other is the set of schools; the edges that connect the students to the schools represent the feasibility of an assignment, *i.e.*, an edge between a student i and a school j represents the availability of both i and j to be matched together. It is usually assumed that a student can be matched at most to one school, and that a school can be matched to as many students up to its capacity. A matching is defined as a subset of the set of edges that respects the capacity constraints of the schools and matches each student to at most one school. If we could decide which matching should be picked, *i.e.* how should students be assigned to schools, it seems reasonable to focus on matchings of maximum cardinality since they match the highest number of students in the education system. In the real world, students have preferences over the schools that may derive from tuition-fees, scholarships, location, et cetera. On the other side, schools may have a ranking over the students which may be a function of past grades, family income, neighborhood and other factors. Exactly because students and schools rank each other, choosing a maximum cardinality matching may yield an assignment that makes both sides of the market dissatisfied. If this may be the case, then pairs of student-school can override the matching and achieve a better outcome. In order to avoid this scenario, a stable matching can be chosen, *i.e.*, loosely speaking, a matching that respects the preferences of the *agents* in the market. In this setting, the actions that the agents take are of two kinds: 1) reporting their ranking, and 2) reporting their capacities (only the schools). One of the main questions policymakers need to address when building a mechanism, is how to incentivise agents to report their true preferences and capacities. If it is not possible to guarantee that agents can achieve their best outcome by reporting the truth, the mechanism is subject to manipulation from the agents. Ensuring that a mechanism cannot be manipulated guarantees its long term success. Therefore, the concern about manipulability in mechanisms brings us to the next critical question.

Is it possible to allocate extra capacities in a way that the mechanism is not manipulable? (2)

Since the beginning of the 21st century, the stable matching setting started to be used as a mathematical model for the school admission problem. Redesigning of school admission processes took place in Boston [**5, 6**], New York City [**3**], Hungary [**34**], Singapore [**148**], Chile [**50**] and San Francisco [**15**], to cite a few. The main concern of primary education is to ensure that each pupil could go to school; as mentioned earlier, there are still many countries around the world where not every student has access to education. Therefore, once a budget of extra capacities is destined to primary education, each of these extra capacities should be used to let a student enter the matching. On the other side, it is often the case that philanthropic institutions provide funding for education with the aim of supporting outstanding students. Therefore, there is a trade-off between allocating extra capacities to target access to education or to target merit. In light of the considerations regarding the fairness of the allocation of extra capacities in education, we find ourselves confronted with the following crucial question.

Can we allocate extra capacities with the primary goal of guaranteeing access? Can we allocate extra capacities on a merit basis? (3)

Complexity

Providing a computation-time efficient algorithm is crucial for policymakers, since it allows them to solve the problem at-hand with several configurations of parameters and choose the one that fits the best their data-set and setting. Therefore, the emphasis on computational efficiency in addressing capacity expansion issues underscores the significance of the ensuing question.

What is the computational complexity of allocating optimally a budget of capacities for the benefit of the students? (4)

As some school districts face a spike in subscriptions, others need to address the problem of managing under-demanded schools [**65**]. Finding a way to aggregate classes would benefit the public budget, help improve the schedule of teachers and may enable students to interact more with their peers. To summarize, this problem can be formulated through the following question.

How should we reduce capacities in the schools while obtaining the best possible matching for the students? And how difficult is it to solve this problem? (5)

Another well-known application of the matching problem is the market of hospitals and residents. When residents apply to hospitals, the hospitals that are under-subscribed in a stable matching are under-subscribed in every stable matching. This phenomenon is called "Rural Hospital Theorem" [**132, 68, 134, 107**] and it poses a serious threat for policymakers who need to ensure that all hospitals can be run with a minimum number of residents. In order to limit the demand for positions in Japanese urban areas, the centralized institution managing the hospital-resident matching provides maximal regional quotas [**85**]. If the centralized institution has at its disposal extra capacities in each region, the problem becomes how to impact the national market the most. The challenge of impacting the national market from a regional standpoint leads us to a series of pertinent questions.

How should we allocate extra regional capacities to obtain a national maximum cardinality stable matching, and a matching that benefits the most the students (or residents)? Finally, how difficult is it to find the solutions to the aforementioned problems? (6)

Clearly, the previous questions can also be posed when capacities should be reduced.

Sibling Priorities

Recently, the stable matching setting has found even more domains of application that range beyond the education system. Applications include refugee resettlement [**52, 17, 14**], healthcare rationing [**122, 25**], market for rabbis [**41**], and online dating [**74**]. A problem that refugee resettlement and hospital-resident matching have in common is *complementarities*, *i.e.*, the joint application of candidates who would like to be matched to the same location. In the case of the hospital-resident matching, allowing joint applications from couples was one of the main reasons that led to the redesign of the mechanism in the U.S.A. [**136**]. This is a problem faced also by families of refugees, who cannot be split and whose application should be taken in consideration in an aggregated fashion. Considering the real-world practice of schools giving priority to students with siblings already enrolled, it raises the following relevant question.

How should the priority of a student in a school change if a sibling is enrolled in that school? (7)

Addressing this question would give the foundations for understanding how to match families in a market where preferences rather than being static, change depending on the tentative assignment of relatives. Providing a matching that accounts for sibling priorities would certainly improve qualitatively the outcome of families participating in the admission process.

Contributions

The results contained in this thesis span in the fields of computational complexity, mathematical programming, matching theory and game theory. Chapters 3, 4 and 5 contain each a paper whose common thread is the study of stable matching when its building blocks become dynamic rather than being fixed. Specifically, in Chapter 3 we answer Question 4, thus yielding the computational complexity backbone of Chapter 4; additionally, in this chapter, through Questions 5 and 6 we formalize and present new problems for which policymakers have already expressed interest in the real world. Chapter 4 addresses Questions 1-3, the main results being the formulation of the capacity expansion problem, an efficient cutting-plane to solve it and experimental results on real world data that provide evidence of our theoretical results; moreover, we provide a thorough computational study on artificial data in which we put in comparison different formulations and establish the advantage of our cutting-plane algorithm. Finally, in Chapter 5 we present for the first time Question 7 and we study it from a modeling and algorithmic perspective.

We are currently working alongside the institution that manages the Chilean school admission process to implement the results in Chapter 4 and Chapter 5 at a national level.

First Paper

In Chapter 3 we investigate the two problems of capacity expansion and reduction from a computational complexity standpoint. These two problems have a very similar structure, therefore the proof of one's intractability can be used for the other problem as well.

It may seem natural to think that the problem of optimally allocating a budget of extra capacities for the benefit of the students is solvable in polynomial time: It may be sufficient to allocate the extra capacities to the most popular schools. We provide a counter-example that shows that allocating extra capacities to the most popular school may yield a sub-optimal solution. Indeed, our goal is to prove that the capacity expansion problem is NP-hard. In order to achieve this goal, we start by observing that finding a minimum cost (weakly) stable matching when there are ties in the preference lists is NP-hard and not approximable within $O(n^{1-\varepsilon})$ for every $\varepsilon > 0$, unless P=NP, where n is the number of schools. This result is particularly interesting because it establishes a divide in the computation of a minimum cost stable matching as soon as we assume preference lists may contain ties. Recall that if there are no ties, the problem can be solved in polynomial time via the DA algorithm. Our reduction to the capacity expansion problem will be from the problem of finding a minimum-cost (weakly) stable matching when there are ties, and in order to do that, we introduce a new structure that we call village. In the end, we prove that the problem of optimally expanding capacities with respect to the students' benefit is not approximable within $O(n^{1/6-\varepsilon})$ for every $\varepsilon > 0$, unless P=NP, where n is the number of schools. This result implies that no polynomial time approximation algorithm can be provided with a constant factor approximation guarantee. By exploiting the structural similarities between the problem of expanding and reducing capacities, we show that also the latter problem is not approximable within $O(n^{1/6-\varepsilon})$ for every $\varepsilon > 0$, unless P=NP.

In Chapter 3, we also study the capacity variation problem (*i.e.*, expansion or reduction) when the budget is partitioned in regions, *i.e.*, sets of schools. We show that both the problem of finding a minimum cost stable matching and that of finding a maximum cardinality one are NP-hard, the former being also not approximable within a constant factor, unless P=NP.

Remark. In this paper, we use the language of the resident-hospital stable matching problem, since the computational complexity literature has historically focused on this application. However, in the other papers, we change language; in the second and third paper, we refer to the many-to-one stable matching problem as the School Choice problem, as this is the main motivation for these contributions.

Second Paper

In Chapter 4 of this dissertation, we prove that under stability constraints, optimizing the individual welfare of each student is equivalent to optimizing the social welfare of all the students. This result is particularly important since it allows to obtain the best stable matching for each student (as in the Deferred Acceptance algorithm) by optimizing a mathematical programming model that has the aggregated welfare of the students as its objective. This key result, allows us to obtain the same outcome of the Deferred Acceptance (DA) algorithm via a mathematical programming model; there are several advantages of using a mathematical programming formulation rather than an adaptation of the DA algorithm, as is often done in the literature. Remarkably, a mathematical programming formulation is more flexible to adaptations than the DA algorithm. As discussed earlier, the fundamentals of the stable matching setting are the students, the schools, their capacities and the preferences of both schools and students. From a practical perspective, it is often the case that capacities are not fixed, indeed, a centralized decision maker may have the power to vary capacities in order to optimize their objective function. In Chapter 4, we introduce several mathematical programming formulations that model the problem of capacity expansion. On one hand, running the DA algorithm would need an exponential search in the space of all the possible allocations of extra capacities. On the other hand, solving a mathematical programming formulation can be done with an off-the-shelf solver which exploits the combinatorial structure of the problem. The first mathematical programming formulation, which is quadratic, is an adaptation of the classical formulation for the stable matching problem to the capacity expansion setting. Since the quadratic terms add extra non-convexities to the formulations, in Chapter 4 we introduce a novel mixed-integer linear programming formulation inspired by the work of Baïou and Balinski [**26**]; this formulation has a number of stability constraints exponential in the number of schools, which makes it impractical to solve in its entirety. Hence, we propose a cutting-plane approach through which these constraints can be separated by the algorithm proposed in [**26**]. Thanks to a series of key structural results that we provide, we propose a novel separation algorithm that finds the most violated constraint; this algorithm can be also applied to the original stable matching problem. Experimental results on artificial data show that the resulting cutting-plane algorithm outperforms the generalizations of the current state-of-the-art formulations for the stable matching problem.

We can enrich the objective function by adding a penalty for each unassigned student. When we solve the problem of allocating optimally extra capacities in the Magallanes region of Chile[1] with a high penalty for each unassigned student (*e.g.*, the length of the preference list plus one), we observe that the model tends to allocate the extra capacities to let previously unassigned students enter the matching. From a theoretical standpoint, we prove that when the penalty is sufficiently high (*e.g.*, the square of the number of schools), the stable matching obtained is a student-optimal maximum cardinality one. Experimentally, on the Chilean dataset, we also observe that a low penalty (*e.g.*, zero), induces the allocation of extra capacities to improve the matching of students who would be matched anyway to a school. We also provide a general theoretical result which states that for a negative penalty equal to the number of schools, the matching obtained from the allocation of extra capacities is one of minimum cardinality. Thus,

[1]Dataset from 2018, we focus on the Pre-K entry level.

the extra capacities are allocated primarily to improve the students who would be already matched in the education system. Finally, we show that the capacity expansion problem is not immune to manipulations from both sides of the market. Nonetheless, we prove that it is practically non-manipulable in a large market setting.

Third Paper

In Chapter 5, we address the problem of finding a stable matching under *dynamic priorities*, *i.e.*, when priorities are updated based on the current assignment; we use school choice with siblings as a motivating example. To accomplish this, we introduce a model where students belong to (potentially different) grade levels and may have siblings applying to the system (potentially in different levels). It is assumed that each family reports preferences over the assignment of their members and, on the schools' side, individual students are ranked according to a strict preference list (tie-breaker).

Motivated by the Chilean school choice problem, two types of sibling priority are introduced: (i) static, whereby students who have a sibling currently enrolled but not participating in the admission process get prioritized; and (ii) dynamic, whereby students with a sibling who is also participating in the admission process get prioritized. In both cases, the priorities of a student are with respect to the school with a sibling. Notably, the term *dynamic priorities* is motivated by its application to a proposed matching (just like justified envy). In simple terms, a matching respects dynamic priorities if no student can be moved to a preferred school that is matched with one of her siblings. Therefore, the concept of dynamic priorities introduces a series of challenges, since they may even override the standard definition of *justified envy*.

To overcome these challenges, an order of priorities among groups of students must be established. First, it must be established if a student with priorities can take the spot of any student in the school or not: In the first case, priorities are *absolute*; on the other hand, if a student with priority cannot take the spot of a student with a better ranking than the sibling providing the priority, then priorities are *partial*. Second, in order to assess which students of two competing families will enter the school, the notion of dependent/independent justified envy is introduced: Dependent priority gives access to the students with the sibling with the highest ranking, while independent priority gives access to students with the highest ranking. Based on these definitions, several notions of stability are introduced, and we show that a stable matching may not exist even under very simple assumptions. Nevertheless, we show that a stable matching under dynamic priorities may exist if families strictly prefer their siblings being assigned together and either (i) families have at most two members participating in the admission process or (ii) there is a single grade level. Moreover, we introduce a new collection of mechanisms that find such a stable matching with dynamic priorities. Finally, we discuss other properties of the mechanism, and we show that finding a maximum cardinality stable matching under dynamic priorities is NP-hard.

This work on dynamic priorities contributes to the literature in several ways. To the best of our knowledge, this is the first work to formalize different types of siblings' priorities and also the first to introduce the idea of dynamic priorities. The main contribution of Chapter 5 is the introduction of a novel notion of stability under dynamic priorities, where priorities are contingent on the matching. Moreover, we also provide complexity results for a stable matching problem with complementarities and dynamic priorities. Although we focus on school choice as a motivating example, these results and insights may prove to be helpful in the design of matching mechanisms where priorities depend on the assignment of others, such as in daycare assignments, college admissions, refugee resettlement, among others.

Organization

This manuscript is organized as follows. In Chapter 1, we present the essential background information on computational complexity, matching theory, and mathematical optimization. This foundational knowledge is crucial for a comprehensive understanding of the subsequent chapters. Chapter 2 is devoted to the literature review, where we outline the scientific works over which our contribution builds on and the papers related to our contributions. The first paper is presented in Chapter 3, the second in Chapter 4 and, finally, the third one in Chapter 5. Finally, concluding remarks and future research directions can be found in Chapter 6.

Chapter 1

Background

In this chapter, we introduce the fundamental concepts necessary to present our results. We begin by providing, in Section 1.1, an overview on the basics of computational complexity, the study of the tractability of solving a mathematical problem. Section 1.2 is dedicated to the introduction of matching theory, with a particular focus on stable matchings in bipartite graphs. After introducing the main definitions, we present known fundamental structural results. From a general perspective, researchers have focused their efforts in devising rules (mechanisms) capable of outputting such matchings. Moreover, it is often desirable to optimize the mechanism's output according with a predefined objective. Therefore, in Section 1.3, we introduce the classical mathematical programming formulations of the stable matching problem.

1.1. Computational Complexity

An algorithm is a sequence of instructions that, given an input, produces an output. One of the key questions regarding an algorithm is understanding its computational complexity, *i.e.*, the amount of resources needed to run it in terms of space and time. Specifically, for the latter, given an algorithm, we want to know in how many elementary operations the output will be produced, and we want to express this number as a function of the input size. The primary objective of an algorithm is to address a specific problem. Therefore, within the domain of computational complexity, our task is to assess the feasibility of finding an efficient algorithm capable of solving the problem efficiently in both time and space.

We introduce the well-known *knapsack* problem to illustrate the concept of complexity class of decision problems. An instance of the knapsack problem consists of a weight tolerance T, a threshold value K and a set of objects $\{o_j\}_{j\in J}$ such that each object o_j has a value v_j and a weight w_j. The question we want to address is if there is a set of objects whose total weight is not greater than the weight tolerance T, and whose total value is greater or equal than K. The knapsack problem just introduced is presented as a decision problem; a *decision problem* is a set of input instances for which we pose a question whose answer is *yes* or *no*. In the following, we provide a formal description of the knapsack decision problem.

Problem 1 (Knapsack)**.**
INSTANCE: A positive integer weight tolerance T, a positive integer threshold value K, a family of indices J, and triplets of object-value-weight $\{(o_j,v_j,w_j)\}_{j\in J}$ with v_j and w_j positive integers for every $j \in J$.
QUESTION: Is there a subset J' of J such that $\sum_{j\in J'} w_j \leq T$ and $\sum_{j\in J'} v_j \geq K$?

There are many variants of the knapsack problem, we compare now the classical knapsack problem stated above and its continuous version, where a fraction of an item can be taken. In the continuous version, the objects of the knapsack problem are divisible, so a solution maximizing the total value can be determined by simply selecting the objects by descending order of value per unit of weight, until the tolerance T is reached; the object at which the tolerance is reached or exceeded is fractionally selected. Since the main step needed to compute a solution of maximum total value for the continuous knapsack is the sorting of the objects by their value per unit of weight, clearly, this problem can be solved in polynomial time. We denote as P the class of decision problems that can be solved in polynomial time.

Otherwise, for the classic knapsack problem, where the objects are not divisible (for example they are cars, computers, et cetera), no polynomial time algorithm is known. NP is the class of decision problems for which, given a proposed solution, one can verify in polynomial time whether the solution is correct or not. In other words, a problem is in NP if it is efficiently verifiable by a deterministic algorithm in polynomial time once a non-deterministic algorithm provides a potential solution. For example, if we are given a combination of objects to put in the knapsack, we can verify in polynomial time whether the chosen combination does not violate the weight tolerance T and if the total sum of the values is at least equal to K. Note that the class of problems in P is included in the class NP, however it is widely believed that this inclusion is strict.

We introduced the knapsack problem as a decision problem, however, rather than finding a set of objects that satisfies both the weight limit and the threshold value, we may want to find the set of objects that *maximises* the overall value, subject to the weight constraint. More broadly, an optimization problem, where the goal is to find a feasible solution optimizing a certain objective, has associated a natural decision version. Given a target value b, the latter corresponds to answering the question: Is there a feasible solution attaining an objective value better or as good as b?

Given two classes of problems A and B, we say that we can *reduce* A to B, denoted as A$\leq$B, when there is a polynomial time transformation from any instance of $I_A \in A$ into an instance of $I_B \in B$ such that I_A is a yes instance of A if and only if I_B is a yes instance of B. We say that a problem is NP-hard when all the problems in NP can be reduced to it. It is broadly assumed that an NP-hard problem is an intractable problem. Moreover, if an NP-hard problem is also in

NP, then we say that the problem is NP-complete. For example, the knapsack problem (with indivisible objects) is an NP-complete problem [**88**].

1.1.1. Approximation Algorithms

In this thesis, we delve into optimization problems classified as NP-hard. Consequently, it becomes pertinent to investigate their efficient resolution through approximation algorithms. In what follows, we proceed to introduce the concept of polynomial time approximation algorithms.

Definition 1.1.1. Given an optimization problem *A*, an instance I of *A*, and a feasible solution x of instance I, the *performance ratio* of the feasible solution value $\mathit{obj}_I(x)$ with respect to the optimal solution $\mathit{opt}_A(I)$ is

$$R_A(I,x) = \frac{\mathit{obj}_I(x)}{\mathit{opt}_A(I)}$$

if the optimization is a minimization, and

$$R_A(I,x) = \frac{\mathit{opt}_A(I)}{\mathit{obj}_I(x)}$$

if the optimization is a maximization.

Moreover, we say that *A* can be *approximated within* an approximation factor c if there is a polynomial time algorithm p such that for every instance I of the problem *A*, p outputs a feasible solution x such that $R_A(I,x) \leq c$. If the approximation factor c is a real number, then we say that c is a *constant approximation factor*.

Note that when an optimization problem can be approximated within a constant factor 1, then the problem can be solved in polynomial time. It is believed that it is not possible to find such an approximation factor for an NP-hard optimization problem; however, for a given NP-hard optimization problem we may find an approximation algorithm with an approximation factor of $1+\varepsilon$, for $\varepsilon > 0$. An optimization problem A is considered to possess a *fully polynomial time approximation scheme* (FPTAS) when it is equipped with an approximation algorithm that takes as input an instance I and a constant $\varepsilon > 0$ such that, within polynomial time, dependent on both $\frac{1}{\varepsilon}$ and the size of I, this approximation algorithm produces a feasible solution that approximates problem A with an approximation factor of $1 + \varepsilon$. For example, the optimisation version of the knapsack problem has an FPTAS [**75**]. When we relax the requirement on the approximation algorithm in the FPTAS definition to the case that it runs in polynomial time only in the size of the instance I, then we obtain the definition of *polynomial time approximation scheme* (PTAS).

Let us introduce the decision problem of finding a maximum cardinality independent set in a planar graph.

Problem 2 (Independent set)**.**

INSTANCE: An integer K, and planar graph (V, E), where V is a set of vertices and E a set of edges incident to vertices in V.
QUESTION: Is there a subset of vertices V' of cardinality at least K for which no two vertices in V' are joined by an edge in E?

Interestingly, the optimization problem of finding a maximum cardinality independent set in a planar graph has a PTAS but no FPTAS [**28**].

It may happen that some NP-hard optimization problems do not admit any approximation algorithm with a constant factor. Typically, a proof of such a case is provided by a gap-introducing reduction from an NP-complete problem. Let us give an example. A propositional formula in conjunctive normal form (CNF) is a conjunction of clauses made of disjunctions of literals, where a literal is a proposition variable or its negation; when each clause in a CNF formula has size at most three, then we have a 3CNF formula.[1]

Problem 3 (3CNF).
INSTANCE: A 3CNF formula θ.
QUESTION: Is there an instantiation of the propositional variables in the formula θ that satisfies the formula?

The 3CNF decision problem is NP-complete [**88**]. Assume we have a decision version of a minimization problem A with targe value b, to which 3CNF can be reduced in polynomial time; specifically, there is a reduction function σ that given a 3CNF formula it outputs an instance of the problem A, with the additional following property: If for a given 3CNF formula θ there is *no* boolean assignment that satisfies it, then the optimal value of the A instance satisfies $\mathrm{opt}_{\mathsf{A}}(\sigma(\theta)) > b$; otherwise, $\mathrm{opt}_{\mathsf{A}}(\sigma(\theta)) \leq a$, where $a < b$. If we could provide an approximation algorithm for A with approximation guarantee at most $\frac{b}{a}$, then, thanks to the reduction σ, we could determine whether a 3CNF formula is satisfiable in polynomial time, thus proving P=NP. This method for proving that an optimization problem does not have an approximation algorithm within a certain factor is known as the *gap technique*. We present the gap technique for a minimization problem.

Theorem 1.1.2. Let A be an NP-hard decision problem, let B be a minimization problem, and let σ be a polynomial time reduction from the set of instances of A into the set of instances of B that satisfies the following two conditions for fixed rationals $a < b$:

- Every YES-instance of A is mapped into an instance of B with optimal objective value at most a.
- Every NO-instance of A is mapped into an instance of B with optimal objective value greater than b.

[1]An example of a 3CNF formula is $\theta = (c_1 \vee \neg c_2 \vee c_3) \wedge (c_4 \vee \neg c_2 \vee c_3)$ where c_1, c_2, c_3, c_4 are booleans.

Then problem B does not have a polynomial time α-approximation algorithm with worst case ratio $\alpha \leq \frac{b}{a}$, unless P=NP. Especially, problem B does not have a PTAS unless P=NP.

1.2. Matching Theory

In this section we review the basic definitions and results in matching theory. In Section 1.2.1 we introduce the concept of matching on a graph and some fundamental results. Then, in Section 1.2.2, we focus on stable matchings with preferences, and, in particular, on the property of stability.

1.2.1. Matchings on Graphs

In this section we present the fundamental results on matching theory under preferences; for further details we suggest consulting the textbooks [**138, 107**]. The fundamental structure in which we will be developing our discussion is the *graph*. A graph (V, E) is a structure consisting of a set of vertices (or nodes) V and a set of edges E between pairs of vertices; we will be interested in *bipartite graphs*. A bipartite graph is a graph $(\mathcal{S},\mathcal{C}, E)$ where the set of vertices is composed of two sets, $\mathcal{S}$, the vertices on the left, and $\mathcal{C}$, the vertices on the right; the edges in E connect vertices in $\mathcal{S}$ to vertices in $\mathcal{C}$. The bipartite graph provides a simple graphical model to represent many real life problems; for example, the allocation of resources (the vertices in $\mathcal{C}$) to agents (the vertices in $\mathcal{S}$) is decided by choosing which subset of E are selected. One of the most important concepts used to establish relations among vertices is called *matching*. A matching of a graph (V, E) is a subset μ of E in which there are no two edges incident with the same vertex. If a vertex $v \in V$ is paired in a matching μ, we say that v is *matched* (or *assigned*), otherwise v is *exposed* (or *unassigned* or *unmatched*). If an edge $\{v,v'\} \in \mu$ then we denote the set of vertices matched to v in μ as $\mu(v)$, in this case $\mu(v) = \{v'\}$; if v'' is an exposed vertex, then $\mu(v'') = \emptyset$. In general, we define $\mu(V') = \{v : \mu(v') = \{v\}$ for $v' \in V'\}$ where $V' \subseteq V$. We are often concerned with finding a matching which is maximum in cardinality among all the possible matching of a given graph; such a matching is called *maximum matching* (note that a graph may have several distinct maximum matchings). An *alternating path* in a graph G with respect to a matching μ is a path p in G in which alternatively one edge is in μ and the successive edge is not in μ. If an alternating path p with respect to μ has both end-vertices exposed, then p is called an *augmenting path*. This idea leads to the following fundamental theorem which characterizes the maximum matchings in a given graph.

Theorem 1.2.1 ([**33, 123**]). Let (V, E) be a graph. Every matching in the graph is a maximum matching if and only if it does not admit an augmenting path.

A matching is *perfect* when all the vertices in the graph are matched. Concerning bipartite graphs, the following result provides a sufficient and necessary condition for a perfect matching to exist.

Theorem 1.2.2 (**[73]**)**.** Let $(\mathcal{S},\mathcal{C},E)$ be a bipartite graph. Let μ be a matching in which all vertices of $\mathcal{S}$ are matched if and only if $|\mu(\mathcal{S}')| \geq |\mathcal{S}'|$ for every $\mathcal{S}' \subseteq \mathcal{S}$. Moreover, μ is a perfect matching if and only if $|\mathcal{S}| = |\mathcal{C}|$ and $|\mu(\mathcal{S}')| \geq |\mathcal{S}'|$ for every $\mathcal{S}' \subseteq \mathcal{S}$.

Let us now introduce the state-of-the-art computational result for determining maximum matchings in a general graph.

Theorem 1.2.3 (**[60, 114, 151]**)**.** Let (V,E) be a graph where $|E| = m$ and $|V| = n$. A maximum matching can be found in $O(\sqrt{n} \cdot m)$ time.

1.2.2. Matchings Under Preferences

The bipartite graph framework serves as a suitable formalism for the modeling of many mathematical and real-world scenarios, including but not limited to the assignment of students to schools, graduate doctors to hospitals, families to public housing, matchmaking endeavors, online advertisement distribution, and many other problems. All the aforementioned examples require an additional superstructure to the graph in order to complete the description of the problem. In fact, in each of these examples, some of the vertices of the bipartition represent agents that have preferences over the vertices of the other side: Students (graduate doctors) have preferred schools (hospitals), schools (hospitals) prefer to admit good students (doctors); families might prefer to be assigned to certain neighborhoods. Usually, when preferences are expressed by the agents on one side over the agents on the opposite side, we are in the framework of one-sided matchings with preferences; when both sides of the bipartition have preferences, then we have two-sided matchings with preferences. In this standard model, vertices represent agents, and the action that the agents can take is to express their preferences and their capacities. In this thesis, we focus our attention on two-sided matchings with preferences.

In general, the agents can express their preferences through orders via binary relations; in this thesis, unless stated otherwise, we assume that preferences are expressed as preorders. The Ranking Order List (ROL), or preference list, of an agent might include ties and can be incomplete (i.e., not all the agents in the other side of the bipartition are ranked). A vertex ranks only the vertices with which it is connected through an edge. This implies that whenever vertices v and v' are linked through an edge, then they rank each other. Note that a weight function on a bipartite graph, which provides a value to each edge incident to a vertex, yields ROLs. Given a vertex v, we define the set of the acceptable vertices for vertex v as $A(v) = \{v' \in V : \{v,v'\} \in E\}$. Next, we introduce the concept of capacity function, which describes, for each vertex, the maximum number of vertices with which it can be paired (e.g. how many students a school can enroll).

Definition 1.2.4. Let $(\mathcal{S},\mathcal{C},E)$ be a bipartite graph where $V = \mathcal{S} \sqcup \mathcal{C}$. A *capacity function* associated to $(\mathcal{S},\mathcal{C},E)$ is a function $\mathbf{q} : V \to \mathbb{N}$ that associates a capacity $q(v)$ to vertex v; a graph with an associated capacity function is said *capacitated*.

Given a subset μ of E, we say that a vertex v is *under-subscribed, full, over-subscribed* if $|\mu(v)|$, *i.e.* the cardinality of $\mu(v)$, is less, equal or greater than $\mathbf{q}(v)$ respectively. A *matching* (or *assignment*) μ is a subset of E such that $|\mu(v)| \leq \mathbf{q}(v)$ for every vertex in the capacitated graph, where $\mathbf{q}$ is the associated capacity function.

When the capacity function has value one for every vertex, then we have the *one-to-one* matching problem. If, the capacity function has values greater or equal than one for a side, then we have the *many-to-one* matching problem.

Now we have all the elements to define the School Choice (SC) problem, also known as the College Admission (CA) problem or Hospital-Residents (HR) problem, which was defined by Gale and Shapley in [**67**]; the School Choice problem will be the main subject of investigation throughout this thesis.

The School Choice problem (SC) in a capacitated bipartite graph with ROLs $(\mathcal{S},\mathcal{C},E,\mathbf{r},\mathbf{q})$ is a many-to-one stable matching problem endowed with a profile of preference lists $\mathbf{r} = (r_i)_{i\in V}$ where r_i is the preference list of vertex i. We call *students* the vertices in $\mathcal{S}$ which have capacity one; we call *schools* the vertices in $\mathcal{C}$, which have capacity greater than or equal to one. The one-to-one stable matching problem is usually called the Stable Marriage problem (SM), where the students are called *men* and the schools are called *women*.[2]

The problem of finding a matching under the additional assumption that agents (vertices) have preferences needs to be tackled through a more refined concept than just looking for a maximum cardinality matching. In fact, we want to avoid the scenario in which, after a matching is chosen, some agents could obtain a mutually better result by making a private arrangement. This case holds when there is at least a pair of agents that *blocks* the matching.

Definition 1.2.5. Let I be an instance of SC in which the ROLs have strict preferences and let μ be a matching of SC. We say that a pair $\{s,u\} \in E \setminus \mu$ is a blocking pair if the following two conditions are satisfied:

- student s is unassigned or prefers school u over school $\mu(s)$,
- school u is under-subscribed or prefers student s over at least one student in $\mu(u)$.

The matching μ is said to be stable if it does not admit a blocking pair.

The problem of finding a stable matching in an SC instance with strict and complete preferences was solved in the seminal paper by Gale and Shapley [**67**]. In Algorithm 1 we present their algorithm to solve the stable matching problem in $O(m)$, where m is the number of acceptable pairs (edges) in the given instance. This method is known as the Deferred Acceptance (DA)

[2] Note that this is an old terminology.

algorithm. Note that the correctness of the DA algorithm is a proof of existence of a stable matching for every SC instance with strict and complete preference lists.

Algorithm 1 Deferred Acceptance algorithm

Input: An instance of SC ($\mathcal{S}$,$\mathcal{C}$, E, **r**,**q**).
Output: A stable matching μ.

Initialize: $\mu = \emptyset$, label all students as *unassigned*
while there is an unassigned student s who has still a school to apply to **do**
    $u :=$ the school most preferred by student s, to which s has not yet applied.
    if u is under-subscribed **then**
        the pair (s,u) is included in the matching μ.
        s is labeled matched.
    else if there is s' matched to u s.t. u prefers s to s' **then**
        the pair (s,u) is included in μ.
        the pair (s',u) is deleted from μ.
        s is labeled matched.
        s' is labeled unassigned.
return μ.

The DA algorithm was initially formulated by Gale and Shapley for the one-to-one stable matching problem. Here, we decided to directly provide the algorithm generalised for the SC problem. It is important to note that the stable matching found by the algorithm is the optimal stable matching for the students: For every student all the other stable matchings lead to same or a worse assignment with respect to their preference lists. From a dual perspective, this formulation of the algorithm provides the worst stable matching for the schools, i.e., each school is at least better off in any other stable matching. The DA algorithm can be also formulated in a way to provide the optimal stable matching for the schools (which is the worst stable matching for the students). The key idea is that the set of vertices which proposes the match, is the one which obtains the optimal stable matching. Let us see how these concepts apply in an example.

Example 1.2.6. Consider an instance of SC with a set of four students $\{s_i : i = 1,2,3,4\}$ and a set of three schools, together with their capacities, $\{(u_i,q(i)) : i = 1,2,3\}$ where $q_1 = q_3 = 1$ and $q_2 = 2$. We describe the preference lists of each agent in the following table. Note that the preference lists are linear orders in which the leftmost element in the list is the most preferred and the rightmost element in the list is the least preferred. For example, student s_1 prefers u_1 over u_2 and u_2 over u_3. In this thesis we adopt two notations for representing preference lists; for example, the preference list of student s_1 can be represented as u_1, u_2, u_3 or as $u_1 \succ_{s_1} u_2 \succ_{s_1} u_3$, where the relation $\succ_{s_1}$ is the ranking of student s_1 over pairs of schools.

	Students' ROLs		Schools' ROLs and capacities	
$s_1:$	$u_1, u_2, u_3.$	$u_1:$	$s_1, s_2, s_3, s_4.$	$q_(1) = 1$
$s_2:$	$u_1, u_2, u_3.$	$u_2:$	$s_1, s_3, s_2, s_4.$	$q_(2) = 2$
$s_3:$	$u_1, u_3, u_2.$	$u_3:$	$s_1, s_2, s_3, s_4.$	$q_(3) = 1$
$s_4:$	$u_1, u_3, u_2.$			

The optimal stable matching for the students is $\{(s_1,u_1),(s_2,u_2),(s_3,u_3),(s_4,u_2)\}$, and the optimal stable matching for the schools is $\{(s_1,u_1),(s_3,u_2),(s_2,u_3),(s_4,u_2)\}$. Note that the matching $\{(s_1,u_2),(s_3,u_1),(s_2,u_3),(s_4,u_2)\}$, despite being a maximum matching, is not a stable matching; in fact, the pair (s_1,u_1) is a blocking pair because both agents s_1 and u_1 would prefer to be matched to each other rather than to the agents with which they are currently matched. □

Insightful results have been proved regarding the properties of the matched and unassigned (under-subscribed) students (schools) in an SC problem with strict and complete ROLs.

Theorem 1.2.7 (Rural Hospital Theorem, [**132, 68, 134, 107**]). For a given instance of SC with strict and complete preference lists, the following properties hold:

- The same students are assigned in all stable matchings;
- Each school is assigned the same number of students in all stable matchings;
- Every school that is under-subscribed in one stable matching is assigned exactly the same set of students in all stable matchings.

We denote the SC problem where the preference lists are strict and incomplete as SCI. Within the framework of the one-to-one matching problem, there is a bijection between the set of stable matchings in an instance I of where ROLs are strict and complete and the set of stable matchings in I when the ROLs are strict and incomplete [**107**].

Theorem 1.2.8 ([**107**]). Let I be an instance of one-to-one stable matching problem with n_1 students and n_2 schools, where the preference lists are strict and complete. There exists an instance I' with strict and incomplete ROLs of the same problem when preferences are of size n, where $n = \max\{n_1, n_2\}$, such that the stable matchings in I are in a bijective correspondence with the stable matchings in I'.

Often, in the real world, agents rank equally two vertices on the other side; therefore, we need a more flexible representation of the preference lists in order to account for ties. A *partially ordered set* (poset), is a set of elements with a reflexive, anti-symmetric and transitive order; a *linear order* is a poset in which all elements are comparable. We call SCT the SC problem in which the preferences are organized as a weak order. Since in the SCT problem it may happen that two agents are ranked in the same position, we must reformulate the notion of stability.

Definition 1.2.9. Let μ be a matching in an instance of the SCT problem. Let (s_i,u_j) be a pair in $E \setminus M$, we say that (s_i,u_j) blocks (or that it is a blocking pair of) M if one of the following conditions is satisfied according to the required level of stability

- Weak stability:
  - (i) s_i is unassigned or prefers u_j to her assigned school in M, and
  - (ii) u_j is under-subscribed or prefers s_i to its worst assigned student in M;
- Strong stability: **either**
  - (i) s_i is unassigned or prefers u_j to her assigned school in M, and
  - (ii) u_j is under-subscribed or prefers s_i to its worst assigned student in M or is indifferent between them;

  or
  - (i) s_i is unassigned or prefers u_j to her assigned school in M or is indifferent between them, and
  - (ii) u_j is under-subscribed or prefers s_i to its worst assigned student in M;
- Super stability
  - (i) s_i is unassigned or prefers u_j to her assigned school in M or is indifferent between them, and
  - (ii) u_j is under-subscribed or prefers s_i to its worst assigned student in M or is indifferent between them.

It is immediate to verify that a super stable matching in an instance of SCT is strongly stable and that strong stability implies weak stability.

The last key notions that we present in this section regard some well-known metrics that have been adopted to guide the search among the set of stable matchings. Note that the following metrics are given for the one-to-one stable matching problem.

Definition 1.2.10. Let I be an instance of the one-to-one stable matching problem and let μ be a stable matching, where $\mathcal{S}$ and $\mathcal{C}$ are the sets of students and schools in I, respectively.

Let s be an agent matched in μ, we define the rank of $\mu(s)$, $r_{s,\mu(s)}$, for agent s as the position of $\mu(s)$ in the preference list of s. We define the *regret* of μ as

$$regret(\mu) = \max_{s \in \mathcal{S}_\mu \cup \mathcal{C}_\mu} r_{s,\mu(s)}$$

where $\mathcal{S}_\mu$ and $\mathcal{C}_\mu$ are the students and schools matched in μ, respectively. The stable matching μ is said to be a minimum regret stable matching if $regret(\mu)$ is minimum over all stable matchings in I.

The *cost for the students* relative to μ is

$$cost^{\mathcal{S}}(\mu) = \sum_{s \in \mathcal{S}_\mu} r_{s,\mu(s)}.$$

Similarly, we can define the cost for the schools $cost^{\mathcal{C}}(\mu)$ relative to μ. More generally, the *cost* of a matching μ is $cost(\mu) = cost^{\mathcal{S}}(\mu) + cost^{\mathcal{C}}(\mu)$. A stable matching μ is an *egalitarian stable matching* if $cost(\mu)$ is minimum over all the stable matchings in I.

The *sex-equality* measure of μ, is defined as

$$d(\mu) = |cost^{\mathcal{S}}(\mu) - cost^{\mathcal{C}}(\mu)|.$$

A stable matching μ is said to be a *sex-equal stable matching* if $d(\mu)$ is minimum over all the stable matchings in I.

Consider an instance I of the one-to-one stable matching problem, and consider the student-optimal stable matching μ. Assume we introduce a new school $c^\star$ in I, thus obtaining the expanded instance $I^\star$ where the preferences of the students are the same as in I with the inclusion of school $c^\star$ therein. Note that $c^\star$ could also be a number of capacities to be added to a school in I. How does it change the student-optimal stable matching $\mu^\star$ with respect to μ? Observe that the set of students $\mathcal{S}$ in I is the same set of students in $I^\star$. It is known that every student in $\mathcal{S}$ weakly prefers the matching $\mu^\star$ to the matching μ [**89, 68**]. This property is known as the *entry comparative static*. Recently, Kominers has shown that a similar result holds in the many-to-one stable matching problem.

Theorem 1.2.11 ([**100**])**.** Let I be an instance of the many-to-one stable matching problem and I^* the instance where the capacities of some schools in I are expanded. Let μ and μ^* be the student-optimal stable matchings of instance I and I^*, respectively. Every student weakly prefers their matching in μ^* to their matching in μ.

1.2.3. Strategy-proofness

So far, we discussed the problem of finding a stable matching in an instance of the SC problem. Stability is a a desirable property for a matching since it ensures that student-school pairs of do not get better off by circumventing the mechanism. However, stability alone is not enough to guarantee that agents cannot manipulate the mechanism once they have enough information about how the mechanism works and some information about other agents' preferences and capacities. In this section, we introduce some fundamental notions to analyze when a mechanism is incentive compatible.

Every student in the SC problem is an agent that decides the preference list to present when taking part to the matching market. Every school is an agent that decides not only its preference list, but also how many capacities to report. Therefore, given a mechanism, *i.e.*, a procedure for the selection of a matching, there is a game played by the agents. In this game, each student (school) chooses an action consisting of picking the preference list (and capacity) to be revealed to the market; the goal of each agent is to optimize their matching with respect to their true preferences and capacities.

Designing a mechanism where the optimal action of each agent is reporting the true preferences and capacities is a fundamental objective of policymakers. A mechanism that has this property is called *strategy-proof*. The DA algorithm, which runs in polynomial time, can be devised to produce the student-optimal stable matching or the school-optimal stable matching. Once the DA is oriented in favor of the students, thus outputting the student-optimal stable matching, the students cannot obtain a better stable matching than that achieved by reporting their true preferences.

Theorem 1.2.12 ([**131**])**.** Consider the many-to-one stable matching problem. The student-optimal stable matching is weakly Pareto optimal for the students in the set of all matchings. That is, there can be no matching (even an unstable matching) that all students strictly prefer to the student-optimal stable matching.

The previous result provides a key insight in the set of possible outcomes that the students can achieve, hence providing the following guarantee of strategy-proofness for the students.

Theorem 1.2.13 ([**56, 131**])**.** Consider the many-to-one stable matching problem. In the game induced by the student-oriented DA algorithm, in which each player states a preference list, it is a dominant strategy for each student to state her true preferences.

Interestingly, a similar result does not hold for the schools when their capacities are greater than one [**56, 131**]. The previous positive result is counter-balanced by the following, which establishes an impossibility over the number of agents for which a mechanism can be strategy-proof.

Theorem 1.2.14 ([**131**])**.** Consider the one-to-one stable matching problem. No mechanism outputting a stable matching exists for which stating the true preferences is a dominant strategy for every agent.

When we restrict ourselves to the one-to-one SC problem, it has been shown that the schools can manipulate their preferences to turn the student-optimal stable matching into the school-optimal stable matching. In the framework of the one-to-one stable matching problem, a *winning coalition* is a set of agents that coordinate their preferences to manipulate the matching into their-side-optimal stable matching. A minimum winning coalition is a minimum-size winning coalition.

Theorem 1.2.15 ([**115**])**.** In any instance of the SC problem, the minimum winning coalition has cardinality at most $\lfloor \frac{n}{2} \rfloor$ where n is the number of schools.

Interestingly, schools can manipulate the matching not only by reporting false preferences, but also by under-reporting the number of capacities.

Theorem 1.2.16. [**144**] Suppose there are at least two schools and three students. Then there exists no matching mechanism that outputs a matching that is stable and non-manipulable via capacities.

As shown by these results, providing a mechanism that is strategy-proof for at least one side of the matching market is a crucial task that policymakers have to address.

In the context of capacity expansion, it becomes crucial assessing whether the entry comparative static is immune from manipulation. Let I be an instance and $c^\star$ a new school, the mechanism that selects the student-optimal stable matching $\mu^\star$ in the expanded instance $I^\star$ is strategy-proof for the students. However, as Question 2 tries to suggest, when the ranking of school $c^\star$ becomes a decision variable rather than being part of the input, it is not obvious anymore whether the mechanism that selects the stable matching and the creation of the new school is still strategy-proof.

1.3. Mathematical Programming

Mathematical programming (or mathematical optimization) is the field of mathematics that develops algorithmic methods to solve decision problems aiming to optimize some criteria. To define a mathematical programming problem, first we need to define the variables of the problem, which indicate the actions controlled by the decision-maker. The variables may have a continuous or discrete domain of existence. Once the decisions are established, we need to set the criteria guiding our decision-making, *i.e.*, we introduce an objective function. The goal of the optimization can be to minimize or a maximize of the objective function. The domain of the variables can be further restricted by introducing additional constraints. Once a problem is formulated as a mathematical programming model, it can be solved with an off-the-shelves optimization solver, such as the open source SCIP solver [**12, 70**].

In this section, we present some of the most important mathematical programming formulations for bipartite matching problems under preferences.

1.3.1. One-to-one Stable Matching

The first known mathematical programming formulations of the one-to-one stable matching problem were given by Gusfield and Irving [**71**] and Vande Vate [**150**]. Later on, Rothblum [**139**] provided a characterization of the stable matchings as the extreme points of a polytope. The relevance of these contributions is supported by the fact that the SC problem can be solved not only with the DA algorithm in polynomial time, but also using a linear mathematical programming formulation. Linear programming formulations offer the extra flexibility of introducing a linear objective function to be optimized, and importantly, they can be solved in polynomial time. Therefore, a linear model of theone-to-one matching problem can also be addressed in polynomial time through a mathematical programming formulation, which, in addition, provides greater adaptability compared to the DA algorithm.

We introduce a mathematical programming formulation for the one-to-one stable matching problem using the notation of the SC problem for ease of exposition; note that, at this point, the only difference between the two problems lies in the fact that the capacities of all the agents in theone-to-one matching problem are equal to one. Assume the set $\mathcal{S}$ of students and the set $\mathcal{C}$ of schools are of the same size, and every agent can be matched to at most one agent on the other side. Let $\mathcal{E} \subseteq \mathcal{S} \times \mathcal{C}$ denote the set of feasible pairs in an instance of the one-to-one stable matching problem with strict and incomplete lists. Note that in the following general formulation we aim at maximizing a general weight function associated with every feasible student-school pair.

$$\max_{\mathbf{x}} \quad \sum_{(s,c)\in\mathcal{E}} w_{s,c} \cdot x_{s,c} \tag{1.3.1a}$$

$$s.t. \quad \sum_{c\in\mathcal{C}:(s,c)\in\mathcal{E}} x_{s,c} \leq 1, \qquad \forall\ s \in \mathcal{S}, \tag{1.3.1b}$$

$$\sum_{s\in\mathcal{S}:(s,c)\in\mathcal{E}} x_{s,c} \leq 1, \qquad \forall\ c \in \mathcal{C}, \tag{1.3.1c}$$

$$x_{s,c} + \sum_{c'\succ_s c} x_{s,c'} + \sum_{s'\succ_c s} x_{s',c} \geq 1, \qquad \forall\ (s,c) \in \mathcal{E} \tag{1.3.1d}$$

$$x_{s,c} \in [0,1], \qquad \forall\ (s,c) \in \mathcal{E} \tag{1.3.1e}$$

where $w_{s,c}$ is the weight assigned to the edge (s,c). The vector of decision variables $\mathbf{x}$ decides the matching between students and schools: $x_{s,c}$ is 1 if student s is matched to c, and 0 otherwise, for every $(s,c) \in \mathcal{E}$. The optimization is a maximization of the objective function (1.3.1a). Note that when $w_{s,c} = 1$, we seek the maximum cardinality matching. Constraints (1.3.1b) and (1.3.1c) establish the capacity constraints on the agents, while Constraints (1.3.1d) characterize the stability condition. Finally, Constraints (1.3.1e) describe the original domain of existence of the decision variables. Rothblum [**139**] proves that all the extreme points of the polytope defined by the constraints of Formulation (1.3.1) are binary vectors; when a mathematical programming formulation satisfies this property, it is said to be a *perfect* formulation. Hence, an optimal x is always binary.

1.3.2. Many-to-one Stable Matching

We can extend Formulation (1.3.1) to the many-to-one stable matching problem. Note that now every school c has a capacity $q(c) \geq 1$.

$$
\begin{aligned}
\max_{\mathbf{x}} \quad & \sum_{(s,c)\in\mathcal{E}} w_{s,c} \cdot x_{s,c} && && (1.3.2a)\\
s.t. \quad & \sum_{c\in\mathcal{C}\colon (s,c)\in\mathcal{E}} x_{s,c} \leq 1, && \forall\ s \in \mathcal{S}, && (1.3.2b)\\
& \sum_{s\in\mathcal{S}\colon (s,c)\in\mathcal{E}} x_{s,c} \leq q(c), && \forall\ c \in \mathcal{C}, && (1.3.2c)\\
& q(c)x_{s,c} + q(c)\cdot \sum_{c'\succ_s c} x_{s,c'} + \sum_{s'\succ_c s} x_{s',c} \geq q(c), && \forall\ (s,c) \in \mathcal{E} && (1.3.2d)\\
& x_{s,c} \in \{0,1\}, && \forall\ (s,c) \in \mathcal{E}. && (1.3.2e)
\end{aligned}
$$

First of all, note that Constraints (1.3.2b) and (1.3.2c) are similar to Constraints (1.3.1b) and (1.3.1c) as they ensure that the capacities of the students and schools are satisfied. Second, it is worth mentioning that we should also take in consideration the capacities of the schools in the stability Constraints (1.3.2d). Finally, since the formulation 1.3.2 is not a perfect formulation, we need to include binary requirements on the variables (Constraints (1.3.2e)). In order to be able to relax the binary constraints on the variables, we need to provide a new formulation of the many-to-one stable matching problem. This problem is addressed in the next section.

1.3.2.1. **Baïou-Balinski Formulation**. Baïou and Balinski [**26**] provide a Linear Programming formulation for the SC problem which is based on a graphical intuition of the concept of stability. Given an instance of the SC problem graph $(\mathcal{S},\mathcal{C},E,\mathbf{r},\mathbf{q})$ with linear preferences $\mathbf{r}$, we can represent it in a matrix form by assigning to each vertex in $\mathcal{S}$ a column and to each vertex in $\mathcal{C}$ a row. The preferences of an agent are expressed through arrows: An horizontal (vertical) arrow from column (row) i to column (row) j means that the school (student) in the corresponding row (column) prefers student (school) j to student (school) i.

Example 1.3.1. Consider the following example with two schools u_1, u_2 of capacities respectively one and two, and three students s_1, s_2, s_3. All the students prefer school u_1 over school u_2; the preference list of school u_1 is s_1, s_3, s_2, and the preference list of school u_2 is s_3, s_2, s_1. In Figure 1, we assign a row to each school and a column to each student. Further, we represent horizontal (vertical) preferences with blue (green) arrows. Specifically, the blue (green) arrows yield schools' (students') preferences, the red (black) vertices are the matched (unassigned) pairs, and the $q(u_i)$ $(i = 1,2)$ are the capacities of the schools. Note that the given matching is stable. □

Given the above graphical interpretation of the SC problem, Baïou and Balinski provide the corresponding concept of stability as follows:

Lemma 1.3.2 ([**26**]). Let $I = (\mathcal{S},\mathcal{C},E,\mathbf{r},\mathbf{q})$ be an instance of the SC problem and let μ be a stable matching of I. A pair (s_j,u_i) is unassigned in the corresponding graphical representation

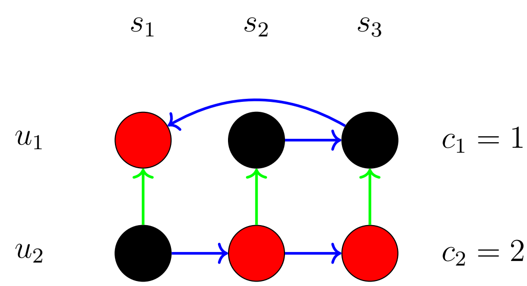


Fig. 1. Example of the Baïou and Balinski's formulation.

of the stable matching μ if and only if it is followed (i.e., preferred) in row i by $q(i)$ matched pairs or it is followed in column j by one matched pair.

In Figure 2, we provide a graphical representation of the stability concept elaborated when Lemma 1.3.2 is used. In order to provide the stability constraints for the LP model of SC , Baïou and Balinski define the following graphical structures.

Definition 1.3.3. Let $(\mathcal{S}, \mathcal{C}, E, \mathbf{r}, \mathbf{q})$ be an instance of the SC problem. We define the set $\Gamma^+ = \{(s,u) \in \mathcal{E}\colon (s,u)$ has at least $q(u) - 1$ successors in row $u\}$. Given a pair (s_j,u_i), we define the the set Γ^- as the complementary set of Γ^+.
Let (s_j,u_i) be a pair in Γ^+, we define the following structures:

- The *shaft* of (s_j,u_i), denoted $S(s_j,u_i)$, is the set of all pairs that follow (s_j,u_i) in row i including (s_j,u_i).
- The *tooth* of (s_j,u_i), denoted $T(s_j,u_i)$, is the set of all pairs that follow (s_j,u_i) in column j including (s_j,u_i).
- A *comb* of (s_j,u_i), denoted $C(s_j,u_i)$, is the union of the shaft $S(s_j,u_i)$, the tooth $T(s_j,u_i)$ and the teeth of $q(u_i) - 1$ successors of (s_j,u_i) in $S(s_j,u_i)$. Note that for a given (s_j,u_i) there might be multiple combs.

We define the set of the combs in a row u by $\mathcal{C}_u = \{C(s,u)$: for every pair $(s,u) \in \Gamma^+$ that is in row $u\}$, and we define the set of all combs by *Combs* $= \bigcup_{u \in \mathcal{C}} \mathcal{C}_u$.

For example, in both graphs of Figure 2 the shaft of (s_j,u_i) is $S(s_j,u_i) = \{(s_j,u_i), (s_{j+1},u_i), (s_{j+2},u_i)\}$. Instead, the tooth of (s_j,u_i) is different in the two graphs; in the first graph of Figure 2, $T(s_j,u_i) = \{(s_j,u_i)\}$, and in the second graph $T(s_j,u_i) = \{(s_j,u_i), (s_j,u_{i-1})\}$. Finally, in the first graph of Figure 2 school u_i has only one comb, $C(s_j,u_i) = \{(s_j,u_i), (s_{j+1},u_i), (s_{j+2},u_i)\}$; in the second graph school u_i has two combs, $C(s_j,u_i)_1 = S(s_j,u_i) \cup T(s_j,u_i) \cup T(s_{j+1},u_i)$ and $C(s_j,u_i)_2 = S(s_j,u_i) \cup T(s_j,u_i) \cup T(s_{j+2},u_i)$. Additionally, in Figure 2, we provide two different examples of stable matchings corresponding to the two cases of the stability condition for an unassigned pair (s_j,u_i). The blue (green) arrows yield schools' (students') preferences, and the red (black) vertices are the matched (unassigned) pairs. A matching is stable if every unassigned pair is followed in the row of school u_i by $q(u_i)$ assigned pairs, or if it is followed in column s_j by one assigned pair.

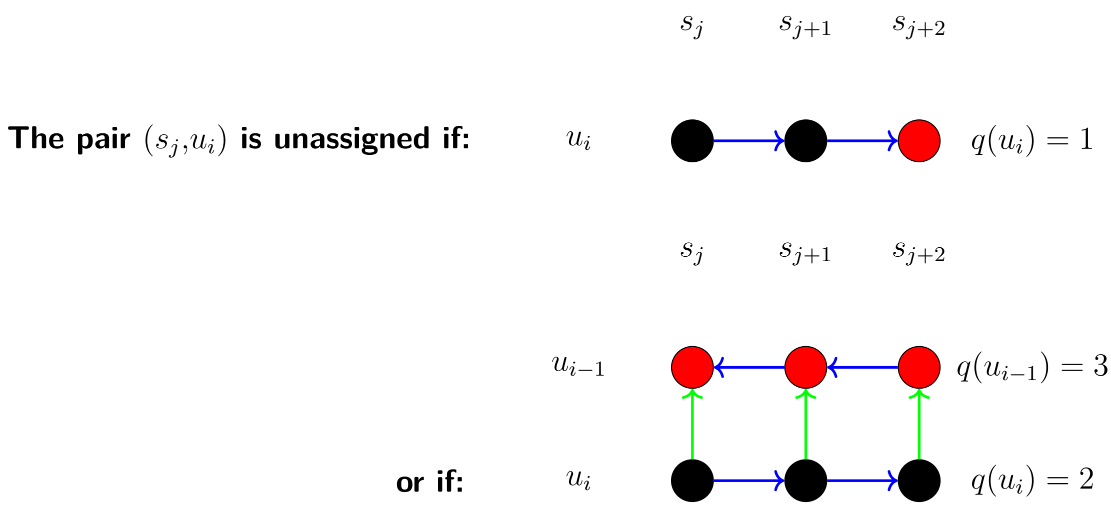


Fig. 2. Example of the Baïou and Balinski's definition of stability.

We can now present the linear polyhedron for the set of stable matchings of SC introduced in [**26**].

$$\max_{\mathbf{x}} \quad \sum_{(s,c)\in\mathcal{E}} w_{s,c}\cdot x_{s,c} \tag{1.3.3a}$$

$$s.t. \quad \sum_{u\in\mathcal{C}:\,(s,u)\in\mathcal{E}} x_{s,u} \le 1 \qquad \forall s\in\mathcal{S} \tag{1.3.3b}$$

$$\sum_{s\in S:\,(s,u)\in\mathcal{E}} x_{s,u} \le q(u) \qquad \forall u\in\mathcal{C} \tag{1.3.3c}$$

$$\sum_{(s,u)\in C} x_{s,u} \ge q(u) \qquad \forall u\in\mathcal{C}, \forall C\in\mathcal{C}_u \tag{1.3.3d}$$

$$x_{s,u} \ge 0 \qquad \forall (s,u)\in\mathcal{E} \tag{1.3.3e}$$

where, Constraints (1.3.3b) impose that each student can be matched with at most one school, Constraints (1.3.3c) yield a limit on the number of students matched to each school, and Constraints (1.3.3d) are the stability (comb) constraints. Finally, Constraints (1.3.3e) define the non-negativity of each assignment. The following theorem establishes that the vertices of the polytope described by Formulation (1.3.3) are the stable matchings of the SC problem.

Theorem 1.3.4 ([**26**])**.** The polyhedron described via constraints (1.3.3) corresponds to the stable polytope of the SC problem.

Therefore, the extreme points of the polyhedron described by Constraints (1.3.3) are integer.

Chapter 2

Literature Review

In this chapter, we provide a literature review in matching theory under preferences with a particular focus on the works related to our papers. The next three sections are dedicated to seminal work on stable matching problems. In Section 2.1, we start by describing the historical context in which matching markets under preferences became relevant, emphasizing their real-world application. Then, in Section 2.2, we briefly review computational complexity results related to the School Choice problem, since our work focuses on this type of matching markets. Section 2.3 focuses on the works using Mathematical Programming to tackle stable matching problems. Finally, Section 2.4, briefly reviews the works related to quota expansion and matchings with complementarities, enabling us to position the contribution of this thesis.

2.1. Historical Context

Since the publication of the seminal work by Gale and Shapley [**67**], researchers have focused their attention on studying the properties of bipartite markets under preferences, such as the design of strategy-proof mechanisms and the determination of equilibria for existing mechanisms, *e.g.*, [**56, 89, 131**]. The interest in this type of markets is not only theoretical. Indeed, it has been observed that mechanisms in which stability is not taken into consideration are usually not successful in the long term. A famous example is the labor market for medical interns and residents, also known as the hospital-resident matching market. The hospital-resident matching market has been running since the beginning of the 20th century, and is now managed by the National Resident Matching Program (NRMP). From its inception, the decentralized matching mechanism demonstrated significant inefficiency, adversely impacting both residents and hospitals alike [**132**]. Following multiple unsuccessful attempts to rectify the issues in the hospital-resident market through various countermeasures and rule changes, a new centralized mechanism was introduced in 1951, which proved to be highly effective [**138**]. Ex-post, it was discovered that the reason for the success of this mechanism was precisely the fact that it guaranteed stability [**132**]. Interestingly, the algorithm used by the NRMP since 1951 was the same algorithm that was independently introduced by Gale and Shapley [**67**], which is known as the Deferred Acceptance (DA) algorithm (Algorithm 1). As the understanding of the properties of this mechanism advanced across decades, prospective residents started worrying about the manipulability of the mechanism, *i.e.*, whether it was a dominant action to report their true preferences rather than strategizing in order to improve their matching. Specifically, as described in details in Section 1.2.2, the DA algorithm can be tuned to provide the best stable matching for all the hospitals or, dually, the best stable matching for all the residents. If, for instance, it is adopted the version that favors the hospitals, then the residents could achieve a better matching by reporting a false ranking of their favorite hospitals.[1] The property of a mechanism not being subject to manipulability is crucial for guaranteeing that merit is the main factor that determines the matching of a resident. In order to restore confidence into the system, the NRMP redesigned the mechanism to provide a stable matching that favors the residents [**136**], and therefore avoids manipulation from the residents' side. Since the successful implementation of the NRMP, several hospital-resident markets worldwide have taken cues from the NRMP's centralized matching approach, incorporating state-of-the-art insights from the field of matching theory. Examples of such adoption can be seen in Japan [**84, 87**] and Canada [**69**].

Another rapidly growing field of application concerns the problem of assigning students to schools, also known as the School Choice (SC) problem. Over the past few decades, many educational systems around the world have been adopting mechanisms suggested by matching theory to assign students at each level of education. Examples comprise the education system in Hungary [**34, 35**], in Chile [**50**], in South Korea [**23**] and in China [**155**], daycare admission in Denmark [**90**], public school admission in Boston [**5**], in New Orleans [**1**] and in San Francisco [**15**], university admission in Germany [**44**], in Spain [**126**] and in Turkey [**27**].

The list of current successful applications of matching mechanisms in bipartite markets under preferences also include examples outside the education domain. For example, the matching of rabbis [**41**], online dating [**74, 82**], labour market for lawyers in Germany [**54**], healthcare rationing [**122**], refugee resettlement [**52**], systems to assign cadets [**145, 106, 125, 154**], for studying stability in supply chains [**119**] and metal-only engineering change order synthesis [**81**].

2.2. Computational Complexity

The DA algorithm finds the student-optimal (or the school-optimal) stable matching in polynomial time [**67**]. In general, the main focus of the literature has been on finding a maximum cardinality stable matching, which can be easily obtained when there are incomplete and strict preference lists; that is because the Rural Hospital Theorem (Theorem 1.2.7) guarantees that all stable matchings have the same size. Similarly, when preference lists are complete and contain

[1] Note that, the DA algorithm presented in Algorithm 1, we can replace schools by hospitals and students by residents to fit the setting of the hospital-resident matching market.

ties, then all weakly stable matchings have the same size; indeed, if ties are broken arbitrarily and we find a stable matching with DA, then such a stable matching is complete and also weakly stable in the original instance with ties.

Once we include simultaneously in SC both incomplete lists and ties, the problem of finding a maximum cardinality stable matching becomes NP-hard even under very restrictive conditions [**79, 108**]. In terms of approximation ratios, the best known factor is $\frac{3}{2}$ [**110, 92, 120**] and the best lower bound is $\frac{33}{29}$ [**153**]. When ties are only on one side, there exists a $\frac{25}{17}$ approximation algorithm [**80**].

2.3. Mathematical Optimization

Mathematical Optimization, also called Mathematical Programming, is the discipline that develops mathematical methods of analysis to guide decision-making in a constrained environment. Domains of application range from energy markets, to scheduling, resource allocation, and transportation. In the context of matching problems, the use of mathematical programming dates back to at least Dantzing [**51**], who studied the matching problem in the form of the optimal assignment problem.

At the end of the 1980s, researchers started investigating mathematical programming formulations of the stable matching problem; the first known formulations are given by Gusfield and Irving [**71**] and Vande Vate [**150**]. Vande Vate [**150**] provides a linear description of the convex hull of the characteristic vectors of one-to-one stable matchings showing that the vertices of the polytope of stable matchings are integer tuples. This result is built upon the algorithm that finds stable matchings through rotations by Irving et al. [**77**]. The result of Vande Vate is extended to the case of one-to-one stable matching problem with incomplete lists by Rothblum [**139**]. Moreover, the author studies the polytope of stable matchings in the SC case. Roth et al. [**137**] prove the same results in an alternative way, providing a new perspective on the subject.

Exploring in greater depth the relation between the Gale-Shapley algorithm and Mathematical Programming, Abeledo and Rothblum [**11**] prove that the Gale-Shapley algorithm is an application of the dual simplex method. Their result is shown for the version of the Gale-Shapley algorithm in which the proposals of the agents are made successively. In the original version of the algorithm the authors assume that the proposals are simultaneous, but it has been proven that the two versions provide the same result [**112**].

In another paper, Abeledo et al. [**10**] prove that a fractional stable matching can be represented through a convex combination of (integral) stable matchings and that such fractional stable matchings yield a lattice structure (as proven in [**137, 147**]). In [**30**], Balinski and Ratier provide instances of one-to-one stable matching that have an exponential number of stable matchings (more precisely, as many stable matchings as the n-th Fibonacci number, where n is the number of agents in one side of the graph); this result showcases the importance of the selection of a specific stable matching, specially in a context where it is required fair decision making.

Concerning the relation between the SC problem and Mathematical Programming, Baïou and Balinski [**26**] characterize through linear inequalities the convex hull of its stable matchings, obtaining the so-called admission polytope (recall Section 1.3.2). The authors prove that the vertices of the admission polytope are integer valued (hence, we can drop the integrality constraints). We generalize their formulation of the SC problem in Chapter 4. Additionally, Baïou and Balinski also provide a polynomial time separation algorithm.

Sethuraman et al. [**140**] expand the work by Baïou and Balinski [**26**] from a geometric perspective. They further prove that a fractional stable matching can be decomposed as a convex combination of (integral) stable matchings.

Mixed Integer Programming (MIP) approaches for SCTI matching are developed through a series of papers that produce pre-processing heuristics [**78**] and exact methodologies [**53**] for optimizing and speeding up the computation of the maximum cardinality stable matching. Recent works providing MIP formulation for SCTI also include Agoston et al. [**157**].

Finally, Fleiner [**63**] provides a linear programming formulation of the many-to-many stable matching problem and thus, a characterization of the convex polytope of stable matchings in this case.

2.4. Related Problem Variants

In our research, we examine different versions of the School Choice problem, considering factors like changing school quotas and family preferences. To set the stage, we begin by discussing the existing literature in this field.

The design of a stable mechanism when the number of participants of one side is increased has already been investigated through the lens of game theory. In particular, for the one-to-one stable matching problem, this is known as the *entry comparative static*; to illustrate, it is known that when a new school (with capacity one, since we are in the one-to-one setting) is added to the instance, all students are matched *weakly better* [**89, 68, 138**]. Recently, Kominers [**100**] extends this result to the school choice problem. In [**31**], the authors show that the DA algorithm is invariant with respect to improvements of the students' position in the preference lists of the schools. Within the existing body of literature, as elucidated in Section 1.2.3, there has been a considerable focus on crafting matching mechanisms that encourage participants to express their genuine preferences. For instance, Sönmez [**144**] proves that schools can manipulate the mechanism in their favor by falsely reporting a reduced capacity. Moreover, Romm [**127**] proves that manipulation is still possible even if the reported capacities are enforced during the admission process; this is particularly interesting since it seems natural that by reducing a school capacity it follows that the outcome for that school would be worse. In Chapters 4 and 3, our contributions are focused on the dynamics of school capacity adjustments. Diverging from the previously mentioned literature, our research is dedicated to investigating the development of mechanisms that guide the optimal expansion or reduction of school capacities, under the assumption that all agents report their true preferences and capacities. We prove that this assumption is reasonable given that our mechanism is *strategy-proof in the large* [**24**].

In the case of the hospital-resident matching problem, allowing applications from couples was one of the main reasons that led to the redesign of the mechanism in the U.S.A. [**136**]. This is a problem faced also by families of refugees, who cannot be split and whose application should be taken in consideration in an aggregated fashion. These problems belong to the literature on stable matchings with complementarities, for which a solution may fail to exist [**132**]. Indeed, Ronn proves that given an instance of the SC problem with complementarities, establishing if there is a stable matching is NP-complete [**129**]. To overcome this limitation, Klaus et al. [**93, 95**] enrich the SC setting by assuming that preferences are weakly responsive; they show that this assumption guarantees the existence of a stable assignment. Another noteworthy path to mitigate the potential non-existence of a stable matching is providing an upper bound on the number of extra capacities that must be assigned in the market to guarantee the existence of a stable matching with complementarities [**117**]. All these approaches maintain the assumption that preferences are static and not subject to change. In fact, what we observe in the real world is that schools tend to admit first students that have siblings already enrolled in one of their programs, host countries tend to receive families of refugees without splitting them and universities tend to hire couples altogether. Even if the members in a couple are considered as an indivisible entity rather than individuals, McDermid and Manlove [**111**] prove that the problem of deciding the existence of a stable matching is still NP-complete. On a more positive light, Dur et al. [**57**] prove existence when pairs of agents are indivisible and each member of the pair applies to a different level; for instance, when siblings apply to different grades. Finally, in the context of the hospital-resident problem, even in the presence of single tie-breaker (or master-list) for the hospitals, Biró et al. [**37**] show that the problem of deciding the existence of a stable matching is NP-complete when there are complementarities. Chapter 5 aligns with the field of stable matchings involving complementarities. However, in the specific context of School Choice, our approach differs from the existing literature. We treat families of students as divisible, but we introduce dynamic priorities to capture the families' desire for their siblings to be placed together in the same school.

Chapter 3

Capacity Variation in Many-to-one Stable Matching

by

Federico Bobbio[1], Margarida Carvalho[2], Andrea Lodi[3], and Alfredo Torrico[4]

([1]) CIRRELT, DIRO Université de Montréal
([2]) CIRRELT, DIRO Université de Montréal
([3]) Jacobs Technion-Cornell Institute, Cornell Tech
([4]) CDSES, Cornell University

The authors dedicate this paper to Gerhard Woeginger (1964-2022), an outstanding computer scientist.

Prologue: In this chapter, we address the problem of assessing the computational complexity of expanding optimally capacities subject to a budget (Question 4); note that when the preferences of the students are all the same, the problem is trivial since it suffices to allocate all the extra-capacities to the school ranked first by all the students. Then, we also show that the same proof can be used to show the problem of reducing capacities to impact the least the matching for the students (Question 5); similarly, when all the students have the same preferences, it suffices to reduce the capacities of the least preferred schools (starting from the last) until all the budget is met. Finally, we investigate the previous two questions when the budget of capacities is partitioned among subsets of schools. In this chapter, we also introduce and study the aforementioned questions when the objective function is the maximization of the number of matched students.

Article details: This article was submitted to the Journal of Computer and System Sciences.

Contributions of the authors: Federico Bobbio participated in all the stages of the work, being the main author of the paper. The idea to study not only the computational complexity of the capacity expansion problem but also of the capacity reduction problem along with the problems with regional quotas came from Federico Bobbio. He devised the proofs presented in the paper as well as its first draft.
Margarida Carvalho, Andrea Lodi and Alfredo Torrico revised the proofs as well as the paper.

Résumé. Le problème de l'appariement stable plusieurs à un constitue l'abstraction fondamentale de plusieurs marchés d'appariement réels tels que le choix de l'école et l'affectation des résidents d'un hôpital. Ces problèmes impliquent deux groupes d'agents, souvent appelés résidents et hôpitaux. La configuration classique suppose que les agents classent le côté opposé et que les capacités des hôpitaux sont fixes. On sait que l'augmentation de la capacité d'un seul hôpital améliore l'allocation finale des résidents. En revanche, la réduction de la capacité d'un seul hôpital détériore l'allocation des résidents. Dans ce travail, nous étudions la complexité de calcul de la recherche de la variation optimale des capacités des hôpitaux qui conduit au meilleur résultat pour les résidents, sous réserve de stabilité et d'un budget de variation des capacités. Tout d'abord, nous montrons que le problème de décision consistant à trouver l'expansion optimale de la capacité est NP-complet et que le problème d'optimisation correspondant est inapproximable en $n^{\frac{1}{6}-\epsilon}$, où n est le nombre de résidents. Ce résultat est valable pour des préférences strictes et complètes, et même si nous allouons des capacités supplémentaires à des ensembles disjoints d'hôpitaux. Deuxièmement, nous obtenons des résultats analogues d'inapproximabilité pour le problème de la réduction des capacités. Ces résultats de complexité sont cruciaux, car ils empêchent le développement d'un algorithme d'approximation à facteur constant en temps polynomial pour s'attaquer à un problème réel d'un grand intérêt pour les décideurs politiques.
Mots clés : Appariement stable de plusieurs à un, problème des résidents d'hôpitaux, expansion des capacités, réduction des capacités, complexité de calcul

Abstract. The many-to-one stable matching problem provides the fundamental abstraction of several real-world matching markets such as school choice and hospital-resident allocation. These problems involve two sets of agents, often referred to as residents and hospitals. The classical setup assumes that the agents rank the opposite side and that the capacities of the hospitals are fixed. It is known that increasing the capacity of a single hospital improves the residents' final allocation. On the other hand, reducing the capacity of a single hospital deteriorates the residents' allocation. In this work, we study the computational complexity of finding the optimal variation of hospitals' capacities that leads to the best outcome for the residents, subject to stability and a capacity variation budget. First, we show that the decision problem of finding the optimal capacity expansion is NP-complete and the corresponding optimization problem is inapproximable within $n^{\frac{1}{6}-\epsilon}$, where n is the number of residents. This result holds under strict and complete preferences, and even if we allocate extra capacities to disjoint sets of hospitals. Second, we obtain analogous computational inapproximability results for the problem of capacity reduction. These complexity results are crucial, as they hinder the development of a polynomial time constant-factor approximation algorithm for tackling a real-world problem of great interest for policymakers.
Keywords: Many-to-one stable matching, hospital-resident problem, capacity expansion, capacity reduction, computational complexity

3.1. Introduction

The stable matching problem in a bipartite graph is a classical problem that requires a few key ingredients to be defined: Two disjoint sets of nodes, a set of edges connecting the nodes in the two sets, a capacity function for each node and a weight function for each pair of edge-node. The weight function is often interpreted as the ranking preferences of the node over the edges. A matching is a collection of edges that respects the capacity of each node, and it is said to be stable when such collection cannot be simultaneously improved (w.r.t. the weight function) for two nodes of every connected pair in the graph. It has been shown that a stable matching always exists via a polynomial-time algorithm [**67**], and that all the stable matchings have the same cardinality even if the graph is not complete [**132, 68, 134**]. If we assume the graph to be complete and we assume the weight function can assign the same value to multiple edges incident to a common node, *i.e.*, some edges are tied in the ranking of the node, then the problem of finding a maximum cardinality stable matching remains polynomial. Nonetheless, as soon as we relax also the assumption that the bipartite graph is complete, then the problem of finding a stable matching of maximum cardinality becomes NP-hard [**108**].

Given the simplicity of the setting in which the problem is defined, the stable matching problem has found multiple applications in the real-world. Perhaps, the education admission process is the most famous application setting in many countries all over the world, such as daycare admission in Denmark [**90**], school and hospital-resident allocation in the USA [**5, 6, 3, 132, 136**], school and university admission in Hungary [**34, 36**] and Chile [**105, 50**], school admission in Singapore [**148**], university admission in China [**155**], Germany [**44**] and Spain [**126**]. Moreover, in the last years, there has been a growing interest in the use of the stable matching problem to facilitate the integration of refugees in hosting countries [**52, 17, 14**], or optimizing the rationing of healthcare services such as ventilators during the pandemic [**122, 25**].

Most of these applications assume that one of the two sets of nodes can be matched more than once, which is known as the many-to-one stable matching (HR) problem, where HR stands for hospital-resident. Henceforth, we refer to one set of nodes in the graph as hospitals and the other nodes as residents. In this setting, the weight function represents the preferences of the residents over the hospitals, and vice versa. The HR problem and its multiple variants, have been widely studied in the literature from different perspectives: From a polyhedral [**26, 27**] and algorithmic [**67**] perspective, to geometry [**140**], mathematical programming [**150**], combinatorics [**97**], fixed-point methods [**146**] and graph theory [**29**].

In the standard version of HR, the capacity of each hospital is fixed and known in advance. The decision-maker in charge of the selection of a matching (assignment) does not have control over these quota. However, there are multiple real-life situations in which the variation of the size of the market, expansion or reduction, could play a significant role. For example, the University of Tsukuba recently restructured the course offering by allocating a budget of capacities to courses starting from zero [**101**]. From a methodological standpoint, Bobbio et al. [**38, 39**] were the first to provide an exact mathematical programming formulation to find the optimal matching and allocation of capacities for the benefit of the residents. Policymakers are often required to produce multiple possible scenarios before choosing how to allocate a budget. This requires the use of algorithms that output multiple solutions within a short time-window. For instance, in the Chilean school admission system, the centralized clearinghouse needs to produce several possible scenarios before deciding how to allocate scholarships [**50**]. Moreover, if finding an optimal solution, *i.e.*, a feasible assignment of hospital capacities and an associated stable matching accordingly with a given objective, turns out to be time consuming, decision-makers may resort to polynomial-time algorithms with constant-factor guarantees. In this paper, we address precisely the question of how difficult it is to find such a solution, and whether it is possible to have approximation guarantees when looking for sub-optimal solutions. Roughly speaking, for the expansion of the market, we study the following question:

> «*Let $B \in \mathbb{Z}_+$ be a non-negative integer. Given that B extra capacities should be added to the hospitals, which are the hospitals whose capacities should be expanded to obtain the best stable matching for the residents?* »

On the other hand, in certain cases one side could be under-demanded, *i.e.*, as it has been observed with schools [**65**], therefore a reduction in the spots may improve the finances of the policymaker, while minimizing the impact on the education system for the residents. Indeed, in the second part of this work, we focus our attention on the reduction of capacities in the market. To put it simply, we study the following question:

«Let $B \in \mathbb{Z}_+$ be a non-negative integer. Given that B capacities should be reduced from the hospitals, which are the spots that should be reduced to obtain the best stable matching for the residents? »

We primarily focus on a rank-based metric (the weight function) to choose the best matching for the residents. We also study the variants of the problems above under a cardinality-based metric, which has been widely studied in the literature [**107**].

3.1.1. Contributions and Organization

This paper is organized as follows. In Section 3.2, we introduce the formal notation, the problem of expanding capacities (Problem 4), and the problem of reducing capacities (Problem 5). Once we have established the notation and main definitions, in Section 3.3 our main focus is to establish the complexity of Problem 4. To achieve this result, we first observe, in Corollary 3.3.2, that determining the stable matching of minimum average rank for the residents in the presence of ties is NP-hard and is not approximable within $\bar{n}^{1-\varepsilon}$, for any $\varepsilon > 0$, where $\bar{n}$ is the number of residents. This result is fundamental since it puts a boundary on the computability of the stable matching of minimum average rank, which is well known to be polynomially solvable when there are no ties via the DA algorithm.[1] Note that Corollary 3.3.2, which is a natural consequence of Theorem 7 in [**108**], is the stepping stone needed to build our main proof. The remainder of Section 3.3 is devoted to study the complexity of the capacity expansion problem. All our results are proven in the special case in which the initial capacity of every hospital is one. Indeed, even under very restrictive assumptions, finding the allocation of extra capacities to the hospitals that lead to a minimum average rank stable matching is NP-hard, and for any $\varepsilon > 0$, it cannot be approximated within a factor of $(\bar{n})^{\frac{1}{6}-\varepsilon}$ unless P=NP (Theorem 3.3.1). Our complexity proof is based on a new structure that we call *village*. Each village is assigned some extra capacities, and the preferences of the hospitals and residents in a village ensure that the extra capacities can be optimally allocated only in a specific way. Then, in Section 3.4, we study the capacity reduction problem. We prove that this problem is NP-hard, and for any $\varepsilon > 0$, it cannot be approximated within a factor of $(\bar{n})^{\frac{1}{6}-\varepsilon}$, unless P=NP (Theorem 3.4.1). The proof follows a similar reasoning as in Theorem 3.3.1. We exploit again the structure of the village, with the caveat that every hospital starts with a capacity of two seats. In Section 3.5, we study several variants of Problems 4 and 5. Specifically, we partition the set of hospitals and allocate (remove) a certain amount of capacities to (from) each set of the partition. Theorem 3.5.2 shows that, even when we partition the set of hospitals and we allocate to (remove from) each set at most one spot, finding the optimal allocation is an NP-hard problem. Moreover, we prove that the optimization version of the problem is not approximable within a factor of $\bar{n}^{1-\varepsilon}$, for any $\varepsilon > 0$ (Theorem 3.5.2). The equivalent results for the reduction problem are shown in Theorem 3.5.4. We provide similar

[1]Note that, as proven in [**38, 39**], when preferences are strict, minimizing the average rank of the residents is equivalent to find the resident-optimal stable matching.

results to the variant of the problems that considers as an objective function the cardinality of the matching, Theorems 3.5.3 and 3.5.5, respectively. Finally, concluding remarks can be found in Section 3.6. A summary of our results and relevant results from the literature can be found in Table 1. The first top three entries in the column of maximum cardinality problems are results from: first, [**132, 68, 134, 107**], second, [**107**] and, third, [**108**]. The first top result from the second column is shown in [**67, 113, 56, 131**].

	Decision version of the problem	
Framework	Maximum cardinality	Average rank
HR/HRI	Polynomial	Polynomial
HRT	Polynomial	Inapprox. ([**108**] and Sec. 3.3)
HRTI	NP-complete	Inapprox. ([**108**] and Sec. 3.3)
HR capacity variation	Trivial	Inapprox. (Sec. 3.3 and 3.4)
HR cap. var. subsets	Trivial	Inapprox. (Sec. 3.5)
HRI cap. var. subsets	NP-complete (Sec. 3.5)	Inapprox. (Sec. 3.5)

Table 1. Summary of our contributions and relevant computational complexity results from the literature.

Note: HR corresponds to the many-to-one stable matching problem, the suffixes I and T stand for incomplete preference lists and for preference lists with ties, respectively.

3.1.2. Related Work

General context. In their seminal paper, Gale and Shapley [**67**] introduce the stable matching problem and provide a polynomial-time algorithm, known as the deferred acceptance (DA) algorithm. The DA algorithm computes an assignment such that there is no pair of agents that would simultaneously prefer to be paired to each other rather than being in their current assignment; this is known as a *stable matching*. In practice, the DA mechanism has been extensively used to improve admission processes, *e.g.*, see [**3, 34**]. For further details on stable matching mechanisms, see [**138, 107**]. In general, the main focus of the literature has been on finding the maximum cardinality stable matching, which can be efficiently obtained when there are incomplete preference lists[2] without ties or complete preference lists that include ties [**107**].[3] Once we assume both, incomplete lists and ties, the problem of finding the maximum cardinality stable matching becomes NP-hard, even under very restrictive conditions [**108**]. In terms of approximation ratios,

[2]Not all the agents are ranked. In the case of incomplete preference lists, the Rural Hospital Theorem holds [**132, 68, 134, 107**], which states that all the stable matchings have the same cardinality.

[3]Some agents in the preference list are ranked equally. In the case of preference lists with ties, all the weakly stable matchings are complete (under the assumption that the cardinalities on the two sides of the bipartition are equal). Weak stability means there is no pair of agents that strictly prefer to be matched to each other rather than being in their current assignment.

the best known factor is $\frac{3}{2}$ [**110, 92, 120**] and the best lower bound is $\frac{33}{29}$ [**153**]. In general, the existence of a stable matching is not even guaranteed in some of the most important variations of the HR problem, such as the HR problem with couples [**129, 37**].

Capacity variation. The design of a stable matching mechanism, when the number of participants of one side is increased, is known as the *entry comparative static*. In [**89, 68, 138, 100**], the authors prove that when a new hospital is added to the instance (or some extra capacities are added parametrically), all residents are matched *weakly better*. A substantial part of the literature has focused on strategy-proof matching mechanisms, *i.e.*, on matching mechanisms that incentivize participants to reveal their true preferences. Sönmez [**144**] proves that hospitals can manipulate the stable matching in their favor by falsely reporting a reduced capacity. Moreover, Romm [**127**] proves that the stable matching mechanism can still be manipulated even if the reported capacities (which may be different from the actual ones) are used during the admission process. Another problem related to ours is addressed by Yahiro and Yokoo [**152**], where the authors consider a profile of "resources" that can be allocated to "projects" (hospitals) and focus on designing strategy-proof and efficient mechanisms. Nguyen and Vohra [**117**] study the problem of ensuring stability in a matching market with couples, and find that by adding at most 4 extra capacities, the existence of a stable matching is guaranteed. Surpassing the approach of considering extra capacities a parameter, as assumed in the entry comparative static, Bobbio et al. [**38, 39**] consider for the first time the allocation of extra capacities to hospitals as a decision variable rather than a parameter; in their paper, the authors propose mixed-integer programming techniques to solve the problem of jointly allocating extra capacities while performing the matching. Another heuristic solution methodology to optimize the outcome for the residents is devised in [**9**]. Recently, also Dur and Van der Linden [**58**] study the problem of allocating capacities, the authors propose a general framework to devise a mechanism and they study the game theoretic properties of it. Finally, Kumano and Kurino [**101**] study the problem from both a theoretical and practical perspective. Their work was used to guide the reallocation of quotas at the University of Tsukuba in Japan. Our work focuses on providing the computational complexity landscape of the problem tackled in [**38, 39, 9, 58, 101**], as well as its counterpart where existing hospital spots are removed, and other variants. The main variant that we study is the allocation of extra capacities on a regional level. This is motivated by problems where some regions receive more residence applications than others. For instance, Kamada and Kojima [**85**] studies matching mechanisms that impose regional quotas for the Japan Residency Matching Program. Our work differs from theirs, since our goal is to optimize the quotas rather than imposing them. To the best of our knowledge the problem of reducing capacities has not yet been studied in the stable matching literature.

3.2. Preliminaries and Problem Definition

The many-to-one stable matching problem consists of a set of residents $\mathcal{S} = \{i_1, \ldots, i_{|\mathcal{S}|}\}$, a set of hospitals $\mathcal{C} = \{j_1, \ldots, j_{|\mathcal{C}|}\}$ and a set of edges $\mathcal{E}$ between $\mathcal{S}$ and $\mathcal{C}$. A resident and a hospital are linked by an edge in $\mathcal{E}$ if they deem each other acceptable. In this work, we assume (if not otherwise stated) that every resident-hospital pair is acceptable, *i.e.*, $\mathcal{E} = \mathcal{S} \times \mathcal{C}$. Each hospital $j \in \mathcal{C}$ has a non-negative integer capacity $c_j \in \mathbb{Z}_+$ that represents the maximum number of residents that hospital j can admit. In this setting, a *matching* M is a subset of $\mathcal{E}$ in which each hospital j appears in at most c_j pairs and each resident appears in at most one pair. We denote by $M(i)$ and $M(j)$ the hospital assigned to resident i and the subset of residents assigned to hospital j, respectively.

An instance Γ of the HR problem corresponds to a tuple $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{c} \rangle$, where $\mathbf{c} \in \mathbb{Z}_+^{\mathcal{C}}$ is the vector of capacities and $\succ$ corresponds to the profile of preferences that residents have over hospitals and vice-versa. Specifically, we assume that the preference list of each resident is a linear order. We use the notation $j \succ_i j'$ to describe when resident i prefers hospital j over hospital j'. We assume that every agent is *individually rational*, *i.e.*, every agent prefers the proposed assignment than to be unmatched. Concerning the preference list of every hospital, we assume it is a *responsive* linear order over the power-set of the residents [**133**].[4]If a preference list is a responsive linear order, it can be fully described by the linear order over single residents. We write $i \succ_j i'$ to denote when hospital j prefers resident i over i'. Whenever the context is clear, we drop the subscript in $\succ$. We emphasize that in the HR problem, unless otherwise stated, the preference lists are complete and strict (there are no *ties*). Under these assumptions, the length of the preference list of each agent, hospital or resident, is exactly the size of the other side of the bipartition. Therefore, preference lists can be interpreted in terms of rankings. Formally, for each resident $i \in \mathcal{S}$ and hospital $j \in \mathcal{C}$, we denote by $r_{i,j} \in \{1, \ldots, |\mathcal{C}|\}$ the rank of hospital j in the list of resident i. According to this notation, for example, the most preferred hospital has the lowest ranking. Analogously, we define $r_{j,i} \in \{1, \ldots, |\mathcal{S}|\}$ for all $j \in \mathcal{C},\ i \in \mathcal{S}$.

Given a matching M, we say that a pair $(i,j) \in \mathcal{E}$ is a *blocking pair* if the following two conditions are satisfied: (1) resident i is unassigned or prefers hospital j over $M(i)$, and (2) $|M(j)| < c_j$ or hospital j prefers resident i over at least one resident in $M(j)$. The matching M is said to be *stable* if it does not admit a blocking pair. Gale and Shapley [**67**] showed that every instance of the HR problem admits a stable matching that can be found in polynomial-time by the deferred acceptance method, also known simply as the Deferred Acceptance algorithm. In particular, this algorithm can be designed to prioritize the residents in the following sense: Let M and M' be two different stable matchings, we say that a resident i *weakly prefers* M over M'

[4]For any two subsets of residents $\mathcal{S}', \mathcal{S}''$, we denote that hospital j prefers $\mathcal{S}'$ over $\mathcal{S}''$ as $\mathcal{S}' \succ_j \mathcal{S}''$. A preference relation of a hospital is responsive if for every $\mathcal{S}' \subseteq \mathcal{S}$ with $|\mathcal{S}'| \leq c_j$, $s' \in \mathcal{S}'$ and $s'' \notin \mathcal{S}'$, we have that (i) $\mathcal{S}' \succ_j \mathcal{S}' \cup \{s''\} \setminus \{s'\}$ if and only if $\{s'\} \succ_j \{s''\}$, and (ii) $\mathcal{S}' \succ_j \mathcal{S}' \setminus \{s'\}$ if and only if $\{s'\} \succ_j \emptyset$. Therefore, a responsive preference list can be obtained from the linear order over singletons.

if $M(i) \succ_i M'(i)$ or $M(i) = M'(i)$. Then, the DA algorithm can be adapted to compute the unique stable matching that is weakly preferred by all residents over all the other possible stable matchings. Such unique stable matching is called *resident-optimal*.
Notation. To ease the exposition, we avoid using the symbol $\succ$ when presenting a preference list, instead we simply separate agents by "," and use the convention that the leftmost agents are the most preferred. For instance, we represent the preference list $w \succ w' \succ w''$ as w,w',w''. Throughout this work, for a given integer $k \geq 1$, we use the shorthand $[k] := \{1, \dots, k\}$. Finally, unless otherwise stated, we use indices i for residents and j for hospitals.

3.2.1. Problem Definition

In this work, we focus on the stable matchings that minimize the *average hospital rank* (or *cost*). Throughout the paper, we assume that the total capacity of the hospitals is at least the total number of residents, *i.e.*, $\sum_{j \in \mathcal{C}} c_j \geq |\mathcal{S}|$. If this assumption does not hold, we must define the cost of an un-assigned resident [**38, 39**]. A natural option is to add an artificial hospital with large capacity that is ranked last by every resident. Therefore, un-assigned residents will be allocated to the artificial hospital whose rank is $|\mathcal{C}| + 1$. Note that as a consequence of our assumption, $\sum_{j \in \mathcal{C}} c_j \geq |\mathcal{S}|$, there may be hospitals that do not fill their quota. The average hospital rank (for the residents) of a matching M is defined as

$$\text{AvgRank}(M) := \sum_{(i,j) \in M} r_{i,j}, \tag{3.2.1}$$

where, without loss of generality, we do not divide by the number of hospitals ranked by each resident.[5] We consider Expression (3.2.1) as our objective function, since one can easily show that a stable matching M is resident-optimal if, and only if, it is a stable matching of minimum average hospital rank [**38**].

In our first problem, initially introduced in [**38**], we aim to improve the allocation of residents by increasing the capacity of the hospitals. For a non-negative vector $\mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}}$, we denote by $\Gamma_{\mathbf{t}} = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{c} + \mathbf{t} \rangle$ an instance of the HR problem in which the capacity of each hospital $j \in \mathcal{C}$ is $c_j + t_j$. Observe that $\Gamma_{\mathbf{0}}$ corresponds to the original instance Γ with no capacity expansion. Formally, we define the capacity expansion problem as follows.

[5]Note that our main results are given for complete preference lists, which is a special case of the incomplete preference setting. Indeed, since we are assuming that the overall number of capacities is greater than the number of residents, every resident will be matched to a school. Moreover, every resident ranks the same number of hospitals ($|\mathcal{C}|$), therefore, we would be dividing every addend by the same constant $|\mathcal{C}|$.

Problem 4 ($\textsc{Min-Avg}_{\textsc{exp}}$ HR).

INSTANCE: A *HR* instance $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{c}\rangle$, a non-negative integer expansion budget $B \in \mathbb{Z}_+$, and a target value $K \in \mathbb{Z}_+$. QUESTION: Is there a non-negative vector $\mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}}$ and a matching $M_{\mathbf{t}}$ such that $$\textit{AvgRank}(M_{\mathbf{t}}) \leq K,$$ where $\mathbf{t}$ satisfies $\sum_{j\in\mathcal{C}} t_j = B$ and $M_{\mathbf{t}}$ is a stable matching in instance $\Gamma_{\mathbf{t}}$?

Given parameters B and K, Problem 4 aims to determine the existence of an allocation of B extra spots through vector $\mathbf{t}$ such that there is a stable matching with an average hospital rank of at most K.[6]

In our second problem, we aim to find the reduction of the hospitals' capacities such that the final average hospital rank is the lowest possible, *i.e.*, that has the least impact on the allocation of residents. As before, for a non-negative vector $\mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}}$, we denote by $\Gamma_{-\mathbf{t}} = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{c} - \mathbf{t}\rangle$ an instance of the HR problem in which the capacity of each hospital $j \in \mathcal{C}$ is $c_j - t_j$. Formally, we define our second problem as follows.

Problem 5 ($\textsc{Min-Avg}_{\textsc{red}}$ HR).

INSTANCE: A *HR* instance $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{c}\rangle$, a non-negative integer reduction budget $B \in \mathbb{Z}_+$ such that $-B + \sum_{j\in\mathcal{C}} c_j \geq \lvert\mathcal{S}\rvert$ and a target value $K \in \mathbb{Z}_+$. QUESTION: Is there a non-negative vector $\mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}}$ and a matching $M_{\mathbf{t}}$ such that $$\textit{AvgRank}(M_{\mathbf{t}}) \leq K,$$ where $\mathbf{t}$ satisfies $\sum_{j\in\mathcal{C}} t_j = B$ and $c_j - t_j \geq 0$ for every $j \in \mathcal{C}$, and $M_{\mathbf{t}}$ is a stable matching in instance $\Gamma_{-\mathbf{t}}$?

Note that in Problem 5, we have the additional constraint that the capacity of every hospital should remain non-negative after removing spots, *i.e.*, $c_j - t_j \geq 0$ for all $j \in \mathcal{C}$. We further assume that the sum of the reduced hospitals' capacities is greater than or equal to the number of residents, *i.e.*, $-B + \sum_{j\in\mathcal{C}} c_j \geq \lvert\mathcal{S}\rvert$. As in Problem 4, if this assumption does not hold, we can transform the instance by adding an artificial hospital with a large capacity (which is ranked last in every resident's list) and by only allowing the reduction of capacities to the original hospitals.

[6] Note that we can also require that $\sum_{j\in\mathcal{C}} t_j \leq B$; however, if we want to include a penalty for the unassigned students as suggested in Chapter 4, then, it may be useful to enforce the allocation of extra capacities to avoid choosing the optimal solution in which no capacities are distributed.

3.3. The Capacity Expansion Problem

Our main result in this section establishes the computational complexity and inapproximability of Problem 4. Denote by $\textsc{Min-Avg}_{\textsc{exp}}$ HR OPT the optimization version of Problem 4, *i.e.*, the problem of finding the allocation of extra spots and the stable matching in the expanded instance that minimizes AvgRank. Formally, our main result is the following.

Theorem 3.3.1. $\textsc{Min-Avg}_{\textsc{exp}}$ HR is NP-complete. Moreover, unless *P=NP*, for any $\varepsilon > 0$, $\textsc{Min-Avg}_{\textsc{exp}}$ HR OPT cannot be approximated within a factor of $\overline{n}^{(\frac{1}{6}-\varepsilon)}$, where $\overline{n}$ is the number of hospitals.

To provide insights on the difficulty of Problem 4, we present an example of how allocating one extra capacity ($B = 1$) using intuitive approaches may yield a sub-optimal solution. In real life instances, certain hospitals may be "more popular" than others, namely, some hospitals are the most preferred according to well-known voting methods such as majority count or Borda count [**156, 43**]. Thus, when $B = 1$, a natural approach is to assign the additional spot to the hospital that is preferred by the majority count or by the Borda count. However, as the following example shows, this is not necessarily optimal.
Counter example for the majority and for the Borda count. Let $\mathcal{S} = \{i_1,i_2,i_3,i_4,i_5,i_6\}$ and $\mathcal{C} = \{j_1,j_2,j_3,j_4\}$. What follows are the preference lists of the residents and of the hospitals.

$$i_1 : j_2 \succ j_1 \succ j_3 \succ j_4 \qquad j_1 : i_1 \succ i_2 \succ \cdots \succ i_6$$
$$i_2 : j_2 \succ j_3 \succ j_1 \succ j_4 \qquad j_2 : i_1 \succ i_2 \succ \cdots \succ i_6$$
$$i_3 : j_3 \succ j_2 \succ j_4 \succ j_1 \qquad j_3 : i_1 \succ i_2 \succ \cdots \succ i_6$$
$$i_4 : j_1 \succ j_4 \succ j_3 \succ j_2 \qquad j_4 : i_1 \succ i_2 \succ \cdots \succ i_6$$
$$i_5 : j_1 \succ j_4 \succ j_3 \succ j_2$$
$$i_6 : j_1 \succ j_4 \succ j_3 \succ j_2$$

Hospitals j_1, j_2 and j_3 have each capacity 1, and hospital j_4 has capacity 3. The resident-optimal stable matching is $M = \{(i_1,j_2),(i_2,j_3),(i_3,j_4),(i_4,j_1),(i_5,j_4),(i_6,j_4)\}$ with AvgRank$(M) = 11$. Now, consider Problem 4 with $B = 1$ and $K = 9$. For this instance, an intuitive solution is allocating the extra spot to j_1, which is the most preferred hospital according to both Majority vote and Borda vote; the allocation of one extra capacity to j_1 is sub-optimal. Indeed, if we expand the capacity $c_{j_1} = 1$ to $c_{j_1} = 2$, then resident i_5 would be assigned to hospital j_1, which leaves an extra spot in hospital j_4. This solution reduces the average hospital rank by 1 unit and the resulting matching does not meet the target $K = 9$. Instead, if we expand the capacity of j_2 to 2, then resident i_2 is admitted by hospital j_2, leaving an empty spot in hospital j_3 that is filled by resident i_3; the resulting matching has an average hospital rank of

8. □

As the previous example shows, the allocation of one extra spot is not trivial when we try to solve it by just looking at the residents' preferences. However, we can still solve this problem, where $B = 1$, in polynomial-time by doing an exhaustive search in combination with the DA algorithm. To achieve this, we compute the resident-optimal stable matching using DA in the instance $\Gamma_{\mathbf{t}}$ with $\mathbf{t} = \mathbf{1}_j$ for each $j \in \mathcal{C}$, where $\mathbf{1}_j \in \{0,1\}^{\mathcal{C}}$ is the indicator vector whose j-th component is 1 and the rest is 0. Once we obtain the cost for each $j \in \mathcal{C}$, we output the resident-optimal stable matching of minimum average hospital rank. Finally, we compare with our target K to decide if such an allocation exists or not. Since the DA algorithm's runtime complexity is $O(|\mathcal{S}| \cdot |\mathcal{C}|)$ [**67**], then exhaustive search runs in $O(|\mathcal{S}| \cdot |\mathcal{C}|^2)$. Whether this can be improved remains an open question.

To prove Theorem 3.3.1, we first study a variant of the *egalitarian stable marriage* problem [**107**]. Formally, the stable marriage (SM) problem corresponds to the HR problem where $c_j = 1$ for all $j \in \mathcal{C}$. We use SMT to indicate the version of SM when ties are present in the preference lists. A tie appears when an agent allocates in the same position of the list two different participants of the opposite side. For example, if the preference list for resident i is $j_4, \left(j_1, j_3\right), j_2$,[7] then the rankings are $r_{i,j_4} = 1$, $r_{i,j} = 2$ for $j \in \{j_1,j_3\}$ and $r_{i,j_2} = 4$. For the SMT problem, stable matchings can be defined in several ways, but in this paper we consider *weak stability* [**107**]. Formally, a matching M is weakly stable if there is no pair such that both agents *strictly* prefer each other over their allocation in M. An egalitarian stable matching is a stable matching that minimizes the total sum of the rankings, *i.e.*, $\sum_{(i,j)\in M}[r_{i,j} + r_{j,i}]$. Manlove et al. [**108**] proved that the problem of finding an egalitarian stable matching for SMT is not approximable within $\overline{n}^{1-\epsilon}$, for any $\epsilon > 0$, unless P $=$ NP, where $\overline{n}$ is the number of hospitals. For more details, we refer to Theorem 7 in [**108**].

Let us define the following variant of the egalitarian SMT problem.

Problem 6 ($\textsc{Min-w SMT}$)**.**

INSTANCE: An *SMT* instance $\Gamma = \langle \mathcal{S},\mathcal{C}, \succ, \mathbf{c}\rangle$ with $c_j = 1$ for all $j \in \mathcal{C}$ and a target value $K \in \mathbb{Z}_+$.
QUESTION: Is there a weakly stable matching M such that *AvgRank*$(M) \leq K$?

We use $\textsc{Min-w SMT opt}$ to denote the optimization version of $\textsc{Min-w SMT}$, *i.e.*, the problem of finding a weakly stable matching that minimizes AvgRank. Using the ideas in [**108**], we can obtain the following result for $\textsc{Min-w SMT}$.

[7] Throughout this chapter, $\left(\cdot\right)$ brackets denote a tie.

Lemma 3.3.2. $\textsc{Min-w SMT}$ is NP-complete. Moreover, for any $\varepsilon > 0$, $\textsc{Min-w SMT opt}$ is not approximable within a factor of $\overline{n}^{1-\varepsilon}$, unless $\mathrm{P} = \mathrm{NP}$, where $\overline{n} = |\mathcal{C}|$. This result holds even if ties are only on residents' side, there is at most one tie per list, and each tie is of length two.

For completeness, we provide the proof of this corollary in the Appendix. Let us now provide a sketch of the steps to prove Theorem 3.3.1. Given an instance Γ of $\textsc{Min-w SMT}$, we construct the following instance $\hat{\Gamma}$ of $\textsc{Min-Avg}_{\textsc{exp}}$ HR: For every resident in Γ that has ties in its preference list, we create a *village* of residents and hospitals with different capacities and strict preferences. Then, we create multiple copies of each of these villages. In Lemma 3.3.5, we prove that the construction can be done in polynomial-time and it leads to an associated stable matching in the new instance. Let M be the stable matching of minimum average hospital rank in $\textsc{Min-w SMT}$; in Lemma 3.3.6, we prove that the associated stable matching $\hat{M}_{\mathbf{t}}$ in $\hat{\Gamma}$ is in fact the stable matching of minimum average hospital rank in $\textsc{Min-Avg}_{\textsc{exp}}$ HR.

3.3.1. Design of the Instance

First, we observe that $\textsc{Min-w SMT}$ is NP-complete even if ties occur only among the preference lists of residents, in each preference list there is at most one tie of length 2, and it is positioned at the head of the list. For more details, we refer to Remark 3.7.1 in the Appendix. Throughout this section, we assume that an instance of SMT satisfies these properties. Now, we introduce a polynomial transformation from such an instance of $\textsc{Min-w SMT}$ to an instance of $\textsc{Min-Avg}_{\textsc{exp}}$ HR.

Let $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{c}\rangle$ be an instance of $\textsc{Min-w SMT}$ such that $|\mathcal{S}| = |\mathcal{C}| = n$. Let $L \leq n$ be the number of residents with ties in their preference list. The set of residents is partitioned in two sets $\mathcal{S} = \mathcal{S}' \cup \mathcal{S}''$, where $\mathcal{S}'$ is the set of residents with a tie of length two at the head of the preference list and $\mathcal{S}''$ is the set of residents with a strict preference list. Henceforth, we fix an ordering of the residents in $\mathcal{S}$ and denote $\mathcal{S}' = \{i_1, \ldots, i_L\}$ and $\mathcal{S}'' = \{i_{L+1}, \ldots, i_n\}$. Since preference lists are complete, observe that in any weakly stable matching every resident is matched.[8]

In the following, we create an instance $\hat{\Gamma} = \langle \hat{\mathcal{S}}, \hat{\mathcal{C}}, \hat{\succ}, \hat{\mathbf{c}}\rangle$ of $\textsc{Min-Avg}_{\textsc{exp}}$ HR with a specific target value and budget.
Hospitals and residents. First, we add n^4 copies of the residents in $\mathcal{R}''$, $\overline{\mathcal{R}}'' = \{i^k_{L+1}, \ldots, i^k_n\}_{k\in[n^4]}$, and we introduce the set of residents $\mathcal{A} = \{a_1, \ldots, a_{n^6}\}$. Moreover, we create n^4 copies of the hospitals in $\mathcal{C}$: $\overline{\mathcal{C}} = \{j^k_1, \ldots, j^k_n\}_{k\in[n^4]}$. We also introduce a set $\mathcal{Z} = \{z_1, \ldots, z_{n^6}\}$ of hospitals of size n^6, and a set $\mathcal{X} = \{x_1, \ldots, x_{n^7}\}$ of hospitals of size n^7.

Recall that we index the residents in $\mathcal{S}'$ as $i_1, \ldots, i_L$. For every resident $i_\ell \in \mathcal{S}'$ ($\ell \in [L]$), and $k \in [n^4]$ we introduce a structure $\mathcal{B}^k_\ell$ that we call *village*, which is composed of

[8] This follows from the hypothesis that preference lists are complete and that the total capacity of the hospitals can accommodate all the residents.

- A set of residents $\mathcal{W}_\ell^k = \{w_{\ell,e}^k\}_{e\in[n]}$.
- A resident y_ℓ^k.
- Two sets of hospitals $\mathcal{V}_\ell^k = \{v_{\ell,e}^k\}_{e\in[n]}$ and $\overline{\mathcal{V}}_\ell^k = \{\overline{v}_{\ell,e}^k\}_{e\in[n]}$. Two sets of residents $\mathcal{U}_\ell^k = \{u_{\ell,e}^k\}_{e\in[n]}$ and $\overline{\mathcal{U}}_\ell^k = \{\overline{u}_{\ell,e}^k\}_{e\in[n]}$.

We denote as $\mathcal{V} := \bigcup_{\ell=1}^L \bigcup_{k=1}^{n^4} \mathcal{V}_\ell^k$, $\overline{\mathcal{V}} := \bigcup_{\ell=1}^L \bigcup_{k=1}^{n^4} \overline{\mathcal{V}}_\ell^k$, $\mathcal{U} := \bigcup_{\ell=1}^L \bigcup_{k=1}^{n^4} \mathcal{U}_\ell^k$ and $\overline{\mathcal{U}} := \bigcup_{\ell=1}^L \bigcup_{k=1}^{n^4} \overline{\mathcal{U}}_\ell^k$. We also denote by $\mathcal{Y} := \bigcup_{\ell=1}^L \bigcup_{k=1}^{n^4} \{y_\ell^k\}$, and $\mathcal{V}^k := \bigcup_{\ell=1}^L \mathcal{V}_\ell^k$. In summary, the new instance $\hat{\Gamma}$ is made of a the set of residents

$$\hat{\mathcal{S}} = \overline{\mathcal{S}}'' \cup \{\mathcal{W}_\ell^k\}_{\ell\in[L],k\in[n^4]} \cup \mathcal{Y} \cup \mathcal{A} \cup \mathcal{U} \cup \overline{\mathcal{U}},$$

and a set of hospitals

$$\hat{\mathcal{C}} = \overline{\mathcal{C}} \cup \mathcal{X} \cup \mathcal{V} \cup \overline{\mathcal{V}} \cup \mathcal{Z}.$$

The sole purpose of the hospitals in $\mathcal{X}$ is to ensure that there are sufficient capacities for matching all the residents. The set of hospitals $\mathcal{Z}$ is introduced to make costly certain allocations of extra capacities. The sets $\mathcal{V}$ and $\overline{\mathcal{V}}$ are used to leverage stability and ensure that multiple allocations of extra spots yield sub-optimal solutions.

Capacity vector. Every hospital $j \in \hat{\mathcal{C}}$ has capacity one, *i.e.*, $\hat{c}_j = 1$.

Preference lists. We now proceed to construct the preference lists in $\hat{\Gamma}$.

Given a resident $i_\ell \in \mathcal{R}'$ with $\ell \in [L]$, let $\big(j_{\sigma_1}, j_{\sigma_2} \big), j_{\sigma_3} \ldots$ be her ranking of the hospitals in the original instance Γ (recall that the $(\cdot)$ parenthesis symbolizes the tie at the head of the list). We provide the preference lists of the residents and hospitals in village $\mathcal{B}_\ell^k$ with $\ell \in [L]$ and $k \in [n^4]$, namely

$$
\begin{aligned}
w_{\ell,1}^k &: v_{\ell,1}^k, \overline{\mathcal{V}}_\ell^k \setminus \{\overline{v}_{\ell,1}^k\}, \mathcal{V}_+^{k+1}[n^2], j_{\sigma(\ell,1)}^k, \mathcal{Z}, \ldots, \mathcal{X} \\
w_{\ell,2}^k &: v_{\ell,2}^k, \overline{\mathcal{V}}_\ell^k \setminus \{\overline{v}_{\ell,2}^k\}, \mathcal{V}_+^{k+1}[n^2+1], j_{\sigma(\ell,2)}^k, \mathcal{Z}, \ldots, \mathcal{X} \\
w_{\ell,e}^k &: v_{\ell,e}^k, \overline{\mathcal{V}}_\ell^k \setminus \{\overline{v}_{\ell,e}^k\}, \mathcal{V}_+^{k+1}[(n^2-1)e+2], j_{\sigma(\ell,e)}^k, \mathcal{Z}, \ldots, \mathcal{X} \qquad e \in \{3, \ldots, n\} \\
y_\ell^k &: \overline{v}_{\ell,2}^k, \overline{v}_{\ell,1}^k, \overline{v}_{\ell,3}^k, \ldots, \overline{v}_{\ell,n}^k, \mathcal{V}_+^{k+1}, \mathcal{Z}, \ldots, \mathcal{X} \\
v_{\ell,e}^k &: u_{\ell,e}^k, \mathcal{A}, w_{\ell,e}^k, \{w_{\ell,e}^{k'}\}_{k'\neq k}, \{\mathcal{W}_\ell^k\}_{\ell,k} \setminus \{w_{\ell,e}^{k'}\}_{k'}, \ldots, \mathcal{Y} \qquad e \in [n] \\
u_{\ell,e}^k &: v_{\ell,e}^k, \ldots \qquad e \in [n] \\
\overline{v}_{\ell,e}^k &: \overline{u}_{\ell,e}^k, \mathcal{A}, \mathcal{W}_\ell^k \setminus \{w_{\ell,e}^k\}, y_\ell^k, \{\mathcal{W}_\ell^{k'}\}_{\ell,k'\neq k} \setminus \{w_{\ell,e}^{k'}\}_{k'\neq k}, \ldots, \mathcal{Y} \setminus \{y_\ell^k\} \qquad e \in [n] \\
\overline{u}_{\ell,e}^k &: \overline{v}_{\ell,e}^k, \ldots \qquad e \in [n]
\end{aligned}
$$

where $j_{\sigma(\ell,e)}$ is the hospital listed in position e by resident i_ℓ, and $\mathcal{V}_+^k$ is the set of hospitals $\mathcal{V}$ listed as follows: $(\mathcal{V}^k, \ldots \mathcal{V}^{n^4}, \mathcal{V}^1, \ldots, \mathcal{V}^{k-1})$, for $k \in [n^4]$; note that when $k \geq n^4$ then we take $\mathcal{V}_+^k$ for $k \pmod{n^4}$. By $\mathcal{V}_+^k[n]$, we denote the first n elements of the ordered set $\mathcal{V}_+^k$. The purpose of positioning set $\mathcal{V}_+^{k+1}$ in the preference lists of the residents $w_{\ell,e}^k$, $(e \in [n])$, is to ensure that we can mimic $r_{i_\ell,j}$, $j \in \mathcal{C}$, of the original instance while also ensuring that multiple allocation of extra capacities to the same hospital are sub-optimal. The symbol "…" means that the agents on the other side of the bipartition not explicitly listed are ranked strictly and arbitrarily.

Now, we present the preference list of a hospital $j_e^k \in \overline{\mathcal{C}}$, which is a modification of the preference list of hospital $j_e \in \mathcal{C}$. We modify the original preference list of j_e by substituting every resident $i_\ell \in \mathcal{R}'$, $\ell \in [L]$, with resident $w_{\ell,r}^k$ where r is the position of hospital j_e in the preference list of resident i_ℓ (e.g., $r = 2$ if j_e is the second hospital listed in the tie at the head of the preference list of i_ℓ); at the head of the preference list of j_e^k, we position $\mathcal{A}$. Concerning the residents in $\mathcal{R}''$, we substitute every $i_\ell \in \mathcal{R}''$, ($\ell \in \{L+1, \ldots, n\}$), with i_ℓ^k. Then, hospital j_h^k ranks arbitrarily a strict ordering of the remaining residents, and at the end $\mathcal{Y}$.

We introduce the preference list of a resident $i_\ell^k \in \overline{\mathcal{S}}''$ by modifying the strict preference list of the corresponding resident $i_\ell \in \mathcal{S}''$: $j_{\sigma(\ell,1)}, \ldots, j_{\sigma(\ell,n)}$ in Γ. The preference list of i_ℓ^k in our new instance $\hat{\Gamma}$ is

$$\mathcal{V}_+^{k+1}[n^2+2], j_{\sigma_1}^k, \mathcal{V}_+^{k+1}[n^2+2,1], j_{\sigma_2}^k, \mathcal{V}_+^{k+1}[n^2+2,2], \ldots$$
$$\ldots, \mathcal{V}_+^{k+1}[n^2+2, n-1], j_{\sigma_n}^k, \mathcal{Z} \ldots, \mathcal{X}$$

where $V_+^{k+1}[b,c]$ is the set of n^2-1 hospitals from $\mathcal{V}_+^{k+1}$ that is listed from index $b + (n^2-1)\cdot c$ onward.[9]

The preference lists of residents and hospitals in $\mathcal{A}, \mathcal{Z}$ are as follow for every $p \in [n^6]$

$$z_p : a_p, \ldots$$
$$a_p : z_p, \ldots.$$

The preference lists of hospitals in $\mathcal{X}$ are arbitrary. Finally, it is worth noting that every stable matching in $\hat{\Gamma}$ will include the following pairs: $\{(u_{\ell,e}^k, v_{\ell,e}^k), (\overline{u}_{\ell,e}^k, \overline{v}_{\ell,e}^k)\}_{\ell \in [L], e \in [n], k \in [n^4]}$, and $\{(a_p, z_p)\}_{p \in [n^6]}$, as they rank each other in the first position.

Target value and budget. Consider an instance of $\textsc{Min-w SMT}$ with a given target K, where L is the number of residents with a tie at the head of their preference list. We define the target $\overline{K} = n^6 \cdot K + n^6 + 4n^5 L + 3n^5 - n^4 L$ and budget $B = L \cdot n^5$ for $\textsc{Min-Avg}_{\text{EXP}}$ HR.

Finally, note that $\hat{\mathcal{S}} = \overline{\mathcal{S}}'' \cup \{\mathcal{W}_\ell^k\}_{\ell \in [L], k \in [n^4]} \cup \mathcal{Y} \cup \mathcal{A} \cup \mathcal{U} \cup \overline{\mathcal{U}}$ and $\hat{\mathcal{C}} = \overline{\mathcal{C}} \cup \mathcal{X} \cup \mathcal{V} \cup \overline{\mathcal{V}} \cup \mathcal{Z}$. Therefore, the instance $\hat{\Gamma}$ consists of $((n-L)n^4 + L(n+1)n^4 + n^6 + 2n^5 L) + (n^5 + n^7 + 2n^5 L + n^6)$ $= n^7 + 2n^6 + n^5(2+5L)$ residents and hospitals, which is $O(n^7)$; therefore the construction can be done in polynomial-time.

Example 3.3.3. We present an example of an instance of $\textsc{Min-w SMT}$ and provide its reduced instance to an instance of $\textsc{Min-Avg}_{\text{EXP}}$ HR. Consider an instance I with $K = 3$, $\mathcal{C} = \{j_1, j_2, j_3\}$ and $\mathcal{S} = \{i_1, i_2, i_3\}$. Every agent has capacity 1, and the preference lists are as follows.

[9] Note that if we go over the limit, we consider again the modulo class.

$$
\begin{array}{ll}
i_1 : (j_1, j_2), j_3 & j_1 : i_1, i_2, i_3 \\
i_2 : (j_3, j_2), j_1 & j_2 : i_2, i_1, i_3 \\
i_3 : j_2, j_1, j_3 & j_3 : i_1, i_3, i_2
\end{array}
$$

Let us now provide the reduced instance of $\textsc{Min-Avg}_{\textsc{exp}}$ HR. The residents are $\overline{\mathcal{S}}'' = \{i_3\}$, $\{\mathcal{W}_\ell^k\}_{\ell \in [L], k \in [n^4]} = \{w_{l,1}^k, w_{l,2}^k, w_{l,3}^k\}_{\ell \in \{1,2\}, k \in [81]}$, $\mathcal{Y} = \{y_l^k\}_{\ell \in \{1,2\}, k \in [81]}$, $\mathcal{A} = \{a_1, \ldots, a_{729}\}$, $\mathcal{U} = \{u_{l,1}^k, u_{l,2}^k, u_{l,3}^k\}_{\ell \in \{1,2\}, k \in [81]}$, $\overline{\mathcal{U}} = \{\overline{u}_{l,1}^k, \overline{u}_{l,2}^k, \overline{u}_{l,3}^k\}_{\ell \in \{1,2\}, k \in [81]}$. The hospitals are $\overline{\mathcal{C}} = \{j_1^k, j_2^k, j_3^k\}_{k \in [81]}$, $\mathcal{X} = \{x_1, \ldots, x_{2187}\}$, $\mathcal{V} = \{v_{l,1}^k, v_{l,2}^k, v_{l,3}^k\}_{\ell \in \{1,2\}, k \in [81]}$, $\overline{\mathcal{V}} = \{\overline{v}_{l,1}^k, \overline{v}_{l,2}^k, \overline{v}_{l,3}^k\}_{\ell \in \{1,2\}, k \in [81]}$, $\mathcal{Z} = \{z_1, \ldots, z_{729}\}$. The budget of extra capacities is $B = 486$, and the objective value is $\overline{K} = 2187 + 729 + 1944 + 729 - 162 = 5427$. The preferences of the agents in the new instance are as follows.

$$
\begin{array}{lll}
 & a_1 : z_1, \ldots & z_1 : a_1, \ldots \\
 & \vdots & \vdots \\
 & a_{729} : z_{729}, \ldots & z_{729} : a_{729}, \ldots \\
 & & x_1 : \ldots \\
 & & \vdots \\
 & & x_{2187} : \ldots \\
\forall k \in [81] : & & \\
 & i_3^k : r_{i_3^k} & j_1^k : w_{1,1}^k, w_{2,2}^k, i_3^k, \ldots \\
 & & j_2^k : w_{2,3}^k, w_{1,2}^k, i_3^k, \ldots \\
 & & j_3^k : w_{1,3}^k, i_3^k, w_{2,1}^k, \ldots \\
 & \text{residents in } \mathcal{B}_1^k & \text{hospitals in } \mathcal{B}_1^k \\
 & \text{residents in } \mathcal{B}_2^k & \text{hospitals in } \mathcal{B}_2^k
\end{array}
$$

where the preference list of i_3^k is $r_{i_3^k} = v_1^{k+1}, v_2^{k+1}, v_3^{k+1}, \ldots, v_2^{k+4}, j_2^k, v_3^{k+4}, \ldots, v_1^{k+7}, j_1^k, v_2^{k+7}, \ldots, v_3^{k+9}, j_3^k, \mathcal{Z}, \ldots, \mathcal{X}$.

To illustrate, we present the preference lists of the residents in $\mathcal{B}_1^k$ and the preference lists of the hospitals in $\mathcal{B}_1^k$. Given $k \in [81]$, the preference lists of the residents in $\mathcal{B}_1^k$ are as follows.

$$
w_{1,1}^k : v_{1,1}^k, \overline{v}_{1,2}^k, \overline{v}_{1,3}^k, v_{1,1}^{k+1}, \ldots, v_{1,3}^{k+3}, j_1^k, \mathcal{Z}, \ldots, \mathcal{X}
$$

$$
\begin{aligned}
w_{1,2}^k &: v_{1,2}^k, \overline{v}_{1,1}^k, \overline{v}_{1,3}^k, v_{1,1}^{k+1}, \ldots, v_{1,1}^{k+4}, j_2^k, \mathcal{Z}, \ldots, \mathcal{X} \\
w_{1,3}^k &: v_{1,3}^k, \overline{v}_{1,1}^k, \overline{v}_{1,2}^k, v_{1,1}^{k+1}, \ldots, v_{1,3}^{k+9}, j_3^k, \mathcal{Z}, \ldots, \mathcal{X} \\
y_1^k &: \overline{v}_{1,2}^k, \overline{v}_{1,1}^k, \overline{v}_{1,3}^k, V_+^{k+1}, \mathcal{Z}, \ldots, \mathcal{X} \\
u_{1,1}^k &: v_{1,1}^k, \ldots \\
u_{1,2}^k &: v_{1,2}^k, \ldots \\
u_{1,3}^k &: v_{1,3}^k, \ldots \\
\overline{u}_{1,1}^k &: \overline{v}_{1,1}^k, \ldots \\
\overline{u}_{1,2}^k &: \overline{v}_{1,2}^k, \ldots \\
\overline{u}_{1,3}^k &: \overline{v}_{1,3}^k, \ldots
\end{aligned}
$$

Given $k \in [81]$, the preference lists of the hospitals in $\mathcal{B}_1^k$ are as follows.

$$
\begin{aligned}
v_{1,1}^k &: u_{1,1}^k, \mathcal{A}, w_{1,1}^k, \{w_{1,1}^{k'}\}_{k' \neq k}, \{\mathcal{W}_\ell^k\}_{\ell,k} \setminus \{w_{1,1}^{k'}\}_{k'}, \ldots, \mathcal{Y} \\
v_{1,2}^k &: u_{1,2}^k, \mathcal{A}, w_{1,2}^k, \{w_{1,2}^{k'}\}_{k' \neq k}, \{\mathcal{W}_\ell^k\}_{\ell,k} \setminus \{w_{1,2}^{k'}\}_{k'}, \ldots, \mathcal{Y} \\
v_{1,3}^k &: u_{1,3}^k, \mathcal{A}, w_{1,3}^k, \{w_{1,3}^{k'}\}_{k' \neq k}, \{\mathcal{W}_\ell^k\}_{\ell,k} \setminus \{w_{1,3}^{k'}\}_{k'}, \ldots, \mathcal{Y} \\
\overline{v}_{1,1}^k &: \overline{u}_{1,1}^k, \mathcal{A}, \mathcal{W}_1^k \setminus \{w_{1,1}^k\}, y_1^k, \{\mathcal{W}_1^{k'}\}_{1,k' \neq k} \setminus \{w_{1,1}^{k'}\}_{k' \neq k}, \ldots, \mathcal{Y} \setminus \{y_1^k\} \\
\overline{v}_{1,2}^k &: \overline{u}_{1,2}^k, \mathcal{A}, \mathcal{W}_1^k \setminus \{w_{1,2}^k\}, y_1^k, \{\mathcal{W}_1^{k'}\}_{1,k' \neq k} \setminus \{w_{1,2}^{k'}\}_{k' \neq k}, \ldots, \mathcal{Y} \setminus \{y_1^k\} \\
\overline{v}_{1,3}^k &: \overline{u}_{1,3}^k, \mathcal{A}, \mathcal{W}_1^k \setminus \{w_{1,3}^k\}, y_1^k, \{\mathcal{W}_1^{k'}\}_{1,k' \neq k} \setminus \{w_{1,3}^{k'}\}_{k' \neq k}, \ldots, \mathcal{Y} \setminus \{y_1^k\}
\end{aligned}
$$

For $k \in [81]$, the preference lists of the residents in $\mathcal{B}_2^k$ and the preference lists of the hospitals in $\mathcal{B}_2^k$ are similar. □

Remark 3.3.4. Note that if no extra spots are assigned in our new instance, the set of hospitals $\mathcal{X}$ ensures that the residents are always matched.[10] Matching the residents to the hospitals in $\mathcal{X}$ leads to a higher average hospital rank. In the following section, we prove that it is optimal to assign $L \cdot n^5$ extra capacities to the hospitals in $\mathcal{V}$, whose initial capacity is one.

3.3.2. Useful Lemmata

In this section we provide two insightful lemmata which will be useful to the proof of our main result; recall that we are considering a budget $B = L \cdot n^5$, where $L = |\mathcal{S}'|$ is the number of residents with a tie in their preference list.

[10]This could be done also by adding a copy of themselves at the end of their list. Matching with oneself is another usual way of representing unassigned residents.

In the next lemma, we prove that the reduction previously introduced maps a weakly stable matching of SMT into a stable matching of HR, where the average rank of the new matching is proportional to the average rank of the one in SMT. The proof can be found in Appendix 3.7.2.

Lemma 3.3.5. For every weakly stable matching M in Γ with *AvgRank*$(M) = K_M$, there is an allocation $\mathbf{t}$ respecting the budget $B = L \cdot n^5$ and a stable matching $\hat{M}_{\mathbf{t}}$ in $\hat{\Gamma}_{\mathbf{t}} = \langle \hat{\mathcal{S}}, \hat{\mathcal{C}}, \hat{\succ}, \hat{\mathbf{c}} + \mathbf{t} \rangle$ with *AvgRank*$(\hat{M}_{\mathbf{t}}) = n^6 \cdot K_M + n^6 + 4n^5 L + 3n^5 - n^4 L$.

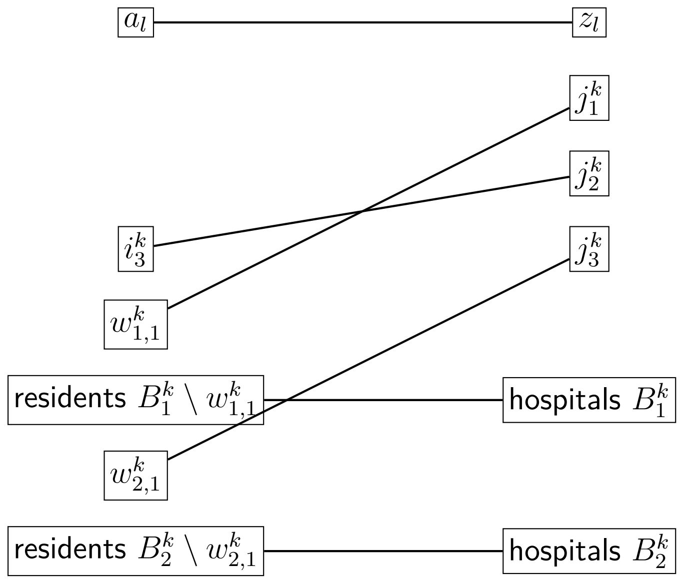
In the next result, we show that the allocation vector and the stable matching constructed in the proof of Lemma 3.3.5 correspond to the solution with the minimum average hospital rank, as long as the original matching of Γ is of minimum average hospital rank. The proof can be found in Appendix 3.7.3.

Lemma 3.3.6. Consider a weakly stable matching M in Γ of minimum average hospital rank. Then, the allocation $\mathbf{t}$ and the stable matching $\hat{M}_{\mathbf{t}}$ constructed in Lemma 3.3.5 constitute a solution of minimum average hospital rank for $\hat{\Gamma}$ when $B = L \cdot n^5$.

Example 3.3.7. Given a weakly stable matching in the instance of $\textsc{Min-w SMT}$ from Example 3.3.3, we find a stable matching in the reduced instance of $\textsc{Min-Avg}_{\textsc{exp}}$ HR.

We consider the weakly stable matching $\mu = \{(i_1, j_1), (i_2, j_3), (i_3, j_2)\}$, which has cost 3. The reduction allocates one extra capacity to each school $v^k_{1,2}, v^k_{1,3}, \overline{v}^k_{1,1}$ of village $\mathcal{B}^k_1$, and one extra capacity to each school $v^k_{2,2}, v^k_{2,3}, \overline{v}^k_{2,1}$ of village $\mathcal{B}^k_2$, for every $k \in [81]$. Note that a total of 486 extra capacities are allocated, thus exhausting the overall budget B. Therefore, the matching induced by the reduction is made of the following pairs: $\{(a_i, z_i)\}_{i \in [729]}$, and, for every $k \in [81]$, $\{(w^k_{l,2}, v^k_{l,2}), (w^k_{l,3}, v^k_{l,3}), (y^k_l, \overline{v}^k_{l,1})\}_{\ell \in \{1,2\}}$, $(w^k_{1,1}, j^k_1)$, $(w^k_{2,1}, j^k_3)$, (i^k_3, j^k_2), $\{(u^k_{l,h}, v^k_{l,h}), (\overline{u}^k_{l,h}, \overline{v}^k_{l,h})\}_{l \in [2], h \in [3]}$. The overall cost of the matching just described is $729 + 81 \cdot (2 \cdot (1 + 1 + 2) + 13 + 13 + 12 + 2 \cdot 3 \cdot (1 + 1)) = 5427$, which is exactly equal to the objective value $\overline{K}$ of the reduced instance. In Figure 1 we provide a graphic illustration of the matching in the reduced instance, note that $w^k_{1,1}$ ($w^k_{2,1}$) is the only resident of village B^k_1 (B^k_2) that is not matched with a school in their own village. □

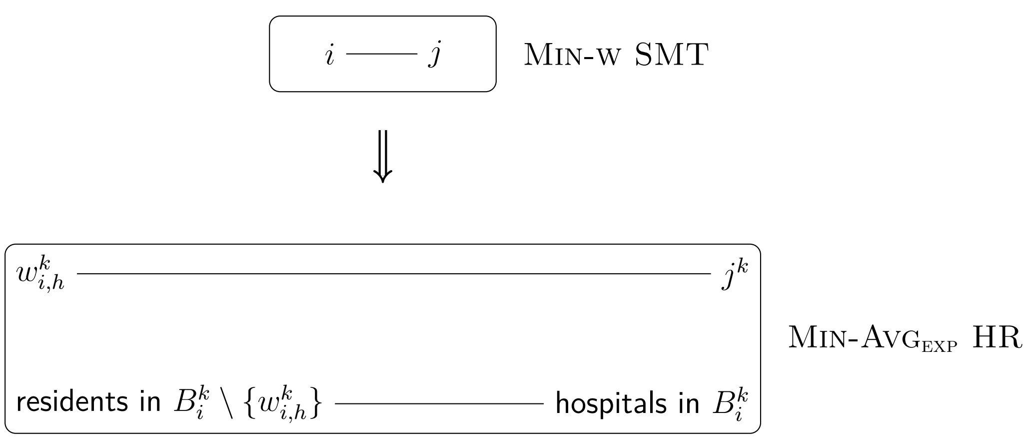
As illustrated in Example 3.3.3 and 3.3.7, the role of the village is to reproduce the functioning of ties in a stable matching where agents rank no ties. Indeed, for every resident i that ranks a tie in an instance of $\textsc{Min-w SMT}$, we create a village in the new instance of $\textsc{Min-Avg}_{\textsc{exp}}$ HR: The village contains n copies of resident i and it contains other hospitals and residents. When resident i is matched to hospital j in the instance of $\textsc{Min-w SMT}$, the overall structure of the village allows to match exactly one copy of i to a copy of j. In this way, we mimic the original matching while adding a fixed cost of the village to the total cost of the matching. We display this idea in Figure 2: Assume resident i ranks hospital j at position h and i has a preference list that includes a tie; if i and j are matched in the instance $\textsc{Min-w SMT}$, then, in the reduced



Note: $l \in [729]$ and $k \in [81]$.

Fig. 1. Example of stable matching in the reduced instance.

instance, we match the copy $w_{i,h}^k$ to the hospital-copy j^k; note that the rest of the village B_i^k is matched to itself. Remarkably, we can match the remaining residents of the village B_i^k to the hospitals in the same village in a stable way. All the agents in the village B_i^k are displayed in colour red.



Note: Resident i ranks hospital j in position h, and resident i has a tie in the preference list. If i is matched with j, then, in the reduced instance, we match $w_{i,h}^k$ to j^k and all the other residents of village B_i^k are matched to the hospitals of B_i^k.

Fig. 2. Illustration of the key idea of the village.

3.3.3. NP-completeness of the capacity expansion problem

In the following, we prove the main result of this section, Theorem 3.3.1.

Proof of Theorem 3.3.1. $\textsc{Min-Avg}_{\text{EXP}}$ HR is clearly in NP since given $\mathbf{t}$ and a matching $\hat{M}_\mathbf{t}$ in instance $\hat{\Gamma}_\mathbf{t}$, we can verify in polynomial-time whether $\hat{M}_\mathbf{t}$ is stable, whether the budget B is allocated and whether its objective value is less than the target value. We now show that $\textsc{Min-Avg}_{\text{EXP}}$ HR is NP-complete.

From Lemma 3.3.2, we know that $\textsc{Min-w}$ SMT is NP-complete. Consider the reduction given in Section 3.3.1. In the constructed instance $\hat{\Gamma}$ of $\textsc{Min-Avg}_{\text{EXP}}$ HR, we set the budget to $B = L \cdot n^5$ and the target value to $\overline{K} = n^6 \cdot K + n^6 + 4n^5 L + 3n^5 - n^4 L$, where K is the target value of the instance of $\textsc{Min-w}$ SMT.

First, suppose that the answer to the instance of $\textsc{Min-w}$ SMT is NO, *i.e.*, there is no weakly stable matching M with an average hospital rank less or equal than K. Let M be a weakly stable matching with minimum average hospital rank K_M; note that $K_M > K$. Next, we prove that there is no allocation of extra positions and a stable matching in the respective instance $\hat{\Gamma}$ of $\textsc{Min-Avg}_{\text{EXP}}$ HR with an objective value less or equal than $n^6 \cdot K + n^6 + 4n^5 L + 3n^5 - n^4 L$. Indeed, in Lemma 3.3.5, we show that there is an allocation $\mathbf{t}$ and a stable matching $\hat{M}_\mathbf{t}$ with AvgRank($\hat{M}_\mathbf{t}$) $= n^6 \cdot K_M + n^6 + 4n^5 L + 3n^5 - n^4 L$. In Lemma 3.3.6, we prove that this is a minimum average hospital rank for $\hat{\Gamma}$ since M is an optimal matching. Therefore, $n^6 \cdot K_M + n^6 + 4n^5 L + 3n^5 - n^4 L > n^6 \cdot K + n^6 + 4n^5 L + 3n^5 - n^4 L = \overline{K}$, which means that the answer for the instance of $\textsc{Min-Avg}_{\text{EXP}}$ HR is also NO.

On the other hand, consider a YES instance of $\textsc{Min-w}$ SMT. Then, there is a weakly stable matching M with an average hospital rank of $K_M \leq K$. Therefore, the allocation $\mathbf{t}$ and the stable matching $\hat{M}_\mathbf{t}$ in $\hat{\Gamma}$ constructed in Lemma 3.3.5 have an objective value of $K_M + n^6 + 4n^5 L + 3n^5 - n^4 L \leq n^6 \cdot K + n^6 + 4n^5 L + 3n^5 - n^4 L = \overline{K}$. Hence, the instance of $\textsc{Min-Avg}_{\text{EXP}}$ HR has a YES answer.

Let us prove now that, for any ε, $\textsc{Min-Avg}_{\text{EXP}}$ HR $\textsc{opt}$ is not approximable within a factor of $(\overline{n})^{1/6-\epsilon}$, where $\overline{n}$ is the number of hospitals, unless P=NP. Consider an instance Γ of $\textsc{Min-w}$ SMT with n hospitals and n residents, where $L \geq 1$ of the residents have a tie in their preference list. Let M^{yes} and M^{no} be the stable matchings of minimum average hospital rank for the cases in which the answer of the decision problem $\textsc{Min-w}$ SMT is YES and NO, respectively. Lemma 3.3.2 implies that, for any $\varepsilon > 0$, AvgRank(M^{no}) $\geq n^{1-\epsilon} \cdot$AvgRank($M^{yes}$). Now, consider the reduction presented in Section 3.3.1 from instance Γ to an instance $\hat{\Gamma}$ of $\textsc{Min-Avg}_{\text{EXP}}$ HR. Lemma 3.3.6 implies that there are allocations $\mathbf{t}$ and $\mathbf{t}'$, and matchings $\hat{M}^{yes}_\mathbf{t}$ and $\hat{M}^{no}_{\mathbf{t}'}$ for the respective YES and NO answers of $\textsc{Min-Avg}_{\text{EXP}}$ HR such that

$$\text{AvgRank}(\hat{M}^{yes}_\mathbf{t}) = n^6 \cdot \text{AvgRank}(M^{yes}) + n^6 + 4n^5 L + 3n^5 - n^4 L,$$
$$\text{AvgRank}(\hat{M}^{no}_{\mathbf{t}'}) = n^6 \cdot \text{AvgRank}(M^{no}) + n^6 + 4n^5 L + 3n^5 - n^4 L.$$

Recall that the reduction in Section 3.3.1 constructs $\hat{\Gamma}$ with $\overline{n} := |\hat{\mathcal{C}}| \leq 2n^6$ residents. Then, for any $\varepsilon > 0$, we have

$$\frac{\mathsf{AvgRank}(\hat{M}^{no}_{\mathbf{t}'})}{\mathsf{AvgRank}(\hat{M}^{yes}_{\mathbf{t}})} = \frac{n^6 \cdot \mathsf{AvgRank}(M^{no}) + n^6 + 4n^5L + 3n^5 - n^4L}{n^6 \cdot \mathsf{AvgRank}(M^{yes}) + n^6 + 4n^5L + 3n^5 - n^4L} \geq$$

$$\geq c \cdot \frac{\mathsf{AvgRank}(M^{no})}{\mathsf{AvgRank}(M^{yes})} \geq c \cdot n^{1-\epsilon} \geq c \cdot \left(\frac{\overline{n}}{2}\right)^{\frac{1-\varepsilon}{6}}$$

for some constant $c > 0$. This completes the proof.

$\square$

3.4. The Capacity Reduction Problem

In this section, we focus on Problem 5 that looks for the reduction of capacities such that the residents' allocations are impacted the least. Our main result establishes the computational complexity of this problem. Formally, our result is the following.

Theorem 3.4.1. $\textsc{Min-Avg}_{\textsc{red}}$ HR is NP-complete. Moreover, for any $\varepsilon > 0$, $\textsc{Min-Avg}_{\textsc{red}}$ HR OPT cannot be approximated within a factor of $\bar{n}^{\frac{1}{6}-\varepsilon}$, where $\bar{n}$ is the number of residents, unless *P=NP*.

Proof. First, recall that Problem 5 assumes that reducing the capacities of hospitals does not leave any resident un-assigned.

Clearly $\textsc{Min-Avg}_{\textsc{red}}$ HR is in NP, since for a given vector $\mathbf{t}$ and a matching $M_{\mathbf{t}}$, we can verify in polynomial-time whether $\mathbf{t}$ satisfies the constraint on the number of spots to be removed, whether $M_{\mathbf{t}}$ is stable in $\Gamma_{-\mathbf{t}}$ and if the target value is attained. Now, we focus on showing that the problem is NP-complete.

The rest of the proof follows the same reasoning exposed in the proof of Theorem 3.3.1. We build a reduction from an instance Γ of Min-w SMT into an instance $\hat{\Gamma}$ of $\textsc{Min-Avg}_{\textsc{red}}$ HR. We assume that Γ satisfies $|\mathcal{S}| = |\mathcal{C}| = n$, ties occur only in residents' lists, and each of their preference list has at most one tie of length 2 positioned at the head of it. Recall also that we denoted by $\mathcal{S}'$ the set of residents with a tie in their preference list and by $\mathcal{S}''$ the set of residents with strict preference lists. The corresponding $\hat{\Gamma}$ is defined as in the reduction presented in the proof of Theorem 3.3.1, with the following difference:

- There is no set $\mathcal{X}$.
- For every *village* $\mathcal{B}^k_\ell$ defined for $i_\ell \in \mathcal{S}'$ and $k \in [n^4]$: Each hospital $v^k_{\ell,e}$ for $e \in [n]$ has capacity 2 and each hospital in $\overline{\mathcal{V}}^k_\ell$ has capacity 2. All the remaining preferences and capacities remain as in Section 3.3.1.

Given a weakly stable matching M in the instance Γ with an average hospital rank K_M, we provide a reduction of the capacities $\mathbf{t}$ that respects the budget $B = n^5 \cdot L$ and we build a stable matching $\hat{M}_{\mathbf{t}}$ in $\hat{\Gamma}_{\mathbf{t}}$ with an average hospital rank $\bar{K} = n^6 K_M + n^6 + 4n^5 L + 3n^5 - n^4 L$.

- *Reduction of capacities.* We remove n spots from each village B_ℓ^k in the following way: Assume that in M, we have the pair (i_ℓ, j), where j is such that $r = r_{i_\ell,j}$. Then, we reduce by 1 the capacities of $v_{\ell,r}^k$ and of each hospital in $\overline{\mathcal{V}}_\ell^k \setminus \{\overline{v}_{\ell,r}^k\}$.
- *Matching.* We build the matching $\hat{M}_{\mathbf{t}}$ as follows. For every pair (j,i_ℓ) in M, if $\ell \leq L$, then we match $(w_{\ell,r}^k, j^k)$, $(y_\ell^k, \overline{v}_{\ell,r}^k)$, $\{(w_{\ell,e}, v_{\ell,e}^k)\}_{e \neq r}$ where $r = \text{index}_{i_\ell}(j)$ and $k \in [n^4]$; otherwise, $\ell > L$, and we match (i_ℓ^k, j^k), where $k \in [n^4]$. The remaining pairs are the same as in the proof of Lemma 3.3.5.

The rest of the proof is analogous to the proofs of Lemma 3.3.5, Lemma 3.3.6, and Theorem 3.3.1.

□

3.5. Extensions

In this section, we investigate the variants of Problems 4 and 5 where the decision-maker has budgets for different subsets of hospitals. In the remainder of this section, we say that $\mathcal{P} = \{\mathcal{C}_1, \ldots \mathcal{C}_q\}$ is a partition of the set of hospitals $\mathcal{C}$ if $\cup_{k \in [q]} \mathcal{C}_k = \mathcal{C}$ and $\mathcal{C}_k \cap \mathcal{C}_{k'} = \emptyset$ for all $k,k' \in [q]$ with $k \neq k'$.

3.5.1. Allocating Extra Spots to a Partition of Hospitals

We generalize Problem 4 to the setting where the set of hospitals is partitioned and we seek to find an allocation of extra spots such that each part has a specific budget. Formally, we study the following problem.

Problem 7 (MIN-AVG$^{\text{SUB}}_{\text{EXP}}$ HR).

INSTANCE: A *HR* instance $\Gamma = \langle \mathcal{S},\mathcal{C}, \succ, \mathbf{c}\rangle$, a partition $\mathcal{P} = \{\mathcal{C}_1, \ldots \mathcal{C}_q\}$ of $\mathcal{C}$, a budget for each part $\{B_k \in \mathbb{Z}_+ \colon k \in [q]\}$, and a non-negative integer target value $K \in \mathbb{Z}_+$.

QUESTION: Is there a non-negative vector $\mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}}$ and a matching $M_{\mathbf{t}}$ such that

$$\textit{AvgRank}(M_{\mathbf{t}}) \leq K,$$

where $\mathbf{t}$ is such that $\sum_{j \in \mathcal{C}_k} t_j \leq B_k$ for each $k \in [q]$ and $M_{\mathbf{t}}$ is a stable matching in instance $\Gamma_{\mathbf{t}}$?

The next result can be directly obtained by considering a single set of hospitals in the partition, *i.e.*, $q = 1$ and $\mathcal{P} = \mathcal{C}$, and by using Theorem 3.3.1.

Corollary 3.5.1. $\textsc{Min-Avg}_{\text{EXP}}^{\text{SUB}}$ HR is NP-complete.

Denote by $\textsc{Min-Avg}_{\text{EXP}}^{\text{SUB}}$ HR $\textsc{opt}$ the optimization version of $\textsc{Min-Avg}_{\text{EXP}}^{\text{SUB}}$ HR, *i.e.*, the problem of finding an allocation of extra capacities and a stable matching in the expanded instance of minimum average hospital rank. In the following result, we show the approximation complexity of $\textsc{Min-Avg}_{\text{EXP}}^{\text{SUB}}$ HR $\textsc{opt}$.

Theorem 3.5.2. For any $\varepsilon > 0$, $\textsc{Min-Avg}_{\text{EXP}}^{\text{SUB}}$ HR $\textsc{opt}$ is not approximable within a factor of $n^{1-\varepsilon}$, unless P=NP, where n is the number of residents. This result holds even if the partition $\mathcal{P} = \{\mathcal{C}_1, \ldots, \mathcal{C}_q\}$ is such that each $\mathcal{C}_k$ contains at most two hospitals and $B_k \in \{0,1\}$ for every $k \in [q]$.

Before proving Theorem 3.5.2, we need to introduce a variant of Problem 7, where the goal is to find a stable matching whose size is at least a certain target. The problem of finding the maximum cardinality stable matching is one of the main focus of the literature [**107**]. We investigate it in relation with capacity expansion when there are incomplete preference lists. Recall that a HR instance with incomplete preference lists means that there is at least one resident or one hospital that does not rank completely the opposite side. Formally, we consider the following problem.

Problem 8 ($\textsc{Max-Card}_{\text{EXP}}^{\text{SUB}}$ HRI)**.**

$\textsc{instance}$: A *HR* instance $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{c} \rangle$ with incomplete preference lists, a partition $\mathcal{P} = \{\mathcal{C}_1, \ldots, \mathcal{C}_q\}$ of $\mathcal{C}$, a budget for each part $\{B_k \in \mathbb{Z}_+ : k \in [q]\}$ and a non-negative integer target value $K \in \mathbb{Z}_+$.
$\textsc{question}$: Is there a non-negative vector $\mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}}$ and a matching $M_{\mathbf{t}}$ such that

$$|M_{\mathbf{t}}| \geq K,$$

where $\mathbf{t}$ is such that $\sum_{j \in \mathcal{C}_k} t_j \leq B_k$ for each $k \in [q]$ and $M_{\mathbf{t}}$ is a stable matching in instance $\Gamma_{\mathbf{t}}$?

Recall that if we consider complete preference lists, the problem above becomes trivial since all stable matchings have the same size. We prove the following result.

Theorem 3.5.3. $\textsc{Max-Card}_{\text{EXP}}^{\text{SUB}}$ HRI is NP-complete, even if the partition $\mathcal{P} = \{\mathcal{C}_1, \ldots, \mathcal{C}_q\}$ is such that each $\mathcal{C}_k$ is of size at most two and $B_k \in \{0,1\}$ for every $k \in [q]$.

The proof of this result can be found in the Appendix. Let us now focus on the proof of Theorem 3.5.2.

$\textsc{Proof of Theorem}$ 3.5.2. Let $\varepsilon > 0$ and define $a = \lceil (3/\varepsilon) \rceil$. We consider an instance Γ of $\textsc{Max-Card}_{\text{EXP}}^{\text{SUB}}$ HRI in which the set of hospitals is $\mathcal{C}$, the set of residents is $\mathcal{S}$ (w.l.o.g., we assume $|\mathcal{C}| = |\mathcal{S}| = n$), every $\mathcal{C}_k$ is of size at most two and $B_k \in \{0,1\}$ for every $k \in [q]$. We denote by O_j (resp. O_i) the preference list of hospital j (resp. resident i). We assume that the target value K is equal to n.

We now build an instance $\hat{\Gamma}$ of $\textsc{Min-Avg}^{\text{SUB}}_{\text{EXP}}$ HR $\textsc{opt}$. Let us define $A = n^{a-1}$. In this instance, the set of hospitals is $\left(\bigcup_{h=1}^{A} \mathcal{C}^h\right) \cup \mathcal{C}^0$ where $\mathcal{C}^h = \{j_1^h, \ldots, j_n^h\}$ is a copy of $\mathcal{C}$ for $h \in [A]$, and $\mathcal{C}^0 = \{j_1^0, \ldots, j_{n^a}^0\}$. The set of residents is $\left(\bigcup_{h=1}^{A} \mathcal{S}^h\right) \cup \mathcal{S}^0$ where $\mathcal{S}^h = \{i_1^h, \ldots, i_n^h\}$ is a copy of $\mathcal{S}$ for $h \in [A]$, and $\mathcal{S}^0 = \{i_1^0, \ldots, i_{n^a}^0\}$. Now that the hospitals are introduced, we need to establish how their set is partitioned; for every pair $(\mathcal{C}_k, B_k)$ in Γ, we establish the pair $(\mathcal{C}_k^h, B_k^h)$ in $\mathcal{C}^h$ for $k \in [q]$ and $h \in [A]$. The hospitals in $\mathcal{C}^0$ have all capacity 1; concerning the capacities of the other hospitals, those that are in a pair have each capacity one with an extra budget of one, and those hospitals that are in a singleton, have capacity one and no extra budget. For $j \in \mathcal{C}$ and $h \in [A]$, we denote by O_j^h the preference list obtained by substituting in the preference list O_j the residents in $\mathcal{S}$ with the residents in $\mathcal{S}^h$. We define similarly O_i^h for every resident i in $\mathcal{S}^h$ where $h \in [A]$. The preference lists of the hospitals and residents in $\hat{\Gamma}$ are as follows:

$$
\begin{array}{ll}
j_h^0 : i_h^0, \ldots & h \in [n^a] \\
j_s^h : O_{j_s}^h, \mathcal{S}^0, \ldots & s \in [n],\, h \in [A] \\
i_h^0 : j_h^0, \ldots & h \in [n^a] \\
i_s^h : O_{i_s}^h, \mathcal{C}^0, \ldots & s \in [n],\, h \in [A],
\end{array}
$$

where the dots "…" in the preference lists mean that the remaining agents on the other side of the bipartition are ranked strictly and arbitrarily.

Our $\textsc{Min-Avg}^{\text{SUB}}_{\text{EXP}}$ SM instance comprises $2n^a$ residents, so that $\bar{n} := 2n^a$; the target value is $K' = \lfloor n^{a+2}/2 \rfloor$. The remainder of the proof follows the same reasoning as the proof of Lemma 3.3.2, which can be found in the Appendix. □

3.5.2. Removing Spots from a Partition of Hospitals

Similar to the problems presented in the previous section, we now study the generalization of Problem 5 where the set of hospitals is partitioned in q parts and each part has a budget for the removal of spots. Specifically, we consider the following problem.

Problem 9 ($\text{MIN-AVG}^{\text{SUB}}_{\text{RED}}$ HR).

INSTANCE: A *HR* instance $\Gamma = \langle \mathcal{S},\mathcal{C}, \succ, \mathbf{c}\rangle$, a partition $\mathcal{P} = \{\mathcal{C}_1, \dots \mathcal{C}_q\}$ of $\mathcal{C}$, a budget for each part $\{B_k \in \mathbb{Z}_+ \colon k \in [q]\}$, and a non-negative integer target value $K \in \mathbb{Z}_+$. QUESTION: Is there a non-negative vector $\mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}}$ and a matching $M_{\mathbf{t}}$ such that $$\textit{AvgRank}(M_{\mathbf{t}}) \leq K,$$ where $\mathbf{t}$ is such that $\sum_{j \in \mathcal{C}_k} t_j \geq B_k$ and $c_j - t_j \geq 0$ for $k \in [q]$, and $M_{\mathbf{t}}$ is a stable matching in instance $\Gamma_{\mathbf{t}}$?

For Problem 9, we prove the following inapproximability result.

Theorem 3.5.4. For any $\varepsilon > 0$, $\text{MIN-AVG}^{\text{SUB}}_{\text{RED}}$ HR OPT is not approximable within a factor of $n^{1-\varepsilon}$, unless P=NP, where n is the number of residents. This result holds even with a partition in which each part $\mathcal{C}_k$ contains at most two hospitals and $B_k \in \{0,1\}$ for every $k \in [q]$.

To prove Theorem 3.5.4, we need to study the analogous version of Problem 8 for the capacity reduction setting. Formally, we define the following problem.

Problem 10 ($\text{MAX-CARD}^{\text{SUB}}_{\text{RED}}$ HRI).

INSTANCE: A *HR* instance $\Gamma = \langle \mathcal{S},\mathcal{C}, \succ, \mathbf{c}\rangle$ with incomplete preference lists, a partition $\mathcal{P} = \{\mathcal{C}_1, \dots \mathcal{C}_q\}$ of $\mathcal{C}$, a budget for each part $\{B_k \in \mathbb{Z}_+ \colon k \in [q]\}$, and a non-negative integer target value $K \in \mathbb{Z}_+$. QUESTION: Is there a non-negative vector $\mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}}$ and a matching $M_{\mathbf{t}}$ such that $$\lvert M_{\mathbf{t}}\rvert \geq K,$$ where $\mathbf{t}$ is such that $\sum_{j \in \mathcal{C}_k} t_j \geq B_k$ and $c_j - t_j \geq 0$ for $k \in [q]$, and $M_{\mathbf{t}}$ is a stable matching in instance $\Gamma_{\mathbf{t}}$?

In particular, we show the following result.

Theorem 3.5.5. $\text{MAX-CARD}^{\text{SUB}}_{\text{RED}}$ HRI is NP-complete. This result holds even with a partition in which each part $\mathcal{C}_k$ is of size at most two and $B_k \in \{0,1\}$ for every $k \in [q]$.

PROOF. The proof is analogous to the proof of Theorem 3.5.3 with the difference that every hospital in each part $\mathcal{C}_k$ has capacity 1. $\square$

PROOF OF THEOREM 3.5.4. The proof follows a similar reasoning as the proof of Theorem 3.5.2 with the difference that every hospital in each part $\mathcal{C}_k$ has capacity 1. $\square$

3.6. Conclusions

How should a centralized institution optimally manage a variation in the capacities of the hospitals? The case in which capacities are increased while staying within a budget has gained recent interest [**38, 58, 101**]. However, the computational complexity of solving its optimization variant was an open question. Indeed, also the problem of reducing capacities optimally may have a strong impact in the real-world [**65**]. Finally, in recent years, researchers attempted to address the concentration of residents' applications in urban areas by establishing quotas or by resource redistribution on a regional level [**83**], for which there was not yet a clear understanding on the computational limitations posed by this problem.

The novelty of our work is that we proceed in our investigation from two points of view: Capacity expansion and capacity reduction. To the best of our knowledge, we are the first to propose the problem of reducing capacities in the framework of matching with stability. Remarkably, we outline that the two problems of expanding and reducing capacities are deeply interconnected. Moreover, we also investigate the case in which the variation of capacities may happen only on a local level.

Our first result, establishes the approximation hardness of the problem of finding the resident-optimal stable matching in the presence of ties. Our theorem defines a boundary on the complexity of the resident-optimal stable matching, which is well known to be polynomial-time solvable when there are no ties. We use this result as the first building block in the construction of the main proof of the paper: The approximation hardness of the problem of allocating optimally extra capacities to the hospitals to reduce the average hospital rank. Our proof introduces a crucial structure, the *village*, that enables us to manage the subtleties of the allocation of extra capacities. The problem of allocating extra resources is not easier when we restrict the distribution of capacities to a partition of the hospitals. If the objective of the problem is the cardinality of the stable matching, we prove that it is NP-complete when the problem has incomplete lists. If the objective is the average hospital rank, the corresponding optimization problem cannot be approximated within a meaningful factor.

The problem of reducing the capacities is equally interesting from both a practical and theoretical perspective. We show that the capacity reduction problem is NP-complete when the goal is finding the maximum cardinality stable matching. Then, we also prove that as we partition the set of hospitals from which we should reduce the capacities, and to each set we allocate a number of seats to be removed, the problem remains hard to solve if we still want to obtain the stable matching with the minimum cost for the students. For this latter problem, we prove that its optimization version is also inapproximable.

We believe these results are significant because they emphasize the existence of an underlying structure in the stable matching problem which governs both the capacity expansion and reduction. Unveiling the properties of this structure is certainly an open question worth being explored. We also believe that another interesting future direction of research is understanding the role of meta-rotations in the capacity variation problem, see e.g., [**71, 32, 49**].

Appendix

3.7. Missing proofs

The following problem is useful for the proofs that we provide in this Appendix.

Problem 11 (MAX-CARD HRTI).

INSTANCE: An *HRTI* instance $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{c} \rangle$ with $c_j = 1$ for all $j \in \mathcal{C}$, $\lvert\mathcal{C}\rvert = \lvert\mathcal{S}\rvert$ and a non-negative integer target value $K \in \mathbb{Z}_+$. QUESTION: Is there a weakly stable matching M such that $\lvert M\rvert \geq K$?

Recall that HRTI corresponds to the problem with ties and incomplete preference lists. Manlove et al. [**108**] proved that MAX-CARD HRTI is NP-complete. As the next remark states, this result holds even if ties are at the head of the preference list, only on one side of it, at most one tie per list, and each tie is of length 2.

Remark 3.7.1. After the proof of Lemma 1 in [**108**], the authors showed that the problem MAX-CARD HRTI can be simplified to the case in which ties are only on one side of the bipartition and are at the end of the preference list. Since the ties of the new instance created in Lemma 1 from [**108**] are of length at most two, we can use the same reasoning to assume instead, without loss of generality, that in an instance of MAX-CARD HRTI and the corresponding MIN-W SMT instance of Lemma 3.3.2 ties occur only at the *head* of a preference list.

3.7.1. Proof of Lemma 3.3.2

In this section, we prove that MIN-W SMT is NP-complete and its optimization version cannot be approximated within a constant factor. The proof is inspired by the proof of Theorem 7 in [**108**]. The result in [**108**] is stated in the traditional notation of the stable marriage problem where both sides are defined as women and men, instead of residents and hospitals, respectively. To keep coherence with the previous work, for this proof we also denote both sides as women and men.

PROOF OF LEMMA 3.3.2. Clearly, MIN-W SMT is in NP. Given $\varepsilon > 0$, let $a = \lceil(3/\varepsilon)\rceil$. From Theorem 2 in [**108**], we know that, when ties occur on the women's side only, and each tie has length two, MAX-CARD HRTI is NP-complete. Consider an instance of Problem 11 with $\mathcal{C} = \{m_1, m_2, \ldots, m_n\}$ and $\mathcal{S} = \{w_1, w_2, \ldots, w_n\}$. We assume that the target value K is equal to n, since it was shown that even for this target value the problem is NP-complete. Let O_h

(resp. R_h) denote the preference list of man m_h (resp. woman w_h) for $h \in [n]$. Next, we build an instance of $\textsc{Min-w SMT}$. Let $C := n^{a-1}$, then

- the set of men is $\mathcal{C}' = \mathcal{C}^0 \cup \left(\bigcup_{h=1}^{C} \mathcal{C}^h\right)$ with $\mathcal{C}^0 = \{m_1^0, m_2^0, \ldots, m_{n^a}^0\}$ and $\mathcal{C}^h = \left\{m_1^h, m_2^h, \ldots, m_n^h\right\}$ for $h \in [C]$;
- the set of women is $\mathcal{S}' = \mathcal{S}^0 \cup \left(\bigcup_{h\in[C]} \mathcal{S}^h\right)$ with $\mathcal{S}^0 = \{w_1^0, w_2^0, \ldots, w_{n^a}^0\}$ and $\mathcal{S}^h = \left\{w_1^h, w_2^h, \ldots, w_n^h\right\}$ for $h \in [C]$;
- for each $h \in [n]$ and $s \in [C]$, let O_h^s be the preference list obtained from O_h by replacing woman w_k in O_h by the corresponding woman w_k^s, for every $k \in [n]$. We refer to the women in O_h^s as the proper women for m_h^s. Similarly, we define R_h^s and the proper men for w_h^s. The preference lists for $\mathcal{C}'$ and $\mathcal{S}'$ are

$$\begin{aligned}
m_h^0 &: w_h^0 \ldots & & h \in [n^a] \\
m_h^s &: O_h^s, \mathcal{S}^0 \ldots & & h \in [n],\, s \in [C] \\
w_h^0 &: m_h^0 \ldots & & h \in [n^a] \\
w_h^s &: R_h^s, \mathcal{C}^0 \ldots & & h \in [n],\ s \in [C]
\end{aligned}$$

  where the dots "…" in the preference lists mean that the remaining agents on the other side of the bipartition are ranked strictly and arbitrarily, and the sets mean that the agents within are ranked according to their indices;
- the target value is $K' = \lfloor (n^{a+2})/2 \rfloor$.

Our $\textsc{Min-w SMT}$ instance comprises $2n^a$ men and $2n^a$ women, so that $\bar{n} := 2n^a$. Note also that the only ties in $\textsc{Min-w SMT}$ occur in the preference lists of women w_h^s for $h \in [n],\ s \in [C]$. Moreover, there is at most one tie per list, and each tie has length 2.

Suppose that we have a YES instance for $\textsc{Max-Card HRTI}$, *i.e.*, there is a stable matching M with $|M| = n$. We create a matching M' in $\textsc{Min-w SMT}$ as follows: For every $h \in [n^a]$, we add the pair (m_h^0, w_h^0) to M', and for each $s \in [n]$, we add the pair $\left(m_s^\ell, w_k^\ell\right)$ to M' for all $\ell \in [C]$, where $(m_s, w_k) \in M$. Note that M' is stable for our $\textsc{Min-w SMT}$ instance. We also have that

$$\mathsf{AvgRank}\,(M') \le n^a + n^{a-1} n^2 \le \left\lfloor \frac{n^{a+2}}{2} \right\rfloor = K',$$

since, without loss of generality, we may choose $n \ge 3$. Therefore, the objective value in $\textsc{Min-w SMT}$ satisfies the target of K'.

On the other side, let us suppose that we have a NO instance for $\textsc{Max-Card HRTI}$, *i.e.*, it does not have a stable matching of cardinality n. Then, in any stable matching M' of $\textsc{Min-w SMT}$, it holds that, for every $s \in [C]$, there is some $h \in [n]$ for which w_h^s is not matched to one of her proper men. Nonetheless, in M', m_h^0 and w_h^0 must be partners, for every $h \in [n^a]$. Therefore, there is some $h \in [n]$ such that $r_{w_h^s, M'(w_h^s} > n^a$. Hence, $\mathsf{AvgRank}\,(M') > n^{2a-1} > K'$ for any stable matching of our $\textsc{Min-w SMT}$ instance.

Therefore, the existence of a polynomial-time approximation algorithm for $\textsc{Min-w SMT opt}$ whose approximation ratio is as good as $\left(2n^{2a-1}\right)/n^{a+2} = 2n^{a-3}$ would give a polynomial-time algorithm for determining whether $\textsc{Max-Card HRTI}$ has a stable matching in which everybody is matched (*i.e.*, $K = n$). To conclude, we note that $2n^{a-3} = \left(2/2^{1-3/a}\right)\bar{n}^{1-3/a} > \bar{n}^{1-3/a} > \bar{n}^{1-\varepsilon}$, which ends the proof. □

3.7.2. Proof of Lemma 3.3.5

Proof. Let M be a (complete) weakly stable matching in Γ. Recall that $\mathcal{S}' = \{i_1,\ldots,i_L\}$ is the set of residents in Γ with a single tie at the head of the list. Let $\text{index}_{i_\ell}(j)$ be the index of hospital j in resident i_ℓ preference list, *i.e.*, $\text{index}_{i_\ell}(j)$ is equal to 1 or 2 if j is ranked first, otherwise is equal to the rank of j. Define the following set of indices in M:

$$\mathtt{Idx}(M) = \{(\ell,r) :\ r = \text{index}_{i_\ell}(j),\ (i_\ell,j) \in M \cap (\mathcal{S}' \times \mathcal{C})\}.$$

The set $\mathtt{Idx}(M)$ contains the information of the pairs $\mathcal{S}' \times \mathcal{C}$ that are matched in M. Given M and $\mathtt{Idx}$, we now define the following sets that will be helpful in this proof. For every $k \in [n^4]$, we define the set of residents

$$\mathcal{W}_M^k = \{w_{\ell,r}^k \in \mathcal{W} :\ (\ell,r) \in \mathtt{Idx}(M)\},$$

and define the sets of hospitals

$$\begin{aligned}\overline{\mathcal{V}}_M^k &= \{\overline{v}_{\ell,r}^k \in \overline{\mathcal{V}} :\ (\ell,r) \in \mathtt{Idx}(M)\},\\ \mathcal{V}_M^k &= \{v_{\ell,e}^k \in \mathcal{V} :\ (\ell,r) \in \mathtt{Idx}(M),\ e \in [n] \setminus \{r\}\}.\end{aligned}$$

We now provide an allocation of extra spots $\mathbf{t}$ with a total budget $B = L \cdot n^5$ and a stable matching $\hat{M}_{\mathbf{t}}$ in $\hat{\Gamma}_{\mathbf{t}}$.

- *Allocation of extra spots.* For every $k \in [n^4]$, we assign n extra positions to each hospital in $\overline{\mathcal{V}}_M^k \cup \mathcal{V}_M^k$. For the rest of the hospitals, we assign no extra capacities. Formally, for $k \in [n^4]$, we have
$$t_v = \begin{cases} 1 & v \in \overline{\mathcal{V}}_M^k \cup \mathcal{V}_M^k \\ 0 & \text{otherwise}\end{cases}$$
Since $L = |\mathcal{S}'|$, all of the extra positions $B = L \cdot n^5$ are used.
- *Matching.* For each $(\ell,r) \in \mathtt{Idx}(M)$ with j such that $r = \text{index}_{i_\ell}(j)$ in Γ, we match the following pairs in $\hat{M}_{\mathbf{t}}$: $(w_{\ell,r}^k,j^k)$, $(y_\ell^k, \overline{v}_{\ell,r}^k)$, and $(w_{\ell,e}^k, v_{\ell,e}^k)$ for $e \in [n] \setminus \{r\}$ and $k \in [n^4]$. Note that if j is ranked first by i_ℓ, the hospital is listed first or second in the tie. If j is listed first, then $r = 1$ and we match the pairs $\{(w_{\ell,1}^k,j^k)\}_{k\in[n^4]}$, otherwise, $r = 2$ and we match the pairs $\{(w_{\ell,2}^k,j^k)\}_{k\in[n^4]}$. For each $(i,j) \in M$ with $i \in \mathcal{S}''$, we match the pairs $\{(i^k,j^k)\}_{k\in[n^4]}$ in $\hat{M}_{\mathbf{t}}$, where j^k is a copy of j in $\overline{\mathcal{C}}$; recall that $\mathcal{S}''$ is the set of residents

with a strict preference list. Formally, matching $\hat{M}_{\mathbf{t}}$ is as follows:

$$\begin{aligned}\hat{M}_{\mathbf{t}} =&\{(i^k,j^k) :\ (i,j) \in M \cap (\mathcal{S}'' \times \mathcal{C}) \text{ and } k \in [n^4]\}\\ &\cup \{(w^k_{\ell,r},j^k) :\ r = \text{index}_{i_\ell}(j),\ (i_\ell,j) \in M \cap (\mathcal{S}' \times \mathcal{C}) \text{ and } k \in [n^4]\}\\ &\cup \{(y^k_\ell, \overline{v}^k_{\ell,r}) :\ (\ell,r) \in \mathtt{Idx}(M) \text{ and } k \in [n^4]\}\\ &\cup \{(w^k_{\ell,e},v^k_{\ell,e}) :\ (\ell,r) \in \mathtt{Idx}(M),\ e \in [n] \setminus \{r\} \text{ and } k \in [n^4]\}\\ &\cup \{(u^k_{\ell,e},v^k_{\ell,e}), (\overline{u}^k_{\ell,e},\overline{v}^k_{\ell,e})\}_{\ell\in[L],e\in[n],k\in[n^4]}\\ &\cup \{(a_p,z_p)\}_{p\in[n^6]}.\end{aligned}$$

The only hospitals that are matched to two residents are those that receive one extra capacity according to vector $\mathbf{t}$. All the other hospitals are matched at most to one resident. Therefore, $\hat{M}_{\mathbf{t}}$ is a matching. Let us verify that $\hat{M}_{\mathbf{t}}$ is a stable matching in $\hat{\Gamma}_{\mathbf{t}}$. First, note that residents $i^k \in \overline{\mathcal{S}}''$ and hospitals $j^k \in \overline{\mathcal{C}}$ cannot create blocking pairs because of their stability in M. Now, let us check the stability of the pairs in each village $\mathcal{B}^k_\ell$, where $i_\ell \in \mathcal{S}'$ with $\ell \in [L]$, and $k \in [n^4]$. The pairs matched in village $\mathcal{B}^k_\ell$ are $(w^k_{\ell,r},j^k)$, $(w^k_{\ell,e}, v^k_{\ell,e})$, $(y^k_\ell, \overline{v}^k_{\ell,r})$ and $\{(u^k_{\ell,p}, v^k_{\ell,p}), (\overline{u}^k_{\ell,p}, \overline{v}^k_{\ell,p})\}_{p\in[n]}$ where $e \in [n] \setminus \{r\}$.

- The agents in the pair $(w^k_{\ell,r},j^k)$ are not part of any blocking pair; in fact, $w^k_{\ell,r}$ cannot be matched to any of the hospitals in $\{v^k_{\ell,r}\} \cup \overline{\mathcal{V}}^k_\ell \setminus \{\overline{v}^k_{\ell,r}\}$ because they have capacity one and they are all matched to their most favorite resident in $\mathcal{U} \cup \overline{\mathcal{U}}$. Also j^k cannot be part of a blocking pair. Indeed, all the residents $w^k_{\ell',r'}$ ranked in its preference list before $w^k_{\ell,r}$ are matched to hospitals of the form $v^k_{\ell',r'}$ that they rank first or to another hospital j'^k they prefer over j^k (due to the stability of M in Γ).
- For $e \in [n] \setminus \{r\}$, $w^k_{\ell,e}$ ranks $v^k_{\ell,e}$ first, and $v^k_{\ell,e}$ has capacity two and ranks $w^k_{\ell,e}$ second. Hence the $n-1$ pairs $(w_{\ell,e}, v^k_{\ell,e})$ do not include any agent who may be part of a blocking pair.
- If $r = 2$, then y^k_ℓ ranks $\overline{v}^k_{\ell,r}$ first, and $\overline{v}^k_{\ell,r}$ cannot be matched to any of the residents in $\mathcal{W}^k_\ell \setminus \{w^k_{\ell,r}\}$ because of the previous point. Moreover, $\overline{v}^k_{\ell,r}$ has capacity two and is also matched to $\overline{u}^k_{\ell,r}$. Therefore, the agents in the pair $(y^k_\ell, \overline{v}^k_{\ell,r})$ do not take part in a blocking pair when $r = 2$. If $r \neq 2$, then y^k_ℓ ranks $\overline{v}^k_{\ell,r}$ second or more, and y^k_ℓ cannot be matched to $\overline{v}^k_{\ell,q}$ for $q \in \{2,1,3,\dots,r-1\}$: Each $\overline{v}^k_{\ell,q}$ has capacity one and it is already matched to $\overline{u}^k_{\ell,q}$. On the other hand, as we mentioned before, $\overline{v}^k_{\ell,r}$ cannot create a blocking pair with any of the residents in $\mathcal{W}^k_\ell \setminus \{w^k_{\ell,r}\}$ because these residents are matched to their most preferred hospitals. Therefore, the agents in the pair $(y^k_\ell, \overline{v}^k_{\ell,r})$ do not create blocking pairs when $r \neq 2$.
- The pairs $\{(u^k_{\ell,p}, v^k_{\ell,p}), (\overline{u}^k_{\ell,p}, \overline{v}^k_{\ell,p})\}_{p\in[n]}$ do not involve any agent who could be part of a blocking pair. First, each of these listed pairs matches two agents that rank each other

first. Second, for those hospitals that have capacity two, from the previous points, we conclude that there are no residents with which they could create a blocking pair.

Therefore, $\hat{M}_{\mathbf{t}}$ is a stable matching in $\hat{\Gamma}_{\mathbf{t}}$.

Next, we compute the average hospital rank in M and $\hat{M}_{\mathbf{t}}$. In M, we can distinguish whether a resident is in $\mathcal{S}''$ or $\mathcal{S}'$ and we can distinguish if a resident is matched to a hospital ranked first or not. Let G'' be the average hospital rank of residents in $\mathcal{S}''$, K'_t be the average hospital rank of the residents in $\mathcal{S}'$ that are matched to a hospital in their ties, and K'_s be the average hospital rank of the residents in $\mathcal{S}'$ that are matched to a hospital they rank third or more. Note that K'_t is also the number of residents from $\mathcal{S}'$ matched to a hospital they rank first. The average hospital rank of M is $K_M = G'' + K'_t + K'_s$.

We now prove that, given $k \in [n^4]$ and $(\ell,r) \in \mathtt{Idx}(M)$, the average rank of the residents in village $\mathcal{B}^k_\ell$ is: $(n + n^2 \cdot r + 3) + (n-1) + (2n)$. The first term comes from the residents matched in the two pairs $(w^k_{\ell,r}, j^k)$ and $(y^k_\ell, \overline{v}^k_{\ell,r})$: if $r = 1$ then $w^k_{\ell,1}$ ranks j^k at position $n + n^2 + 1$ and y^k_ℓ ranks $\overline{v}^k_{\ell,1}$ at position 2, thus totaling $n + n^2 + 3$; if $r = 2$ then $w^k_{\ell,2}$ ranks j^k at position $n + n^2 + 2$ and y^k_ℓ ranks $\overline{v}^k_{\ell,2}$ at position 1, thus totaling $n + n^2 + 3$; if $r > 2$ then $w^k_{\ell,r}$ ranks j^k at position $n + (n^2-1)r + 2 + 1$ and y^k_ℓ ranks $\overline{v}^k_{\ell,2}$ at position r, thus totaling $n + n^2 + 3$. The second term comes from the $(n-1)$ residents matched in the pairs $\{(w^k_{\ell,e}, v^k_{\ell,e})\}_{e \in [n] \setminus \{r\}}$. The third term comes from the $2n$ residents matched in the pairs $\{(u^k_{\ell,p}, v^k_{\ell,p}), (\overline{u}^k_{\ell,p}, \overline{v}^k_{\ell,p})\}$. Note that $(n + n^2 \cdot r + 3) + (n-1) + (2n) = n^2 \cdot r + 3 + (4n-1)$. On the other hand, each resident i^k_ℓ for $\ell \in \{L+1, \ldots, n\}$ is matched to a hospital j^k, and w.l.o.g. j is ranked r by i_ℓ in Γ; therefore, i^k_ℓ ranks j^k at position $(n^2+2) + 1 + (r-1) \cdot (n^2-1) + (r-1) = n^2 \cdot r + 3$, where the first term comes from the cardinality of the first set ranked by i^k_ℓ, the second term comes from the first j ranked by i_ℓ, the third term comes from the number of sets of the form $\mathcal{V}^{k+1}_+[n^2+2,e]$ (each of cardinality n^2-1) positioned before j^k, and the last term is the number of hospitals j'^k ranked before j^k.

Therefore, the residents in copy k have an average rank of $[n^2 \cdot G'' + 3(n-L)] + n^2 \cdot (K'_t + K'_s) + [3 + (4n-1)] \cdot (L) = n^2 \cdot K_M + 3(n - L + L) + (4n-1)L = n^2 \cdot K_M + 4nL + 3n - L$, where the first term comes from the residents i^k_ℓ for $\ell \in \{L+1, \ldots, n\}$ (note that when we sum the contribution of each such resident, we have the sum $\sum_{\ell \in \{L+1,\ldots,n\}} n^2 \cdot r_\ell + 3$, where r_ℓ is $\text{index}_{i_\ell}(M(i_\ell))$), the second term comes from the residents $\{w^k_{\ell,r}, y^k_\ell\}$ for $(\ell,r) \in \mathtt{Idx}(M)$,[11] and the third term comes from the remaining residents in the villages indexed with k and the constant 3 multiplied by the number of students in villages matched to a hospital j^k. Since $k \in [n^4]$, then $\text{AvgRank}(\hat{M}_{\mathbf{t}})$ is $n^4 \cdot (n^2 \cdot K_M + 4nL + 3n - L) + n^6$, where the last term comes from the residents in $\mathcal{A}$. Hence $\text{AvgRank}(\hat{M}_{\mathbf{t}}) = n^6 \cdot K_M + n^6 + 4n^5 L + 3n^5 - n^4 L$.

□

[11] Note that when we sum the contribution of each resident in every village that is not matched to a hospital $v^k_{\ell,e}$, we have the sum $\sum_{\ell \in \{L+1,\ldots,n\}} n^2 \cdot r_\ell + 3$, where r_ℓ is $\text{index}_{i_\ell}(M(i_\ell))$).

3.7.3. Proof of Lemma 3.3.6

PROOF. Let M be a stable matching in Γ of minimum average hospital rank. Recall the instance $\hat{\Gamma}$ constructed in Section 3.3.1, the allocation

$$t_v = \begin{cases} 1 & v \in \overline{\mathcal{V}}_M^k \cup \mathcal{V}_M^k \\ 0 & \text{otherwise,} \end{cases}$$

and the matching

$$\begin{aligned} \hat{M}_{\mathbf{t}} = &\{(i^k,j^k) \colon\ k \in [n^4],\ (i,j) \in M \cap (\mathcal{S}'' \times \mathcal{C})\} \\ &\cup \{(w^k_{\ell,r},j^k) \colon\ k \in [n^4],\ r = r_{i_\ell,j},\ (i_\ell,j) \in M \cap (\mathcal{S}' \times \mathcal{C})\} \\ &\cup \{(y^k_\ell,\overline{v}^k_{\ell,r}) \colon\ k \in [n^4],\ (\ell,r) \in \texttt{Idx}(M)\} \\ &\cup \{(w^k_{\ell,e},v^k_{\ell,e}) \colon\ k \in [n^4],\ (\ell,r) \in \texttt{Idx}(M),\ e \in [n] \setminus \{r\}\} \\ &\cup \{(u^k_{\ell,e},v^k_{\ell,e}),(\overline{u}^k_{\ell,e},\overline{v}^k_{\ell,e}) \colon\ k \in [n^4],\ e \in [n]\} \\ &\cup \{(a_p,z_p) \colon\ p \in [n^6]\}, \end{aligned}$$

constructed in Lemma 3.3.5. Denote by $\overline{K} = n^6 \cdot K_M + n^6 + 4n^5L + 3n^5 - n^4L$, which is the average rank of $\hat{M}_{\mathbf{t}}$ in $\hat{\Gamma}$. Now, we will prove that any other feasible allocation $\tilde{\mathbf{t}}$ with total budget $B = L \cdot n^5$ and any stable matching $\hat{M}_{\tilde{\mathbf{t}}}$ in the expanded instance $\hat{\Gamma}_{\tilde{\mathbf{t}}}$ have AvgRank$(\hat{M}_{\tilde{\mathbf{t}}}) \geq \overline{K}$.

Given allocation $\mathbf{t}$, we start by observing that it is not optimal to move one extra capacity from a hospital $v^k_{\ell,e}$ or $\overline{v}^k_{\ell,e}$ to a hospital in $\mathcal{X} \cup \mathcal{Z}$. Indeed, $\mathcal{X}$ already has n^7 positions available, but since it is at the end of the preference list of every resident, it would be sub-optimal to match a resident to a hospital in it. Similarly, it would be sub-optimal to allocate an extra-capacity to $\mathcal{Z}$, since all the residents are already matched to a hospital they prefer to any hospital in $\mathcal{Z}$.

Regarding the hospitals in $\overline{\mathcal{C}}$, let us assume we move a capacity from $\overline{v}^k_{\ell,e}$ to a hospital $j^{k'} \in \overline{\mathcal{C}}$ with $l \in [L]$, $e \in [n]$, $k, k' \in [n^4]$. First, note that, in the best case, the extra allocation of one capacity to $j^{k'}$ will improve the matching of all the residents in copy k' from $n^2 \cdot K_M + 4nL + 3n - L$ to $n^2 + 4nL + 3n - L$ if every $w^{k'}_{\ell,1}$ and every $i^{k'}_\ell$ is matched to the $\overline{j}^{k'}$ ranked first by i_ℓ for $\ell \in [n]$. Since y^k_ℓ was matched to $\overline{v}^k_{\ell,e}$, then y^k_ℓ will be necessarily matched after $\mathcal{V}^{k+1}_+$, which has cardinality $O(n^5 \cdot L)$. Therefore, the improvement in the cost $(O(n^2 \cdot K_M - n^2) \leq O(n^4))$ is smaller than the cost of rematching y^k_ℓ $(O(n^5 \cdot L))$. Similarly, if the extra capacity is taken from some $v^k_{\ell,e}$, then the corresponding $w^k_{\ell,e}$ that was matched to it, will take the place of y^k_ℓ in $\overline{v}^k_{\ell,r}$, thus re-matching y^k_ℓ after $\mathcal{V}^{k+1}_+$. As before, we would have an additional cost that is greater than any possible benefit.

Given the fact above, we only have to focus on feasible allocations to hospitals that belong to $\mathcal{V}$. In the following, we analyze why a different allocation of extra capacities in $\mathcal{V}$ does not lead to a stable matching with a lower average hospital rank. First, we prove that re-arranging one extra capacity differently within a village does not yield a stable matching with a lower cost;

we denote by $M^\star$ the stable matching obtained as a consequence of the re-organization of extra capacities in a village $\mathcal{B}_\ell^k$. Note that there are n extra capacities to allocate in village $\mathcal{B}_\ell^k$ and $n+1$ residents that could benefit of the re-allocation. It is evident that it is sub-optimal to not allocate one extra capacity to a hospital $\overline{v}_{\ell,r'}^k$ to match y_ℓ^k; in fact, otherwise, y_ℓ^k would be matched after $\mathcal{V}_+^{k+1}$ with an additional cost of $O(n^5 \cdot L)$. Therefore, the only possibility is to allocate one extra capacity so that a certain $v_{\ell,r'}^k$ does not receive one extra capacity. Assume $w_{\ell,r'}^k$ is the resident of village $\mathcal{B}_\ell^k$ that is not matched to $v_{\ell,r'}^k$ in $M^\star$, and assume that $j_{\sigma(\ell,r)}^k$ is the hospital to which resident $w_{\ell,r}^k$ is matched in $\hat{M}$. Therefore, we move one extra capacity from $v_{\ell,r}^k$ to $v_{\ell,r'}^k$. As a consequence, $(w_{\ell,r}^k, v_{\ell,r}^k)$ are matched and $w_{\ell,r'}^k$ is matched to $\overline{v}_{\ell,r}^k$, leaving y_ℓ^k to be matched after $\mathcal{V}_+^{k+1}$, making the re-allocation of one extra capacity within the village sub-optimal. At this point, the only re-allocation of one extra capacity that may improve the objective is that obtained by transferring one extra capacity from one village to another village.

Consider $i_\ell, i_{\ell'} \in \mathcal{S}'$. We now analyze the effects of moving one extra capacity from village $\mathcal{B}_{\ell'}^k$ to village $\mathcal{B}_\ell^{k'}$. The reason why we are analyzing these transfers of extra capacities is because the corresponding residents are not necessarily matched with their top choice so their ranking and the overall average ranking may improve.

- From $\overline{v}_{\ell,r}^k$ to $v_{\ell',r'}^{k'}$. In the best case, the reallocation of extra capacities in k' improves the matching by $O(n^4)$. Nonetheless, y_ℓ^k, who was previously matched to $\overline{v}_{\ell,r}^k$, is matched after $\mathcal{V}_+^{k+1}$, thus increasing the average rank by at least $O(L \cdot n^5)$. Note that y_ℓ^k cannot be matched to $v_{\ell',r'}^{k'}$ since there are at least n residents in copy $k'-1$ who would rather be matched to $v_{\ell',r'}^{k'}$ than to a $j^{k'}$.
- From $\overline{v}_{\ell,r}^k$ to $\overline{v}_{\ell',r'}^{k'}$. As mentioned earlier, the allocation of one extra capacity to copy k' may improve the matching by at most $O(n^4)$, but the additional cost of re-matching y_ℓ^k after $\mathcal{V}_+^{k+1}$ is of $O(L \cdot n^5)$.
- From $v_{\ell,e}^k$ to $\overline{v}_{\ell',r'}^{k'}$ or to $v_{\ell',r'}^{k'}$. In both cases copy k' improves at most by $O(n^4)$ and $w_{\ell,e}^k$ takes the spot of y_ℓ^k at $\overline{v}_{\ell',e}^k$, thus re-matching y_ℓ^k after $\mathcal{V}_+^{k+1}$ with an additional cost of $O(L \cdot n^5)$.

Now, let us prove that it is sub-optimal to re-allocate multiple extra capacities among $\mathcal{V} \cup \overline{\mathcal{V}}$. Note that since the hospitals in $\mathcal{Z} \cup \overline{H}$ are always ranked by the residents after at least $O(n^2)$ hospitals in $\mathcal{V} \cup \overline{\mathcal{V}}$, it will follow that it would also be sub-optimal to re-allocate multiple extra capacities to hospitals in $\mathcal{Z} \cup \overline{H}$; finally, it would also hold that it is sub-optimal to re-allocate extra capacities to hospitals in $\mathcal{Z} \cup \overline{H} \cup \mathcal{V} \cup \overline{\mathcal{V}}$, thus terminating the proof.

First, assume that as a consequence of reallocating extra capacities, some villages may have less than $n \cdot L$ assigned extra capacities and others more than $n \cdot L$ extra capacities. Let $\mathcal{B}_\ell^k$ be a village with $m' < n-1$ extra capacities, and define $m = n - m' > 1$ the number of "missing" extra capacities. In the best case, each of the m extra capacities will ameliorate the matching in a different copy k', with an overall improvement in the cost of $O(m \cdot n^4)$; note that allocating more than one extra capacity to the same copy k' could not improve the average rank in k' by more than $O(n^2 \cdot K_M) \leq O(n^4)$. Given that in village $\mathcal{B}_\ell^k$ there are less than $n-1$ extra capacities, it follows that at least two distinct residents $w_{\ell,e}^k$ and $w_{\ell,f}^k$ are not matched to $v_{\ell,e}^k$ and $v_{\ell,f}^k$, thus taking the priority to be matched to $\overline{\mathcal{V}}_\ell^k$ if one of the hospitals there has an extra capacity; this implies that y_ℓ^k must not be matched to $\overline{\mathcal{V}}_\ell^k$. Therefore, y_ℓ^k may be matched to a hospital $v_{\ell'',r''}^{k''}$ in $\mathcal{V}_+^{k+1}$ or may be matched after $\mathcal{V}_+^{k+1}$. The resident y_ℓ^k can be matched to a hospital $v_{\ell'',r''}^{k''}$ only if at least $n+2$ extra capacities are allocated to $v_{\ell'',r''}^{k''}$. Indeed, before y_ℓ^k, $v_{\ell'',r''}^{k''}$ prefers

(1) n residents $\{w_{\ell''',r'''}^{k''-1}, i_{\ell^v}^{k''-1}\}_{\ell'''\in L, \ell^v \in \{L+1,\ldots,n\}, r'''\in[n]}$ from copy $(k''-1) \mod n^4$ which are matched to n hospitals $j^{k''-1}$ they rank worst than $v_{\ell'',r''}^{k''}$, and

(2) $w_{\ell'',r''}^{k''}$.

Therefore, in order to match y_ℓ^k to a hospital $v_{\ell'',r''}^{k''}$ in $\mathcal{V}_+^{k+1}$, we would need to allocate at least $n+2$ extra capacities to $v_{\ell'',r''}^{k''}$; these $n+2$ would come from at least two villages, hence matching at least two residents $y_\ell^{k''''}$ after $\mathcal{V}$. Hence, w.l.o.g. we can assume that y_ℓ^k is matched after $\mathcal{V}_+^{k+1}$, contributing with an additional cost of $O(n^5 \cdot L)$. Note that at least $m-1$ residents $w_{\ell,e}^k$ from village $\mathcal{B}_\ell^k$ are not matched to their most preferred hospitals $v_{\ell,e}^k$ nor to any hospital in $\mathcal{V}_\ell^k$. Therefore, these $m-1$ residents $w_{\ell,e}^k$ may be matched to some hospital $v_{\ell,e'}^{k+1}$ in $\mathcal{V}_+^{k+1}[n^3]$. For a $w_{\ell,e}^k$ to be matched to $v_{\ell,e'}^{k+1}$, hospital $v_{\ell,e'}^{k+1}$ must receive at least two extra capacities, one for matching $w_{\ell,e'}^{k+1}$ and the other for matching $w_{\ell,e}^k$. Hence, the extra capacity that was originally allocated to $w_{\ell,e}^k$ would be given to $v_{\ell,e'}^{k+1}$, and rather than improving the average rank of copy $k+1$ (which is un-affected), the allocation would match worst $w_{\ell,e}^k$. Therefore, everyone of the $m-1$ residents $w_{\ell,e}^k$ should be matched not to a hospital in $\mathcal{V}_+^{k+1}[n^3]$. If $w_{\ell,e}^k$ is matched to a hospital j^k, then the resident i^k who was previously matched to it, would be matched after $\mathcal{Z}$ with an additional cost of $O(n^6)$. Therefore, w.l.o.g., each of the $m-1$ residents $w_{\ell,e}^k$ will be matched after $\mathcal{Z}$ (if they are matched to $\mathcal{Z}$, then one extra capacity should be allocated to $\mathcal{Z}$) with an additional average cost of $O((m-1)\cdot n^6)$, which is greater than the benefit of $O(m \cdot n^4)$.

Second, many extra capacities may be allocated differently than suggested by the reduction of Lemma 3.3.5, with the condition that every village receives n extra capacities. Note that in every village $\mathcal{B}_\ell^k$ at least one extra capacity should be allocated to $\overline{\mathcal{V}}_\ell^k$, this must be the case because otherwise y_ℓ^k would be matched after $\mathcal{V}_+^{k+1}$ with an additional cost of $O(n^5 \cdot L)$, which is greater than any gain that may be achieved by re-allocating differently the $L \cdot n$ extra capacities in copy k (recall that the average cost of copy k is $O(n^2 \cdot K_M)$. Then, note that it is also sub-optimal to allocate more than one extra capacity to $\overline{\mathcal{V}}_\ell^k$, because at least one resident $w_{\ell,e}^k$ would be matched to it; hence, it would be more convenient to match $w_{\ell,e}^k$ to $v_{\ell,e}^k$ by allocating one extra capacity to hospital $v_{\ell,e}^k$. Hence, in every village there will be a resident $w_{\ell,e}^k$ who is not matched to $v_{\ell,e}^k$ because this hospital did not receive any extra capacity. Such $w_{\ell,e}^k$ cannot be matched to any of the hospitals in $\mathcal{V}_+^{k+1}[n^3]$, unless we would provide at least two extra capacities to it (thus matching a $y_{\ell'}^{k'}$ after $\mathcal{V}_+^{k'+1}$ with an extra cost of $O(n^5 \cdot L)$). Therefore, $w_{\ell,e}^k$ can be matched to hospital $j_{\sigma(\ell,e)}^k$, or after $\mathcal{Z}$ (with an extra cost of $O(n^6)$. So, for a fixed copy k, we would have that one resident in every of the L villages in copy k, may be matched at best to a hospital j^k; on the other hand, also the residents $\{i_\ell^k\}_{\ell \in \{L+1,\dots,n\}}$ are matched at best to a hospital j^k. This implies that when each village receives n extra capacities, the best way of allocating them is by having a resident in each village matched to a hospital j^k (recall that each of these hospitals has capacity one). Hence, at best, n residents in copy k (at most one from each village) are matched to the n hospitals j^k, thus projecting a matching M' in the original instance Γ. M' must be a weakly stable matching of Γ since the preferences of the hospitals in $\overline{H}$ rank the residents in copy k according to the original preferences in Γ. Note that since M was a minimal average rank matching, then AvgRank$(M) \leq$ AvgRank(M'), which implies that

$$n^2 \cdot \text{AvgRank}(M) + 4nL + 3n - L \leq n^2 \cdot \text{AvgRank}(M') + 4nL + 3n - L, \tag{3.7.1}$$

where the left-hand side of equation 3.7.1 is the average rank of copy k according to the reduction as per Lemma 3.3.5, while the term on the right-hand side is the average rank of copy k obtained following the reasoning above.

Therefore, there is no allocation $\tilde{\mathbf{t}} \neq \mathbf{t}$ and a stable matching $\hat{M}_{\tilde{\mathbf{t}}}$ with an objective value strictly lower than $\overline{K}$.

□

3.7.4. Proof of Theorem 3.5.3

In this section, we prove that $\text{Max-Card}^{\text{SUB}}_{\text{EXP}}$ HRI is NP-complete.

Proof of Theorem 3.5.3. We build a polynomial reduction from an instance of Max-Card HRTI with target value K, where ties are only on the hospital side, they are at the head of the preference list and are of length two. Let $\mathcal{C}$ and $\mathcal{S}$ be the set of hospitals and residents in Γ, respectively; $\mathcal{C} = \mathcal{C}' \cup \mathcal{C}''$, where $\mathcal{C}'$ is the set of hospitals with a tie at the head of the preference list and $\mathcal{C}''$ is the set of hospitals with a strict preference list.

We build an instance $\hat{\Gamma} = \langle \hat{\mathcal{S}}, \hat{\mathcal{C}}, \hat{\succ}, \hat{\mathbf{c}} \rangle$ of $\text{Max-Card}^{\text{SUB}}_{\text{EXP}}$ HRI as follows:

- The set of residents is composed of a copy of $\mathcal{S}$ that we denote by $\overline{\mathcal{S}}$ and two sets of residents $\mathcal{U} = \{u_j \colon \forall j \in \mathcal{C}'\}$ and $\overline{\mathcal{U}} = \{\overline{u}_j \colon \forall j \in \mathcal{C}'\}$, each of size $|\mathcal{C}'|$;
- The set of hospitals $\hat{\mathcal{C}}$ consists of a copy of $\mathcal{C}''$ and the set $\tilde{\mathcal{C}} = \{j' : \ j \in \mathcal{C}'\} \cup \{j'' : \ j \in \mathcal{C}'\}$, *i.e.*, we make two copies per hospital in $\mathcal{C}'$. Each hospital in $\tilde{\mathcal{C}}$ has capacity 1 and each hospital in $\mathcal{C}''$ has capacity 1;
- For each resident in $\overline{\mathcal{S}}$, we keep the preference list that she has in the original instance Γ, with the exception that each $j \in \mathcal{C}'$ in her preference list is replaced by j' if she does not appear in the tie. If she is the first resident listed in the tie of $j \in \mathcal{C}'$, then we replace the hospital j in the preference list by j'; otherwise, if the resident is listed second in the tie of $j \in \mathcal{C}'$, then we replace the hospital j in the preference list by j'';
- Every resident u_j only ranks hospital j', and every resident $\overline{u}_j$ only ranks hospital j'';

- For the hospitals in $\mathcal{C}''$, we maintain their preference lists of Γ over the residents in $\mathcal{S}$. For a hospital $j \in \mathcal{C}'$ with a preference list $(i_{\sigma_1}, i_{\sigma_2})$,$i_{\sigma_3}, \ldots$,$i_{\sigma_s}$, the preference list of j' becomes $u_{j'}$,i_{σ_1},i_{σ_2},$i_{\sigma_3}, \ldots$,i_{σ_s} and that of j'' becomes $\overline{u}_{j''}$,i_{σ_2},i_{σ_1},$i_{\sigma_3}, \ldots$,i_{σ_s};
- For each hospital $j \in \mathcal{C}''$, we create a set $\mathcal{C}_j = \{j\}$ with $B_j = 0$. For every hospital $j \in \mathcal{C}'$, we create a set $\mathcal{C}_j = \{j', j''\}$ with $B_j = 1$. Clearly, the sets $\mathcal{C}_j$ induce a partition of the set of hospitals $\hat{\mathcal{C}}$.
- The target value is $K + 2L$, where $L = |\mathcal{C}'|$.

First of all, note that for every $j \in \mathcal{C}'$, the pairs $(u_{j'}, j')$ and $(\overline{u}_{j''}, j'')$ will always be matched in every stable matching of instance $\hat{\Gamma}$.

Let M be a weakly stable matching of the Max-Card HRTI instance. We will show that there is a feasible allocation of the capacities $\mathbf{t}$ and a stable matching $M_\mathbf{t}$ in $\hat{\Gamma}_\mathbf{t}$ with cardinality $|M| + 2L$, and thus, establishing the problems' equivalence. For every pair (i,j) in M, we have to distinguish whether $j \in \mathcal{C}'$ or $j \in \mathcal{C}''$. If $j \in \mathcal{C}''$, then we add the corresponding pair (i,j) to $M_\mathbf{t}$; recall that for a hospital $j \in \mathcal{C}''$, $B_j = 0$. Otherwise, $j \in \mathcal{C}'$. If $i \neq i_{\sigma_2}$, then we allocate the extra capacity of part $\mathcal{C}_j$ to j' and we match the pair (i,j'). If, instead, $i = i_{\sigma_2}$, then we match the pair (i,j'') by assigning the extra capacity of part $\mathcal{C}_j$ to j''. If there is a hospital $j \in \mathcal{C}'$ that has not been assigned to any resident, then we allocate the extra capacity of part $\mathcal{C}_j$ to j'.

Note that $M_\mathbf{t}$ is stable indeed. If not, there must be a blocking pair (i,j). Note that j must be in some $\mathcal{C}_k$ given that those subsets form a partition of $\hat{\mathcal{C}}$. Indeed, in each set $\mathcal{C}_k$ exactly one hospital has one extra capacity 1, and for $k = j$, j is exactly such hospital. If $|\mathcal{C}_j| = 1$, then $j \in \mathcal{C}''$ and, thus, it has exactly the same preference list that it has in the instance Γ; therefore the corresponding pair (i,j) in M is a blocking pair, which yields a contradiction. If $|\mathcal{C}_j| = 2$, then we have to distinguish whether $j = j'$ or $j = j''$. If $j = j'$, then we find that (i,j) is a blocking pair in M. Otherwise, if $j = j''$, then (i,j) is a blocking pair if and only if $i = i_{\sigma_2}$ since it is the only resident ranking j'' in $\hat{\Gamma}$. The pair (i,j'') could be a blocking pair only if j'' has capacity 2; the extra capacity $B_j = 1$ was assigned to j'' in accordance with the reduction. Therefore (i,j'') is already matched in $M_\mathbf{t}$ and (i,j'') cannot be a blocking pair.

Note that the reduction is injective. Moreover, given a weakly stable matching M in instance Γ of cardinality greater than or equal to K, it immediately follows that the reduction finds a stable matching of cardinality at least $K + 2L$.

On the other hand, assume there is no weakly stable matching of cardinality greater than K. Let us assume, by contradiction that there is an allocation t of extra capacities and a stable matching $M_\mathbf{t}$ in $\hat{\Gamma}_t$ of cardinality at least $K + 2L$. Without loss of generality, if for a hospital $j \in \mathcal{C}'$, both corresponding hospitals j' and j'' are not matched to any other resident than $u_{j'}$ and $\overline{u}_{j''}$ in $M_\mathbf{t}$, then we assume that the extra capacity is allocated to j'. Let M' be the matching that we build in instance Γ copying from $M_\mathbf{t}$: Every hospital $j \in \mathcal{C}''$ is matched as in $M_\mathbf{t}$, and every hospital $j \in \mathcal{C}'$ is matched to the same resident $i \in \mathcal{R}$ to which the copy of j in $\hat{\Gamma}$ that receive the extra spot is matched to (if the hospital is matched to two residents). Note that the matching M' has cardinality at least K, next we prove that it is also weakly stable. Indeed, if it would not be stable, then there would be a blocking pair (i,j). If $j \in \mathcal{C}''$, then the pair (i,j) is also a blocking pair in $\hat{\Gamma}$, which is not possible. Otherwise, $j \in \mathcal{C}'$ and it may be that j is not matched or that j is matched to another resident $\hat{i}$ that j ranks worst than i. In the first case, note that the extra capacity in $\hat{\Gamma}_t$ is allocated to j', which is matched only to $u_{j'}$, therefore (i, j') is also a blocking pair of $M_{\mathbf{t}}$, contradicting the assumption of stability. Finally, (i,j) is a blocking pair of M, where j is matched to $\hat{i}$ and j *strictly* prefers i over $\hat{i}$. Without loss of generality, assume the extra capacity is assigned to j', therefore $(j', \hat{i})$ are matched together in $M_{\mathbf{t}}$. Let us denote by i_{σ_1},i_{σ_2} the two residents ranked first in the tie of hospital j. If i is neither i_{σ_1} nor i_{σ_2}, then (i,j') is a blocking pair of $M_{\mathbf{t}}$ too since i is also more preferred than $\hat{i}$ by j'. Otherwise, i is one of the two residents i_{σ_1} or i_{σ_2}. If $i = i_{\sigma_1}$, then also j' ranks i better than $\hat{i}$ (recall that j' is also matched to $u_{j'}$ that is ranked first), therefore (i,j') is also a blocking pair in $M_{\mathbf{t}}$. Otherwise, $i = i_{\sigma_2}$, and (i,j) could be a blocking pair of M' only if $\hat{i}$ is ranked third or worst, which means that $\hat{i}$ is ranked fourth or worst by j', thus making (i,j') a blocking pair of $M_{\mathbf{t}}$ too, which is absurd.

To conclude, note that the created instance introduces a polynomial number of hospitals, residents, preferences and pairs $\{(\mathcal{C}_j, B_j)\}_{j \in \mathcal{C}}$ in the input. Moreover, it can be verified in polynomial-time that: (1) the vector of allocation $\mathbf{t}$ satisfies the corresponding constraints and (2) the constructed stable matching has a cardinality greater than or equal to the target value. $\square$

Chapter 4

Capacity Planning in Stable Matching

by

Federico Bobbio[1], Margarida Carvalho[2], Andrea Lodi[3], Ignacio Rios[4], and Alfredo Torrico[5]

([1]) CIRRELT, DIRO Université de Montréal
([2]) CIRRELT, DIRO Université de Montréal
([3]) Jacobs Technion-Cornell Institute, Cornell Tech
([4]) School of Management, The University of Texas at Dallas
([5]) CDSES, Cornell University

Prologue: In this work, for the first time to the best of our knowledge, we address Question 1, i.e., the problem of allocating optimally extra capacities in the many-to-one stable matching framework. Furthermore, we address Question 2, which poses the critical issue of whether the distribution of capacities is vulnerable to manipulation. Finally, applying our framework to the School Choice setting, we provide a model that empowers policymakers, letting them allocate capacities to pursue access or improvement (Question 3).

Article details: This work received the runner-up award at the CORS 2023 Student Paper Open Competition. A preliminary version of this paper was accepted at MATCH-UP 2022 and EAAMO 2022. Additionally, a similar version to the one presented in this chapter was accepted at EC'23; we chose to publish only an abstract for this conference. This work is currently submitted to Operations Research.

Contributions of the authors: Federico Bobbio participated in all the stages of the work, being the main author of the paper. In particular, he led the literature review, proposed the compact mathematical programming models, their linearizations and the theoretical analyses of their linear relaxations. From an algorithmic perspective, he provided structural results and the new separation method. From the perspective of mechanism properties, at early stages of the project, he provided results in terms of number of students accessing and improving their outcomes and strategyproofness, which were later refined in collaboration with co-authors. With respect to computational experiments, he contributed in all of its steps. He also wrote the first draft of the paper.
Margarida Carvalho provided the idea of the project. She also contributed with the non-compact formulation and its validity. She participated in the revision and editing of the paper.
Andrea Lodi contributed with the initial conceptualization of the problem, provided directions for the experimental and modeling work. He participated in the revision and editing of the paper.
Ignacio Rios contributed on the modeling and on devising the properties of the mechanism. He also contributed in the experimental setting. He participated in the revision and editing of the paper.
Alfredo Torrico contributed to the initial conceptualization of the problem, he gave guidance on the algorithmic implementation since the early stages and contributed to the decision making in every aspect of the project. He participated in the revision and editing of the paper.

RÉSUMÉ. Nous introduisons le problème de l'augmentation conjointe des capacités des écoles et de la recherche d'une affectation optimale des étudiants sur le marché élargi. En raison de l'impossibilité de résoudre efficacement le problème avec les méthodes classiques, nous généralisons les formulations existantes de programmation mathématique des contraintes de stabilité à notre cadre, dont la plupart aboutissent à des programmes à contraintes quadratiques entières. En outre, nous proposons une nouvelle formulation de programmation linéaire en nombres entiers mixtes qui est exponentiellement grande sur la taille du problème. Nous montrons que ses contraintes de stabilité peuvent être séparées en exploitant la fonction objective, ce qui conduit à un algorithme efficace de plan de coupe. Nous concluons l'analyse théorique du problème en discutant de certaines propriétés du mécanisme. Sur le plan algorithmique, nous évaluons les performances de nos approches dans une étude détaillée, et nous constatons que notre méthode de plan de coupe est plus performante que notre généralisation des approches mixtes en nombres existantes. Nous proposons également deux heuristiques qui sont efficaces pour les grandes instances du problème. Enfin, nous utilisons les données du système chilien de choix des écoles pour démontrer l'impact de la planification de la capacité dans des conditions de stabilité. Nos résultats montrent que chaque place supplémentaire peut bénéficier à plusieurs élèves et qu'il est possible de cibler efficacement l'affectation d'élèves précédemment non affectés ou d'améliorer l'affectation de plusieurs élèves grâce à des chaînes d'amélioration. Ces résultats permettent au décideur d'ajuster l'algorithme d'appariement afin de fournir une application équitable.
Mots clés : Appariement stable, planification des capacités, choix de l'école, programmation en nombres entiers

ABSTRACT. We introduce the problem of jointly increasing school capacities and finding a student-optimal assignment in the expanded market. Due to the impossibility of efficiently solving the problem with classical methods, we generalize existent mathematical programming formulations of stability constraints to our setting, most of which result in integer quadratically-constrained programs. In addition, we propose a novel mixed-integer linear programming formulation that is exponentially large on the problem size. We show that its stability constraints can be separated by exploiting the objective function, leading to an effective cutting-plane algorithm. We conclude the theoretical analysis of the problem by discussing some mechanism properties. On the computational side, we evaluate the performance of our approaches in a detailed study, and we find that our cutting-plane method outperforms our generalization of existing mixed-integer approaches. We also propose two heuristics that are effective for large instances of the problem. Finally, we use the Chilean school choice system data to demonstrate the impact of capacity planning under stability conditions. 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. These insights empower the decision-maker in tuning the matching algorithm to provide a fair application-oriented solution.
Keywords: Stable matching, capacity planning, school choice, integer programming

4.1. Introduction

Centralized mechanisms are becoming the standard approach to solve several assignment problems. Examples include 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 [**138**] and [**135**], finding a stable matching is crucial for the clearinghouse's success, long-term sustainability and also ensures some notion of fairness as it eliminates so-called *justified-envy*.[1]

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 [**124**]. 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. In addition, school administrators often report how many open seats they have in each grade based on their current enrollment and the size of their classrooms. However, they could switch classrooms of different sizes to modify the seats offered on each level. Finally, in both school choice and college admissions, affirmative action policies include special seats for under-represented groups (such as lower-income students, women in STEM programs, etc.) that are allocated based on students' preferences and chances of succeeding.

As the previous discussion illustrates, capacities may be flexible, and it may be natural to incorporate them as a decision to further improve the assignment process. By jointly deciding capacities and the allocation, the clearinghouse can leverage the knowledge about agents' preferences to achieve different goals. On the one hand, one possible goal is to maximize *access*, i.e., to choose an allocation of capacities that maximizes the total number of agents being assigned.

[1]See [**128**] for a discussion on the differences between *stability* and *no justified-envy*.

[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 [**45**], over-subscribed courses often increase their capacities, while under-subscribed ones are merged or canceled.

[3]According to the results of a nationwide survey, 22% of schools in the US experienced some degree of overcrowding, and 8% had enrollments that exceeded their capacity by more than 25% [**65**].

This objective is especially relevant in some settings, such as school choice, where the clearinghouse aims to ensure that each student is assigned to some school. On the other hand, the clearinghouse may wish to prioritize *improvement*, i.e., to enhance the assignment of high-priority students. This objective is common in merit-based settings such as college admissions and the hospital-resident problem. 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 when capacities are fixed.

We refer to the *capacity planning* problem as the problem of allocating capacities while at the same time choosing a matching. While the computation of a student-optimal stable matching can be done in polynomial time using the well-known Deferred Acceptance (DA) algorithm [**67**], the computation of a student-optimal matching under capacity planning is theoretically intractable (Theorem 3.3.1 of Chapter 3, [**40**]). Therefore, we face two important challenges: (i) devising a framework that takes into account students' preferences when making capacity decisions; and (ii) designing an algorithm that efficiently computes exact solutions of large-scale instances of the problem. The latter is particularly relevant for a policymaker that aims to test and balance access vs. improvement, since different settings and input parameters can lead to extremely different outcomes. Therefore, expanding modeling capabilities (such as the inclusion of capacity expansion) and the methodologies for solving them are crucial to amplify the flexibility of future matching mechanisms.

4.1.1. Contributions and Paper Organization

Our work combines a variety of methodologies and makes several contributions that we now describe in detail.

Model and mechanism analysis. To capture the problem described above, we introduce a novel 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 [**67**]. 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 mechanism's monotonicity.

Exact solution methods. First, we formulate our problem as an integer quadratically-constrained program by extending existing approaches. We provide two different linearizations to improve the computational efficiency and show that one has a better linear relaxation. This is particularly important as tighter relaxations generally indicate faster running times when using commercial solvers. Preliminary computational results motivated us to devise a novel formulation of the problem and, subsequently, design a cutting-plane method to solve larger instances exactly. Specifically, we introduce a mixed-integer extended formulation where the extra capacity allocations are determined by binary variables and the assignment variables are relaxed. This formulation is of exponential size and, consequently, we solve it by a cutting-plane method that uses as a starting point the relaxation of the extended formulation without stability constraints. In each iteration, such a cutting-plane method relies on two stable matchings, one fractional and one integral. The first one is the current optimal assignment solution, while the second matching is obtained by applying the DA algorithm on the expanded market defined by the current optimal integral capacity allocation. These two stable matchings serve as proxies to guide the secondary process of finding violated constraints. The search for the most violated constraints focuses on a considerably smaller subset of them by exploiting structural properties of the problem.

We emphasize that this separation method does not contradict the hardness result presented in Chapter 3 (Theorem 3.3.1), since this algorithm relies on the solution of a mixed-integer formulation. Our separation algorithm somewhat resembles the method proposed by Baïou and Balinski [**26**] for the standard setting of the stable matching problem with no capacity expansion. However, the search space in our setting is significantly larger as we have an exponential family of constraints defined over a pseudopolynomial-sized space. Our efficient algorithm relies on two main aspects: The stable matchings that are used as proxies and a series of new structural results. The sum of all these technical enhancements ensures that our cutting-plane method outperforms the benchmarks obtained by adapting the formulations in the existing literature to the capacity planning setting, and solved by state-of-the-art solvers.

Heuristic solution methods. As shown in Chapter 3 (Theorem 3.3.1), the problem is NP-hard and cannot be approximated within a $\mathcal{O}\left(n^{\left(\frac{1}{6}-\varepsilon\right)}\right)$ factor, where n is the number of students, unless P=NP. Moreover, from our collaborations with the Chilean agencies, we realized that many real-life instances could not be solved in a reasonable time, even using our cutting plane algorithm. This motivated us to study efficient heuristics that possibly provide near-optimal solutions. In particular, we focus on two heuristics: First, we consider the standard *Greedy* algorithm for set functions that sequentially adds one extra seat in each iteration to the school that leads to the largest marginal improvement in the objective function. Our second heuristic, called *LPH*, proceeds in two steps: (i) solves the problem without stability constraints to find the allocation of extra seats, and (ii) finds 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 close-to-optimal solution with LPH being the fastest. Moreover, LPH outperforms Greedy in terms of optimality gap when the budget of extra seats increases and does that in a negligible amount of time. Hence, LPH could be a good approach to quickly solve large-scale instances of the problem.

Practical insights and societal impact. To illustrate the benefits of embedding capacity decisions, we use data from the Chilean school choice system and we adapt our framework to solve the problem including all the specific features described by Correa et al. [**50**]. First, we show that each additional seat can benefit multiple students. 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. Our results show that the students' matching is improved even if we upper bound the total number of additional seats per school. Finally, we show that our model can be extended in several interesting directions, including the addition of costs to expand capacities, the addition of secured enrollment, the planning of classroom assignment to different grades, etc.

Given these positive results, we are currently collaborating with the institutions in charge of implementing the Chilean school choice and college admissions systems to test our framework in the field. The computational time reduction enabled by our method(s) has been critical to evaluate both the assignment of extra seats to schools and the rules of affirmative action policies in college admissions, as assessing the impact of these policies requires thousands of simulations to understand their effects under different scenarios. Moreover, our model can be easily adapted to tackle other policy-relevant questions. For instance, it can be used to optimally decide how to decrease capacities, as some school districts are experiencing large drops in their enrollments [**149**]. Our model could also be used to optimally allocate tuition waivers under budget constraints, as in the case of Hungary's college admissions system. Finally, our methodology could also be used in other markets, such as refugee resettlement [**52, 17, 14**]—where local authorities define how many refugees they are willing to receive, but they could increase their capacity given proper incentives—or healthcare rationing [**122, 25**]—where policymakers can make additional investments to expand the resources available. These examples further illustrate the importance of jointly optimizing stable assignments and capacity decisions since it can answer crucial questions in numerous settings.

Organization of the paper. The remainder of the paper is organized as follows. In Section 4.2, we provide a literature review. In Section 4.3, we formalize the stable matching problem within the framework of capacity planning; then, we present our methodologies to solve the problem, including the compact formulations and their linearizations, our novel non-compact formulation and our cutting-plane algorithm; and we conclude this section by discussing some properties of our mechanism. In Section 4.4, we provide a detailed computational study on a synthetic dataset. In Section 4.5, we evaluate our framework using Chilean school choice data. Finally, in Section 4.6, we draw some concluding remarks. All the proofs, examples, extensions and additional discussions can be found in the Appendix.

4.2. Related Work

Gale and Shapley [**67**] introduced the well-known Deferred Acceptance algorithm, which finds a stable matching in polynomial time for any instance of the problem. Since then, the literature on stable matchings has extensively grown and has focused on multiple variants of the problem. For this reason, we focus on the most closely related work, and we refer the interested reader to [**107**] for a broader literature review.

Mathematical programming formulations. The first mathematical programming formulations of the stable matching problem were studied in [**71, 150, 139**] and [**137**]. Baïou and Balinski [**26**] provided thereafter an exponential-size linear programming formulation describing the convex hull of the set of feasible stable matchings. Moreover, they gave a polynomial-time separation algorithm. Kwanashie and Manlove [**104**] presented an integer formulation of the problem when there are ties in the preference lists (i.e., when agents are indifferent between two or more options). Kojima et al. [**99**] introduced a way to represent preferences and constraints to guarantee strategy-proofness. Agoston et al. [**13**] proposed an integer model that incorporates upper and lower quotas. Delorme et al. [**53**] devised new mixed-integer programming formulations and pre-processing procedures. More recently, Agoston et al. [**157**] proposed similar mathematical programs and used them to compare different policies to deal with ties. In our computational experiments, we consider the adaptation of these formulations to capacity planning which then form the baselines of our approach.

Capacity expansion. Our paper is the first to introduce the problem of optimal capacity planning in the context of stable matching. After this manuscript's preliminary version was released, there has been subsequent work in the same setting. Bobbio et al. [**40**] (Chapter 3) studied the capacity planning problem's complexity and other variations. The authors showed that the decision version of the problem is NP-complete and inapproximable within a $\mathcal{O}\left(n^{(\frac{1}{6}-\varepsilon)}\right)$ factor, unless P=NP, where n is the number of students. Abe et al. [**9**] studied a heuristic method to solve the capacity planning problem that relies on the Upper Confidence Tree (a Monte Carlo tree search method) searching over the space of capacity expansions. Dur and Van der Linden [**58**], 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 discuss 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 potentially recover any Pareto efficient allocation, including the ones returned by our mechanism. However, their algorithm cannot be generalized to achieve a specific outcome. Our methodology is flexible enough to enable policymakers to target a particular goal when deciding how to allocate the extra seats, including access, improvement, or any other objective beyond student optimality. Finally, Dur and Van der Linden [**58**] 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. Another related capacity expansion model is considered in Kumano and Kurino [**101**], where the authors study and implement 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 et al. [**8**], a large body of literature has studied different elements of the school choice problem, including the use of different mechanisms such as DA, Boston, and Top Trading Cycles [**5, 121, 2**]; the use of different tie-breaking rules [**4, 18, 20**]; the handling of multiple and potentially overlapping quotas [**102, 143**]; the addition of affirmative action policies [**61, 72**]; and the implementation in many school districts and countries [**5, 46, 50, 16**]. Within this literature, the closest papers to ours are those that combine the optimization of different objectives with finding a stable assignment. Caro et al. [**47**] introduced an integer programming model to make school redistricting decisions. Shi [**141**] proposed a convex optimization model to decide the assortment of schools to offer to each student to maximize the sum of utilities. Ashlagi and Shi [**22**] 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 [**42**] presented an optimization model to find the best stable and incentive-compatible match that maximizes any combination of welfare, diversity, and prioritizes the allocation of students to their neighborhood school. Finally, Feigenbaum et al. [**62**] introduced a novel mechanism to efficiently reassign vacant seats after an initial round of a centralized assignment and used data from the NYC high school admissions system to showcase its benefits.

4.3. Model

We formalize the stable matching problem using school choice as an illustrating example. Let $\mathcal{S} = \{s_1, \ldots, s_n\}$ be the set of n students, and let $\mathcal{C} = \{c_1, \ldots, c_m\}$ be the set of m schools.[4] Each student $s \in \mathcal{S}$ reports a strict preference order $\succ_s$ over the elements in $\mathcal{C} \cup \{\emptyset\}$. Note that we allow for $\emptyset \succ_s c$ for some $c \in \mathcal{C}$, so students may not include all schools in their preference list.

[4]To facilitate the exposition, we assume that all students belong to the same grade, e.g., pre-kindergarten.

In a slight abuse of notation, we use $|\succ_s|$ to represent the number of schools to which student s applies and prefers compared to being unassigned, and we use $c' \succeq_s c$ to represent that either $c' \succ_s c$ or that $c' = c$. On the other side of the market, each school $c \in \mathcal{C}$ ranks the set of 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 of $q_c \in \mathbb{Z}_+$ seats, and we assume that $\emptyset$ has a sufficiently large capacity.

Let $\mathcal{E} \subseteq \mathcal{S} \times \left(\mathcal{C} \cup \{\emptyset\}\right)$ be the set of feasible pairs, with $(s,c) \in \mathcal{E}$ meaning that s includes school c in their preference list.[5] 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. A matching μ is stable if it has no *blocking pairs*, i.e., there is no pair $(s,c) \in \mathcal{E}$ that 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: (1) student s prefers school c over $\mu(s) \in \mathcal{C} \cup \{\emptyset\}$, and (2) $|\mu(c)| < q_c$ or there exists $s' \in \mu(c)$ such that $s \succ_c s'$, i.e., c prefers s over s'.

For any instance $\Gamma = \langle \mathcal{S},\mathcal{C}, \succ, \mathbf{q}\rangle$, the DA algorithm [**67**] can find in polynomial time the unique stable matching that is *weakly preferred* by every student, also known as the student-optimal stable matching. Moreover, 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 4.9.1, we formally describe the DA version that finds the student-optimal stable matching.

Let $r_{s,c}$ be the position (rank) of school $c \in \mathcal{C}$ in the preference list of student $s \in \mathcal{S}$, and let $r_{s,\emptyset}$ be a parameter that represents a penalty for having student s unassigned.[6] In Lemma 4.3.1 we show that, for any instance $\Gamma = \langle \mathcal{S},\mathcal{C}, \succ, \mathbf{q}\rangle$, we can find the student-optimal stable assignment by solving an integer linear program whose objective is to minimize the sum of students' preference of assignment and penalties for having unassigned students. The proof can be found in Appendix 4.7.1.

Lemma 4.3.1. Given an instance $\Gamma = \langle \mathcal{S},\mathcal{C}, \succ, \mathbf{q}\rangle$, finding the student-optimal stable matching is equivalent to solving the following integer program:

$$\min_{\mathbf{x}} \quad \sum_{(s,c)\in\mathcal{E}} r_{s,c} \cdot x_{s,c} \tag{4.3.1a}$$

$$s.t. \quad \sum_{c:(s,c)\in\mathcal{E}} x_{s,c} = 1, \qquad \forall\, s \in \mathcal{S}, \tag{4.3.1b}$$

$$\sum_{s:(s,c)\in\mathcal{E}} x_{s,c} \le q_c, \qquad \forall\, c \in \mathcal{C}, \tag{4.3.1c}$$

[5]To ease notation, we assume that students include $\emptyset$ at the bottom of their preference list, and we assume that any school $c \in \mathcal{C}$ not included in the preference list is such that $\emptyset \succ_s c$.

[6]Note that the penalty $r_{s,\emptyset}$ may be different from the ranking of $\emptyset$ in student s preference list. As such, the penalty does not directly affect the stability condition.

$$q_c x_{s,c} + q_c \cdot \sum_{c' \succ_s c} x_{s,c'} + \sum_{s' \succ_c s} x_{s',c} \geq q_c, \quad \forall\ (s,c) \in \mathcal{E}, c \succ_s \emptyset \tag{4.3.1d}$$

$$x_{s,c} \in \{0,1\}, \qquad \forall\ (s,c) \in \mathcal{E}. \tag{4.3.1e}$$

Note that Formulation (4.3.1) can be solved in polynomial time using the reformulation and the separation algorithm proposed by Baïou and Balinski [**26**].[7]

Notation. In the remainder of the paper, we use bold to write vectors and italic for one-dimensional variables; for instance, $\mathbf{t} = (t_1, \ldots, t_m)$.

4.3.1. Capacity Expansion

In Formulation (4.3.1), the goal is to find a student-optimal stable matching. In this section, we adapt this problem to incorporate capacity expansion decisions. Let $\mathbf{t} = \{t_c\}_{c \in \mathcal{C}} \in \mathbb{Z}_+^{\mathcal{C}}$ be the vector of additional seats allocated to each school $c \in \mathcal{C}$, and let $\Gamma_{\mathbf{t}} = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} + \mathbf{t} \rangle$ be the instance of the problem in which the capacity of each school c is $q_c + t_c$.[8] For a non-negative integer B, the *capacity expansion problem* consists in finding an allocation $\mathbf{t}$ that does not violate the budget B and a stable matching μ in $\Gamma_{\mathbf{t}}$ that minimizes the sum of preferences of assignment for the students and penalties for having unassigned students. Given budget $B \in \mathbb{Z}_+$, this can be formalized as

$$\min_{\mathbf{t},\mu} \left\{ \sum_{(s,c) \in \mu} r_{s,c} \ : \ \mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}},\ \sum_{c \in \mathcal{C}} t_c \leq B,\ \mu \text{ is a stable matching in instance } \Gamma_{\mathbf{t}} \right\}. \tag{4.3.2}$$

In other words, an optimal allocation of extra seats in Formulation (4.3.2) leads to a student-optimal stable matching whose objective value is the best among all feasible capacity expansions.

Remarks. First, note that Formulation (4.3.2) is equivalent to Formulation (4.3.1) when $B = 0$. Second, Formulation (4.3.2) may have multiple optimal assignments (see Appendix 4.8.1). Third, observe that an optimal allocation may not necessarily use the entire budget since the objective value may no longer improve, e.g., if we assign every student to their top preference. Finally, it is important to highlight that students' preferences are an input of the problem; thus, capacity decisions are made in response. This assumption is suitable when the planner can implement these capacity decisions in a shorter timescale, such as decisions on adding additional seats in a course/grade/program, merging neighboring schools, or re-organizing classroom assignments to different courses/grades depending on their popularity. Nevertheless, our framework can also help evaluate longer-term policies, especially when the number of applicants and their preferences are consistent over time. Moreover, our framework is flexible enough to accommodate any other objective beyond student optimality, such as minimizing implementation costs (e.g., adding

[7][**104**] and subsequent papers name Formulation (4.3.1) as MAX-HRT. Although in some cases the objective function may differ and they may consider ties or other extensions, the set of constraints they study capture the same requirements as our constraints.

[8]Note that $\Gamma_{\mathbf{0}}$ corresponds to the original instance Γ with no capacity expansion.

teachers, portable classrooms, etc.), transportation costs, students' estimated welfare, or any combination of goals. As discussed in Section 4.5.3, our framework can be used to decide the optimal number of reserved seats to offer to under-represented groups or determine the minimum requirements that schools should meet regarding these affirmative action policies.

Motivated by the theoretical complexity of our problem and the absence of constant factor approximation algorithms (in Appendix 4.7.5 we show that also sub-modularity is not a viable way), we now concentrate on mathematical programming approaches to solve Formulation 4.3.2.

4.3.2. Compact Formulation

Recall that $\mathbf{t}$ denotes the vector of extra seats allocated to the schools and that $B \in \mathbb{Z}_+$ is the total budget of additional seats. Let

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

be the set of fractional (potentially non-stable) matchings with capacity expansion. Note that the first condition states that each student must be fully assigned to a school in $\mathcal{C} \cup \{\emptyset\}$. The second condition ensures that updated capacities (including the extra seats) are respected, and the last condition guarantees that the budget is not exceeded.

Let $\mathcal{P}_{\mathbb{Z}}$ be the integer points of $\mathcal{P}$, i.e., $\mathcal{P}_{\mathbb{Z}} = \mathcal{P} \cap \left(\{0,1\}^{\mathcal{E}} \times \{0,\ldots,B\}^{\mathcal{C}} \right)$. Then, we model our problem by generalizing Formulation (4.3.1) to incorporate the decision vector $\mathbf{t}$. As a result, we obtain the following integer quadratically constrained program:

$$\min_{\mathbf{x},\mathbf{t}} \quad \sum_{(s,c)\in\mathcal{E}} r_{s,c} \cdot x_{s,c} \tag{4.3.3a}$$

$$s.t. \quad (t_c + q_c) \cdot \left(1 - \sum_{c' \succeq_s c} x_{s,c'} \right) \leq \sum_{s' \succ_c s} x_{s',c}, \qquad \forall\, (s,c) \in \mathcal{E}, c \succ_s \emptyset \tag{4.3.3b}$$

$$(\mathbf{x},\mathbf{t}) \in \mathcal{P}_{\mathbb{Z}}. \tag{4.3.3c}$$

Constraint (4.3.3b) guarantees that the matching is stable. Note that when $(\mathbf{x},\mathbf{t})$ is allowed to be fractional, this contraint is quadratic and non-convex, which adds an extra layer of complexity on top of the integrality requirements (4.3.3c).

To address the challenge introduced by the quadratic constraints, we linearize them with McCormick envelopes (see Appendix 4.9.2 for a brief background). The quadratic term $t_c \cdot \sum_{c' \succeq_s c} x_{s,c'}$ in constraint (4.3.3b) can be linearized in at least two ways. Specifically, we call

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

$$\min_{\mathbf{x},\mathbf{t},\alpha} \quad \sum_{(s,c)\in\mathcal{E}} r_{s,c}\cdot x_{s,c} \tag{4.3.4a}$$

$$s.t. \quad t_c - \alpha_{s,c} + q_c \cdot \left(1 - \sum_{c' \succeq_s c} x_{s,c'}\right) \leq \sum_{s' \succ_c s} x_{s',c}, \qquad \forall\ (s,c)\in\mathcal{E}, c \succ_s \emptyset \tag{4.3.4b}$$

$$-\alpha_{s,c} + t_c + B\cdot \sum_{c' \succeq_s c} x_{s,c'} \leq B, \qquad \forall\ (s,c)\in\mathcal{E}, c \succ_s \emptyset \tag{4.3.4c}$$

$$\alpha_{s,c} \leq t_c, \qquad \forall\ (s,c)\in\mathcal{E}, c \succ_s \emptyset \tag{4.3.4d}$$

$$\alpha_{s,c} \leq B\cdot \sum_{c' \succeq_s c} x_{s,c'}, \qquad \forall\ (s,c)\in\mathcal{E}, c \succ_s \emptyset \tag{4.3.4e}$$

$$(\mathbf{x},\mathbf{t}) \in \mathcal{P}_{\mathbb{Z}}, \alpha \geq 0. \tag{4.3.4f}$$

Constraints (4.3.4c), (4.3.4d), (4.3.4e) and the non-negativity constraints for $\alpha_{s,c}$ form the McCormick envelope. The other constraints and the objective function remain the same.

It is well known that whenever at least one of the variables involved in the linearization is binary, the McCormick envelope leads to an equivalent formulation. This is the case for the aggregated linearization since $\sum_{c' \succeq_s c} x_{s,c'} \in \{0,1\}$ due to the constraints in $\mathcal{P}_{\mathbb{Z}}$.

Corollary 4.3.2. The projection of the feasible region given by constraints (4.3.4b)-(4.3.4f) in the variables $\mathbf{t}$ and $\mathbf{x}$ coincides with the region given by constraints (4.3.3b) and (4.3.3c).

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

$$\min_{\mathbf{x},\mathbf{t},\beta} \quad \sum_{(s,c)\in\mathcal{E}} r_{s,c}\cdot x_{sc} \tag{4.3.5a}$$

$$s.t. \quad t_c - \sum_{c' \succeq_s c} \beta_{s,c,c'} + q_c \cdot \left(1 - \sum_{c' \succeq_s c} x_{s,c'}\right) \leq \sum_{s' \succ_c s} x_{s',c}, \qquad \forall\ (s,c)\in\mathcal{E}, c \succ_s \emptyset \tag{4.3.5b}$$

$$-\beta_{s,c,c'} + t_c + B\cdot x_{s,c'} \leq B, \qquad \forall\ (s,c)\in\mathcal{E}, \forall\ c' \succeq_s c \succ_s \emptyset, \tag{4.3.5c}$$

$$\beta_{s,c,c'} \leq t_c, \qquad \forall\ (s,c)\in\mathcal{E}, \forall\ c' \succeq_s c \succ_s \emptyset, \tag{4.3.5d}$$

$$\beta_{s,c,c'} \leq B\cdot x_{s,c'}, \qquad \forall\ (s,c)\in\mathcal{E}, \forall\ c' \succeq_s c \succ_s \emptyset, \tag{4.3.5e}$$

$$(\mathbf{x},\mathbf{t}) \in \mathcal{P}_{\mathbb{Z}}, \beta \geq 0. \tag{4.3.5f}$$

Constraints (4.3.5c), (4.3.5d), (4.3.5e) and the non-negativity constraints for $\beta_{s,c,c'}$ form the McCormick envelope. Similar to the case of Corollary 4.3.2, we know that this is an exact formulation since $x_{s,c'} \in \{0,1\}$.

Corollary 4.3.3. The projection of the feasible region given by constraints (4.3.5b)-(4.3.5f) in the variables $\mathbf{t}$ and $\mathbf{x}$ coincides with the region given by constraints (4.3.3b) and (4.3.3c).

Therefore, Formulation (4.3.4) and (4.3.5) yield the same set of feasible solutions. Interestingly, the feasible region of the relaxed aggregated linearization, i.e., when $(\mathbf{x},\mathbf{t}) \in \mathcal{P}_{\mathbb{Z}}$ in constraint (4.3.4f) is changed to $(\mathbf{x},\mathbf{t}) \in \mathcal{P}$, is contained in the feasible region of the relaxed non-aggregated linearization.

Theorem 4.3.4. The feasible region of the relaxed aggregated linearization model is contained in the feasible region of the relaxed non-aggregated linearization model.

The proof of Theorem 4.3.4 can be found in Appendix 4.7.2. Theorem 4.3.4 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. In Appendix 4.8.3, we provide an example that shows that the inclusion in Theorem 4.3.4 is strict. Since solution approaches to mixed-integer programming formulations are based on the quality of their continuous relaxation, we conclude that the aggregated linearization dominates the non-aggregated one, and thus we expect it to perform better in practice.

As discussed in Section 4.2, there are other variants of Formulation (4.3.1) in the literature. In Appendix 4.10, we generalize the state-of-the-art formulations to the case where $B > 0$, and we note that all of them involve quadratic constraints. Although linearizations similar to the ones applied to Formulation (4.3.3) are possible, they result in a larger number of variables, Big-M constraints, and therefore, in potentially weak relaxations. Hence, in the next section, we provide an alternative mixed-integer programming (MIP) formulation that is non-compact, and we introduce a cutting-plane method to solve it efficiently.

4.3.3. Non-compact Formulation

For any instance Γ, Baïou and Balinski [**26**] describe the polytope of stable matchings (for the standard setting without extra capacities) through an exponential family of inequalities, called *comb constraints*, and provide a polynomial time algorithm to separate them. Inspired by their results, we propose a novel non-compact formulation to incorporate capacity decisions. One of the main challenges is that the formulation proposed by Baïou and Balinski [**26**] does not directly generalize to the capacity planning setting since the comb constraints depend on the capacity of each school, which can be modified in our setting. To address this, we appropriately generalize the definition of a comb and define the family of constraints by using additional decision variables. Generalized comb definition. A tooth $T(s,c)$ with base $(s,c) \in \mathcal{E}$ consists of (s,c) and all pairs (s,c') such that $c' \succ_s c$. 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. For $(s,c) \in \mathcal{E}_c^+(k)$, a shaft with base (s,c) where school c has expansion k is denoted by $S^k(s,c)$ and consists of (s,c) and all pairs (s',c) such that $s' \succ_c s$. For $(s,c) \in \mathcal{E}_c^+(k)$, a comb with base (s,c) where school c has expansion k is denoted by

$C^k(s,c)$ and consists of the union between $S^k(s,c)$ and exactly $q_c + k$ teeth of $(s',c) \in S^k(s,c)$, including $T(s,c)$. Finally, $C_c(k)$ is the family of combs for school $c \in \mathcal{C}$ with expansion k.

Given these definitions, we can model the capacity expansion problem using the following mixed-integer programming formulation:

$$\min_{\mathbf{x},\mathbf{y}} \quad \sum_{(s,c)\in\mathcal{E}} r_{s,c} \cdot x_{s,c} \tag{4.3.6a}$$

$$s.t. \quad \sum_{(s,c')\in C} x_{s,c'} \geq q_c + k \cdot y_c^k, \qquad \forall\, c \in \mathcal{C}\,, \forall\, k = 0, \ldots, B\ , \forall\, C \in C_c(k), \tag{4.3.6b}$$

$$(\mathbf{x},\mathbf{y}) \in \mathcal{P}_{\mathbb{Z}}^{\text{ext}} = \mathcal{P}^{\text{ext}} \cap \left(\{0,1\}^{\mathcal{E}} \times \{0,1\}^{\mathcal{C}\times\{0,\ldots,B\}}\right), \tag{4.3.6c}$$

where

$$\mathcal{P}^{\text{ext}} = \Bigg\{(\mathbf{x},\mathbf{y}) \in [0,1]^{\mathcal{E}} \times [0,1]^{\mathcal{C}\times\{0,\ldots,B\}} \colon \sum_{c:(s,c)\in\mathcal{E}} x_{s,c} = 1\ \forall\, s \in \mathcal{S},$$
$$\sum_{s:(s,c)\in\mathcal{E}} x_{s,c} \leq q_c + \sum_{k=0}^{B} k \cdot\ y_c^k \qquad \forall\, c \in \mathcal{C},\ \forall\, k = 0, \ldots, B\ ,$$
$$\sum_{c\in\mathcal{C}} \sum_{k=0}^{B} k \cdot y_c^k \leq B, \quad \sum_{k=0}^{B} y_c^k = 1\ \forall\, c \in \mathcal{C}\Bigg\}.$$

In Formulation (4.3.6), the decision vector $\mathbf{y}$ is simply the pseudopolynomial description (or unary expansion) of $\mathbf{t}$, i.e., $t_c = \sum_{k=0}^{B} k \cdot y_c^k$. Indeed, note that there is a one-to-one correspondence between the elements of $\mathcal{P}_{\mathbb{Z}}^{\text{ext}}$ and $\mathcal{P}_{\mathbb{Z}}$. Hence, the novelty of Formulation (4.3.6) is on the modeling of stability through the generalized comb constraints (4.3.6b). If $B = 0$, we obtain the comb formulation of Baïou and Balinski [**26**]. Otherwise, when $B > 0$, for each school we need to *activate* these constraints only for the capacity expansion assigned to it. For instance, if school $c \in \mathcal{C}$ has capacity $q_c + k$, we must only enforce the constraints for the combs in $C_c(k)$. This motivates the use of the binary vector $\mathbf{y}$. In Theorem 4.3.5, we show the correctness of our new formulation. The proof can be found in Appendix 4.7.3.

Theorem 4.3.5. Formulation (4.3.6), even with the integrality of $\mathbf{x}$ relaxed, is a valid formulation of Formulation (4.3.2).

Let $\mathcal{P}_{x,\mathbb{Z}}^{\text{ext}}$ be $\mathcal{P}_{\mathbb{Z}}^{\text{ext}}$ with the binary requirement for $\mathbf{x}$ relaxed. Motivated by Theorem 4.3.5, we define the formulation $\text{BB-}\textsc{cap}$ as Formulation (4.3.6) with $\mathcal{P}_{\mathbb{Z}}^{\text{ext}}$ replaced by $\mathcal{P}_{x,\mathbb{Z}}^{\text{ext}}$:

$$\min_{\mathbf{x},\mathbf{y}} \quad \sum_{(s,c)\in\mathcal{E}} r_{s,c} \cdot x_{s,c} \tag{4.3.7a}$$

$$s.t. \quad \sum_{(s,c')\in C} x_{s,c'} \geq q_c + k \cdot y_c^k, \qquad \forall\, c \in \mathcal{C}\,, \forall\, k = 0, \ldots, B\ , \forall\, C \in C_c(k), \tag{4.3.7b}$$

$$(\mathbf{x},\mathbf{y}) \in \mathcal{P}_{x,\mathbb{Z}}^{\text{ext}} = \mathcal{P}^{\text{ext}} \cap \left([0,1]^{\mathcal{E}} \times \{0,1\}^{\mathcal{C}\times\{0,\ldots,B\}}\right). \tag{4.3.7c}$$

One limitation of $\mathrm{BB\text{-}CAP}$ is that it is non-compact. For each school $c \in \mathcal{C}$ and $k = 0,\ldots,B$, the family of combs $C_c(k)$ can have exponential size, and there is a pseudopolynomial (in the size of the input) number of these families. Hence, to cope with the size of $\mathrm{BB\text{-}CAP}$, we present a cutting-plane algorithm and its associated (polynomial-time) separation method.[9] Algorithm 2 describes our cutting-plane approach. The idea is to start by solving a mixed-integer problem (in Steps 2 and 3) that only considers a subset of the comb constraints (selected in Step 1), i.e., a relaxation of $\mathrm{BB\text{-}CAP}$. If the solution to this problem is not stable, then our separation algorithm (in Step 4) allows us to find the most violated comb constraints for each school c, namely

$$C^\star \in \operatorname{argmin}\left\{ \sum_{(s,c')\in C} x^\star_{s,c'} : C \in C_c(t^\star_c) \text{ where } t^\star_c = \sum_{k=0}^{B} k \cdot y_c^{k\star} \right\}, \tag{4.3.8}$$

and these constraints are added to the main problem, which is solved again. This process repeats until no additional comb constraint is added to the main problem, guaranteeing that the solution is stable and optimal (due to Theorem 4.3.5). Note that since the set of stability constraints is finite, Algorithm 2 terminates in a finite number of steps. In the next section, we detail our polynomial time algorithm to solve the separation problem (4.3.8).

Algorithm 2 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 $\mathbf{x}^\star$ and the optimal vector of additional seats $\mathbf{t}^\star$.
1: $\mathcal{J} \leftarrow$ subset of comb constraints (4.3.7b)
2: MP $\leftarrow$ build the main program $\min\limits_{(\mathbf{x},\mathbf{y})\in\mathcal{P}^{\text{ext}}_{x,\mathbb{Z}}\cap\mathcal{J}} \sum\limits_{(s,c)\in\mathcal{E}} r_{s,c}\cdot x_{s,c}$
3: $(\mathbf{x}^\star,\mathbf{y}^\star) \leftarrow$ solve MP
4: $\mathcal{J}' \leftarrow$ subset of constraints (4.3.7b) violated by $(\mathbf{x}^\star,\mathbf{y}^\star)$ ▷ Apply Algorithm 3
5: **if** $\mathcal{J}' \neq \emptyset$ **then**
6: MP $\leftarrow$ add to MP the constraints $\mathcal{J}'$ and go to Step 3
7: **return** $\mathbf{x}^\star$ and $\mathbf{t}^\star$ where $t^\star_c \leftarrow \sum_{k=0}^{B} k\cdot y_c^{k^\star}$ for every $c \in \mathcal{C}$

4.3.3.1. **Separation Algorithm.** A separation algorithm is a method that, given a point and a polyhedron, produces a valid inequality that is violated (if any) by that point. This is our goal in Step 4 of Algorithm 2. In fact, given an infeasible $(\mathbf{x}^\star,\mathbf{y}^\star)$ to $\mathrm{BB\text{-}CAP}$, we aim to find the constraint (4.3.7b) that is the most violated by it for each school.

Since capacities may change, we cannot use the cutting plane procedure by Baïou and Balinski [**26**]: This is why we propose Algorithm 2. However, we could use their separation algorithm, since in this step the capacities are fixed (recall Problem (4.3.8)).[10] Nevertheless, with the aim to

[9]Note that in $\mathrm{BB\text{-}CAP}$, the decision vector $\mathbf{y}$ is binary. Therefore, our separation method works for $\mathbf{x}$ fractional and $\mathbf{y}$ binary. Nonetheless, the fact that the separation runs in polynomial-time does not guarantee that the cutting-plane method runs in polynomial time.
[10]The separation by Baïou and Balinski [**26**] 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., $\mathbf{y}^\star$ is binary).

speed up computations, we introduce a novel separation algorithm that relies on new structural results that guarantee to find the most violated comb constraint.

To formalize these results, we introduce additional notation. Given an optimal solution $(\mathbf{x}^\star, \mathbf{y}^\star)$ obtained from Step 3 in Algorithm 2, let $\mathbf{t}^\star$ be the projection of $\mathbf{y}^\star$ in the original problem, i.e., $t_c^\star = \sum_{k=0}^{B} k \cdot y_c^{k\star}$ for all $c \in C$. In addition, let $\mu^\star(\mathbf{t}^\star)$ be the student-optimal stable matching for instance $\Gamma_{\mathbf{t}^\star}$.[11] To simplify the notation, we assume that $\mathbf{y}^\star$ (and thus $\mathbf{t}^\star$) is fixed, and we use $\mu^\star$ to represent $\mu^\star(\mathbf{t}^\star)$. We say that a school $c \in \mathcal{C}$ is *fully-subscribed* in $\mathbf{x}^\star$ if $\sum_{s:(s,c)\in\mathcal{E}} x_{s,c}^\star = q_c + t_c^\star$; otherwise, we say that school c is *under-subscribed*. Moreover, given a school c, let

$$\text{exceeding}(c) = \{s \in \mathcal{S} : x_{s,c}^\star > 0 \text{ and } \mu_{s,c}^\star = 0\}$$

be the set of exceeding students, i.e., the set of students assigned (possibly fractionally) to school c in $\mathbf{x}^\star$ that are not assigned to c in $\mu^\star$, and let

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

be the set of fully-subscribed schools in both $\mathbf{x}^\star$ and $\mu^\star$ that have a non-empty set of exceeding students. Finally, a student-school pair (s,c) is called a fractional blocking pair for $\mathbf{x}^\star$ if the following two conditions hold: (i) there is a school c' such that $x_{s,c'}^\star > 0$ and $c \succ_s c'$, and (ii) c is not fully-subscribed or there is a student s' such that $x_{s',c}^\star > 0$ and $s \succ_c s'$.[12]

In our first structural result, formalized in Lemma 4.3.6, we show that we can restrict the search for the most violated combs to the schools that are fully-subscribed, reducing significantly the number of schools that we need to check at every iteration of the separation algorithm. All the proofs in this subsection can be found in Appendix 4.7.3.2.

Lemma 4.3.6. Let $(\mathbf{x}^\star,\mathbf{y}^\star)$ be the optimal solution obtained at Step 3 in some iteration of Algorithm 2, and suppose that there is a fractional blocking pair (s,c). Then, c is fully-subscribed in $\mathbf{x}^\star$.

In Appendix 4.8.4, we present an example that shows that the set of fully-subscribed schools is not necessarily the same for $\mathbf{x}^\star$ and $\mu^\star$, and, thus, illustrates the potential of using $\mu^\star$ to also guide the search for violated comb constraints.

Lemma 4.3.7. Let $(\mathbf{x}^\star,\mathbf{y}^\star)$ be the optimal solution obtained at Step 3 in some iteration of Algorithm 2 and $\mu^\star$ the student-optimal stable matching in instance $\Gamma_{\mathbf{t}^\star}$, where $\mathbf{t}^\star$ is the allocation

[11] We can obtain this by applying DA on the expanded instance $\Gamma_{\mathbf{t}^\star}$.

[12] Kesten and Ünver [**91**] 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 [**91**] define ex-ante justified envy of student s towards student s' if both $x_{s,c'}^\star, x_{s',c}^\star > 0$ with $c \succ_s c'$ and $s \succ_c s'$. Note that the definition of fractional blocking pair implies that $x_{s,c}^\star < 1$. Moreover, a blocking pair is also a fractional blocking pair.

defined by $\mathbf{y}^\star$. If $\mathcal{J}$ contains all the comb constraints of every fully-subscribed school in $\mu^\star$, then both matchings $\mathbf{x}^\star$ and $\mu^\star$ coincide.

As we mentioned earlier, the family of comb constraints is exponentially large, so we need to further reduce the scope of our search. Our next structural result, formalized in Lemma 4.3.8, accomplishes this by restricting the search of violated combs only among those schools that are in block($\mathbf{x}^\star$).

Lemma 4.3.8. If $\mathbf{x}^\star$ has a fractional blocking pair, then there is at least one student-school pair (s,c), where c is in *block($\mathbf{x}^\star$)*, such that: (i) c prefers s over the least preferred student in exceeding(c), and (ii) the value of the tooth $T(s,c)$ is smaller than 1, i.e., $\sum\limits_{(s,c')\in T(s,c)} x^\star_{s,c'} < 1$.

Algorithm 3 Separation method

Input: An instance $\Gamma = \langle \mathcal{S},\mathcal{C}, \succ, \mathbf{q} + \mathbf{t}^\star \rangle$ and a (fractional) matching $\mathbf{x}^\star$.
Output: A non-empty set $\mathcal{J}'$ of constraints (4.3.7b) violated by $(\mathbf{x}^\star,\mathbf{y}^\star)$, if it exists.

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

In Algorithm 3, we present our separation method. The algorithm begins by initializing the set of violated combs $\mathcal{J}'$ equal to $\emptyset$. Based on Lemmas 4.3.6, 4.3.7 and 4.3.8, we focus only on the schools in block($\mathbf{x}^\star$). Therefore, at Step 2, we iterate to find the most violated (i.e., least valued) comb of every school in block($\mathbf{x}^\star$). 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 $\in$block($\mathbf{x}^\star$), 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 $\underline{s}$.[13] 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 $\mathbf{x}^\star$ (Step 7). If the list $\mathcal{T}$ does not contain a sufficient number of students to build a comb (i.e., $q_c + t^\star_c$), 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 $s^\star$ as the student in $\mathcal{T}$ with the highest value $T^-(s^\star,c)$ At Step 14, we compare the value of $T^-(s',c)$ with the value of $T^-(s^\star,c)$. 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 $s^\star$ (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 $\mathbf{x}^\star$ 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 4.8.5, we exemplify the application of Algorithm 3.

Note that Algorithm 3 resembles the one introduced in [**26**]. However, there are two key differences: (i) we use fractional and integer stable matchings as proxies to reduce the search space to block($\mathbf{x}^\star$), and (ii) we begin the search of the most violated comb at the head of the preference list of the school rather than at the tail (as in [**26**]), which guarantees that our method finds the most violated comb constraint.[14]

Theorem 4.3.9. Algorithm 3 finds the combs solving Formulation (4.3.8) for every school in block($\mathbf{x}^\star$) in $\mathcal{O}(m \cdot n \cdot \bar{q})$ time, where n is the number of students, m is the number of schools and $\bar{q} = \max\limits_{c \in \text{block}(\mathbf{x}^\star)} \{q_c + t^\star_c\}$. Moreover, Algorithm 3 returns a set of combs such that the corresponding constraints (4.3.7b) are violated by $(\mathbf{x}^\star,\mathbf{y}^\star)$, if $\mathbf{x}^\star$ is not stable for $\Gamma_{\mathbf{t}^\star}$.

Remark 4.3.10. Algorithm 3 can be easily adapted to separate the comb constraints with any linear objective function. Indeed, it is sufficient to remove Steps 4 and 5, and iterate over the whole preference list $\succ_c$ at Step 6.

[13] Note that, since school c is fully-subscribed, we know that there are $q_c + t^\star_c$ students $s' \succeq_c \underline{s}$ assigned to c, and thus the comb constraint is always satisfied.

[14] In Appendix 4.8.6 we show that the separation algorithm by [**26**] may not find the most violated constraint.

4.3.4. Properties of the Mechanism

Now that we have devised a framework to solve the problem, we briefly discuss some properties of the mechanism[15] and the underlying optimal solutions. We provide a thorough discussion of these properties in Appendix 4.7.4

Cardinality. In the standard setting with no capacity decisions, we know from the Rural Hospital theorem [**134**] 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 4.7.4.1, we show that if these values are sufficiently small, then there exists an optimal solution of Formulation 4.3.2 whose allocation of capacities yields a student-optimal stable matching of minimum cardinality among all the possible student-optimal stable matchings (Theorem 4.7.3).[16] 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 4.7.4).

Note that this result is not surprising in hindsight, as the objective function in Formulation (4.3.2) is the weighted sum of the students' preference of assignment and the value of unassigned students. Nevertheless, Theorems 4.7.3 and 4.7.4 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 [**131**] and Dubins and Freedman [**56**] 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 4.7.4.2 (see Proposition 4.7.5). Nevertheless, we also show that our mechanism is strategy-proof in the large [**24**], which guarantees that it is approximately

[15]Mechanism in this case stands for the optimal method that solves Problem (4.3.2).

[16]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 4.8.2 in Appendix 4.8).

optimal for students to report their true preferences for any i.i.d. distribution of students' reports. As Azevedo and Budish [**24**] 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.[17]

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. [**31**] and [**27**] show that this property holds for the student-proposing Deferred Acceptance algorithm in the standard setting. However, in Appendix 4.7.4.3, we show that students can be harmed when adding extra seats if their rank improves in a given school. Nevertheless, we can adapt our framework by incorporating additional constraints that would rule out non-monotone allocations if this is a concern for policymakers.

4.4. Evaluation of Methods on Random Instances

In this section, we empirically evaluate the performance of our methods to assess which formulations and heuristics work better. To perform this analysis, we assume that students have complete preference lists and that the sum of schools' capacities equals the number of students. Since the number of variables and constraints increases with $|\mathcal{E}|$, considering complete preference lists increases the dimension of the problem and, thus, makes it harder to solve, providing a "worst-case scenario" in terms of computing time.[18]

Experimental Setup. We consider a fixed number of students $|\mathcal{S}| = 1000$, and we create 100 instances for each combination of the following parameters: $|\mathcal{C}| \in \{5, 10, 15, 20\}$, $B \in \{0, 1, 5, 10, 20, 30\}$. Specifically, for each instance, we generate preference lists and capacities at random, ensuring that the total number of seats is equal to the number of students and that no school has zero capacity.[19] Our methods were coded in Python 3.7.3, with optimization problems solved by Gurobi 9.1.2 restricted to a single CPU thread and one hour time limit. The scripts were run on an Intel(R) Xeon(R) Gold 6226 CPU on 2.70GHz, running Linux 7.9.[20]

Benchmarks. We compare the performance of the following exact methods:[21]

(1) Quad, which corresponds to the quadratic programming model in Formulation (4.3.3).

[17] Note that Dur and Van der Linden [**58**] 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.

[18] We ran experiments considering shorter preference lists and including correlations between students' preferences. The key insights remain unchanged.

[19] Specifically, we generate preferences uniformly at random, and we generate capacities by first allocating one seat to each school, and then we divide the remaining $|S| - |C|$ seats using a multinomial distribution.

[20] The code and the synthetic instances are available upon request.

[21] We also implemented the linearized versions of MaxHrt-cap and MinCut-cap, and the generalization to $B > 0$ of models available in the literature (e.g., MinBinCut [**157**]). We do not report the results of these comparisons because they are dominated by MaxHrt-cap and MinCut-cap.

(2) AGG-LIN, which corresponds to the aggregated linearization in Formulation (4.3.4).
(3) CPM, which corresponds to our cutting-plane method described in Algorithm 2. Appendix 4.11 discusses Step 1 of Algorithm 2.
(4) MAXHRT-CAP, which corresponds to the formulation introduced in [**53**] and adapted to the capacity expansion setting. We provide the detailed description of this model in Formulation (4.10.1) (Appendix 4.10).
(5) MINCUT-CAP, which corresponds to the formulation introduced in [**157**] and adapted to the capacity expansion setting. We provide the detailed description of this model in Formulation (4.10.17) (Appendix 4.10).

For each simulated instance, we also solve the problem with two natural heuristics, called Greedy and LPH, described in Appendix 4.12.[22] To our knowledge, when $B = 0$ (classic School Choice problem), MAXHRT-CAP and MINCUT-CAP are the state-of-the-art mathematical programming formulations for general linear objectives, and thus the most relevant benchmarks for our methods.[23]

Results. In Table 2, we report the results obtained for the exact methods considered.

First, we observe that QUAD and AGG-LIN take significantly more time to solve the problem compared to the other exact methods. Moreover, QUAD and AGG-LIN were not able to find the optimal solution within one hour in 38.04% and 16.58% of the instances, respectively. This result suggests that our aggregated linearization helps towards having a better formulation, but still is not enough to solve the problem effectively. Second, we observe that, in general, execution times increase with the number of schools and decrease with the budget (after reaching a spike). Finally, and more interestingly, we find that our cutting-plane method (CPM) consistently outperforms the other benchmarks when the number of schools increases, while remaining competitive for $|\mathcal{C}| = 5$. These results suggest that our cutting-plane method is the most effective approach to solve larger instances of the problem.

To analyze the performance of our heuristics, in Figures 1 and 2 we report the average optimality gap and run time obtained for each heuristic, including 95% confidence intervals.[24] On the one hand, from Figure 1, we observe that both heuristics find near-optimal solutions, as their optimality gap is always below 3%. Second, we observe that Greedy performs better in terms of optimality gap for low values of B. However, for larger budgets, we observe that LPH outperforms Greedy both in running time and optimality gap. On the other hand, from Figure 2

[22]We emphasize that in Chapter 3, we show that the problem cannot be approximated within a multiplicative factor of $\mathcal{O}\left(n^{\left(\frac{1}{6}-\varepsilon\right)}\right)$ for every $\varepsilon > 0$, unless P=NP; therefore, these heuristics do not achieve meaningful worst-case approximation guarantees.

[23]Note that, when $B = 0$, using the Deferred Acceptance algorithm is the fastest of all methods. However, when $B > 0$, we focus on mathematical programming formulations because the DA algorithm cannot be adapted to find the optimal capacity allocation in polynomial time, otherwise $\mathrm{P} = \mathrm{NP}$.

[24]Optimality gap corresponds to (HEUR − OPT)/OPT, where HEUR is the objective value obtained by the heuristic and OPT is the optimal value obtained with our exact formulation.

Table 2. Comparison of formulations with respect to solving time.

				$\lvert S \rvert = 1000$		
$\lvert C \rvert$	B	Quad	Agg-Lin	MinCut-cap	MaxHrt-cap	Cpm
5	0	14.15 (4.99)[100]	20.1 (5.04)[100]	1.43 (0.48)[100]	0.47 (0.08)[100]	0.74 (0.58)[100]
5	1	21.63 (9.31)[100]	54.86 (43.15)[100]	1.78 (0.53)[100]	1.35 (0.62)[100]	0.74 (0.49)[100]
5	5	12.58 (3.09)[100]	34.08 (13.22)[100]	2.59 (1.1)[100]	2.67 (2.68)[100]	1.55 (1.15)[100]
5	10	13.23 (5.42)[100]	36.74 (18.55)[100]	2.7 (1.34)[100]	3.11 (3.83)[100]	2.6 (2.3)[100]
5	20	10.36 (3.08)[100]	29.83 (12.0)[100]	1.76 (0.93)[100]	1.27 (1.69)[100]	2.02 (2.2)[100]
5	30	8.91 (2.93)[100]	26.66 (13.11)[100]	1.44 (0.89)[100]	0.76 (1.31)[100]	2.09 (2.62)[100]
10	0	815.66 (649.47)[88]	266.99 (165.54)[99]	8.47 (0.97)[100]	2.65 (1.25)[100]	1.11 (0.09)[100]
10	1	742.09 (697.51)[95]	511.71 (258.3)[100]	10.9 (2.64)[100]	27.6 (15.9)[100]	1.89 (0.71)[100]
10	5	377.12 (193.11)[100]	635.38 (381.81)[100]	19.98 (8.04)[100]	64.21 (63.01)[100]	8.14 (7.41)[100]
10	10	344.9 (211.02)[100]	525.25 (358.91)[100]	18.89 (6.56)[100]	128.86 (201.44)[100]	13.17 (10.67)[100]
10	20	440.22 (336.24)[100]	303.36 (248.32)[100]	15.41 (7.3)[100]	63.77 (124.25)[100]	6.88 (7.1)[100]
10	30	636.55 (619.27)[100]	203.8 (155.09)[100]	12.01 (6.68)[100]	31.28 (75.07)[100]	6.33 (7.19)[100]
$\lvert C \rvert$	B	Quad	Agg-Lin	MinCut-cap	MaxHrt-cap	Cpm
15	0	2534.88 (638.83)[21]	729.08 (552.15)[86]	18.34 (1.85)[100]	7.2 (4.8)[100]	1.69 (0.1)[100]
15	1	1858.86 (808.95)[30]	1704.91 (681.47)[93]	43.94 (23.44)[98]	98.07 (70.53)[98]	3.27 (1.17)[97]
15	5	2063.47 (776.7)[82]	1967.68 (864.77)[74]	101.54 (54.77)[100]	785.1 (850.25)[88]	20.58 (18.31)[100]
15	10	2221.03 (812.56)[63]	1610.95 (874.57)[73]	107.96 (70.0)[99]	1076.85 (932.88)[80]	31.77 (32.8)[99]
15	20	2253.21 (858.44)[62]	1126.02 (780.03)[96]	59.82 (36.34)[100]	584.74 (624.36)[98]	20.93 (26.79)[100]
15	30	1853.41 (943.74)[40]	762.15 (691.37)[95]	51.73 (36.76)[100]	395.75 (635.21)[99]	23.79 (32.39)[100]
20	0	- -[0]	2283.18 (917.36)[23]	36.03 (3.73)[100]	11.96 (6.03)[100]	2.45 (0.16)[100]
20	1	- -[0]	2639.4 (640.23)[43]	112.0 (82.99)[100]	185.32 (113.22)[100]	4.61 (1.47)[100]
20	5	- -[0]	2289.12 (860.57)[25]	352.56 (174.11)[100]	1646.98 (983.62)[49]	34.31 (37.58)[100]
20	10	- -[0]	1720.38 (844.15)[42]	324.23 (193.93)[100]	1690.8 (1063.63)[49]	47.29 (50.86)[100]
20	20	2783.57 (371.67)[2]	1316.49 (817.33)[61]	200.46 (162.72)[100]	1219.15 (1020.87)[68]	71.89 (153.08)[100]
20	30	2105.92 (264.53)[4]	1050.01 (712.37)[92]	94.04 (78.16)[100]	727.63 (844.63)[96]	50.31 (186.93)[100]

Note: The time is in seconds, with 1h time limit. We report (in square brackets) the number of instances (out of 100) solved by each method within one hour. The average time (in seconds) and standard deviation (in parenthesis) of the computing times are computed considering these instances.

we observe that the execution time of Greedy is increasing in the budget, while LPH is almost invariant and takes almost no time.

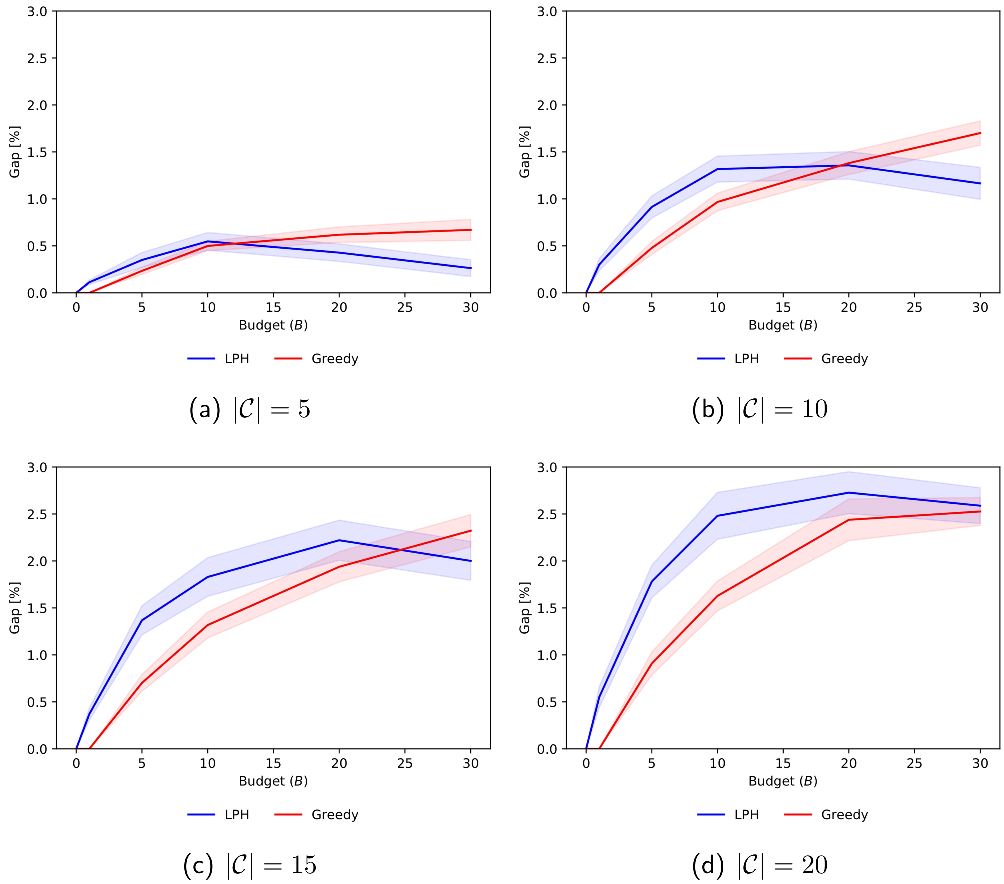


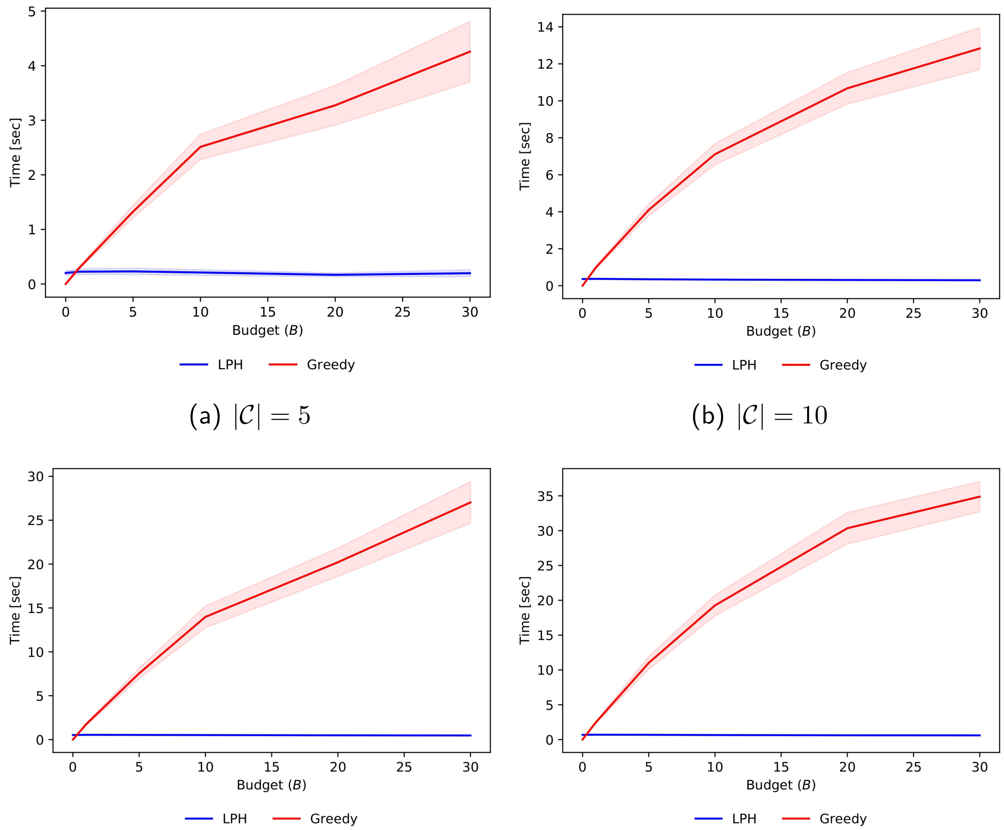
Fig. 1. Optimality Gap for the Heuristics

Overall, our simulation results suggest that CPM is the best approach to solve larger instances of the problem and that LPH is the best heuristic for even larger instances. This heuristic is very relevant in terms of practical use as it is simple to describe, extremely efficient, and capable of achieving good quality solutions.

4.5. Application to School Choice in Chile

To illustrate the potential benefits of capacity expansion, we adapt our framework to the Chilean school choice system. This system, introduced in 2016 in the southernmost region of the country (Magallanes), was fully implemented in 2020 and serves close to half a million students and more than eight thousand schools each year.

The Chilean school choice system is a good application for our methodology for multiple reasons. First, the system uses a variant of the student-proposing Deferred Acceptance algorithm, which incorporates priorities and overlapping quotas. Our framework can include all the features of the Chilean system, including the block application, the dynamic siblings' priority, etc. We refer to Correa et al. [**50**] for a detailed description of the Chilean school choice system and



(a) $|\mathcal{C}| = 5$ (b) $|\mathcal{C}| = 10$ (c) $|\mathcal{C}| = 15$ (d) $|\mathcal{C}| = 20$

Fig. 2. Running Time for the Heuristics

the algorithm used to perform the allocation. Second, the Ministry of Education manages all schools that participate in the system and thus can ask them to modify their vacancies within a reasonable range. Finally, the system is currently being redesigned, and we are collaborating with the authorities to include some of the ideas introduced in our work.

4.5.1. Data and Simulation Setting

We consider data from the admission process in 2018.[25] Specifically, we focus on the southernmost region of the country as it is the region where all policy changes are first evaluated. Moreover, we restrict the analysis to Pre-K for two reasons: (i) it is the level with the highest number of applicants, as it is the first entry level in the system, and (ii) to speed up the computation. In Table 3, we report summary statistics about the instance, and we compare it with the values nationwide for the same year.[26]

[25] All the data is publicly available and can be downloaded from this website.

[26] In our simulations, we consider a total of 1395 students and 49 schools. The difference in the number of students is due to students that are not from the Magallanes region but only apply to schools in that region. The difference in the number of schools is due to some schools offering morning and afternoon seats, whose

Table 3. Details of the instance of the Chilean dataset.

Region	Students	Schools	Applications
Magallanes Pre-K	1389	43	4483
Overall Pre-K	84626	3465	256120
Overall (all levels)	274990	6421	874565

We perform our simulations varying the budget $B \in \{0,1,\ldots,30\}$ and the penalty for unassigned students $r_{s,\emptyset}$. For the latter, we consider two cases: (i) $r_{s,\emptyset} = |\mathcal{C}| + 1$ for all $s \in \mathcal{S}$, and (ii) $r_{s,\emptyset} = |\succ_s| + 1$ for all $s \in \mathcal{S}$. Notice that the two values for $r_{s,\emptyset}$ cover two extreme cases. When $r_{s,\emptyset} = |\mathcal{C}| + 1$ (or any large number), the model will use the extra vacancies to ensure that a student that was previously unassigned gets assigned. In contrast, when $r_{s,\emptyset} = |\succ_s| + 1$, the model will (most likely) assign the extra seat to the school that leads to the largest chain of improvements. Hence, from a practical standpoint, which penalty to use is a policy-relevant decision that must balance *access* and *improvement*.

4.5.2. Results

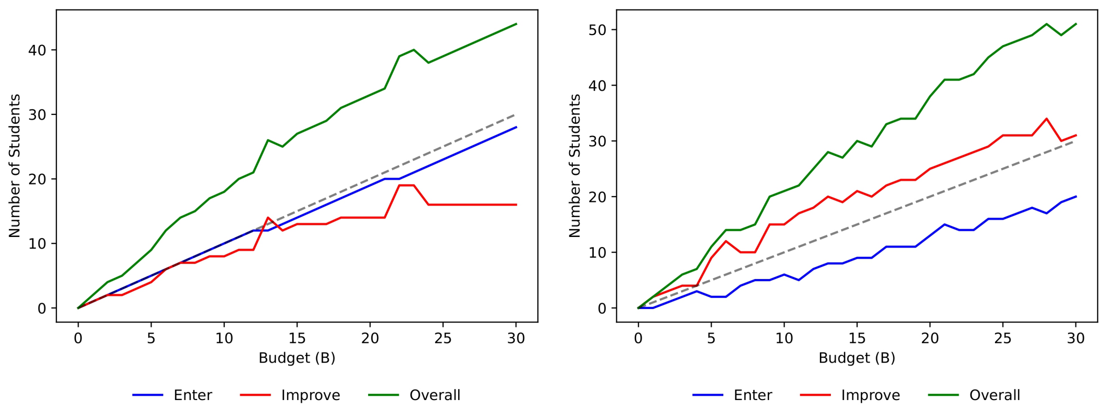
We report our main simulation results in Figure 3. We produced the plots by solving Agg-Lin, given its simplicity and running times below 1 hour for each instance. For each budget, we plot the number of students who (1) *enter* the system, i.e., who are not initially assigned (with $B = 0$), but are assigned to one of their preferences when capacities are expanded; (2) *improve*, i.e., students who are initially assigned to some preference but improve their preference of assignment when capacities are expanded; and (3) *overall*, which is the total number of students who benefit relative to the baseline and is equal to the sum of the number of students who *enter* and *improve*.[27]

First, we confirm that all initially assigned students (with $B = 0$) get a school at least as preferred when we expand capacities. Second, increasing capacities with a high penalty primarily benefits initially unassigned students. In contrast, students who improve their assignments are the ones who most benefit when the penalty is low. Third, we observe that the total number of students who benefit (in green) is considerably larger than the number of additional seats (dashed). The reason is that an extra seat can lead to a chain of improvements that ends either on a student that *enters* the system or in a school that is under-demanded. Finally, we observe that the total number of students who gain from the additional seats is not strictly increasing in the budget. Indeed, the number may decrease if the extra seat allows a student to dramatically enhance their assignment (e.g., moving from being unassigned to being assigned

admissions are separate. Hence, 43 is the total number of unique schools, while 49 is the number of "schools" with independent admissions processes.

[27]For all simulations, we consider a MipGap tolerance of 0.0% and we solve them using the Agg-Lin formulation. By construction, the results are the same if we use any of the other exact methods discussed in Section 4.4.

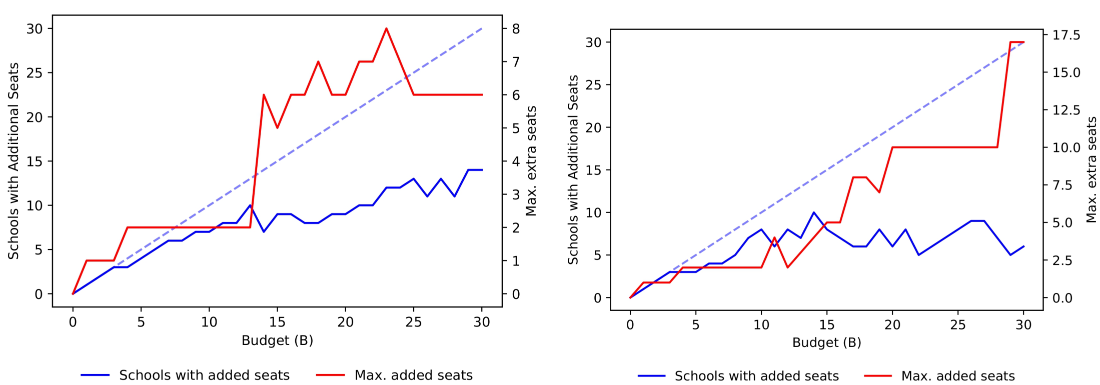
to their top preference). This effect on the objective could be larger than that of a chain of minor improvements involving several students, and thus the number of students who benefit may decrease.



(a) $r_{s,\emptyset} = |\mathcal{C}| + 1$ (b) $r_{s,\emptyset} = |\succ_s| + 1$

Fig. 3. Effect for Students

In Figure 4, we analyze the impact of expanding capacities on the number of schools with increased capacity and the maximum number of additional seats per school. We observe that the number of schools with extra seats remains relatively stable as we increase the budget. In addition, we observe that the maximum number of additional seats in a given school increases with the budget. This is because students' preferences are highly correlated (i.e., students have similar preferences) and, thus, a few over-demanded schools concentrate the extra seats added to the system. Finally, the latter effect is more prominent when the penalty is lower.



(a) $r_{s,\emptyset} = |\mathcal{C}| + 1$ (b) $r_{s,\emptyset} = |\succ_s| + 1$

Fig. 4. Effect for Schools

4.5.2.1. **Practical Implementation.** A valid concern from policymakers is that our approach would assign most extra seats to a few over-demanded schools if preferences are highly correlated and the penalty of having unassigned students is small. As a result, our solution would not be feasible in practice. To rule out this concern, we adapt our model and include the new set of constraints

$$t_c \leq b_c, \quad \forall\, c \in \mathcal{C},$$

where b_c is the maximum number of additional seats that can be allocated to school c.

In Figure 5, we compare the gap between the optimal (unconstrained) solution and the values obtained when considering $b_c \in \{2,5,10\}$ for all $c \in \mathcal{C}$ and $r_{s,\emptyset} = |\succ_s| + 1$ for all $s \in \mathcal{S}$. First, we observe that the gap increases for $b_c \in \{2,5\}$ as we increase the budget, while it does not change for $b_c = 10$. This result suggests that the problem has many optimal solutions and, thus, we can select one that does not over-expand some schools. Second, we observe that the overall gap is relatively low (max of 0.8%), which suggests that we can include the practical limitations for schools without major losses of performance. In Appendix 4.13, we discuss some model extensions to incorporate other relevant aspects from a practical standpoint.

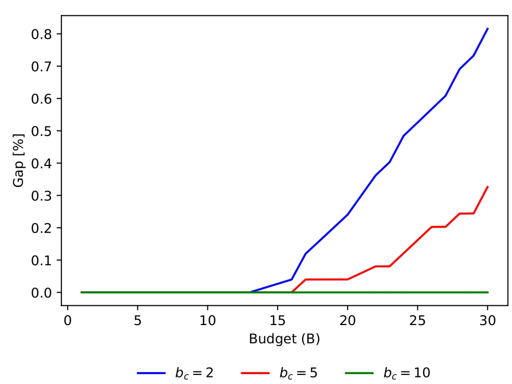


Fig. 5. Effect of Bounds on Expansion

4.5.2.2. **Heuristics.** For each value of the budget, in Figure 6, we report the gap obtained relative to the optimal policy.[28] Consistent with the results in Section 4.4, Greedy performs better than LPH for low values of the budget, but this reverses as the budget increases. Hence, we conclude that LPH can be an effective approach for large instances and large values of B. This could be particularly relevant when applying our framework to other more populated regions, such as the Metropolitan region, where close to 250,000 students apply each year.

[28]We consider the problem with penalty $r_{s,\emptyset} = |\succ_s| + 1$. The results are similar if we consider $r_{s,\emptyset} = |\mathcal{C}| + 1$

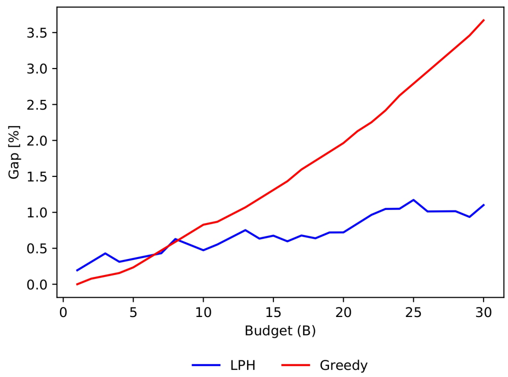


Fig. 6. Heuristics: Optimality Gap

4.5.3. 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.

4.6. Conclusions

We study how centralized clearinghouses can jointly decide the allocation of additional seats and find a stable matching. To accomplish this, we introduce the stable matching problem under capacity planning and devise integer programming formulations for it. We show that all natural formulations involve quadratic constraints and provide linearizations for them. Then, we develop a non-compact mixed-integer linear program (BB-CAP) and prove that it correctly models our problem. Building on this key result, we introduce a cutting-plane algorithm to solve BB-CAP. At the core of our cutting-plane algorithm is a new separation method, which finds the most violated comb constraint for each school in polynomial time and, when the objective is the student-optimal stable matching, prunes the set of schools for which violated comb constraints may exist. Finally, we show and discuss several properties of our mechanism, including how the cardinality of the allocation varies with the penalty (and how this can be used to prioritize different goals), some incentive properties and also the mechanism's monotonicity.

Through an extensive numerical study, we find that our cutting-plane algorithm significantly outperforms all the state-of-the-art formulations in the literature. In addition, we find that one of the two heuristics that we propose (LPH) consistently finds near-optimal solutions in few seconds. These results suggest that our cutting-plane method and the LPH heuristic can be practical approaches depending on the size of the problem. Moreover, we adapted our framework to solve an instance of the Chilean school choice problem. Our results show that each additional seat can benefit multiple students. However, depending on how we penalize having unassigned students in the objective, the set of students who benefit from the extra seats changes. Indeed, we have theoretically shown that if we consider a large penalty, the optimal solution prioritizes *access*, i.e., assigning students that were previously unassigned. In contrast, if that penalty is low, we proved that the optimal solution prioritizes *improvement*, i.e., benefiting students' preference of assignment as much as possible. Hence, which penalty to consider is a policy-relevant decision that depends on the objective of the clearinghouse.

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 4.13 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.

Appendix

4.7. Missing Proofs and Other Results

4.7.1. Proof of Lemma 4.3.1.

We begin by observing that Baïou and Balinski [**26**] show that the feasible region of MAX-HRT corresponds to the set of stable matchings. Therefore, in the following proof, we only need to focus on proving the equivalence between the student-optimal matching and a matching minimizing the sum of the students' rank over the set of stable matchings.

If: In a student-optimal stable matching, each student is assigned to the best school they could achieve in any stable matching [**67**]. Thus, each unassigned student is unassigned in every stable matching. Moreover, by the Rural Hospital Theorem [**131, 132, 68, 134**], the same students are assigned in all stable matchings. Suppose that μ is the student-optimal stable matching but its corresponding binary encoding is not optimal to MAX-HRT. Let x' be an optimal solution MAX-HRT and let $\mu' = \{(s,c) \in \mathcal{E} : x'_{s,c} = 1\}$ be the associated matching. 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 MAX-HRT. 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 MAX-HRT 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 [**67**], 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.

4.7.2. Proof of Theorem 4.3.4

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.

4.7.3. Missing Proofs in Section 4.3.3

4.7.3.1. Proof of Theorem 4.3.5. Before proving Theorem 4.3.5, we show a couple of lemmata. 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 4.7.1. Let $\bar{\mathbf{x}} \in \{0,1\}^{\mathcal{E}}$ be a stable matching for some vector of extra seats $\bar{\mathbf{y}} \in \{0,1\}^{\mathcal{C}\times\{0,\ldots,B\}}$. If $\bar{y}_c^k = 1$ then all constraints (4.3.6b) for the combs in $C_c(k+\alpha)$ with $\alpha > 0$ are satisfied.

Proof. When $\bar{y}_c^k = 1$, the right-hand-side of constraints (4.3.6b) for combs in $C_c(k+\alpha)$ is q_c. Thus, we need to show that the left-hand-side of these constraints is guaranteed to be greater than or equal to q_c. By the definition of comb, for all $C \in C_c(k+\alpha)$ there is $\hat{C} \in C_c(k)$ such that $\hat{C} \subseteq C$. Hence,

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

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

□

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

Lemma 4.7.2. Let $\bar{\mathbf{x}} \in \{0,1\}^{\mathcal{E}}$ be a stable matching for some vector of extra seats $\bar{\mathbf{y}} \in \{0,1\}^{\mathcal{C}\times\{0,\dots,B\}}$. If $\bar{y}_c^k = 1$, then all constraints (4.3.6b) for combs in $C_c(k-\alpha)$ with $\alpha < k$ are satisfied.

Proof. When $\bar{y}_c^k = 1$, the right-hand-side of constraint (4.3.6b) for each comb $C \in C_c(k-\alpha)$ is q_c. Thus, we need to show that the left-hand-side of this constraint is guaranteed to be greater than or equal to q_c. Let (s,c) be the base of the comb C.

First, we show the result when there is $\hat{C} \in C_c(k)$ such that $C \subseteq \hat{C}$. By the definition of generalized comb (see Subsection 4.3.3), C contains the shaft $S^{k-\alpha}(s,c)$. Consequently, since $\hat{C}$ contains C, if (s',c) is the base of $\hat{C}$, then $S^{k-\alpha}(s,c) \subseteq S^k(s',c)$.

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

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

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

- If there is $r \in T$ 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:
$$\sum_{(i,c)\in S^{k-\alpha}(s,c)} \bar{x}_{i,c} = q_c + k.$$
Since $S^{k-\alpha}(s,c) \subseteq C$, 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 $r \in T$, then
$$\sum_{r\in T} \sum_{(s_r,j)\in T_{s_r}} \bar{x}_{s_r,j} = |T| = q_c + k - \alpha \geq q_c.$$
Since $\bigcup_{r\in T} T_{s_r} \subseteq C$ the results holds.

Second, we prove the lemma when no comb of $C_c(k)$ contains C. Then, such situation can only occur if it is not possible to find a shaft $S^k(s',c)$ 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 $C_c(k)$ is empty and there is nothing to prove.

□

We now prove our main result in this section.

Proof. Proof of Theorem 4.3.5. The preceding lemmas lead us to conclude the statement when the integrality of $\mathbf{x}$ is imposed: for each fixed $\mathbf{y}$, only the comb constraints (4.3.6b) associated to the overall capacity of each school are enforced; the remaining are redundant. When the integrality of $\mathbf{x}$ is relaxed, the statement follows directly from Baïou and Balinski [**26**]: when $\mathbf{y}$ is fixed, the comb inequalities for the associated capacities provide the convex hull of stable matchings for $\Gamma_{\mathbf{t}}$ where $t_c = \sum_{k=0}^{B} y_c^k$ for all $c \in \mathcal{C}$. Thus, as long as the number of seats for each school is integer, constraints (4.3.6b) contain the comb constraints from Baïou and Balinski [**26**] for $\Gamma_{\mathbf{t}}$.

□

4.7.3.2. Missing Proofs in Section 4.3.3.1.

Proof. Proof of Lemma 4.3.6. We prove that c is fully-subscribed in $\mathbf{x}^\star$ by contradiction. Let (s,c) be a fractional blocking pair for $\mathbf{x}^\star$, where c is an under-subscribed school in $\mathbf{x}^\star$. Let s be such that $x^\star_{s,c'} > 0$ and $c \succ_s c'$. So s prefers c over c'. Moreover, c has some available capacity which would allow to increase $x^\star_{s,c}$ by decreasing $x^\star_{s,c'}$. Clearly, such modification would decrease the objective value of the main program. This contradicts the optimality of $\mathbf{x}^\star$. □

Proof. Proof of Lemma 4.3.7. Let $\mathcal{C}'$ be the set of fully-subscribed schools in $\mu^\star$ and $\mathcal{S}'$ be the set of students matched to the schools $\mathcal{C}'$ in $\mu^\star$. We denote by $\mathcal{C}''$ and $\mathcal{S}''$ the complementary sets of $\mathcal{C}'$ and $\mathcal{S}'$ respectively. Let us analyze the matches in $\mathcal{S}'$. First, note that for every student $s \in \mathcal{S}'$ we have $\sum_{c \succeq \mu^\star(s)} x^\star_{s,c} = 1$, i.e., student s must be matched in $\mathbf{x}^\star$ to some school that she prefers at least as much as $\mu^\star(s) \in C'$. This is because $\mathcal{J}$ contains all the comb constraints of the schools in $\mathcal{C}'$. Second, every student $s \in \mathcal{S}'$ must be matched in $\mathbf{x}^\star$ to a school that is in $\mathcal{C}'$, i.e., $\sum_{c \in \mathcal{C}'} x^\star_{s,c} = 1$. This is because, otherwise, s would create a blocking pair in $\mu^\star$ with a under-subscribed school in $\mathcal{C}''$ due to our first argument. As a consequence of the second point, we cannot have $x^\star_{s,c} > 0$ 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 $\mathbf{x}^\star$ is given by the comb constraints. Because of optimality, $\mu^\star$ and $\mathbf{x}^\star$ 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 $\mathbf{x}^\star$ in the sub-instance defined by $\mathcal{S}''$ and $\mathcal{C}''$ is given by the integrality of the matching polytope without stability constraints. Now, we will show that the assignments coincide in both $\mu^\star$ and $\mathbf{x}^\star$. By contradiction, assume that the students in $\mathcal{S}''$ are matched differently in $\mathbf{x}^\star$ and in $\mu^\star$. First, if there is a student $s \in \mathcal{S}''$ that prefers a school c over $\mu^\star(s)$, then (s,c) would be a blocking pair in $\mu^\star$ because c is under-subscribed in $\mu^\star$, which is not possible. Therefore, every student $s \in \mathcal{S}''$ weakly prefers $\mu^\star(s)$ over their assignment in $\mathbf{x}^\star$, and, by the hypothesis in our contradiction argument, there is at least one student s that strictly prefers $\mu^\star(s)$ over their match in $\mathbf{x}^\star$ (i.e., the two schools are different). Given that $\mathcal{S}''$ and $\mathcal{C}''$ form a separate sub-instance, we can rearrange the matching $\mathbf{x}^\star$ of every student in $\mathcal{S}''$ by matching the student to her matching in $\mu^\star$. This new matching would have a lower objective than the one of $\mathbf{x}^\star$, which would violate its optimality. Therefore, both $\mu^\star$ and $\mathbf{x}^\star$ also coincide in $\mathcal{S}''$ and $\mathcal{C}''$.

□

PROOF. Proof of Lemma 4.3.8. Suppose that the statement is false. Then, for every school c in block($\mathbf{x}^\star$), and for every student s more preferred by c to the least preferred student in exceeding(c), s has the tooth evaluated to 1. By Lemma 4.3.6, we know that the fractional blocking pairs must all involve a school that is fully-subscribed in $\mathbf{x}^\star$. Among these schools, it is sufficient to focus our attention on the schools $c \in C$ such that there exists $s \in S$ with $x^\star_{s,c} > 0$ and student s is not matched to c in $\mu^\star$. Therefore, for these schools the set of exceeding students is non-empty.

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

□

PROOF. Proof of Theorem 4.3.9.

We start this proof by showing that for each school $c \in$ block($\mathbf{x}^\star$), Steps 3-19 of Algorithm 3 provide the comb solving Formulation (4.3.8) in $\mathcal{O}(n \cdot \bar{q})$.

First, let us observe that the search for the least valued comb in $C_c(t^\star_c)$ terminates in a finite number of steps because the preference list $\succ_c$ is finite. At the beginning of every cycle in Step 6, we select a student s'. By the end of the $(q_c + t^\star_c)$-th iteration, the comb C is the smallest valued comb in $C_c(t^\star_c)$ with a base preferred at least as s' (i.e., the student at the base of the comb is ranked equal or less than s'). If we prove this statement, we prove that the algorithm solves Formulation (4.3.8) for c.

Base: At the $(q_c + t^\star_c)$-th iteration, we create comb C at Step 11. The base of comb C is the student s' that is ranked $q_c + t^\star_c$ by c, and the bases of the other $q_c + t^\star_c - 1$ 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 $l \geq q_c + t^\star_c$, 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 $l+1$, the algorithm selects the comb with lowest value among all combs with a base student ranked less or equal than $(l+1)$. Let s' be the student ranked $l+1$ and let $s^\star$ 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 $T^{-}(s',c) \geq T^{-}(s^\star,c)$, then adding s' in $\mathcal{T}$ would produce a comb with equal or higher value. Otherwise, $T^{-}(s',c) < T^{-}(s^\star,c)$, and adding $'s$ in $\mathcal{T}$ may provide a better comb. If $T^{-}(s',c) < T^{-}(s^\star,c)$, then we create a new comb C' with base in (s',c) and teeth in $\mathcal{T} \setminus \{s^\star\}$. Any other choice of teeth is sub-optimal for the base (s',c) since they are the least valued $q_c + t_c^\star$ teeth among those in the first $l+1$ 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 $l+1$.

If s' is the least preferred exceeding student, then all the combs with base s' are valued at least $q_c + t_c^\star$ since c is a fully subscribed school. Hence, any comb that may be selected with such a base would not be violated, and we can terminate our search.

To establish the correctness of Algorithm 3, it remains to show that it determines the violated combs (if any) that eliminate the input solution. First, the algorithm terminates in a finite number of steps because $|\text{block}(\mathbf{x}^\star)| \leq |\mathcal{C}| = m$ and $| \succ_c | \leq |\mathcal{S}| = n$. Second, by Lemma 4.3.8 we only need to focus our attention on the fully-subscribed schools in block($\mathbf{x}^\star$). Thus, Algorithm 3 returns the most violated comb constraint for those schools.

Running-time analysis. The procedure for finding $\underline{s}$ takes at most n operations. 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 $\mathcal{T}$ is 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 $|\mathcal{T}| = q_c + t_c^\star$. Hence, the number of elementary operations for finding the most violated comb of school c is at most $\mathcal{O}(n \cdot (q_c + t_c^\star))$. To conclude, we observe that, in Algorithm 3, the computation of block($\mathbf{x}^\star$) at Step 2 can take at most $\mathcal{O}(m \cdot n)$ time, i.e., the running time of the Deferred Acceptance algorithm. Moreover, as proved above, finding the least valued comb of a school in block($\mathbf{x}^\star$) can take at most $\mathcal{O}(n \cdot (q_c + t_c^\star))$ time. Hence, Algorithm 3 can take at most $\mathcal{O}(m \cdot n \cdot \bar{q})$ time, where $\bar{q} = \max\limits_{c \in \text{block}(\mathbf{x}^\star)} \{q_c + t_c^\star\}$.

□

4.7.4. Properties of the Mechanism

4.7.4.1. Cardinality. A crucial aspect of the solutions of Formulation 4.3.2 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 larger 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 4.7.3 and 4.7.4 we formalize this intuition.

Theorem 4.7.3. There is a sufficiently small and finite $r_{s,\emptyset}$ for all $s \in \mathcal{S}$ such that the optimal solution of Formulation (4.3.2) 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:

$$\begin{aligned}
\Delta = \sum_{(s,c)\in\mu^*} r_{s,c} - \sum_{(s,c)\in\mu'} r_{s,c} = & \sum_{s\in(M(\mu^*)\cap M(\mu'))} r_{s,\mu^*(s)} - r_{s,\mu'(s)} \\
& + \sum_{s\in M(\mu^*)\setminus M(\mu')} r_{s,\mu^*(s)} - r_{s,\emptyset} + \sum_{s\in M(\mu')\setminus M(\mu^*)} r_{s,\emptyset} - r_{s,\mu'(s)} \\
& + \sum_{s\in S\setminus(M(\mu^*)\cup M(\mu'))} r_{s,\emptyset} - r_{s,\emptyset} \\
& = \sum_{s\in(M(\mu^*)\cap M(\mu'))} r_{s,\mu^*(s)} - r_{s,\mu'(s)} \\
& + \sum_{s\in M(\mu^*)\setminus M(\mu')} r_{s,\mu^*(s)} - r_{s,\emptyset} + \sum_{s\in M(\mu')\setminus M(\mu^*)} r_{s,\emptyset} - r_{s,\mu'(s)} \\
& \geq \sum_{s\in(M(\mu^*)\cap M(\mu'))} (1 - |\succ|_s) \\
& + \sum_{s\in M(\mu^*)\setminus M(\mu')} 1 - r_{s,\emptyset} + \sum_{s\in M(\mu')\setminus M(\mu^*)} r_{s,\emptyset} - |\succ_s| \\
& \geq |(M(\mu^*)\cap M(\mu'))| \cdot (1-\xi) \\
& + |M(\mu^*)\setminus M(\mu')| \cdot (1 - r_\emptyset) + |M(\mu')\setminus M(\mu^*)| \cdot (r_\emptyset - \xi) \\
& = |(M(\mu^*)\cap M(\mu'))| \cdot (1-\xi) + (|M(\mu')\setminus M(\mu^*)| + 1) \cdot (1 - r_\emptyset) \\
& + |M(\mu')\setminus M(\mu^*)| \cdot (r_\emptyset - \xi) \\
& \geq |(M(\mu^*)\cap M(\mu'))| \cdot (1-\xi) + (|M(\mu')\setminus M(\mu^*)|) \cdot (1-\xi) - r_\emptyset \\
& = (|M(\mu^*)\cap M(\mu')| + |M(\mu')\setminus M(\mu^*)|) \cdot (1-\xi) - r_\emptyset \\
& = (|M(\mu^*)\cap M(\mu')| + |M(\mu')\setminus M(\mu^*)|) \cdot (1-\xi) - |\mathcal{S}| \cdot (1-\xi) \\
& = (|M(\mu^*)\cap M(\mu')| + |M(\mu')\setminus M(\mu^*)| - |\mathcal{S}|) \cdot (1-\xi) \\
& \geq 0,
\end{aligned} \tag{4.7.1}$$

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. □

Note that the penalty used in the proof is negative. A negative penalty can be interpreted as the policymaker 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 4.8.2 in Appendix 4.8, this is not the case.

On the other hand, in Theorem 4.7.4, we show that if the penalty values are sufficiently large, then the optimal solution will prioritize *access* by obtaining a student-optimal stable matching of maximum cardinality.

Theorem 4.7.4. There is a sufficiently large and finite $r_{s,\emptyset}$ for all $s \in \mathcal{S}$ such that the optimal solution of Formulation (4.3.2) 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

$$\begin{aligned}
\sum_{s\in\mathcal{S}} r_{s,\mu^*(s)} - \sum_{s\in\mathcal{S}} r_{s,\mu'(s)} &= \sum_{s\in\mathcal{S}:\mu^*(s)\in\mathcal{C}} r_{s,\mu^*(s)} - \sum_{s\in\mathcal{S}:\mu'(s)\in\mathcal{C}} r_{s,\mu'(s)} + \sum_{s\in\mathcal{S}:\mu^*(s)=\emptyset} r_{s,\emptyset} - \sum_{s\in\mathcal{S}:\mu'(s)=\emptyset} r_{s,\emptyset} \\
&= \sum_{s\in\mathcal{S}:\mu^*(s)\in\mathcal{C}} r_{s,\mu^*(s)} - \sum_{s\in\mathcal{S}:\mu'(s)\in\mathcal{C}} r_{s,\mu'(s)} \\
&\quad + \bar{r} \cdot \left[|s \in \mathcal{S} \, : \, \mu^*(s) = \emptyset| - |s \in \mathcal{S} \, : \, \mu'(s) = \emptyset| \right] \\
&> - \sum_{s\in\mathcal{S}} |\succ_s| + \bar{r} \cdot \left[|s \in \mathcal{S} \, : \, \mu^*(s) = \emptyset| - |s \in \mathcal{S} \, : \, \mu'(s) = \emptyset| \right] \\
&> 0.
\end{aligned} \tag{4.7.2}$$

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 large. 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 $(\mathbf{x}^*, \mathbf{t}^*)$. $\square$

4.7.4.2. Incentives. It is direct from [**144**] that our mechanism is not strategy-proof for schools. In Proposition 4.7.5, 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 4.7.5. 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:

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

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. □

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

Proposition 4.7.6. The mechanism is strategy-proof in the large.

PROOF. In an extension of their Theorem 1, [**24**] 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.

Semi-anonimity. As defined in [**24**], 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 $l_s, l_{s'} \in [0,1]$.[29] Note that G is finite because the number of priority groups is finite in most applications.[30] 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 $\Phi^n : T^n \to \Delta(\mathcal{C} \cup \{\emptyset\})^n$ 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 $\tilde{c} \in \Delta(C \cup \{\emptyset\})$ (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 $x^n : (T \times [0,1])^n \to \Delta(\mathcal{C} \cup \{\emptyset\})^n$ such that

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

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

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

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 that 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 $u_{t_s}\left[x_s^n(\mathbf{t},\mathbf{l})\right] \geq u_{t_s}\left[x_{s'}^n(\mathbf{t},\mathbf{l})\right]$, 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. □

4.7.4.3. 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 $k \in \{5, \ldots, n\}$, 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 $s_4 \succ_{c_k} \ldots \succ_{c_k} s_n \succ_{c_k} \emptyset$ for all $k \geq 4$. In addition, we assume that the preferences and priorities of the other agents are:

$$s_1 : c_1 \succ c_3 \succ \emptyset \qquad\qquad c_1 : \; s_1 \succ s_3 \succ \emptyset$$

[29] In 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.

[30] In 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.

$$s_2 : c_2 \succ c_3 \succ \emptyset \qquad\qquad c_2 :\ s_2 \succ s_3 \succ \emptyset$$

$$s_3 : c_1 \succ c_2 \succ \emptyset \qquad\qquad c_3 :\ s_1 \succ s_2 \succ \emptyset$$

$$s_4 : c_4 \succ \emptyset$$

It is easy to see that (when $B = 0$) $\mu^0 = \{(s_1, c_1), (s_2, c_2), (s_3, \emptyset), (s_4, c_4), \ldots, (s_n, c_n)\}$. 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 $\mu^1 = \{(s_1, c_1), (s_2, c_2), (s_3, c_1), (s_4, c_4), \ldots, (s_n, c_n)\}$. 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 $\mu^{00} = \{(s_1, c_1), (s_2, c_3), (s_3, c_2), (s_4, c_4), \ldots, (s_n, c_n)\}$. 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 $\mu^{11} = \{(s_1, c_1), (s_2, c_3), (s_3, c_2), (s_4, c_4), (s_5, c_4), \ldots, (s_n, c_{n-1})\}$. 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.

4.7.5. Complexity

In Chapter 3, we analyze the complexity of Problem (4.3.2) and we 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. Note that for a fixed $\mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}}$, the value

$$f(\mathbf{t}) := \min_{\mu} \left\{ \sum_{(s,c)\in\mu} r_{s,c} :\ \mu \text{ is a stable matching in instance } \Gamma_{\mathbf{t}} \right\} \tag{4.7.4}$$

can be computed 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., [**142**]). As we show in the following result, f is neither lattice submodular nor supermodular.

Proposition 4.7.7. The function $f(\mathbf{t})$ defined in Expression (4.7.4) is neither lattice submodular nor supermodular.

Proof. Let us recall the definition of lattice submodularity. A function $f : \mathbb{Z}_+^{\mathcal{C}} \to \mathbb{R}_+$ 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 $\mathbf{t}$,$\mathbf{t}' \in \mathbb{Z}_+^{\mathcal{C}}$, 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. Now, let us recall Expression (4.7.4)

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

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

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

$\square$

4.8. Missing Examples

4.8.1. 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:

$$
\begin{aligned}
&c\ \ : s_1 \succ s_2 \succ s_3 \succ s_4,\ \forall c \in \mathcal{C}\\
&s_1 : c_1 \succ c_2 \succ c_3\\
&s_2 : c_2 \succ c_1 \succ c_3\\
&s_3 : c_1 \succ c_3 \succ c_2\\
&s_4 : c_2 \succ c_3 \succ c_1.
\end{aligned}
$$

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.

4.8.2. Minimum cardinality matching

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 $k \in \{4, \ldots, n\}$, we assume that their preferences are given by $c_k \succ_{s_k} \emptyset \succ_{s_k} \ldots$, and we assume that the priorities at school c_k are $s_k \succ_{c_k} \emptyset \succ_{c_k} \ldots$, where the $\succ \ldots$ represent an arbitrary ordering of the missing agents after $\emptyset$. In addition, we assume that the preferences and priorities of the other agents are:

$$
\begin{aligned}
&s_1 : c_1 \succ \emptyset \succ \ldots && c_1 :\ s_1 \succ s_2' \succ s_2 \succ \emptyset \succ \ldots\\
&s_2 : c_1 \succ c_3 \succ c_2 \succ \emptyset \succ \ldots && c_2 :\ s_2 \succ s_2' \succ s_3' \succ \emptyset \succ \ldots\\
&s_3 : c_3 \succ c_2 \succ \emptyset \succ \ldots && c_3 :\ s_3 \succ s_3' \succ s_2 \succ \emptyset \succ \ldots\\
&s_2' : c_n \succ \ldots \succ c_4 \succ c_1 \succ \emptyset \succ \ldots &&\\
&s_3' : c_3 \succ \emptyset \succ \ldots. &&
\end{aligned}
$$

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.

4.8.3. Inverse Inclusion in Theorem 4.3.4

The following example shows that the inverse inclusion of the statement in Theorem 4.3.4 does not necessarily hold. Let us consider the set of students $\mathcal{S} = \{s_1,s_2,s_3,s_4,s_5,s_6\}$ and the set of schools $\mathcal{C} = \{c_1, c_2, c_3, c_4\}$. The rankings of the students are as follows:

$$
\begin{array}{ll}
s_1 : c_3 \succ c_4 \succ c_1 \succ c_2 & c_1 : s_1 \succ s_3 \succ s_2 \succ s_5 \succ s_6 \succ s_4 \\
s_2 : c_2 \succ c_1 \succ c_4 \succ c_3 & c_2 : s_4 \succ s_1 \succ s_6 \succ s_5 \succ s_2 \succ s_3 \\
s_3 : c_2 \succ c_1 \succ c_4 \succ c_3 & c_3 : s_2 \succ s_1 \succ s_5 \succ s_6 \succ s_3 \succ s_4 \\
s_4 : c_1 \succ c_3 \succ c_2 \succ c_4 & c_4 : s_6 \succ s_3 \succ s_2 \succ s_4 \succ s_5 \succ s_1 \\
s_5 : c_3 \succ c_1 \succ c_4 \succ c_2 & \\
s_6 : c_1 \succ c_3 \succ c_4 \succ c_2. &
\end{array}
$$

Finally, schools' capacities are $q_{c_1} = q_{c_2} = 1$ and $q_{c_3} = q_{c_4} = 2$. Given that we have to allocate optimally one extra position, the optimal solution for the relaxed aggregated linearization is $x_{s_1c_3} = 1, x_{s_2c_2} = 1, x_{s_3c_1} = 0.16, x_{s_3c_2} = 0.66, x_{s_4c_1} = 0.5, x_{s_4c_2} = 0.16, x_{s_4c_3} = 0.16, x_{s_5c_3} = 0.83, x_{s_6c_1} = 0.5, x_{s_6c_4} = 0.33$ and $\mathbf{t} = (0.16, 0.83, 0, 0)$ with the cost equal to 1.33.

On the other hand, the optimal solution for the relaxed non-aggregated linearization is $x_{s_1c_3} = 0.83, x_{s_1c_4} = 0.08, x_{s_2c_2} = 1, x_{s_3c_1} = 0.11, x_{s_3c_2} = 0.66, x_{s_3c_4} = 0.02, x_{s_4c_1} = 0.66, x_{s_4c_3} = 0.16, x_{s_5c_3} = 1, x_{s_6c_1} = 0.52, x_{s_6c_4} = 0.30$ and $\mathbf{t} = (0.30, 0.66, 0, 0.03)$ with cost equal to 1.03.

4.8.4. 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 $x^\star_{s_1,c_1} = 1$, $x^\star_{s_2,c_2} = 1$, $x^\star_{s_3,c_4} = 1$ and all other entries equal to zero. On the other side, the student-optimal stable matching is $\mu^\star = \{(s_2,c_1), (s_1,c_3), (s_3,c_4)\}$. Note that the set of fully-subscribed schools is mutually not-inclusive.

4.8.5. 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 are

$$\begin{aligned}
s_1 &: c_1 \succ c_4 \succ c_3 \succ c_2 \succ \dots & c_1 &: s_2 \succ s_1 \succ s_3 \succ \dots \\
s_2 &: c_1 \succ c_2 \succ c_3 \succ c_4 \succ \dots & c_2 &: s_2 \succ s_1 \succ s_3 \succ \dots \\
s_3 &: c_4 \succ c_1 \succ c_5 \succ c_6 \succ \dots & c_3 &: s_1 \succ s_2 \succ s_3 \succ \dots \\
s_4 &: c_5 \succ c_1 \succ c_4 \succ c_6 \succ \dots & c_4 &: s_3 \succ s_4 \succ s_5 \succ \dots \\
s_5 &: c_4 \succ c_5 \succ c_6 \succ c_1 \succ \dots & c_5 &: s_4 \succ s_3 \succ s_5 \succ \dots \\
& & c_6 &: s_5 \succ \dots
\end{aligned}$$

The "..." 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 $(\mathbf{x}^\star, \mathbf{t}^\star)$ with $x^\star_{s_1,c_1} = 1$, $x^\star_{s_2,c_2} = 1$, $x^\star_{s_3,c_4} = 1$, $x^\star_{s_5,c_4} = 1$, $x^\star_{s_4,c_5} = 1$, $t^\star_4 = 1$ and all other entries equal to zero. We take the instance $\Gamma_{\mathbf{t}^\star}$ (with the expanded capacities in accordance to $\mathbf{t}^\star$) and the matching $\mathbf{x}^\star$ as the input for Algorithm 3. 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 block($\mathbf{x}^\star$). First, note that the student-optimal stable matching of $\Gamma_{\mathbf{t}^\star}$ 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 $\mathbf{x}^\star$ is $\{c_1, c_4, c_5\}$. The only school that is fully subscribed in both μ and $\mathbf{x}^\star$ 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 $\Gamma_{\mathbf{t}^\star}$. 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 $s^\star$ 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 $\mathbf{x}^\star$ 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 3 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.

4.8.6. Separation Algorithm: Counterexample for Baïou and Balinski (2000)

In this counterexample, we show that Algorithm 4, the separation algorithm provided in [**26**], 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:

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

Let us consider the following optimal matching $\mathbf{x}^\star$ of the main program with no stability constraints (i.e., $\mathcal{J} = \emptyset$) and budget $B = 0$: $x^\star_{s_1,c_1} = 1$, $x^\star_{s_2,c_1} = 1$, $x^\star_{s_3,c_6} = 1$, $x^\star_{s_4,c_6} = 1$, $x^\star_{s_5,c_2} = 1$, $x^\star_{s_6,c_3} = 1$, $x^\star_{s_7,c_4} = 1$, $x^\star_{s_8,c_5} = 1$. Note that $n = 8$, $m = 6$, $\bar{q} = 2$.

Algorithm 4 Separation algorithm from [**26**]

Input: An instance $\Gamma = \langle \mathcal{S}, \mathcal{C}, \succ, \mathbf{q} \rangle$, and a matching $\mathbf{x}^\star$.
Output: A violated comb.

1: $i \leftarrow 1$ ▷ index of the school
2: $\mathcal{Q} \leftarrow \emptyset$ ▷ The students that will make the comb
3: Assign $q_i - 1$ elements of $\mathcal{S}$ to $\mathcal{Q}$ such that $\mathbf{x}^\star((\cup_{c_i' \succ_{s'} c_i}(c_i', s'))) \leq \mathbf{x}^\star(\cup_{c_i' \succ_{s''} c_i}(c_i', s''))$, for every $s' \in \mathcal{Q}$, $s'' \in \mathcal{S} \setminus \mathcal{Q}$
4: $U \leftarrow \mathcal{S} \setminus \mathcal{Q}$, and let s be c_i least preferred applicant
5: If $rank_{c_i}(s) < q_{c_i}$, go to Step 17 ▷ $rank_{c_i}(s)$ is the rank of s for c_i
6: **if** $s \in \mathcal{Q}$ **then**
7: $\quad\bar{s} = \text{argmin}_{s'' \in U}\{\mathbf{x}^\star(\cup_{c_i' \succ_{s''} c_i}(c_i', s''))\}$
8: $\quad U \leftarrow U \setminus \{\bar{s}\}$
9: $\quad\mathcal{Q} \leftarrow (\mathcal{Q} \setminus \{s\}) \cup \{\bar{s}\}$
10: **else**
11: $\quad U \leftarrow U \setminus \{s\}$
12: $C \leftarrow S^0(c_i,s) \cup T(c_i,s) \cup (\cup_{s' \in \mathcal{Q}} T(c_i,s'))$
13: **if** $\mathbf{x}(C) < q_i$ **then**
14: $\quad$Return C
15: **else**
16: $\quad$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: $\quad$Go to Step 2
20: **else**
21: $\quad$Return $\emptyset$

If we use the separation algorithm provided by Baïou and Balinski [**26**], 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^{bb} with base (s_3,c_6) and teeth bases (s_3,c_6) and (s_2,c_6). Note that comb C^{bb} has value 1 in $\mathbf{x}^\star$. It took $\mathcal{O}(m \cdot n^2)$ operations to find it.

However, if we use our Algorithm 3, we have that the only school in block($\mathbf{x}^\star$) is c_6. We find the comb $\bar{C}$ with base in (s_2,c_6) and the other tooth in (s_1,c_6). Note that $\bar{C}$ has value 0 in $\mathbf{x}^\star$ and it took $\mathcal{O}(m \cdot n)$ operations to find it, which is the number of operations needed to find block($\mathbf{x}^\star$).

Therefore, if we use the separation algorithm of Baïou and Balinski [**26**], we find a comb which is not the most violated one.

4.9. Additional Background

4.9.1. The Deferred Acceptance Algorithm

In this section, we recall the Deferred Acceptance algorithm introduced in [**67**].

INPUT:	An instance $\Gamma = \langle \mathcal{S},\mathcal{C}, \succ, \mathbf{q}\rangle$.
OUTPUT:	Student-optimal matching.
Step 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.
Step 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.
Step 3:	If all students are enrolled or have applied to all the schools they rank, return the current matching. Otherwise, go to Step 2.

4.9.2. McCormick Linearization

In this section we describe the McCormick convex envelope used to obtain a linear relaxation for bi-linear terms [**109**]; 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 $x_i \cdot x_j$.

4.10. Other Formulations of the Capacity Expansion Problem

In this section we present the generalizations to Formulation (4.3.2) of two models of the classic School Choice problem.

The first model that we generalize was proposed in [**53**]. Let $g^d(s)$ be the number of ranks for student $s \in \mathcal{S}$,[31] $g^h(c)$ be the number of acceptable students for school $c \in \mathcal{C}$ and y^d_{sk} a binary variable that takes value 1 if student s is assigned to a school of rank at most k, and y^h_{ck} an integer variable indicating how many students of rank at most k are assigned to school c, and z_{ck} a binary variable taking value 1 if school c is fully-subscribed to students that rank at most $k-1$ and 0 otherwise:

$$
\begin{aligned}
\min_{\mathbf{x},\mathbf{t},\mathbf{y},\mathbf{z}} \quad & \sum_{(s,c)\in\mathcal{E}} r_{s,c}\cdot x_{s,c} & & \qquad (4.10.1)\\
s.t. \quad & y^h_{cg^h(c)} \le q_c + t_c, & \forall c \in \mathcal{C} & \qquad (4.10.2)\\
& (q_c+t_c)(1-y^d_{sk}) \le y^h_{cr^h_s(c)}, & \forall s \in \mathcal{S}, k=1,\dots,g^d(s), c \in H^=_k(s) & \qquad (4.10.3)\\
& \sum_{c\in H^=_1(s)} x_{s,c} = y^d_{s1}, & \forall s \in \mathcal{S} & \qquad (4.10.4)\\
& \sum_{c\in H^=_k(s)} x_{s,c} + y^d_{s,k-1} = y^d_{sk}, & \forall s \in \mathcal{S}, k=2,\dots,g^d(s) & \qquad (4.10.5)\\
& \sum_{s\in S^=_1(c)} x_{s,c} = y^h_{c1}, & \forall c \in \mathcal{C} & \qquad (4.10.6)\\
& \sum_{s\in S^=_k(c)} x_{s,c} + y^h_{c,k-1} = y^h_{ck}, & \forall c \in \mathcal{C}, k=2,\dots,g^h(c) & \qquad (4.10.7)\\
& x_{s,c} \le 1 - z_{cr^h_s(c)}, & \forall (s,c) \in \mathcal{E} & \qquad (4.10.8)\\
& z_{ck} \ge z_{ck-1}, & \forall c \in \mathcal{C}, k=2,\dots,g^h(c)+1 & \qquad (4.10.9)\\
& 1 - z_{ck} \le y^d_{sr_{cs}}, & \forall c \in \mathcal{C}, k=2,\dots,g^h(c)+1, s\in S^=_{k-1}(c) & \qquad (4.10.10)\\
& (q_c+t_c) z_{cg^h(c)+1} \le y^h_{cg^h(c)}, & \forall c \in \mathcal{C} & \qquad (4.10.11)\\
& \sum_{c\in C} t_c \le B, & & \qquad (4.10.12)\\
& y^h_{ck} \in \mathbb{Z}_+, & \forall c \in \mathcal{C}, k=1,\dots,g^h(c) & \qquad (4.10.13)\\
& x_{s,c} \in \{0,1\}, & \forall (s,c) \in \mathcal{E} & \qquad (4.10.14)\\
& y^d_{sk} \in \{0,1\}, & \forall c \in \mathcal{C}, k=1,\dots,g^h(c) & \qquad (4.10.15)\\
& z_{ck} \in \{0,1\}, & \forall c \in \mathcal{C}, k=1,\dots,g^h(c)+1 & \qquad (4.10.16)
\end{aligned}
$$

where $r_{s,c}$ is the rank of student s for school c, $H^=_k(s)$ is the set of schools acceptable for student s with rank k and $S^=_k(c)$ is the set of students acceptable for school c with rank k. Constraints (4.10.2) ensure that each school does not have assigned more students than its capacity. Constraints (4.10.3) ensure that if student s is assigned to a school with rank superior to k, then it must be because school c has its quota satisfied. Constraints (4.10.4)- (4.10.7)

[31] $g^d(s)$ is the cardinality of the set of schools that student s deems acceptable.

establish the meaning of the y variables. In [**53**], it is discussed that it is not worth to merge constraints (4.10.13) if there are no ties on the schools' side, i.e., $|H_k^=(s)| = 1$, in order to improve the models' relaxation.

The second model that we generalize was introduced by Agoston et al. [**157**]. The authors propose a binary and continuous cutoff score formulations which they claim to be similar to the integer programs in [**53**].

Let z_c be a non-negative variable denoting the cutoff score of school c. The next formulation is based on the fact that a student-optimal envy-free matching is equivalent to a student-optimal stable matching:

$$\min_{\mathbf{x},\mathbf{t},\mathbf{f},\mathbf{z}} \quad \sum_{(s,c)\in\mathcal{E}} r_{s,c} \cdot x_{s,c} \tag{4.10.17}$$

$$s.t. \quad z_c \leq (1 - x_{s,c}) \cdot (|\mathcal{S}| + 1) + w_{sc}, \qquad \forall\ (s,c) \in \mathcal{E} \tag{4.10.18}$$

$$w_{sc} + \epsilon \leq z_c + \left(\sum_{k:k\succeq_s c} x_{s,k}\right) \cdot (|\mathcal{S}| + 1), \qquad \forall (s,c) \in \mathcal{E} \tag{4.10.19}$$

$$f_c \cdot (q_c + t_c) \leq \sum_{s\in\mathcal{S}:(s,c)\in\mathcal{E}} x_{s,c}, \qquad \forall c \in \mathcal{C} \tag{4.10.20}$$

$$z_c \leq f_c(|\mathcal{S}| + 1), \qquad \forall c \in \mathcal{C} \tag{4.10.21}$$

$$\sum_{c:(s,c)\in\mathcal{E}} x_{s,c} = 1, \qquad \forall s \in \mathcal{S} \tag{4.10.22}$$

$$\sum_{s\in\mathcal{S}} x_{s,c} \leq q_c + t_c, \qquad \forall\ c \in \mathcal{C} \tag{4.10.23}$$

$$\sum_{c\in\mathcal{C}} t_c \leq B, \tag{4.10.24}$$

$$x_{s,c}, t_c \in \mathbb{Z}_+, \qquad \forall s \in \mathcal{S}, \forall c \in \mathcal{C} \tag{4.10.25}$$

$$z_c \geq 0 \qquad \forall\ c \in \mathcal{C} \tag{4.10.26}$$

$$f_c \in \{0,1\} \qquad \forall\ c \in \mathcal{C}. \tag{4.10.27}$$

Constraints (4.10.18) imply that if a student s is matched with school c, then her score has reached the cutoff score; we can establish $w_{sc} = |\mathcal{S}| - r_{c,s}$ where $r_{c,s}$ is the rank of student s in the list of school c. Constraints (4.10.19) ensure the envy-freeness, i.e., if student s is admitted to school c or to any better according to her preference, then it must be the case that she has not reached the cutoff at school c. To complete the notion of stability it is needed to include non-wastefulness (no blocking with empty seats); they do so in constraints (4.10.20) and (4.10.21), where the authors include a binary variable f_c indicating whether school c rejects any student in the solution. A binary cutoff score formulation is also provided in their paper.

4.11. Initialization Details of the Cutting-plane Method

In Algorithm 2, the initialization of $\mathcal{J}$ at Step 1 affects critically the gap at the root node of the Branch-and-Bound algorithm. There are several ways in which we can improve the Algorithm at this step.

In our formulation, we improve the root node gap by adding the two following kinds of constraints:

- The comb constraints (4.3.7b) for the capacity vector $\mathbf{t} = \mathbf{0}$. Since the allocation of extra capacities weakly improves the matching for every student, it is valid to add stability constraints for the zero expansion. The main problem can be initialized with any subset of comb constraints $\mathcal{J}$. For this numerical study, we define $\mathcal{J}$ as the set of comb constraints derived from the student-optimal stable matching obtained when $B = 0$, which can be efficiently obtained using DA. Specifically, let μ^0 be the student-optimal stable matching when $B = 0$, and let $\underline{\mu}^0(c)$ be the lowest priority student admitted to school c (according to $\succ_c$). Then, for each school c such that $|\mu^0(c)| = q_c$, we add to $\mathcal{J}$ the comb constraint involving $C_c = S_{\underline{\mu}^0(c),c} \cup \bigcup_{s:\mu^0(s)=c} T^-_{s,c}$.
- The relaxed stability constraints of Formulation (4.10.1).

4.12. Heuristics

In this section, we present two natural methods: (i) a greedy approach (Grdy), and (ii) an LP-based heuristic (LPH). These two heuristics rely on the computation of a student-optimal stable matching, which can be done in polynomial time using the DA algorithm. The description of DA in Appendix 4.9.1.

Greedy Approach. In Grdy, we explore the fact that the objective function is decreasing in $\mathbf{t}$ and iteratively assign an extra seat to the school leading to the greatest reduction in the objective. More precisely, Grdy performs B sequential iterations. At each iteration, we evaluate the objective function for each possible allocation of one extra seat using DA. Then, the school leading to the lowest objective receives that extra seat. At the end of this procedure, B extra seats are allocated. In Algorithm 5, we formalize our Greedy heuristic.

Algorithm 5 Grdy

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 μ in the expanded instance $\Gamma_{\mathbf{t}}$.

Initialize $\mathbf{t} \leftarrow \mathbf{0}$
while $\sum_{c \in \mathcal{C}} t_c < B$ **do**
  $c^\star \in \text{argmin}\,\{f(\mathbf{t} + \mathbf{1}_c) : \; c \in \mathcal{C}\}$, where f is defined as in Expression (4.7.4)
  $\mathbf{t} \leftarrow \mathbf{t} + \mathbf{1}_{c^\star}$
return $\mathbf{t}$ and μ

In the algorithm above, $\mathbf{1}_c \in \{0,1\}^{\mathcal{C}}$ denotes the indicator vector whose value is 1 in component $c \in \mathcal{C}$ and 0 otherwise. Recall that, for a given $\mathbf{t}$, $f(\cdot)$ can be evaluated in polynomial time using the DA algorithm.

LP-based Heuristic. If we relax the stability constraints, Formulation 4.3.2 can be formulated as a minimum-cost flow problem whose polytope has integer vertices. Once we enrich this problem with the expansion of capacities, the integrality of the vertices is preserved. Hence, LPH starts by solving the linear program that minimizes Objective (4.3.3a), restricted to the set $\mathcal{P}$. As a result, we obtain an allocation of extra seats and an assignment that is not necessarily stable — recall that $\mathcal{P}$ is the space of fractional (potentially non-stable) matchings. Then, using the DA algorithm, we compute the student-optimal stable matching in the new instance that considers the capacity expansion obtained by the linear program. In Algorithm 6, we formalize the LPH heuristic.

Algorithm 6 LPH

Input: An instance $\Gamma = \langle \mathcal{S},\mathcal{C}, \succ, \mathbf{q}\rangle$ and a B.
Output: A feasible allocation $\mathbf{t}$ and a stable matching μ in the expanded instance $\Gamma_{\mathbf{t}}$.
1: Obtain $(\mathbf{x}^*,\mathbf{t}^*) \in$ 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 stable matching μ in instance $\Gamma_{\mathbf{t}^*}$ using the DA algorithm
3: **return** $\mathbf{t}^*$ and μ

4.13. Model Extensions

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

- Adding budget: 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
$$\sum_{c\in\mathcal{C}} t_c \cdot p_c \leq B',$$
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 our model by considering the unary expansion of the variables t_c for $c \in \mathcal{C}$, as we did in BB-CAP. Specifically, let $y_c^k = 1$ if the capacity of school c is expanded in k seats, and $y_c^k = 0$ otherwise. Then, we know that $\sum_{k=0}^{B} k \cdot y_c^k = t_c$, and we must add the constraint that $\sum_{k=0}^{B} y_c^k = 1$ for each $c \in \mathcal{C}$. If the capacity of school c can only be expanded by values in a subset $B' \subseteq B$, we can enforce this by adding the constraints $y_c^k = 0$ for all $k \in B \setminus B'$. 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 by introducing a parameter $m_{s,c}$, which is equal to 1 if student s is currently enrolled in school c, and 0 otherwise, and defining $M = \{s \in \mathcal{S} : m_{s,c} = 1$ for some $c \in \mathcal{C}\}$. Then, we would only have to update a couple of constraints:

$$\begin{aligned} \sum_{s\in\mathcal{S}} x_{s,c} \cdot (1 - m_{s,c}) \leq q_c + t_c, \quad &\forall\, c \in \mathcal{C}, \\ \sum_{c\in\mathcal{C}} x_{s,c} = 1, \qquad &\forall\, s \in M. \end{aligned} \tag{4.13.1}$$

  The first constraint ensures that students currently enrolled do not count towards the capacity of the school they are currently enrolled. The second constraint ensures that all students that are currently enrolled are assigned to some schools (potentially, to the same school they are currently enrolled).
- Room assignment: Schools report the number of vacancies they have for each level. This decision depends on the classrooms they have and their capacity. However, schools decide (before the assignment) what level goes in each classroom, and this determines the number of reported vacancies for that level. This may introduce some inefficiencies, since some levels may be more demanded, and thus assigning a larger classroom may benefit both students in that school but also in others.
- Quota assignment: Many school choice systems have different quotas to serve underrepresented 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.

Chapter 5

Stable Matching with Dynamic Priorities

by

Federico Bobbio[1], Margarida Carvalho[2], Ignacio Rios[3], and Alfredo Torrico[4]

([1]) CIRRELT, DIRO Université de Montréal
([2]) CIRRELT, DIRO Université de Montréal
([3]) School of Management, The University of Texas at Dallas
([4]) CDSES, Cornell University

Prologue: Often, in the real world, when a group of siblings applies for school admission, their priorities are updated as part of a pre-processing routine *before* these new priorities are included in the input of the matching mechanism. In this chapter, we address the problem of updating dynamically the priorities of siblings *inside* the mechanism, with the goal of matching them to the same school (Question 7).

Article details: This paper was accepted at EAMMO 2023; we choose the non-archival option of this conference. An extended version of this paper will be submitted to a peer-reviewed venue.

Contributions of the authors: Federico Bobbio participated in all the stages of the work, being the main author of the paper. In particular, he devised the counter-examples to the non-existence of a stable matching, the proofs on the existence of a stable matching and the algorithms to find it. He also wrote the first draft of the paper.

Margarida Carvalho contributed to the initial conceptualization of the problem. She participated in the revision and editing of the paper, and in the conceptualization of the proofs.
Ignacio Rios provided the idea of the project together with the initial definitions of dynamic priorities. He participated in the revisions and editing of the paper, and in the conceptualization of the proofs.
Alfredo Torrico contributed to the initial conceptualization of the problem. He participated in the revision and editing of the paper, and in the conceptualization of the proofs.

Résumé. Nous étudions le problème de la recherche d'un appariement stable dans le cadre de priorités dynamiques, où la chambre de compensation donne la priorité à certains agents en fonction de l'allocation d'autres agents, et nous utilisons le choix de l'école comme exemple motivant. Pour ce faire, nous introduisons un modèle stylisé de marché d'appariement biparti avec priorité aux familles. Nous soutenons que la notion standard de stabilité ne s'applique pas en présence de priorités dynamiques. Pour ce faire, et motivés par la pratique, nous définissons plusieurs hypothèses sur les préférences des familles et les priorités, nous introduisons différentes notions de stabilité en présence de priorités dynamiques et nous montrons qu'un appariement stable dans ces conditions peut ne pas exister. Du côté positif, nous montrons que de tels appariements existent si les familles préfèrent strictement que leurs membres restent ensemble dans deux situations importantes : (i) lorsque les familles sont au maximum de deux personnes, et (ii) lorsqu'il n'y a qu'un seul niveau scolaire. En outre, nous concevons un mécanisme permettant de trouver de telles affectations stables en temps polynomial. Enfin, nous montrons que le problème de la recherche d'un appariement stable sous des priorités dynamiques de cardinalité maximale est NP-difficile.
Mots clés : Appariement stable, choix de l'école, familles, priorités dynamiques

Abstract. We study the problem of finding a stable matching under dynamic priorities, whereby the clearinghouse prioritizes some agents based on the allocation of others, and we use school choice as a motivating example. To accomplish this, we introduce a stylized model of a two-sided matching market with siblings' priority. We argue that the standard notion of stability does not apply in the presence of dynamic priorities. To address this, and motivated by practice, we define several assumptions on families' preferences and siblings' priorities, introduce different notions of stability under dynamic priorities, and show that a stable matching under these settings may not exist. On the positive side, we show that such matchings exist if families strictly prefer their members to remain together in two important settings: (i) when families are of size at most two, and (ii) when there is a single grade level. In addition, we devise a mechanism to find such stable assignments in polynomial time. Finally, we show that the problem of finding the stable matching under dynamic priorities of maximum cardinality is NP-hard.
Keywords: Stable matching, school choice, families, dynamic priorities

5.1. Introduction

The theory of two-sided many-to-one matching markets, introduced by Gale and Shapley [**67**], provides a framework for solving many large-scale real-life assignment problems. Examples include entry-level labor markets for doctors and teachers, education markets from daycare, school choice and college admissions, and other applications such as refugee resettlement. A common feature in many of these markets is the use of mechanisms that find a stable assignment, as this guarantees that no coalition of agents has incentives to circumvent the match.

In many of these markets, the clearinghouse may be interested in finding a stable allocation, while individual agents may care about their assignment and that of other agents. For instance, in the hospital-resident problem, couples jointly participate and must coordinate to find two positions that complement each other. In school choice, students may prefer to be assigned with their siblings or neighbors. In refugee resettlement, agencies may prioritize allocating families with similar backgrounds (e.g., from the same region or speaking the same language) to the same cities.

One approach to accommodate these joint preferences is to provide priorities contingent on the assignment. For instance, many school choice systems (including NYC, New Haven, Denver, Chile, etc.) consider *sibling priorities*, by which students get prioritized in schools where they have a sibling currently assigned or enrolled. Similarly, in refugee resettlement, families may get higher priority in localities where they have relatives based on *family reunification*. However, most clearinghouses assume that priorities are fixed and known before the assignment process and, thus, cannot accommodate settings in which priorities depend on the current assignment. Similarly, most definitions of stability and justified-envy assume that priorities are fixed and known, and thus, also fail to capture the aforementioned setting.

In this paper, we study the problem of finding a stable matching under *dynamic priorities*, i.e., when priorities are updated based on the current assignment, and we use school choice with siblings as a motivating example. To accomplish this, we first introduce a stylized model where students belong to (potentially different) grade levels and may have siblings applying to the system (potentially in different levels). On the one hand, each family reports preferences over the assignment of their members while, on the other hand, schools prioritize students with siblings already enrolled or currently assigned, and break ties among students in the same priority group (with or without siblings) using a random tie-breaker.

Motivated by the Chilean school choice system, we distinguish two types of sibling priority: (i) static, whereby students who have a sibling currently enrolled but not participating in the admission process get prioritized; and (ii) dynamic, whereby students with a sibling who is also participating in the admission process and is currently assigned get prioritized. Notably, dynamic priorities depend on the current assignment and, thus, must be granted and updated simultaneously while solving the allocation task. This simultaneity introduces a series of challenges, and the standard notion of *justified envy* no longer applies.

To overcome these challenges, we start by simplifying the space of families' preferences and introducing several assumptions that limit how dynamic priorities work. Specifically, we present the concepts of absolute/partial and dependent/independent justified envy to break ties across and within priority groups. Based on these definitions, we introduce several notions of stability and show that a stable matching may not exist. Nevertheless, we show that a stable matching under dynamic priorities exists if families strictly prefer that their siblings are assigned together and either (i) families have at most two members participating in the admission process or (ii) there is a single grade level. Moreover, we introduce a new family of mechanisms that find such a stable matching. Finally, we discuss other properties of the mechanism, and we show that finding a maximum cardinality stable matching under dynamic priorities is NP-hard.

Our work contributes to the literature in several ways. To the best of our knowledge, this is the first work to formalize different types of siblings' priorities and also the first to introduce the idea of dynamic priorities. Consequently, we introduce a novel notion of stability under dynamic priorities, where these are contingent on the matching. We also provide the first complexity results for a stable matching problem with dynamic priorities. Although we focus on school choice as a motivating example, our results and insights may deem helpful in the design of matching mechanisms where priorities depend on the assignment of others, such as in daycare assignments, college admissions, refugee resettlement, among others.

5.1.1. Organization

The remainder of this chapter is organized as follows. In Section 5.2, we discuss the relevant literature. In Section 5.3, we introduce our model. In Section 5.4, we study the existence of a stable assignment under dynamic priorities. In Section 5.5, we study the complexity of finding a maximum cardinality stable matching under dynamic priorities. Finally, in Section 5.6 we conclude.

5.2. Literature Review

Our paper draws from various threads of existing literature. In Section 5.2.1, we delve into the primary stream of research within the stable matching with complementarities field. Following that, in Section 5.2.2, we outline the placement of our contribution within this academic landscape.

5.2.1. General Context

Matching with families. A recent strand of the literature has extended the classic school choice model [**8**] to incorporate families. Dur et al. [**57**] consider a setting where siblings report the same preferences, and assignments are feasible if and only if all family members are assigned to the same school (or all of them are unassigned). The authors argue that justified envy is not an adequate criterion for the problem. Thus, they propose a new solution concept (suitability), show that a suitable matching always exists, and introduce a new family of strategy-proof mechanisms that finds a suitable matching. Correa et al. [**50**] also consider a model with siblings applying to potentially different grades, but assume that each sibling submits their own (potentially different) preference list. In addition, the authors assume that the clearinghouse aims to prioritize the joint assignment of siblings, but they model it as a soft requirement, i.e., an assignment may be feasible even if siblings are not assigned to the same school. To prioritize the joint assignment of siblings, Correa et al. [**50**] introduce (i) the use of lotteries at the family level; (ii) a heuristic that processes grades sequentially in decreasing order, updating priorities in each step to capture siblings' priorities that result from the assignment of higher grades; and (iii) the option for families to report that they prefer their siblings to be assigned to the same school rather than following their individual reported preferences. This last feature, called *family application*, prioritizes the joint assignment of siblings by updating the preferences of younger siblings by adding the school of assignment of their older siblings. The authors show that all these features significantly increase the probability that families get assigned together.

Matching with couples. Our paper is also related to the matching with couples literature, which is commonly motivated by labor markets such as the matching for medical residents. In this setting, couples wish to be matched to the same hospital (or at least to the same region) and hence, they report a joint preference list of pairs of hospitals. For an extension of the stability concept with couples, Roth [**132**] shows that a stable matching may not exist if couples participate. To overcome this limitation, Klaus et al. [**93, 95**] introduce the property of weak responsive preferences and show that this guarantees the existence of a stable assignment. Kojima et al. [**98**] provide conditions under which a stable matching exists with high probability in large markets, and introduce an algorithm that finds a stable matching with high probability which is approximately strategy-proof. Ashlagi et al. [**19**] find a similar result, as they show that a stable matching exists with high probability if the number of couples grows slower than the size of the market. However, the authors also show that a stable matching may not exist if the number of couples grows linearly. Finally, Nguyen and Vohra [**117**] show that the existence of a stable matching is guaranteed if the capacity of the market is expanded by at most a fixed number of spots to the schools.

Matching with complementarities. Beyond families and couples, the matching literature has studied other settings with complementarities. For instance, Ashlagi et al. [**21**] show that using correlated lotteries can increase community cohesion by increasing the probability of neighbors being assigned to the same schools. Dur and Wiseman [**59**] also study the school choice problem with neighbors and show that a stable matching may not exist if students have preferences over joint assignment with their neighbors. Moreover, the authors show that the student-proposing deferred acceptance algorithm is not strategy-proof and propose a new algorithm to address this issue. Kamada and Kojima [**86**] study matching markets where the clearinghouse cares about the composition of the match and thus imposes distributional constraints. The authors show that existing mechanisms suffer from inefficiency and instability and propose a mechanism that addresses these issues while respecting the distributional constraints. Nguyen and Vohra [**118**] also study the problem with distributional concerns but consider these constraints as soft bounds and provide ex-post guarantees on how close the constraints are satisfied while preserving stability. Nguyen et al. [**116**] introduce a new model of many-to-one matching where agents with multi-unit demand maximize a cardinal linear objective subject to multidimensional knapsack constraints, capturing settings such as refugee resettlement, day-care matching, and school choice/college admissions with diversity concerns. The authors show that a pairwise stable matching may not exist and provide a new algorithm that finds a group-stable matching that approximately satisfies all the multidimensional knapsack constraints. Another example in which agents care about other agents' assignment is the affiliate stable matching problem, where, for instance, a college is not only interested in hiring good academic candidates, but also wishes that its graduates find good jobs [**55, 96**].

5.2.2. Our Contributions and the Literature

The problem of guaranteeing the existence of a stable matching in a many-to-one stable matching problem with complementarities is known to be intractable under general assumptions. This problem emerged when the National Resident Matching Program noticed a decrease in the participation of couples in the hospital-resident matching market. In this regard, Roth [**132**] finds an instance of a matching market with couples where there is no stable matching. It is evident that the non-existence of a stable matching with complementarities is due to a combination of the following factors: (i) the way the preferences of the agents are represented, and (ii) the way blocking coalitions are defined (i.e., the definition of stability).

The first step towards a systematization of the preferences' representation was given with the introduction of *responsive preferences* [**133**]. Ronn [**129**] was the first to prove that deciding whether an instance of the hospital-resident problem with couples has a stable matching is NP-complete; he proved this result even when all the hospitals have capacity one and there are no single students. Then, other negative complexity results followed using other preference representations and/or definitions of stability. However, the strength of this result lies in the fact that for any definition of blocking coalition that does not encompass any particular requirement on the preference list of the couples, there is no hope of having existence guarantees for a stable matching. A breakthrough came with the introduction of *weakly responsive prefereneces* [**93, 95**] which proved to be a sufficient property on the ordering of couple's preference lists to guarantee the existence of a stable matching. The key intuition behind this result is that when there are no negative externalities among the members of a couple in the assignment, then we can treat each member as an individual; *e.g.*, if the couple only ranks the jobs in the same metropolitan area, then each of the partners can be considered as a single agent expressing their individual preferences. All the results mentioned consider a representation of couple preferences that allow two partners to be matched to different locations. McDermid and Manlove [**111**] prove that under *consistent* preference lists and for the classic notion of stability (members of a couple are treated as individuals) derived from [**71**], the problem of deciding the existence of a stable matching is still NP-complete. Under restrictive conditions on the stability definition, the authors are able to provide a polynomial-time algorithm that outputs a stable matching or report that none exists. This result highlights that as a consequence of further modifications on the stability condition (point (ii)), then existence can be guaranteed.

On the other side of the spectrum, if members in a couple are considered as an indivisible entity rather than individuals, we find the hospital-resident problem with sizes (HRS), which is a many-to-many matching market problem. McDermid and Manlove [**111**] provide a polynomial-time algorithm that outputs a stable matching under restrictive conditions on the preference lists and capacities, otherwise the problem remains NP-hard. Another example in which existence is ensured when couples are indivisible, is the case in which the members of a pair apply to the same schools but at different levels (point (i)); for example, in the case of siblings applying to different education grades [**57**].

To summarize, the literature, which has been mainly focused on the case of couples (families of size two) has been able to obtain existence of a stable matching when 1) the preferences of the families can be considered in a fashion similar to individual preferences for each family member, or 2) the family is treated as an indivisible block. On one side, we would like to have a *flexible* representation of the preference lists such as in the *weak responsive* assumption, which allows us to match family members to different schools. On the other side, given that *weak responsive preferences* require a complete ranking of all the possible combinations of acceptable locations, we would like to have a *practical* representation of the family preference lists such as in the indivisible family/couple case. The advantage of the flexibility of the former model of preference lists comes with the price that justified envy must be defined and checked among all comparable subsets of family members of cardinality 2; on the other side, the practicality of the indivisible representation comes at the cost of lacking the expressivity to describe the envy of a member in a family who prefers to go to another location as a single individual.

One may hope that by further restricting the domain of preference lists, it would be possible to guarantee the existence of a stable matching. One such way, related to many applications in the real world, is the case in which residents (or students) are ranked by all the hospitals (schools) according to a single tie-breaker (or master-list). However, Biró et al. [**37**] show that even if both the residents in a couple and the hospitals have their preference lists derived from a single tie-breaker, then the problem of the existence of a stable matching is NP-complete.

In this work, we opt for a practical representation of the preference lists of each member in a family, and we provide a notion of stability that is flexible enough to express envy also as a function of the assignment of an individual's family member. The key insight that allows us to achieve this result is that we let the ranking of a student depend also on the matching of their family members, thus providing a notion of justified envy that relies on dynamic priorities. The idea of establishing justified envy on a dynamic notion of priority differs from the existing literature; in fact, justified envy has previously always been defined on the basis of a static representation of preference lists and priorities.

5.3. Model

In this section, we introduce a two-sided matching market model that includes a priority system. To facilitate the exposition, we use school choice with sibling priorities as a concrete application of the model.

Let $\mathcal{S}$ be a finite set of students and $\mathcal{F} \subseteq 2^{\mathcal{S}}$ be a partition of $\mathcal{S}$ where $f \in \mathcal{F}$ is called a *family* and its size is denoted as $|f|$. For $f \in \mathcal{F}$ with $|f| \geq 2$, we say that students s and s' are *siblings* if $s,s' \in f$. If $f \in \mathcal{F}$ is such that $f = \{s\}$, then we say that s has no siblings. With a slight abuse of notation, we define function $f : \mathcal{S} \to \mathcal{F}$ to map a student into their specific family, i.e., each student $s \in \mathcal{S}$ belongs to family $f(s) \in \mathcal{F}$. Note that students s and s' are siblings if $f(s) = f(s')$ and a student s has no siblings when $f(s) = \{s\}$.

Let $\mathcal{C}$ be a finite set of schools and $\mathcal{G}$ be the set of grade levels. We define a function $g : \mathcal{S} \to \mathcal{G}$ that maps a student $s \in \mathcal{S}$ into the grade level $g(s)$ to which they are applying to. With a slight abuse of notation, we denote by $\mathcal{S}^g \subseteq \mathcal{S}$ the set of students applying to grade level $g \in \mathcal{G}$. We assume that each school $c \in \mathcal{C}$ offers $q_c^g \in \mathbb{Z}_+$ seats on grade level $g \in \mathcal{G}$, where $q_c^g = 0$ means that school c does not offer grade g.

Let $\mathcal{E} \subseteq \mathcal{S} \times \{\mathcal{C} \cup \{\emptyset\}\}$ be the set of feasible pairs, i.e., $(s,c) \in \mathcal{E}$ implies that student s and school c deem each other acceptable and $q_c^{g(s)} > 0$; $\emptyset$ represents being unassigned. A matching is an assignment $\mu \subseteq \mathcal{E}$ such that (i) each student is assigned to at most one school in $\mathcal{C}$, and (ii) each school is assigned at most its capacity in each grade level. Formally, for $\mu \subseteq \mathcal{E}$, let $\mu(s) \in \mathcal{C} \cup \{\emptyset\}$ be the school that student s was assigned to, $\mu(f) \subseteq \mathcal{C}$ be the subset of schools where the students of family f were assigned to, i.e., $\mu(f) = \{\mu(s) : \ s \in f\}$, and $\mu(c) \subseteq \mathcal{S}$ be the set of students assigned to school c. Given a grade g, we denote by $\mu^g(c)$ the set of students assigned to school c at grade g. Then, a matching satisfies that (i) $\mu(s) \in \mathcal{C} \cup \{\emptyset\}$ for all students $s \in \mathcal{S}$ and (ii) $|\mu^g(c)| \leq q_c^g$ for all schools $c \in \mathcal{C}$ and grade levels $g \in \mathcal{G}$.[1]

Each family $f = \{s_1, \ldots, s_\ell\} \in \mathcal{F}$ has a strict preference order $\succ_f$ over tuples in $(C \cup \{\emptyset\})^\ell$, which means that $(c_1, \ldots, c_\ell) \succ_f (c'_1, \ldots, c'_\ell)$ implies that family f prefers that its members $s_1, \ldots, s_\ell$ go to schools $c_1, \ldots, c_\ell$ over $c'_1, \ldots, c'_\ell$, respectively; notice that we implicitly assume that students in a family are sorted, thus making the comparison of tuples of schools unambiguous. On the other hand, each school $c \in \mathcal{C}$ has a strict preference order $\succ_c$ over feasible subsets of $\mathcal{S}$ which means that for subsets $S,S' \subseteq \mathcal{S}$ that satisfy grade level capacities, $S \succ_c S'$ denotes that school c prefers students in S over students in S'.

[1]Notice that the model captures other single-level applications such as refugee resettlement, college admissions and the hospital-resident problem.

As Roth [**135**] discusses, a desired property of any matching is stability, i.e., that there is no group of agents that prefer to circumvent their current match and be matched to each other. Given a matching $\mu \subseteq \mathcal{E}$, we say that student s has *justified envy* towards another student s' assigned to school c if (i) $g(s) = g(s')$, (ii) $(c, \mu(f \setminus \{s\})) \succ_f \mu(f)$, and (iii) $(\mu(c) \cup \{s\}) \setminus \{s'\} \succ_c \mu(c)$. In words, the first condition states that both students belong to the same grade level; the second condition implies that the family prefers that $s \in f$ is assigned to c rather than $\mu(s)$, given the assignment of their siblings; and the third condition states that school c prefers the set of student that replaces s' with s. In addition, we say that a matching μ is *non-wasteful* if there is no student $s \in \mathcal{S}$ and school c such that $(c, \mu(f \setminus \{s\})) \succ_f \mu(f)$ and $|\{s' \in \mu(c) \ : \ g(s') = g(s)\}| < q_c^g$. Finally, we say that a matching is *stable* if no student has *justified envy* and it is *non-wasteful*.

To account for *sibling priorities*, we aim to reshape the space of preferences of the schools. Sibling priorities can happen in two forms:

(1) *Static priority*: A family $f \in \mathcal{F}$ has static priority in school c if one or more students in f are applying to c and have a sibling who is currently enrolled in c and is not participating in the admission process.[2] Therefore, school c prefers each student in f over students in $\mathcal{S}$ with no sibling priority.

(2) *Dynamic priority*: A family $f \in \mathcal{F}$ has dynamic priority in school c if two or more students in f are simultaneously applying to c. Therefore, school c prefers those students in f over students in $\mathcal{S}$ with no sibling priority. This type of priority is called dynamic because students get prioritized only if another sibling is assigned to the school, i.e., priorities adapt to the current matching.

Throughout the paper, we often shorten sibling priority as priority. Under static (resp. dynamic) priorities, we say that student s *provides* sibling priority in school c if s is currently enrolled (resp. assigned) in c, and we say that the siblings of s *receive* sibling priority in school c. Note that a student may receive static and dynamic priority in different schools or both types of priority in the same one. For instance, suppose that a family $f = \{s, s'\}$ is applying to schools c and c', and that s and s' have a sibling $s'' \notin \mathcal{S}$ currently enrolled in c and not applying to the system. If s, who receives static priority from s'' in school c, gets assigned to school c' in the current matching,[3] then s' would receive static priority in c and dynamic priority in c'. In contrast, if s gets assigned to c, then s' receives both static and dynamic priority in c. Therefore, we assume that static priority overrules dynamic priority, i.e., a student with potentially both priorities in a given school can only benefit from the static priority.[4] In other words, students are not additionally prioritized if they have siblings enrolled and also siblings currently matched. We borrow this

[2]This means that this sibling is not part of the input $\mathcal{S}$.

[3]This could happen if the family prefers s to be assigned in school c', or it could happen if school c is over-demanded and all the seats are filled with students with static sibling priority.

[4]In the example above, s' would only have static priority.

assumption from practice, as in certain school districts (e.g., in Chile), the clearinghouse prefers to assign students with static priority because their probability of enrollment is higher compared to students without siblings currently enrolled.

Given the above, in practice, these priorities define three disjoint groups of applicants in each school: (i) students with static priority, (ii) students with dynamic priority, and (iii) students with no priority. Within each group, all students are equally preferred by the school, and thus the clearinghouse breaks ties using a random tie-breaker. Note that if there are only students with no priority and families with static priorities, then the random tie-breaker defines a strict order over the whole set of students $\mathcal{S}$ in each school, as the group with siblings will be always prioritized over the group with no siblings. Thus, in this case, for any school $c \in \mathcal{C}$, $\succ_c$ would be as if no student had siblings, but with the group of students with sibling priority placed first in the list and then the rest.[5] This implies the following immediate proposition.

Proposition 5.3.1 ([**67**])**.** If there are no students who can receive dynamic priority, then a stable matching exists.

Given this positive result, we focus our attention on dynamic priorities where the existence may not be guaranteed. Henceforth, we consider the following assumption.

Assumption 5.3.2. No student has static priority in any school. Thus, in each school, the set of students are composed by two disjoint groups of applicants: (i) students with (dynamic) sibling priority, and (ii) students with no priority.

In the remainder of the paper, we use sibling priority and dynamic priority interchangeably. In addition, we assume that schools break ties within each group with a random tie-breaker and we denote by $p_{s,c} \in \mathbb{R}_+$ the value of the random tie-breaker of student s for school c. As opposed to static priorities, the combination of dynamic priorities and random tie-breakers do not define a unique order among any two pair of students for each school, as this pair may change from one priority class to the other depending on the current match of their siblings. In fact, the existence of a stable matching is not guaranteed as shown in [**50**] (see their Proposition 1).

The key insight with dynamic priorities is the dependency of the priorities on the *current matching*. To illustrate, consider a family $f = \{s,s'\}$ and a matching mechanism that, at some step, matches student s to school c and student s' to some school $c' \in \mathcal{C} \cup \{\emptyset\} \setminus \{c\}$ such that $(c,c) \succ_f (c,c')$. Since s and s' are siblings, we say that s' *receives dynamic priority* in c from s; given this priority, the mechanism would attempt the assignment of s' to c in grade level $g(s')$, potentially displacing another student $s'' \notin f$ without priority applying to the same grade $g(s')$. Given that multiple families are simultaneously applying to different schools and grade levels, a stable matching may not exist as we previously mentioned. To address this simultaneity challenge, school districts have either (i) defined an order to process grades, and the clearinghouse updates

[5]In other words, the static priority and the random tie-breaking rule define a unique set ordering $\succ_c$ which translates in a linear preference order.

dynamic priorities before moving to the next grade [**50**]; or (ii) do not consider dynamic priorities. As we discuss in Appendix 5.7, different processing orders of grade levels may lead to different outcomes.

The design of dynamic sibling priorities opens four immediate important questions. First, what is an appropriate notion of stability to capture dynamic priorities? Second, under which assumptions can we guarantee the existence of a stable matching? Third, if such assumptions exist, can we find a stable matching under dynamic priorities efficiently? And finally, what are the properties of these stable matchings? Our goal in the next section is to simplify the space of preferences and formalize how siblings' priorities affect schools' ordering of students, so as to properly define a new notion of stability that considers dynamic priorities.

5.3.1. Simplifying the space of preferences and priorities

The definition of *justified envy* in the previous section assumes that schools have preferences over sets of students, and that families have joint preferences over tuples of schools. However, in most clearinghouses, preferences are not as complex. In practice, they involve students declaring linear preferences over schools, and schools' linear preferences are determined by a combination of random tie-breakers and priority groups. For this reason, in the remainder of the paper, we assume a simplified structure of preferences, as formalized in the following Assumption 5.3.3.

Assumption 5.3.3. We assume the following structure for preferences and tie-breaking rules:

(1) On the students' side, we assume that each family reports a strict preference order over $\mathcal{C} \cup \{\emptyset\}$ and that each family member s follows the same preference order as their family among the schools that offer grade $g(s)$.

(2) On the schools' side, we assume that every school has sibling priority and uses a random tie-breaker to break ties between students across applicant groups (i.e., students with or without sibling priority).

Although Assumption 5.3.3 simplifies the reporting of preferences, the sibling priority needs some limitations to ensure the fairness of the assignment, as the following example illustrates.

Example 5.3.4. Consider an instance with a single grade level, a set of students $\mathcal{S} = \{a_1, a_2, a_3, s_1, s_2, s'_1, s'_2\}$ where $f = \{s_1, s_2\}$ and $f' = \{s'_1, s'_2\}$ are siblings, and a single school c with capacity 4. Moreover, suppose the random-tie breakers of school c are $p_{a_1,c} > p_{a_2,c} > p_{a_3,c} > p_{s_1,c} > p_{s_2,c} > p_{s'_1,c} > p_{s'_2,c}$. Then, one possible matching is $\mu = \{(a_1, c), (a_2, c), (a_3, c), (s_1, c), (s_2, \emptyset), (s'_1, \emptyset), (s'_2, \emptyset)\}$, which is also the student-optimal stable matching (with the classical notion of stability). However, the alternative matchings

$$\mu' = \{(a_1, \emptyset), (a_2, \emptyset), (a_3, \emptyset), (s_1, c), (s_2, c), (s'_1, c), (s'_2, c)\}$$

and

$$\mu'' = \{(a_1, c), (a_2, c), (a_3, \emptyset), (s_1, c), (s_2, c), (s'_1, \emptyset), (s'_2, \emptyset)\}$$

are also feasible in terms of capacity (not in terms of the classical notion of stability); depending on how siblings are prioritized over students with no siblings, one matching would be more desirable than the other. □

Consider Example 5.3.4, note that in matching μ' neither s_1' nor s_2' would be admitted in school c without dynamic priority; s_1' gives priority to s_2' and vice-versa, thus creating a priority cycle; arguably, such a cycle of priorities is not desirable. This differs from the case of family f, because there is a matching μ that only accounts for random-tie breakers (and no sibling priority) in which s_1 is matched to c and, if s_2 was not matched to c, could provide dynamic priority to s_2. To rule out this issue, we restrict our attention to matchings that satisfy the following assumption.

Assumption 5.3.5. A student cannot simultaneously provide and receive sibling priority in a given school.

Note that the assignment μ'' in Example 5.3.4 satisfies Assumption 5.3.5 and thus is a feasible matching with sibling priority. On the other hand, μ' does not satisfy this assumption, because neither s_1' nor s_2' would be assigned in μ' if the other is not part of $\mathcal{S}$.

Given Assumption 5.3.5, the key question is which matchings the clearinghouse prefers. As we saw in Example 5.3.4, sibling priority in μ'' leads to s_2 displacing another students previously assigned in c. Before focusing on which matchings are preferred, we ask ourselves the following: Is a student with sibling priority "allowed" to displace any other student without priority? This question leads us to define two notions of priorities: (1) *Absolute priority*, in which a prioritized student s in school c can displace any other student with no priority, regardless of their random tie-breaker; and (2) *partial priority*, in which a prioritized student s in school c can displace only certain students with no priority.[6]

Both notions of sibling priority have implications in terms of justified-envy and, consequently, for the stability of the matching. Therefore, we formalize the concepts of *absolute* and *partial* justified-envy in Definition 5.3.6. For this, let $P_\mu(s,c) = \max_{a \in f(s)\setminus\{s\}} \{p_{a,c} \ : \ \mu(a) = c, \ a \succ_c s\}$ be the function that returns the highest random tie-breaker among the siblings of student s currently assigned to c. If the student s does not have a sibling currently assigned to c, then we define $P_\mu(s,c) = p_{s,c}$.

Definition 5.3.6 (Absolute and partial justified-envy)**.** Consider a matching $\mu \subseteq \mathcal{E}$.

- A student with sibling priority s has *absolute* justified-envy towards another student s' without sibling priority assigned to school c if (i) $g(s) = g(s')$, (ii) $c \succ_s \mu(s)$, (iii) $f(s') = \{s'\}$, and (iv) there exists a sibling $\bar{s} \in f(s) \setminus \{s\}$ such that $\mu(\bar{s}) = c$.

[6] A common approach used in practice is to assume that a prioritized student moves up in the order of the school until they meet their (highest ranked) siblings, displacing students with a random tie-breaker lower than the sibling who provided them with their priority.

- A student with sibling priority s has *partial* justified-envy towards another student s' without sibling priority assigned to school c if (i) $g(s) = g(s')$, (ii) $c \succ_s \mu(s)$, (iii) $f(s') = \{s'\}$, and (iv) $P_\mu(s,c) > p_{s',c}$.

Note that condition (iii) of both definitions requires that student s' has no siblings; this must be the case since absolute and partial priority describe how the competition between a family and an individual with no siblings ensues. If no students have siblings applying to the system, then both definitions of partial and absolute justified-envy are null and everything is reduced to the standard notion of justified-envy. The concepts of absolute and partial justified-envy allow us to compare students from different applicant groups: students with sibling priority vs. students with no priority. Hence, it remains to describe how to compare students within the same applicant group. Among students with no priority, the schools compare their random tie-breaker and justified-envy is defined as usual. Among students with sibling priority, we define two approaches: (1) the *Dependent rule* where two students s and s' that belong to different families and they have sibling priority in the same school c, then school c compares the tie-breaker of their highest-ranked sibling already matched to c; (2) the *independent rule* where two students s and s' belong to different families and they have sibling priority in the same school c, then school c compares the tie-breaker of each competing student.

In Example 5.3.7 we illustrate the dependent and independent rules.

Example 5.3.7. Consider a single school c with capacity 3, and two families, $f = \{s_1, s_2\}$, $f' = \{s_1', s_2'\}$, with all students applying to the same grade. Moreover, tie-breakers are such that $p_{s_1,c} > p_{s_1',c} > p_{s_2',c} > p_{s_2,c}$. Suppose that $\mu(s_1) = c$ and $\mu(s_1') = c$. As a result, both s_2 and s_2' get sibling priority, but there is only one seat left. If the dependent rule is in place, then $\mu(s_2) = c$ and $\mu(s_2') = \emptyset$, since $p_{s_1,c} > p_{s_1',c}$. On the other hand, if the independent rule is in place, $\mu(s_2) = \emptyset$ and $\mu(s_2') = c$, since $p_{s_2',c} > p_{s_2,c}$.

Note that the dependent and the independent rules are used in practice. On the one hand, the dependent rule is used in Chile [**50**], where the clearinghouse breaks ties at the family level first, and then breaks ties within each family. On the other hand, the independent rule is used in NYC to break ties among students with sibling priority.

Based on the dependent and the independent rule, we define two notions of justified-envy: (i) *dependent-justified envy* and (ii) *independent-justified envy*.

Definition 5.3.8 (Dependent and independent justified-envy)**.** Consider a matching $\mu \subseteq \mathcal{E}$.

- A student with sibling priority s has *dependent* justified-envy toward another student with sibling priority s' from another family assigned to a school c if (i) $g(s) = g(s')$, (ii) $c \succ_s \mu(s)$, and (iii) $P_\mu(s,c) > P_\mu(s',c)$.

- A student with sibling priority s has *independent* justified-envy toward another student s' with sibling priority from another family assigned to a school c if (i) $g(s) = g(s')$, (ii) $c \succ_s \mu(s)$, and (iii) $p_{s,c} > p_{s',c}$.

Among students with sibling priority and from the same family, the school compares their random tie-breaker and justified-envy is defined as usual. In summary, given two students s and s' from two different families, we have:

Student s	Student s'	Justified-envy of s toward s'
No dynamic priority	No dynamic priority	Standard
Dynamic priority	No dynamic priority	Absolute or partial
No dynamic priority	Dynamic priority	Absolute or partial
Dynamic priority	Dynamic priority	Dependent or independent

Given Definitions 5.3.6 and 5.3.8, we can provide four notions of stability, as we formalize in Definition 5.3.9.

Definition 5.3.9. We say that a matching $\mu \subseteq \mathcal{E}$ is *partial-dependent stable* if it is non-wasteful, and if no student has partial and dependent justified-envy. Similarly, we define *absolute-independent stable*, *absolute-dependent stability*, *partial-independent stability* and *partial-dependent stability*.

5.4. Existence

As discussed in [**135**], stability is a desirable property since it correlates with the long-term success of the matching process. Unfortunately, as we show in Propositions 5.4.1 and 5.4.2, a stable matching under any combination of dynamic priorities, according to Definition 5.3.9, may not exist.

Proposition 5.4.1. An absolute-(in)dependent stable matching may not exist, even if families are of size at most two.

PROOF. There are four schools, c_1, c_2, c_3, and c_4, and one single level. The schools c_1 and c_3 have one seat, and both the other schools have two seats. There are five families of students, $f_a = \{a_1, a_2\}$, $f_x = \{x\}$, $f_y = \{y\}$, $f_d = \{d_1, d_2\}$, $f_h = \{h\}$. The preferences of the families (and of each student) are the following, $f_a : c_3 \succ c_4$; $f_x : c_2$; $f_y : c_2$; $f_d : c_1 \succ c_2 \succ c_3$; $f_h : c_4 \succ c_1$. Every school has the same tie-breaker, i.e., the following student ordering $p_{h,c} > p_{d_1,c} > p_{x,c} > p_{y,c} > p_{d_2,c} > p_{a_1,c} > p_{a_2,c}$.

Note there is only one stable matching without sibling priority:

$$\mu = \{(a_1,c_4),(a_2,\emptyset),(x,c_2),(y,c_2),(d_1,c_1),(d_2,c_3),(h,c_4)\}.$$

Clearly, every other matching different from μ in which two siblings are not matched together, is not stable. Notice that the only matchings that may be stable according to sibling priority are those that would match a_1, a_2 in school c_4 (a_1 providing priority to a_2) or d_1, d_2 in school c_2 (d_1 providing priority to d_2).

First, assume we have a matching where a_1 provides priority to a_2 in c_4. a_1,a_2 both prefer c_3 over c_4, so c_3 must be full. But f_d is the only other family that finds c_3 acceptable. Suppose d_i (for $i = 1$,2) is in c_3. Then, the other sibling in f_d cannot be unmatched, otherwise both d_1,d_2 would prefer c_2 over their current assignment, and c_2, with two seats, ranks d_1 second (and h does not rank c_2). Additionally, the other sibling d_j ($j \neq i$) cannot be matched in c_2, otherwise it would provide a priority to d_i, who would prefer to be matched to c_2 rather than c_3. Therefore, d_j must be matched to c_1, but then h has justified envy towards d_j at c_1.

Now assume that d_1 and d_2 are matched together in c_2. They both prefer c_1, so c_1 must be full. Thus, h must be matched with c_1. Since h prefers c_4 and has highest priority at c_4, it must be the case that both a_1 and a_2 are matched with c_4. But this is then wasteful as c_3 is unmatched and is the first choice of family f_a.

□

Proposition 5.4.2. A partial-(in)dependent stable matching may not exist, even if families are of size at most two and there at most two grade levels.

PROOF. There are four schools, c_1, c_2, c_3, and c_4, and two levels g_1 and g_2. At level g_1, schools c_1 and c_3 have one seat, and all the other schools have two seats. At level g_2, c_1 has one seat, and all the other schools have zero seats. There are five families of students, $f_a = \{a_1, a_2\}$, $f_x = \{x\}$, $f_y = \{y\}$, $f_d = \{d_1, d_2\}$, $f_h = \{h_1, h_2\}$. All the students, except for h_2, apply to level g_1. The preferences of the students (which are the same for both levels) are the following, $f_a : c_3 \succ c_4$; $f_x : c_2$; $f_y : c_2$; $f_d : c_1 \succ c_2 \succ c_3$; $f_h : c_4 \succ c_1$. The random tie-breakers are the same for all schools and lead to the following student ordering $p_{h_2,c} > p_{d_1,c} > p_{x,c} > p_{y,c} > p_{d_2,c} > p_{a_1,c} > p_{h_1,c} > p_{a_2,c}$.

Note there is only one stable matching without sibling priority:

$$\mu = \{(a_1,c_4),(a_2,\emptyset),(x,c_2),(y,c_2),(d_1,c_1),(d_2,c_3),(h_1,c_4),(h_2, c_1)\}.$$

Clearly, every other matching different from μ in which two siblings are not matched together, is not stable. Notice that the only matchings that may be stable according to sibling priority are those that would match a_1, a_2 in school c_4 (a_1 providing priority to a_2) or d_1, d_2 in school c_2 (d_1 providing priority to d_2) or h_1, h_2 in school c_1 (h_2 providing priority to h_1).

First, assume we have a matching where a_1 provides priority to a_2 in c_4. Note that a_1,a_2 both prefer c_3 over c_4, so c_3 must be full. But f_d is the only other family that finds c_3 acceptable. Suppose d_i (for $i = 1$,2) is in c_3. Then, the other sibling in f_d cannot be unmatched, otherwise both d_1,d_2 would prefer c_2 over their current assignment, and c_2, with two seats, ranks d_1 second (and h_2 does not rank c_2). Additionally, the other sibling d_j ($j \neq i$) cannot be matched in c_2, otherwise it would provide a priority to d_i, who would prefer to be matched to c_2 rather than c_3. Therefore, d_j must be matched to c_1, but then h_1 has justified envy towards d_j at c_1 since it receives priority from h_2.

Now assume that d_1 and d_2 are matched together at c_2. They both prefer c_1, so c_1 must be full. Thus, h_1 must be matched with c_1. Since h_1 prefers c_4 and has higher priority at c_4 than a_2 (there are only three students that rank c_4 at level g_1: h_1,a_1,a_2), it must be the case that both a_1 and a_2 are matched with c_4. But this is then wasteful as c_3 is unmatched and is the first choice of family f_a.

Finally, assume that h_1 and h_2 are matched together at c_1. h_2 can only be matched at c_1, while h_2 would prefer to be matched with c_4. Therefore, c_4 must be matched with a_1,a_2. But this is then wasteful as c_3 is unmatched and is the first choice of family f_a.

□

One important factor for the non-existence of a partial-(in)dependent stable matching is that priorities across different grade levels can go in any possible direction, i.e., there may be families where the provider of sibling priority is at a lower level and others where the provider is at a higher level. In order to mitigate this, some clearinghouses may impose additional rules, such as the one used in Chile, where there is a specified order (e.g., decreasing) in which grade levels are processed and, consequently, sibling priorities can only move according to that order (e.g., providers are in higher grades and receivers in lower ones). However, as we show in Proposition 5.4.3, the non-existence results hold even if we define an order in which priorities move across grade levels.

Proposition 5.4.3. A partial-(in)dependent stable matching may not exist, even if families are of size at most two and there is a fixed order in which siblings provide priorities between grade levels, and there is a single tie-breaker, i.e., $p_{s,c} = p_s$ for all $c \in \mathcal{C}$. This non-existence result also holds for absolute-(in)dependent stability.

Proof. Let Γ be the instance provided in the proof of Proposition 5.4.2 with the two levels g_1 and g_2. We create another instance Γ' which is a copy of Γ where the families and the schools have different names. Moreover, the names of the two levels are switched and the agents of Γ do not rank those of Γ' and vice-versa. When we juxtapose Γ and Γ' to create a new instance Γ'', we find that for any priority ordering between grade levels g_1 and g_2 there is no partial-(in)dependent stable matching. □

5.4.1. Guaranteed Existence under Refined Family Preferences

An alternative explanation for the non-existence results described in the previous section is that, although students may benefit from the sibling priority, they still aim to be allocated in their most preferred school, regardless of the matching of their siblings. However, in many cases, the primary goal of the families is to get their siblings assigned to the same school. Indeed, some school districts may define as infeasible matchings where siblings are separated [**59**]. In other cases, the clearinghouse may explicitly elicit whether the family wants to prioritize the joint assignment of their siblings over their individual preferences. For instance, the Chilean school choice system allows families to submit a *family application*, whereby the family states that they prefer their siblings to be assigned to the same school over any assignment where this does not happen, even if the siblings end up being assigned together in a lower preferred school (see [**50**] for more details).

To capture these settings, we assume that families lexicographically prefer that their members are assigned to the same school over any other assignment where they are separated. For instance, if family f includes first school c and then c' in their list, then the actual preferences of each member of $s \in f$ can be written as $(c,c) \succ_s (c',c') \succ_s c \succ_s c'$, where the tuples represent that student s prefers to be assigned with at least one sibling and the non-tuples refer to individual preferences. We formalize this in Assumption 5.4.4.

Assumption 5.4.4. Students preferences are lexicographic, so that they first prefer to be assigned with at least one of their siblings, and then to be individually assigned to the schools reported in the family list.

Algorithm 7 Direct matching mechanism

Initialize: $H = \mathcal{S}$, $\mu = \emptyset$
while $H \neq \emptyset$ **do**
  Find: $s^\star = \text{argmax}_{s \in H} \{p_s\}$, $f^\star = f(s^\star) \cap H = \{s^\star, s^\diamond\}$, $g^\star = g(s^\star)$ ▷ Note: $s^\diamond = \emptyset \Leftrightarrow f^\star = \{s^\star\}$
  Initialize: $\underline{c} = \emptyset$
  for $c \in \succ_{f^\star}$ **do** ▷ In decreasing order of pref.
    if $\left|\mu^{g^\star}(c)\right| < q_c^{g^\star}$ **then**
      if $\underline{c} = \emptyset$ **then**
        Update: $\underline{c} \leftarrow c$
      if $s^\diamond = \emptyset$ **then**
        Update: $\mu \leftarrow \{(s^\star, c)\}$, $H \leftarrow H \setminus \{s^\star\}$
        break
      if $s^\diamond \neq \emptyset$ and $\left|\mu^{g(s^\diamond)}(c)\right| < q_c^{g(s^\diamond)}$ **then**
        Update: $\mu \leftarrow \{(s^\star, c), (s^\diamond, c)\}$, $H \leftarrow H \setminus f^\star$
        break
  if $s^\star \in H$ **then**
    Update: $\mu \leftarrow \{(s^\star, \underline{c})\}$, $H \leftarrow H \setminus \{s^\star\}$
return μ.

Note that this assumption only restricts the space of families' preferences and, thus, does not affect the different definitions of stability under dynamic priorities stated in Definition 5.3.6.

This assumption turns to be crucial for establishing the existence of a matching with dynamic priorities. Before we present the theoretical result, we present the mechanism that finds such a matching. This mechanism, outlined in Algorithm 7, extends the Random Serial Dictatorship (RSD) algorithm [**7**] to jointly assign siblings if there is enough capacity to accommodate them. Specifically, the algorithm iterates over students in decreasing order of their random tie-breaker (Step 3). If a student s has no siblings (Step 9), the algorithm matches s to their most preferred school among those with seats left (Step 10). If a student s has a sibling (Step 12), then the algorithm stores (in $\underline{c}$) s's most preferred school with seats left in $g(s)$ and then tries to jointly assign the family $f(s)$ in order of their preferences: If school c has seats left in both grades, then both siblings are assigned to c (Step 13); in contrast, if no such school exists, student s is assigned to $\underline{c}$ (Step 16).

As we show in Theorem 5.4.5, if families' preferences follow Assumption 5.4.4 and families are of size at most two, then a partial-dependent stable matching exists.

Theorem 5.4.5. A partial-dependent stable matching exists when families are of size at most two, their preferences satisfy Assumption 5.4.4 and there is a single tie-breaker, i.e., $p_{s,c} = p_s$ for all $c \in \mathcal{C}$. Moreover, such a matching can be found using Algorithm 7 in $O(|\mathcal{S}|^2 \log |\mathcal{S}| + |\mathcal{S}| \cdot |\mathcal{C}|)$.

PROOF. Since the number of agents is finite and at least one student is removed from H in each iteration, we know that Algorithm 7 finishes. Our proof consists in demonstrating the following statement by induction: At the end of every while iteration, the matching μ is such that no student in $\mathcal{S} \setminus H$ has partial-dependent justified envy.

Basis. At the end of the first iteration, one of the following three cases holds: (i) $\mu(s^\star) = \emptyset$, (ii) $\mu(s^\star) \neq \emptyset$ and $\mu(s') = \emptyset$ for all $s' \in \mathcal{S} \setminus \{s^\star\}$, or (iii) $\mu(s^\star) = \mu(s^\diamond) \neq \emptyset$. In the first case, it means that there is no school listed by $s^\star$ that has an open seat in grade $g^\star$ and, thus, $s^\star$ has no justified-envy as no school has seats open. In the second case, $s^\star$ is matched to their most preferred school with seats left in $g^\star$; if $s^\diamond = \emptyset$, then $s^\star$ has no justified envy because there are no schools that $s^\star$ prefers and that have open seats. If $s^\diamond \neq \emptyset$, then it means that there is no school with seats open in $g(s^\diamond)$ and, thus, the family cannot have dependent justified envy. In the last case, family $f^\star = \{s^\star, s^\diamond\}$ is matched to their most preferred school that can accommodate both siblings at their respective grades. Note that the students in $f^\star$ cannot have justified-envy because the schools they prefer do not have enough seats to accommodate both siblings, and they prefer to be matched together over being separated.

Inductive step. Suppose that after n iterations of the while loop, no student in $\mathcal{S} \setminus H$ has partial-dependent justified envy. We need to show that this is also true at the end of the $n+1$ iteration. If $H = \emptyset$, then this holds by inductive hypothesis. Otherwise, let $s^\star$ be the student in H with the highest random tie-breaker. Then, we start searching for the acceptable schools starting from the most preferred. As soon as we find a school $\underline{c}$ with an open seat in grade $g^\star$

(if any), we record it. If none of the schools listed by family $f^\star$ have an open seat for grade $g^\star$, then $s^\star$ remains unassigned and has no justified envy towards any other matched student since they all have higher tie-breaker. Hence, we assume $\underline{c} \neq \emptyset$. If $s^\star$ has no siblings, then we simply add $(s^\star,\underline{c})$ to μ. Since all the other students previously assigned have a higher random tie-breaker than $s^\star$ and $\underline{c}$ is the most preferred school by $s^\star$ among those with seats left, then $s^\star$ cannot have justified-envy. Finally, if $s^\star$ has a sibling and school $\underline{c}$ cannot accommodate both siblings, we continue to the next preference of the family. If there is no school that can accommodate both siblings in $f^\star$, then we match $s^\star$ to $\underline{c}$ while $s^\diamond$ remains unassigned (recall that, by construction, $s^\diamond \in H$ and, thus, $p_{s^\star} > p_{s^\diamond}$). As before, $f^\star$ cannot have partial justified envy because there is no school that can accommodate both $\{s^\star, s^\diamond\}$ and all students previously assigned have a higher random tie-breaker than both siblings. Otherwise, if there is a school $\tilde{c} \preceq_{s^\star} \underline{c}$ with two open seats for the siblings in $f^\star$, then the algorithm matches both students to $\tilde{c}$. By Assumption 5.4.4, we know that student $s^\star$ prefers being assigned with their sibling in $\tilde{c}$ over being separated from their sibling and being assigned to $\underline{c}$, and we also know that $s^\diamond$ prefers to be matched with $s^\star$ over being separated. Finally, since there is no school they prefer that can accommodate both of them and all previously assigned students have a higher random tie-breaker, we conclude that family $f^\star$ cannot have partial-dependent justified envy.

So far, we have shown that at the end of the while loop we obtain a partial-dependent justify envy free matching μ. To show that this matching is stable, it remains to show that μ is non-wasteful. To find a contradiction, suppose it is not. Then, there exists a pair (s,c) such that $s \succ_c \emptyset$, $c \succ_s \mu(s)$, and $\left|\mu^{g(s)}(c)\right| < q_c^{g(s)}$. If s has no siblings, then we know that c had seats open in grade $g(s)$ in the iteration where s was assigned (because $\left|\mu^{g(s)}(c)\right|$ is non-decreasing in the iterations), so this leads to a contradiction as the algorithm would have assigned s to c. If s has a sibling s', there are two cases. If $\mu(s) \neq \mu(s')$, then it means that there was no school in the family's list that could accommodate both siblings, and thus they were separated. In that case, the algorithm would assign student s to school $\underline{c}$, i.e., their most preferred school with open seats. Since c had opened seats in that iteration, it means that $\underline{c} \succeq_s c \succ_s \mu(s)$, and thus s should have been assigned to $\underline{c}$. Finally, if $\mu(s) = \mu(s') \neq \emptyset$, then s prefers being assigned to c over $\mu(s)$ only if s can get assigned there with s'. However, this did not happen because, when processing student s, school c had no seats left in grade $g(s')$ (otherwise, we would have $\mu(s) = \mu(s') = c$) and, thus, given Assumption 5.4.4, it would not be true that s prefers to be assigned in c over their current assignment $\mu(s)$.

To conclude, note that in the worst case every student has no siblings and needs to apply to every school. Recall that a set of size $|\mathcal{S}|$ can be sorted in $O(|\mathcal{S}| \cdot \log|\mathcal{S}|)$. Steps 2-17 will be done at most $|\mathcal{S}|$ times. Step 3 takes at most $O(|\mathcal{S}| \cdot \log|\mathcal{S}|)$ and Steps 5-14 $O(|\mathcal{C}|)$. Thus, $O(|\mathcal{S}| \cdot (|\mathcal{S}| \cdot \log|\mathcal{S}| + |\mathcal{C}|)) = O(|\mathcal{S}|^2 \log|\mathcal{S}| + |\mathcal{S}| \cdot |\mathcal{C}|)$. □

Note that the proof of Theorem 5.4.5 is constructive, as it provides a mechanism that allows finding a stable matching under dynamic preferences. If there are families of size larger than two, a partial-dependent stable matching may not exist, as Proposition 5.4.6 illustrates. Nevertheless, considering that the vast majority of families that participate in these systems involve at most two siblings,[7] and that families generally prefer that their members go to the same school, this result is of high practical value.

Proposition 5.4.6. A partial-(in)dependent stable matching may not exist, even if families are of size at most three, there are at most two grades, student preferences satisfy Assumption 5.4.4, and there is a single tie-breaker, i.e., $p_{s,c} = p_s$ for all $c \in \mathcal{C}$.

PROOF. Consider an instance of the problem with five schools, c_1, c_2, c_3, c_4, and c_5, and two grade levels $\{g_1,g_2\}$. In g_1, c_1, c_3 and c_5 have capacity one, and all the other schools have capacity two. In g_2, c_3 has capacity one, and all the other schools have capacity zero. In addition, suppose there are five families of students, $f_a = \{a\}$, $f_b = \{b_1, b_1', b_2\}$, $f_e = \{e\}$, $f_d = \{d_1, d_1'\}$, $f_h = \{h\}$. All the students, except for b_2, apply to level g_1. The preferences of the students (which are the same for both levels) are the following, $f_a : c_1 \succ c_2$; $f_b : c_2 \succ c_3$; $f_e : c_4 \succ c_1$; $f_d : c_5 \succ c_4$; $f_h : c_3 \succ c_4$, and the random tie-breakers are such that $p_{b_2} > p_h > p_{d_1} > p_e > p_a > p_{b_1} > p_{b_1'} > p_{d_1'}$.

Note there is only one stable matching without sibling priority, namely $\mu = \{(a,c_1), (b_1,c_2), (b_1',c_2), (b_2,c_3), (e,c_4), (d_1,c_4), (d_1',c_5), (h,c_3)\}$. If we then try to sequentially adjust μ following the preferences and stability assumptions of Theorem 5.4.5, we return to μ. Moreover, any other possible matching with sibling priority is not partial-dependent stable since the only two families that have siblings are f_b and f_d. Family f_b can have siblings matched together only in schools c_2 and c_3, while family f_d can have siblings matched together only in school c_4. Both cases are covered in the dynamics that begins from matching μ. Note that this example still holds if we consider adaptive stability under partial independent justified-envy. □

As we formalize in Theorem 5.4.7, another case in which we can guarantee existence is when there is a single grade level. This case is also of practical relevance, since it would allow us to account for dynamic priorities in the presence of twins, and it would also capture other relevant settings such as daycare and refugee resettlement, which could be thought as having a single grade level.

Proposition 5.4.7. A partial-dependent stable matching exists when there is a single grade level, families preferences satisfy Assumption 5.4.4 regardless of the family sizes, and there is a single tie-breaker, i.e., $p_{s,c} = p_s$ for all $c \in \mathcal{C}$. Such a matching can be found in $O(|\mathcal{S}|^2 \log |\mathcal{S}| + |\mathcal{S}| \cdot |\mathcal{C}|)$.

[7] In Chile, less than 5% of families involve more than two siblings simultaneously participating in the assignment process.

PROOF. The proof follows a similar reasoning as the one of Theorem 5.4.5, provided that Algorithm 7 is updated as follows. First, in Step 3, the set $f^\star$ may have more than two members. In Step 13, the algorithm may try to match as many of the siblings as possible to school c, provided that it has open seats in the corresponding grade levels. This operation can be done safely because there is a single tie-breaker, siblings have the same preferences, and we use the dependent rule to break ties among students with sibling priority. Therefore, the algorithm does not cycle, which indeed may happen in the case of families of size three on multiple levels. $\square$

5.5. Complexity Analysis: Maximum Cardinality

In this section, we analyze the computational complexity of finding a partial-dependent stable matching that is of maximum cardinality, when preferences satisfy Assumption 5.4.4. In this setting, there may be partial-dependent stable matchings of different cardinalities, as we show in the following example.

Example 5.5.1. Suppose that there is one grade level, two schools c_1 and c_2 with capacities three and two, respectively, and four families $f_a = \{a\}, f_b = \{b\}, f_d = \{d\}, f_e = \{e_1, e_2\}$. The preferences of the families are $f_a : c_2 \succ c_1$, $f_b : c_1$, $f_d : c_1$, $f_e : c_1 \succ c_2$, while the random tie-breakers are as follows: for c_1 we have $p_{a,c_1} > p_{b,c_1} > p_{e_1,c_1} > p_{d,c_1} > p_{e_2,c_1}$ and for c_2 we have $p_{e_1,c_2} > p_{e_2,c_2} > p_{a,c_2}$ (f_b and f_d do not apply to c_2, so we do not need tie-breakers for them). Note that we can find two partial-dependent stable matchings of different cardinality: $\mu' = \{(a,c_1), (b,c_1), (d,c_1), (e_1, c_2), (e_2, c_2)\}$ and $\mu'' = \{(a_1,c_2), (b,c_1), (d,\emptyset), (e_1, c_1), (e_2, c_1)\}$.

As the previous example illustrates, the Rural Hospital Theorem [**132, 134**] does not hold in our framework. Therefore, it is essential to understand the complexity of finding the maximum cardinality partial-dependent stable matching. We now formalize the problem of finding such a matching.

Problem 12. Let $\Gamma = \langle \mathcal{S},\mathcal{C},\mathcal{F}, \succ, \mathbf{c}, \{p_{s,c}\}_{s\in\mathcal{S},c\in\mathcal{C}}\rangle$ be an instance of the school choice problem with families, incomplete preference lists for the families, and let $K \in \mathbb{Z}_+$ be a non-negative integer target value. Is there a matching μ such that $|\mu| \geq K$, and μ is a partial-dependent stable matching in Γ?

We denote this problem as $\textsc{Max-Card}_{\text{FAM}}$.

In Theorem 5.5.2, we show that Problem 12 is NP-complete, and we defer the proof to Appendix 5.8. Note that this result also holds if we restrict attention to partial-independent stable matchings.

Theorem 5.5.2. Problem 12 is NP-complete, even if there are families of size at most three and two grade levels.

5.6. Conclusions

Motivated by the context of school choice with sibling priorities, we study the problem of finding a stable matching under dynamic priorities, i.e., students get prioritized if they have siblings participating in the process and who are currently assigned. We start by introducing a model of a matching market where siblings may apply together to potentially different grade levels. We argue that the standard notion of stability may not work if we allow priorities to (adapt) be a function of the matching. As a result, we define a series of assumptions on families' preferences and school priorities tailored to follow existent practices, and we introduce several novel notions of stability under dynamic priorities. Although we show that a stable matching may not exist in general, such a matching exists if families strictly prefer that their members remain together over being separated and that families are of size at most two. Moreover, we show that a stable matching also exists when there is a single grade level for any family size. Finally, we show that finding a maximum cardinality stable matching under dynamic priorities is NP-hard.

Our results show that dynamic priorities must be carefully designed to ensure the existence of a stable assignment. Moreover, several design choices have relevant implications, namely, how to break ties within and across priority groups and how families prioritize the joint assignment of their members. Hence, the insights derived from our work may help design these policies, either in the context of school choice or in other contexts, such as daycare assignments, and refugee resettlement, among others.

Our work opens several directions for future research. First, we are working on extending our existence and complexity results to the other notions of stability introduced in our work. Second, we are working on how to efficiently solve the problem of finding a stable matching under dynamic priorities using mathematical programming tools. Finally, we are collaborating with the Ministry of Education of Chile to showcase the potential benefits of considering dynamic priorities when solving the assignment of students to schools.

Appendix

5.7. Extra discussion on how to process grade levels and others

As proposed in [**50**], one option to handle dynamic priorities is to define an order in which grades are processed and sequentially solve the assignment of each grade level using the student-optimal variant of DA. More specifically, the algorithm in [**50**] starts processing the highest grade (i.e., 12th grade). Then, before moving to the next grade, the sibling priorities are updated, considering the assignment of the grade levels already processed. After processing the final grade level (i.e., Pre-K), this procedure finishes. Notice that this heuristic obtains a stable assignment if the preferences of families satisfy *higher-first*, i.e., each family prioritizes the assignment of their oldest member (see Proposition 2 in [**50**]). However, this is not the case if some families' preferences do not satisfy this condition. In addition, as Example 5.7.1 illustrates, the order in which grades are processed matters.

Example 5.7.1. Consider an instance with two grades $g_1 < g_2$, two schools c_1 and c_2 with one seat in each grade, one family $f = \{f_1,f_2\}$, and two additional students, a_1 and a_2. Students f_1 and a_1 apply to grade g_1, and f_2 and a_2 apply to grade g_2. Finally, the preferences and priorities are:

$$\begin{aligned}
&(c_2,c_1) \succ_f (c_1,c_1) \succ_f (c_2,c_2) \succ_f (c_1,c_2)\\
&c_2 \succ_{a_1} c_1\\
&c_1 \succ_{a_2} c_2\\
&p_{a_1,c_1} > p_{f_1,c_1} \text{ and } p_{a_2,c_1} > p_{f_2,c_1}\\
&p_{a_1,c_2} > p_{f_1,c_2} \text{ and } p_{a_2,c_2} > p_{f_2,c_2}.
\end{aligned} \tag{5.7.1}$$

Since the preferences $\succ_f$ are *responsive*, we can easily derive the related individual preferences $\succ_{f_1}$ and $\succ_{f_2}$, which are $c_2 \succ_{f_1} c_1$ and $c_1 \succ_{f_2} c_2$ [**94, 93**]. We observe that, if grades are processed in decreasing order (as in Chile), we obtain the matching $\mu = \{(f_1, c_2), (a_1, c_1), (f_2, c_2), (a_2, c_1)\}$. In contrast, if we process grades in increasing order, we obtain the matching $\mu' = \{(f_1, c_1), (a_1, c_2), (f_2, c_1), (a_2, c_2)\}$. $\square$

5.8. Missing Proofs in Section 5.5

Our reduction is done from the problem of finding a maximum cardinality stable matching in a market where schools are partitioned in sets and each set of schools receives some extra seats that should be allocated optimally. Note that a similar proof can be given for partial-independent priorities.

Problem 13. Let $\Gamma = \langle \mathcal{S},\mathcal{C},\mathcal{F}, \succ, \mathbf{c}, \{p_{s,c}\}_{s\in\mathcal{S},c\in\mathcal{C}}\rangle$ be an instance of the school choice problem with families, incomplete preference lists for the families, a partition $\mathcal{P} = \{\mathcal{C}_1, \ldots ,\mathcal{C}_q\}$ of $\mathcal{C}$, budget for each part $\{B_k \in \mathbb{Z}_+ \colon k \in [q]\}$, and let $K \in \mathbb{Z}_+$ be a non-negative integer target value. Is there a non-negative allocation vector $\mathbf{t} \in \mathbb{Z}_+^{\mathcal{C}}$ and a matching $\mu_{\mathbf{t}}$ such that $|\mu_{\mathbf{t}}| \geq K$, where $\mathbf{t}$ is such that $\sum_{j\in\mathcal{C}_k} t_j \leq B_k$ for each $k \in [q]$ and μ is a stable matching in the expanded instance Γ_t?

We denote this problem as $\textsc{Max-Card}^{\text{SUB}}_{\text{EXP}}$ HRI.

Proof of Theorem 5.5.2. We provide a reduction from $\textsc{Max-Card}^{\text{SUB}}_{\text{EXP}}$ HRI [**40**], the problem of finding a maximum cardinality stable matching in a instance where preferences may be incomplete, schools are partitioned into subsets and every such set has a budget to expand schools' capacities. From the proof of Theorem 5.2 in [**40**], we deduce the following assumptions on the generic instance of $\textsc{Max-Card}^{\text{SUB}}_{\text{EXP}}$ HRI.

- The partition of schools $\mathcal{P} = \{C_1, \ldots, C_p\}$ is made of subsets of size at most two and for each such C_k there is an extra budget $B_k \leq 1$ for $k \leq l$.
- Every school that is partitioned as a singleton has capacity one and an extra capacity zero. We denote by $\mathcal{C}^\star$ the set of these schools.
- Every school that is partitioned in a subset of cardinality two (C_k), has capacity zero; the budget of extra capacities allocated to such pair of schools is one, $B_k = 1$. We denote by $\mathcal{C}^{\star\star}$ the set of schools with capacity zero; for simplicity we assume that C_i with $i \leq t$ we have the schools with capacity zero. Every school with capacity zero has a preference list of length one, i.e., it ranks only one student; moreover, we make the crucial assumption that one of the two students ranked by the pair of schools is ranked only once by a pair of schools.[8]

The objective in instance Γ is to find a stable matching of cardinality at least K. Given an instance Γ of Problem 12, we build an instance Γ' of $\textsc{Max-Card}_{\textsc{fam}}$ with two grades, g_1, g_2, and families of size at most three. First of all, we create a copy of $\mathcal{S}$ in Γ'. For every school c in $C^\star$ we create a copy of c in Γ' with the same preference list and the same capacity at level g_1. For a school c in $\mathcal{C}^{\star\star}$, let $C_r = \{c'_r, c''_r\}$ be the subset of the partition $\mathcal{P}$ to which c belongs. Assume the preference lists of c'_r, c''_r in Γ are $c'_r : s'_r$ and $c''_r : s''_r$, respectively. In Γ' we create two new schools $\bar{c}_r$ and c^+_r (note that we make no copies of c'_r and c''_r) and three new students y'_r, y''_r and w_r. In Γ', students s'_r, s''_r, w_r and y'_r apply to level g_1, while y''_r applies to level g_2. All the other students in Γ that are not ranked by a school in $\mathcal{C}^{\star\star}$, apply in Γ' to grade g_1. On the other hand, school c^+_r has only one capacity at level g_1 and one capacity at level g_2; school $\bar{c}_r$ has two capacities at level g_1 and no seats at level g_2. We assume s''_r is the student that is only ranked once by a pair of schools in Γ, and we create the family $f_r = \{s''_r, y'_r, y''_r\}$ in Γ'. The preferences of these agents in Γ' are as follows.

- $\bar{c}_r : y'_r \succ s'_r \succ w_r \succ s''_r \succ y''_r$.
- $c^+_r : w_r \succ y'_r \succ y''_r \succ s''_r$.
- $w_r : \bar{c}_r \succ c^+_r$.
- In the original preference list of s''_r we substitute c'_r with $c^+_r \succ \bar{c}_r$. Therefore, the preference list of f_r is the same as s''_r. All the schools ranked by f_r that are not $\{c^+_r, \bar{c}_r\}$, they rank y'_r and y''_r last.
- In the original preference list of s'_r we substitute c''_r with $\bar{c}_r$.

[8]In [**40**],the authors prove that $\textsc{Max-Card}^{\textsc{sub}}_{\textsc{exp}}$ HRI is NP-hard by proving a reduction from the problem of finding the maximum cadinality stable matching with ties and incomplete lists [**108**]. The assumption that we made (the fact that each school in a pair lists only one student), follows from the fact that in the proof of Theorem 2 in [**108**], the preference list of $x_{i,r}$ is only made by a tie of length two. The second assumption (the fact that one of the two students ranked by the pair is only listed by one pair) follows from the fact that $w_{i,r}$ is only ranked by $x_{i,r}$ in the proof of Theorem 2 in [**108**].

From the assumption that s''_r is ranked only once by a pair in Γ, it follows that s''_r has only two siblings, which are exactly y'_r and y''_r. Note that c^+_r and y''_r will always be matched together. Let l be the number of paired sets from $\mathcal{P}$ in Γ (i.e., half the number of schools in $\mathcal{C}^{\star\star}$).

Let M be a stable matching in Γ of cardinality at least K. Our goal is to find a corresponding partial-dependent stable matching M' in Γ' that has a cardinality at least $K+3\cdot l$. Let (c,s) be a pair in M. If c is in $\mathcal{C}^\star$, then we match (c,s) in M'. Otherwise, c is part of a pair $C_r = \{c'_r, c''_r\}$. If $c = c'_r$, then we match $(\bar{c}_r,s'_r)$, $(\bar{c}_r, w_r)$, (c^+_r, y''_r) and y'_r with a school ranked at least as c^+_r. On the other hand, if $c = c''_r$, then we match $(\bar{c}_r,s''_r)$, $(\bar{c}_r, y'_r)$, (c^+_r, w_r), and (c^+_r, y''_r). Finally, it may happen that some schools in $\mathcal{C}^{\star\star}$ may be under-subscribed in Γ; in this case, we match $(\bar{c}_r, w_r)$, (c^+_r, y''_r), and y'_r with a school ranked at least as c^+_r. We need to prove that the matching M' that we just built is partial-dependent stable in Γ'. First, the pairs involving schools in $\mathcal{C}^\star$ directly inherit the stability from M. Note that the set of students $\{y''_r, y'_r\}$ will always be matched: As mentioned earlier, y''_r will always be matched to c^+_r, while y'_r will be matched to a school that is ranked at least as $\bar{c}_r$. Indeed, y'_r can be matched to c^+_r, $\bar{c}_r$ or a school more preferred than these, which is under-subscribed in matching Γ. Similarly, also student w_r is always matched to the set of schools $\{\bar{c}_r, c^+_r\}$. Clearly, y''_r cannot have partial-dependent justified envy because it is assigned to the only acceptable school with a spot in g_2. If s'_r is matched to $\bar{c}_r$, then s''_r cannot have partial-dependent justified envy because it is matched or to a better school than $\bar{c}_r$, or because it does not receive priority from y'_r (which, in this case, is matched to c^+_r - recall that $y'_r \succ_{c^+_r} s''_r$). If, instead, s''_r is matched to $\bar{c}_r$, then s'_r, w_r cannot have partial-dependent justified envy because s''_r uses the priority obtained by y'_r. Also y'_r cannot have partial-dependent justified envy: c^+_r is matched to its preferred student (w_r) and all the schools more preferred than $\bar{c}_r$, prefer s''_r to y'_r. Finally, note that if neither s'_r nor s''_r are matched to $\bar{c}_r$, then none of the students in the pairs $(\bar{c}_r, w_r)$, (c^+_r, y''_r), and $(y'_r, \tilde{c})$ (where $\tilde{c}$ is a school ranked at least as c^+_r) may have partial-dependent justified envy. Therefore, we have built a partial-dependent stable matching in Γ. Note that for every pair of schools in $\mathcal{C}^{\star\star}$, we have introduced three new students that are always matched. Hence, we obtained a matching M' of cardinality $K+3\cdot l$.

Let M' be a partial-dependent stable matching in Γ' of cardinality K'. Given $r \leq l$, w_r will always be matched to a school in $\{\bar{c}_r, c^+_r\}$, y''_r will always be matched to c^+_r, and y'_r to a school ranked at least as $\bar{c}_r$. Therefore, $3\cdot l$ students will always be matched in M'. Our goal is to find a corresponding stable matching M in Γ of cardinality at least $K = K' - 3\cdot l$. For every student s matched to a school c in $C^\star$ in Γ', we match (s,c) in Γ. On the other hand, if a student s is matched to a school $\bar{c}_r$ for a certain $r \leq l$, then, if $s = s'_r$, in Γ we match (s'_r,c'_r) by allocating one extra capacity to c'_r; otherwise, if $s = s''_r$, in Γ we match (s''_r,c''_r) by allocating one extra capacity to c''_r. If some $\bar{c}_r$ does not match any student of the form s'_r, s''_r, then both students are matched to a school they rank better; hence, in Γ, we can allocate the extra spot arbitrarily to c'_r and c''_r. We now prove the following statement: If some $\bar{c}_r$ does not match any student of the form s'_r, s''_r, then both students are matched to a school they rank better. If the previous statement is true, then the matching M in Γ is stable since it mimics the stability of M' without needing the students of the form $\{y'_r, y''_r, w_r\}_{r \leq l}$ which only make sure we can do the other inclusion; hence, we can obtain M as a restriction of matching M' in Γ' without considering the students $\{y'_r, y''_r, w_r\}_{r \leq l}$. To prove the statement, let us assume that there is $r \leq l$ for which s'_r and s''_r are not matched to $\bar{c}_r$ and at least one of the two students is matched to a school ranked worst. Assume s'_r is matched to $\tilde{c}$ such that $\bar{c}_r \succ_{s'_r} \tilde{c}$. Then, $(s'_r, \bar{c}_r)$ is a blocking pair in Γ' since $\bar{c}_r$ has capacity two and s'_r is ranked second by $\bar{c}_r$. Therefore, s'_r is matched to a school more preferred than $\bar{c}_r$. Now let us assume that s''_r is matched to $\tilde{c}$ such that $\bar{c}_r \succ_{s''_r} \tilde{c}$. Then, $(s''_r, \bar{c}_r)$ is a blocking pair in Γ' for one of the following two reasons: 1) y'_r is matched to $\bar{c}_r$ and s''_r can use sibling priority, 2) there is one empty capacity in $\bar{c}_r$ therefore the condition of non-wastefulness is not met. Finally, note that the students matched in Γ' that are not of the form $\{y'_r, y''_r, w_r\}_{r \leq l}$ are exactly K, which is the number of students that we matched in M.

Note that the proof holds even if we set $K = n$, i.e., we are looking for a perfect stable matching.

To build our new instance, we have introduced $3 \cdot l$ new students, and we have substituted l schools with $2 \cdot l$ new schools. Since $l \leq |\mathcal{S}|$, the reduction that we built is polynomial.

□

Chapter 6

Conclusions and Future Work

To establish a matching market, we require several essential components: The ensemble of participating agents, the permissible pairings between them, their capacities (limiting their engagements), and the rankings reflecting each agent's pairing preferences. These fundamental elements form the very essence of defining a matching market. These components collectively form the essential framework for understanding and modeling matching markets, allowing us to analyze and optimize allocation processes (matching mechanisms) in various real-world scenarios. In the realm of education admission systems, numerous pertinent real-world challenges remain elusive within the confines of existing matching market formalisms. As a compelling illustration, consider the impending scenario projected for the year 2029, where an unprecedented surge in student applications to the higher education institutions of Québec is anticipated [**48**]. Alas, a disheartening predicament looms: The grim prospect that approximately 40,000 aspiring scholars may find themselves without the coveted opportunity for enrollment, a predicament that underscores the urgency for novel methodologies and solutions within the domain of education admission.

In this thesis, we formulate and address the following questions. How should scarce resources be allocated to expand in the most impactful way the higher education system? There are also cases in which school redistricting is necessary for a reduction in the application rate. How should a policymaker reduce the education budget or incorporate several schools while affecting the least the education system? In general, how can we allow the highest number of students to access education? And, how should we promote academic merit? Siblings often apply together; how can we match family members to the same school for the benefit of families?

In the conventional approach of the stable matching problem, capacities and priorities are typically regarded as input parameters. In this thesis, we depart from this conventional perspective and offer fresh, conceptually sound paradigms that can inform policymaking strategies for education systems. The theme of this thesis revolves around the examination of many-to-one matching scenarios wherein capacities and priorities play an integral role in determining a stable matching. *Dynamic capacities* and *dynamic priorities* herald an innovative and simple approach for addressing the aforementioned inquiries with promising practical benefits. Our algorithmic contribution is based on a thorough theoretical understanding of the problems we introduce. Our methods, which are effective and exact, empower decision makers.

6.1. Dynamic Capacities

In our first two works, Chapters 3 and 4, we consider the many-to-one stable matching problem, where capacities vary and are allocated subject to a budget. Using the terminology of the many-to-one School Choice model, we introduce the problem of allocating optimally extra capacities to the schools for the benefit of the students, while also deciding the matching between them (schools and students). Allocating extra capacities to the most popular school based solely on popularity may result in a sub-optimal solution. Indeed, we prove that this optimization problem cannot be approximated within $O(n^{\frac{1}{6}-\varepsilon})$, where n is the number of schools. This strong theoretical limitation, motivated us to focus on mathematical programming tools that could effectively solve the problem in practice. We introduce a new mixed-integer linear formulation with a pseudo-polynomial number of stability constraints in the input of the problem. Despite the exponential number of constraints, we devise a separation algorithm that exploits the structure of the problem and yields a state-of-the-art cutting plane method. Indeed, our cutting plane method outperforms the direct resolution of the generalization of known mathematical programming formulations in a synthetic data-set. Moreover, our methodology is effective in solving instances based on data from the Chilean school choice system. Solving these cases in a short time-frame is of practical interest, as it enables policymakers to analyze different scenarios such as budget values. In this direction, future work could focus on speeding up our methodology, in particular by investigating improvements to our formulation and separation step.

From the modeling perspective, we observe that the introduction of penalties for unassigned students allows the policymaker to tune the model according to their necessities: Do they need to prioritize access of new students in the system or the improvement on the basis of merit? We show that our formulation captures this trade-off between access and improvement by both a theoretical and experimental perspective on real-world data from the Chilean school admission system. We are currently collaborating with the Chilean institutions to embed our framework in their system.

A crucial remark is that as soon as capacities are treated as a variable, there may be multiple optimal solutions when the objective is the minimization of the students' cost. This entails that two different optimal allocations of capacities can benefit different sets of schools and different sets of students. In this regard, a relevant open question is the following.

Which measures should a policymaker adopt to break ties between optimal solutions?

For instance, should a policymaker choose the allocation of capacities that assigns extra capacities to a disadvantaged neighborhood or the allocation that expands the capacities of a popular school? Furthermore, if the policymaker needs to target disadvantaged students with a certain profile, which optimal solution is the one more adherent to their policy? These are some of the questions that a policymaker needs to address when considering the improvement of a community, or, more simply, when it needs to impact the education of students via targeted scholarships.

6.2. Dynamic Priorities

In real-world education admission systems, schools policies often give priority to matching siblings to the same school by assigning additional scores to students with siblings. As a result, these priorities are typically incorporated into the students' rankings before the matching process occurs.

In this thesis, we study many-to-one matching markets in which priorities are part of the decision process rather than being given as an input. We embed this idea in the framework of stable matching with complementarities, which is one of the most challenging problems in matching under preferences. In the realm of matching theory with complementarities, the literature has been rife with negative outcomes over the past four decades. However, it is noteworthy that a few scholarly contributions have successfully delineated the specific conditions under which the existence of a stable matching can be assured within this intricate context. In this thesis, which focuses on the context of the school choice admission process, we successfully identify definitions of sibling priority that ensure the existence of a stable matching and those that do not; moreover, we provide mechanisms that find such stable matchings. Our next steps involve two key aspects: First, we aim to determine the conditions under which the Rural Hospital Theorem holds, ensuring that all stable matchings have the same size. Second, we want to assess whether our proposed mechanisms are strategyproof. Additionally, existing literature has demonstrated that by introducing a fixed number of extra capacities, it becomes possible to guarantee the existence of a stable matching in markets with complementarities. This observation prompts a question in scenarios involving priorities:

Can we establish the minimum number of extra capacities required to ensure the existence of a stable matching when sibling priorities are considered, regardless of the specific priority definition?

Sibling priority seems to be a natural notion to embed in a school choice system involving elementary school to high school students. However, it may be more difficult to justify this type of family priority for college education or for the hospital-resident problem with couples; indeed, both of these settings usually involve a stronger component of merit in their admission process, while the current definition of sibling priority may violate some standard notions of merit by allowing siblings to take the spots of students with a higher ranking. Moreover, in the hospital-resident setting, it would be more natural to talk about *partner priority*, in line with the original literature of stable matching with complementarities. An intriguing question for future research could be as follows.

How should we formulate partner priority in a merit-based setting?

In conclusion, this thesis has embarked upon a comprehensive exploration of matching markets, shedding light on their intricate dynamics when incorporating capacities and priorities. By shifting the traditional paradigm and introducing innovative concepts like dynamic capacities and dynamic priorities, we have not only extended the theoretical boundaries of matching theory but also provided valuable insights with practical implications, particularly in the context of education systems. Our findings underscore the necessity for reevaluating and reformulating existing policies to meet the evolving demands of modern society.

References

[1] Atila Abdulkadiroglu, Yeon-Koo Che, Parag A Pathak, Alvin E Roth, and Olivier Tercieux. Minimizing justified envy in school choice: the design of New Orleans' OneApp. Technical report, National Bureau of Economic Research, 2017.

[2] Atila Abdulkadiroğlu, Yeon-Koo Che, and Yosuke Yasuda. Resolving Conflicting Preferences in School Choice: The Boston Mechanism Reconsidered. *American Economic Review*, 101(1):399–410, feb 2011.

[3] Atila Abdulkadiroğlu, Parag A Pathak, and Alvin E Roth. The New York City high school match. *American Economic Review*, 95(2):364–367, 2005.

[4] Atila Abdulkadiroğlu, Parag A. Pathak, and Alvin E. Roth. Strategy-proofness versus efficiency in matching with indifferences: Redesigning the NYC high school match. *American Economic Review*, 99(5):1954–1978, dec 2009.

[5] Atila Abdulkadiroğlu, Parag A Pathak, Alvin E Roth, and Tayfun Sönmez. The Boston public school match. *American Economic Review*, 95(2):368–371, 2005.

[6] Atila Abdulkadiroglu, Parag A Pathak, Alvin E Roth, and Tayfun Sönmez. Changing the Boston school choice mechanism, 2006.

[7] Atila Abdulkadiroğlu and Tayfun Sönmez. Random serial dictatorship and the core from random endowments in house allocation problems. *Econometrica*, 66(3):689–701, 1998.

[8] Atila Abdulkadiroğlu and Tayfun Sönmez. School choice: A mechanism design approach. *American Economic Review*, 93(3):729–747, may 2003.

[9] Kenshi Abe, Junpei Komiyama, and Atsushi Iwasaki. Anytime capacity expansion in medical residency match by Monte Carlo tree search. *arXiv preprint arXiv:2202.06570*, 2022.

[10] Hernán G Abeledo, Yosef Blum, and Uriel G Rothblum. Canonical monotone decompositions of fractional stable matchings. *International Journal of Game Theory*, 25(2):161–176, 1996.

[11] Hernán G Abeledo and Uriel G Rothblum. Courtship and linear programming. *Linear algebra and its applications*, 216:111–124, 1995.

[12] Tobias Achterberg. SCIP: solving constraint integer programs. *Mathematical Programming Computation*, 1:1–41, 2009.

[13] Kolos Csaba Ágoston, Péter Biró, and Iain McBride. Integer programming methods for special college admissions problems. *Journal of Combinatorial Optimization*, 32(4):1371–1399, 2016.

[14] Narges Ahani, Tommy Andersson, Alessandro Martinello, Alexander Teytelboym, and Andrew C Trapp. Placement optimization in refugee resettlement. *Operations Research*, 69(5):1468–1486, 2021.

[15] Maxwell Allman, Itai Ashlagi, Irene Lo, Juliette Love, Katherine Mentzer, Lulabel Ruiz-Setz, and Henry O'Connell. Designing school choice for diversity in the San Francisco Unified School District. In *Proceedings of the 23rd ACM Conference on Economics and Computation*, pages 290–291, 2022.

[16] Maxwell Allman, Itai Ashlagi, Irene Lo, Juliette Love, Katherine Mentzer, Lulabel Ruiz-Setz, and Henry O'Connell. Designing school choice for diversity in the San Francisco Unified School District. In *Proceedings of the 23rd ACM Conference on Economics and Computation*. ACM, jul 2022.

[17] Tommy Andersson and Lars Ehlers. Assigning Refugees to Landlords in Sweden: Efficient, Stable, and Maximum Matchings. *The Scandinavian Journal of Economics*, 122(3):937–965, 2020.

[18] Nick Arnosti. Short lists in centralized clearinghouses. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*. ACM, jun 2015.

[19] Itai Ashlagi, Mark Braverman, and Avinatan Hassidim. Stability in large matching markets with complementarities. *Operations Research*, 62(4):713–732, 2014.

[20] Itai Ashlagi, Afshin Nikzad, and Assaf Romm. Assigning more students to their top choices: A comparison of tie-breaking rules. *Games and Economic Behavior*, 115:167–187, may 2019.

[21] Itai Ashlagi and Peng Shi. Improving community cohesion in school choice via correlated-lottery implementation. *Operations Research*, 62(6):1247–1264, dec 2014.

[22] Itai Ashlagi and Peng Shi. Optimal allocation without money: An engineering approach. *Management Science*, 62(4):1078–1097, apr 2016.

[23] Christopher Avery, Soohyung Lee, and Alvin E Roth. College admissions as non-price competition: The case of South Korea. Technical report, National Bureau of Economic Research, 2014.

[24] Eduardo A Azevedo and Eric Budish. Strategy-proofness in the large. *The Review of Economics Studies*, 86(1):81–116, 2018.

[25] Haris Aziz and Florian Brandl. Efficient, fair, and incentive-compatible healthcare rationing. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pages 103–104, 2021.

[26] Mourad Baïou and Michel Balinski. The stable admissions polytope. *Mathematical Programming*, 87(3):427–439, 2000.

[27] Mourad Baïou and Michel Balinski. Student admissions and faculty recruitment. *Theoretical Computer Science*, 322(2):245–265, 2004.

[28] Brenda S Baker. Approximation algorithms for NP-complete problems on planar graphs. *Journal of the ACM (JACM)*, 41(1):153–180, 1994.

[29] Michel Balinski and Guillaume Ratier. Of stable marriages and graphs, and strategy and polytopes. *SIAM review*, 39(4):575–604, 1997.

[30] Michel Balinski and Guillaume Ratier. Graphs and marriages. *The American Mathematical Monthly*, 105(5):430–445, 1998.

[31] Michel Balinski and Tayfun Sönmez. A tale of two mechanisms: student placement. *Journal of Economic theory*, 84(1):73–94, 1999.

[32] Vipul Bansal, Aseem Agrawal, and Varun S Malhotra. Polynomial time algorithm for an optimal stable assignment with multiple partners. *Theoretical Computer Science*, 379(3):317–328, 2007.

[33] Claude Berge. Two theorems in graph theory. *Proceedings of the National Academy of Sciences of the United States of America*, 43(9):842, 1957.

[34] Péter Biró. Student admissions in Hungary as Gale and Shapley envisaged. *University of Glasgow Technical Report TR-2008-291*, 2008.

[35] Péter Biró, Tamás Fleiner, Robert W Irving, and David F Manlove. The college admissions problem with lower and common quotas. *Theoretical Computer Science*, 411(34-36):3136–3153, 2010.

[36] Péter Biró and Sofya Kiselgof. College admissions with stable score-limits. *Central European Journal of Operations Research*, 23(4):727–741, 2015.

[37] Péter Biró, David F Manlove, and Iain McBride. The hospitals/residents problem with couples: Complexity and integer programming models. In *Experimental Algorithms: 13th International Symposium, SEA 2014, Copenhagen, Denmark, June 29–July 1, 2014. Proceedings 13*, pages 10–21. Springer, 2014.

[38] Federico Bobbio, Margarida Carvalho, Andrea Lodi, Ignacio Rios, and Alfredo Torrico. Capacity planning in stable matching: An application to school choice. *arXiv preprint arXiv:2110.00734*, 2021.

[39] Federico Bobbio, Margarida Carvalho, Andrea Lodi, Ignacio Rios, and Alfredo Torrico. Capacity planning in stable matching: An application to school choice. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, 2023.

[40] Federico Bobbio, Margarida Carvalho, Andrea Lodi, and Alfredo Torrico. Capacity variation in the many-to-one stable matching problem. *arXiv preprint arXiv:2205.01302v1*, 2022.

[41] Lawrence Bodin and Aaron Panken. High tech for a higher authority: The placement of graduating rabbis from Hebrew Union College—Jewish Institute of Religion. *Interfaces*, 33(3):1–11, 2003.

[42] Aaron L. Bodoh-Creed. Optimizing for distributional goals in school choice problems. *Management Science*, 66(8):3657–3676, aug 2020.

[43] Felix Brandt, Vincent Conitzer, Ulle Endriss, Jérôme Lang, and Ariel D Procaccia. *Handbook of computational social choice*. Cambridge University Press, 2016.

[44] Sebastian Braun, Nadja Dwenger, and Dorothea Kübler. Telling the truth may not pay off: An empirical study of centralized university admissions in Germany. *The BE Journal of Economic Analysis & Policy*, 10(1), 2010.

[45] Eric Budish, Gerard Cachon, Judd Kessler, and Abraham Othman. Course match: A largescale implementation of approximate competitive equilibrium from equal incomes for combinatorial allocation. *Operations Research*, 65(2):314–336, 2016.

[46] Caterina Calsamiglia and Maia Güell. Priorities in school choice: The case of the Boston mechanism in Barcelona. *Journal of Public Economics*, 163:20–36, 2018.

[47] F Caro, T Shirabe, M Guignard, and A Weintraub. School redistricting: embedding GIS tools with integer programming. *Journal of the Operational Research Society*, 55(8):836–849, 2004.

[48] Lea Carrier. Les cégeps face à une vague d'élèves. `https://www.lapresse.ca/contexte/2022-04-03/les-cegeps-face-a-une-vague-d-eleves.php`, 2022-04-03. [Published online on La Presse; accessed 2023-09-30].

[49] Christine Cheng, Eric McDermid, and Ichiro Suzuki. A unified approach to finding good stable matchings in the hospitals/residents setting. *Theoretical Computer Science*, 400(1-3):84–99, 2008.

[50] José Correa, Natalie Epstein, Rafael Epstein, Juan Escobar, Ignacio Rios, Nicolás Aramayo, Bastián Bahamondes, Carlos Bonet, Martin Castillo, Andres Cristi, Boris Epstein, and Felipe Subiabre. School Choice in Chile. *Operations Research*, 70(2):1066–1087, 2022.

[51] George Bernard Dantzig. *Linear programming and extensions*, volume 48. Princeton University Press, 1998.

[52] David Delacrétaz, Scott Duke Kominers, and Alexander Teytelboym. Refugee resettlement. *University of Oxford Department of Economics Working Paper*, 2016.

[53] Maxence Delorme, Sergio García, Jacek Gondzio, Joerg Kalcsics, David Manlove, and William Pettersson. Mathematical models for stable matching problems with ties and incomplete lists. *European Journal of Operational Research*, 277(2):426–441, 2019.

[54] Philipp D. Dimakopoulos and C. Philipp Heller. Matching with waiting times: The German entry-level labor market for lawyers. *Games and Economic Behavior*, 115:289–313, 2019.

[55] Samuel Dooley and John P Dickerson. The affiliate matching problem: On labor markets where firms are also interested in the placement of previous workers. *arXiv preprint arXiv:2009.11867*, 2020.

[56] Lester E Dubins and David A Freedman. Machiavelli and the Gale-Shapley algorithm. *The American Mathematical Monthly*, 88(7):485–494, 1981.

[57] Umut Dur, Thayer Morrill, and William Phan. Family ties: School assignment with siblings. *Theoretical Economics*, 17(1):89–120, 2022.

[58] Umut Dur and Martin Van der Linden. Capacity design in school choice. *Available at SSRN 3898719*, 2021.

[59] Umut Mert Dur and Thomas Wiseman. School choice with neighbors. *Journal of Mathematical Economics*, 83:101–109, 2019.

[60] Jack Edmonds. Paths, trees, and flowers. *Canadian Journal of mathematics*, 17:449–467, 1965.

[61] L Ehlers. School choice with control. Cahiers de recherche 13-2010 CIREQ, 2010.

[62] Itai Feigenbaum, Yash Kanoria, Irene Lo, and Jay Sethuraman. Dynamic matching in school choice: Efficient seat reassignment after late cancellations. *Management Science*, 66(11):5341–5361, 2020.

[63] Tamás Fleiner. On the stable b-matching polytope. *Mathematical Social Sciences*, 46(2):149–158, 2003.

[64] Organisation for Economic Co-operation and Development. *Literacy for Life: Further Results from the Adult Literacy and Life Skills Survey*. OECD Publishing, 2011.

[65] National Center for Education Statistics. Condition of America's Public School Facilities: 1999. Access here., April 1999.

[66] UNESCO Institute for Statistics (UIS). Government expenditure on education. `https://data.worldbank.org/indicator/SE.XPD.TOTL.GD.ZS`, 2023. [Published online on World Bank Open Data; accessed 2023-11-01].

[67] David Gale and Lloyd S Shapley. College admissions and the stability of marriage. *The American Mathematical Monthly*, 69(1):9–15, 1962.

[68] David Gale and Marilda Sotomayor. Some remarks on the stable matching problem. *Discrete Applied Mathematics*, 11(3):223–232, 1985.

[69] John Gallinger, Michel Ouellette, Eric Peters, and Lisa Turriff. Carms at 50: Making the match for medical education. *Canadian Medical Education Journal*, 11(3):e133, 2020.

[70] Gerald Gamrath, Daniel Anderson, Ksenia Bestuzheva, Wei-Kun Chen, Leon Eifler, Maxime Gasse, Patrick Gemander, Ambros Gleixner, Leona Gottwald, Katrin Halbig, et al. The SCIP optimization suite 7.0. 2020.

[71] Dan Gusfield and Robert W Irving. *The stable marriage problem: structure and algorithms*. MIT press, 1989.

[72] Isa E. Hafalir, M. Bumin Yenmez, and Muhammed A. Yildirim. Effective affirmative action in school choice. *Theoretical Economics*, 8(2):325–363, may 2013.

[73] Philip Hall. On representatives of subsets. *Classic Papers in Combinatorics*, pages 58–62, 1987.

[74] Günter J Hitsch, Ali Hortaçsu, and Dan Ariely. Matching and sorting in online dating. *American Economic Review*, 100(1):130–163, 2010.

[75] Oscar H Ibarra and Chul E Kim. Fast approximation algorithms for the knapsack and sum of subset problems. *Journal of the ACM (JACM)*, 22(4):463–468, 1975.

[76] Our World in Data. Average years of education for 15-64 years male and female youth and adults.

[77] Robert W Irving, Paul Leather, and Dan Gusfield. An efficient algorithm for the "optimal" stable marriage. *Journal of the ACM (JACM)*, 34(3):532–543, 1987.

[78] Robert W Irving and David F Manlove. Finding large stable matchings. *Journal of Experimental Algorithmics (JEA)*, 14:1–2, 2010.

[79] Kazuo Iwama, Shuichi Miyazaki, Yasufumi Morita, and David Manlove. Stable marriage with incomplete lists and ties. In *International Colloquium on Automata, Languages, and Programming*, pages 443–452. Springer, 1999.

[80] Kazuo Iwama, Shuichi Miyazaki, and Hiroki Yanagisawa. A 25/17-approximation algorithm for the stable marriage problem with one-sided ties. *Algorithmica*, 68(3):758–775, 2014.

[81] Iris Hui-Ru Jiang and Hua-Yu Chang. Ecos: Stable matching based metal-only eco synthesis. *IEEE transactions on very large scale integration (VLSI) systems*, 20(3):485–497, 2011.

[82] Kedar Joshi and Sushil Kumar. Matchmaking using fuzzy analytical hierarchy process, compatibility measure and stable matching for online matrimony in India. *Journal of Multi-Criteria Decision Analysis*, 19(1-2):57–66, 2012.

[83] Yuichiro Kamada and Fuhito Kojima. Improving efficiency in matching markets with regional caps: The case of the Japan residency matching program. *Discussion Papers, Stanford Institute for Economic Policy Research*, volume 1, 2010.

[84] Yuichiro Kamada and Fuhito Kojima. Stability and strategy-proofness for matching with constraints: A problem in the Japanese medical match and its solution. *American Economic Review*, 102(3):366–370, 2012.

[85] Yuichiro Kamada and Fuhito Kojima. Efficient matching under distributional constraints: Theory and applications. *American Economic Review*, 105(1):67–99, 2015.

[86] Yuichiro Kamada and Fuhito Kojima. Efficient matching under distributional constraints: Theory and applications. *The American Economic Review*, 105(1):67–99, 2015.

[87] Yuichiro Kamada and Fuhito Kojima. Stability and strategy-proofness for matching with constraints: A necessary and sufficient condition. *Theoretical Economics*, 13(2):761–793, 2018.

[88] RM KARP. Reducibility among combinatorial problems. *Complexity of Computer Computations*, pages 85–103, 1972.

[89] Alexander S Kelso Jr and Vincent P Crawford. Job matching, coalition formation, and gross substitutes. *Econometrica: Journal of the Econometric Society*, pages 1483–1504, 1982.

[90] John Kennes, Daniel Monte, Norovsambuu Tumennasan, et al. *The daycare assignment problem*. Department of Economics and Business Economics, Aarhus BSS, 2011.

[91] Onur Kesten and M Utku Ünver. A theory of school-choice lotteries. *Theoretical Economics*, 10(2):543–595, 2015.

[92] Zoltán Király. Linear time local approximation algorithm for maximum stable marriage. *Algorithms*, 6(3):471–484, 2013.

[93] Bettina Klaus and Flip Klijn. Stable matchings and preferences of couples. *Journal of Economic Theory*, 121(1):75–106, 2005.

[94] Bettina Klaus, Flip Klijn, and Jordi Massó. Some Things Couples always wanted to know about stable matchings (but were afraid to ask). Working Papers 78, Barcelona School of Economics, September 2003.

[95] Bettina Klaus, Flip Klijn, and Toshifumi Nakamura. Corrigendum to "stable matchings and preferences of couples"[J. Econ. Theory 121 (1)(2005) 75–106]. *Journal of Economic Theory*, 144(5):2227–2233, 2009.

[96] Marina Knittel, Samuel Dooley, and John Dickerson. The dichotomous affiliate stable matching problem: Approval-based matching with applicant-employer relations. In Lud De Raedt, editor, *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, pages 356–362. International Joint Conferences on Artificial Intelligence Organization, 7 2022. Main Track.

[97] Donald Ervin Knuth. Marriages stables. *Technical report*, 1976.

[98] Fuhito Kojima, Parag A Pathak, and Alvin E Roth. Matching with couples: Stability and incentives in large markets. *The Quarterly Journal of Economics*, 128(4):1585–1632, 2013.

[99] Fuhito Kojima, Akihisa Tamura, and Makoto Yokoo. Designing matching mechanisms under constraints: An approach from discrete convex analysis. *Journal of Economic Theory*, 176:803–833, 2018.

[100] Scott Duke Kominers. Respect for improvements and comparative statics in matching markets. Technical report, Working Paper, 2019.

[101] Taro Kumano and Morimitsu Kurino. Quota adjustment process. Technical report, Institute for Economics Studies, Keio University, 2022.

[102] Ryoji Kurata, Naoto Hamada, Atsushi Iwasaki, and Makoto Yokoo. Controlled school choice with soft bounds and overlapping types. *Journal of Artificial Intelligence Research*, 58(1):153–184, jan 2017.

[103] Mark Kutner, Elizabeth Greenburg, Ying Jin, and Christine Paulsen. The Health Literacy of America's Adults: Results from the 2003 National Assessment of Adult Literacy. NCES 2006-483. *National Center for education statistics*, 2006.

[104] Augustine Kwanashie and David F Manlove. An integer programming approach to the hospitals/residents problem with ties. In *Operations Research Proceedings 2013*, pages 263–269. Springer, 2014.

[105] Tomás Larroucau, I Ríos, and Alejandra Mizala. The effect of including high school grade rankings in the admission process for Chilean universities. *Revista de Investigación Educacional Latinoamericana*, 52262(1):95–118, 2015.

[106] Jay Liebowitz and James Simien. Computational efficiencies for multi-agents: a look at a multi-agent system for sailor assignment. *Electronic Government, an International Journal*, 2(4):384–402, 2005.

[107] David Manlove. *Algorithmics of matching under preferences*, volume 2. World Scientific, 2013.

[108] David F Manlove, Robert W Irving, Kazuo Iwama, Shuichi Miyazaki, and Yasufumi Morita. Hard variants of stable marriage. *Theoretical Computer Science*, 276(1-2):261–279, 2002.

[109] G. P. McCormick. Computability of global solutions to factorable nonconvex programs: Part I - convex underestimating problems. *Mathematical Programming*, 10:147–175, 1976.

[110] Eric McDermid. A 3/2-approximation algorithm for general stable marriage. In *Automata, Languages and Programming: 36th International Colloquium, ICALP 2009, Rhodes, Greece, July 5-12, 2009, Proceedings, Part I 36*, pages 689–700. Springer, 2009.

[111] Eric J McDermid and David F Manlove. Keeping partners together: algorithmic results for the hospitals/residents problem with couples. *Journal of Combinatorial Optimization*, 19:279–303, 2010.

[112] David G McVitie and Leslie B Wilson. Stable marriage assignment for unequal sets. *BIT Numerical Mathematics*, 10(3):295–309, 1970.

[113] David G McVitie and Leslie B Wilson. The stable marriage problem. *Communications of the ACM*, 14(7):486–490, 1971.

[114] Silvio Micali and Vijay V Vazirani. An O (v| v| c| E|) algorithm for finding maximum matching in general graphs. In *21st Annual Symposium on Foundations of Computer Science (FOCS 1980)*, pages 17–27. IEEE, 1980.

[115] Ndiamé Ndiaye, Sergey Norin, and Adrian Vetta. Descending the stable matching lattice: How many strategic agents are required to turn pessimality to optimality? In *International Symposium on Algorithmic Game Theory*, pages 281–295. Springer, 2021.

[116] Hai Nguyen, Thành Nguyen, and Alexander Teytelboym. Stability in matching markets with complex constraints. *Management Science*, 67(12):7438–7454, 2021.

[117] Thành Nguyen and Rakesh Vohra. Near-feasible stable matchings with couples. *American Economic Review*, 108(11):3154–69, 2018.

[118] Thành Nguyen and Rakesh Vohra. Stable matching with proportionality constraints. *Operations Research*, 67(6):1503–1519, 2019.

[119] Michael Ostrovsky. Stability in supply chain networks. *American Economic Review*, 98(3):897–923, 2008.

[120] Katarzyna Paluch. Faster and simpler approximation of stable matchings. *Algorithms*, 7(2):189–202, 2014.

[121] Parag A Pathak and Tayfun Sönmez. Leveling the Playing Field: Sincere and Sophisticated Players in the Boston Mechanism. *American Economic Review*, 98(4):1636–1652, Aug 2008.

[122] Parag A Pathak, Tayfun Sönmez, M Utku Ünver, and M Bumin Yenmez. Fair allocation of vaccines, ventilators and antiviral treatments: leaving no ethical value behind in health care rationing. In *Proceedings of the 22nd ACM Conference on Economics and Computation*, pages 785–786, 2021.

[123] Julius Petersen. Die theorie der regulären graphs. *Acta Mathematica*, 15(1):193, 1891.

[124] I. Rios, T. Larroucau, G. Parra, and R. Cominetti. Improving the Chilean College Admissions System. *Operations Research*, 69(4):1186–1205, July-August 2021.

[125] Paul A Robards. Applying two-sided matching processes to the united states navy enlisted assignment process. Technical report, Naval Postgraduate School Monterey CA, 2001.

[126] Antonio Romero-Medina. Implementation of stable solutions in a restricted matching market. *Review of Economic Design*, 3(2):137–147, 1998.

[127] Assaf Romm. Implications of capacity reduction and entry in many-to-one stable matching. *Social Choice and Welfare*, 43(4):851–875, 2014.

[128] Assaf Romm, Alvin E Roth, and Ran I Shorrer. Stability vs. no justified envy. *No Justified Envy (March 6, 2020)*, 2020.

[129] Eytan Ronn. NP-complete stable matching problems. *Journal of Algorithms*, 11(2):285–304, 1990.

[130] Max Roser and Esteban Ortiz-Ospina. Literacy. *Our World in Data*, 2016. https://ourworldindata.org/literacy.

[131] Alvin E Roth. The economics of matching: Stability and incentives. *Mathematics of Operations Research*, 7(4):617–628, 1982.

[132] Alvin E Roth. 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, 1984.

[133] Alvin E Roth. The college admissions problem is not equivalent to the marriage problem. *Journal of economic Theory*, 36(2):277–288, 1985.

[134] Alvin E Roth. On the allocation of residents to rural hospitals: a general property of two-sided matching markets. *Econometrica: Journal of the Econometric Society*, pages 425–427, 1986.

[135] Alvin E. Roth. The economist as engineer: Game theory, experimentation, and computation as tools for design economics. *Econometrica*, 70(4):1341–1378, July 2002.

[136] Alvin E Roth and Elliott Peranson. The redesign of the matching market for American physicians: Some engineering aspects of economic design. *American Economic Review*, 89(4):748–780, 1999.

[137] Alvin E Roth, Uriel G Rothblum, and John H Vande Vate. Stable matchings, optimal assignments, and linear programming. *Mathematics of Operations Research*, 18(4):803–828, 1993.

[138] Alvin E. Roth and Marilda A. Oliveira Sotomayor. *Two-sided matching: A study in game-theoretic modeling and analysis*. Cambridge Univ. Press, Cambridge, MA, 1990.

[139] Uriel G Rothblum. Characterization of stable matchings as extreme points of a polytope. *Mathematical Programming*, 54(1):57–67, 1992.

[140] Jay Sethuraman, Chung-Piaw Teo, and Liwen Qian. Many-to-one stable matching: geometry and fairness. *Mathematics of Operations Research*, 31(3):581–596, 2006.

[141] Peng Shi. Assortment planning in school choice. 2016.

[142] Tasuku Soma and Yuichi Yoshida. A generalization of submodular cover via the diminishing return property on the integer lattice. *Advances in neural information processing systems*, volume 28, 2015.

[143] T Sönmez and MD Yenmez. Affirmative action with overlapping reserves. 2019.

[144] Tayfun Sönmez. Manipulation via capacities in two-sided matching markets. *Journal of Economic theory*, 77(1):197–204, 1997.

[145] Tayfun Sönmez and Tobias B Switzer. Matching with (branch-of-choice) contracts at the United States military academy. *Econometrica*, 81(2):451–488, 2013.

[146] Ashok Subramanian. A new approach to stable matching problems. *SIAM Journal on Computing*, 23(4):671–700, 1994.

[147] Chung-Piaw Teo and Jay Sethuraman. The geometry of fractional stable matchings and its applications. *Mathematics of Operations Research*, 23(4):874–891, 1998.

[148] Chung-Piaw Teo, Jay Sethuraman, and Wee-Peng Tan. Gale-Shapley stable marriage problem revisited: strategic issues and applications. In *International Conference on Integer Programming and Combinatorial Optimization*, pages 429–438. Springer, 1999.

[149] Jill Tucker. SFUSD enrollment plummets this year, doubling peak pandemic declines, new data shows. https://www.sfchronicle.com/sf/article/SFUSD-enrollment-plummets-this-year-doubling-17073854.php, April 2022.

[150] John H Vande Vate. Linear programming brings marital bliss. *Operations Research Letters*, 8(3):147–153, 1989.

[151] Vijay V Vazirani. A theory of alternating paths and blossoms for proving correctness of the general graph maximum matching algorithm. *Combinatorica*, 14(1):71–109, 1994.

[152] Kentaro Yahiro and Makoto Yokoo. Game theoretic analysis for two-sided matching with resource allocation. In *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, pages 1548–1556, 2020.

[153] Hiroki Yanagisawa. Approximation algorithms for stable marriage problems. *Ph.D. Thesis, Kyoto University*, 2007.

[154] Wei Yang, JA Giampapa, and Katia Sycara. Two-sided matching for the us navy detailing process with market complication. Technical report, Technical Report CMU-RI-TR-03-49, Robotics Institute, Carnegie-Mellon University, 2003.

[155] Haibo Zhang. Analysis of the Chinese college admission system. *The University of Edinburgh*, 2010.

[156] William S Zwicker. Introduction to the theory of voting. *Handbook of Computational Social Choice*, pages 23–56, 01 2016.

[157] Kolos Csaba Ágoston, Péter Biró, Endre Kováts, and Zsuzsanna Jankó. College admissions with ties and common quotas: Integer programming approach. *European Journal of Operational Research*, 2021.