

Integer programming methods for special college admissions problems*

Kolos Csaba Ágoston¹, Péter Biró^{2,†} and Iain McBride^{3,‡}

¹ *Department of Operations Research and Actuarial Sciences,
Corvinus University of Budapest, H-1093, Fővám tér 13-15., Budapest, Hungary
Email: kolos.agoston@uni-corvinus.hu.*

² *Institute of Economics, Research Centre for Economic and Regional Studies,
Hungarian Academy of Sciences, H-1112, Budaörsi út 45, Budapest, Hungary, and
Department of Operations Research and Actuarial Sciences,
Corvinus University of Budapest, H-1093, Fővám tér 13-15., Budapest, Hungary
Email: peter.biro@krtk.mta.hu.*

³ *School of Computing Science, University of Glasgow
Sir Alwyn Williams Building, Glasgow G12 8QQ, UK.
Email: i.mcbride.1@research.gla.ac.uk.*

Abstract. We develop Integer Programming (IP) solutions for some special college admission problems arising from the Hungarian higher education admission scheme. We focus on four special features, namely the solution concept of stable score-limits, the presence of lower and common quotas, and paired applications. We note that each of the latter three special feature makes the college admissions problem NP-hard to solve. Currently, a heuristic based on the Gale-Shapley algorithm is being used in the Hungarian application. The IP methods that we propose are not only interesting theoretically, but may also serve as an alternative solution concept for this practical application, and other similar applications. We finish the paper by presenting a simulation using the 2008 data of the Hungarian higher education admission scheme.

Keywords: College admissions problem, integer programming, stable score-limits, lower quotas, common quotas, paired applications, simulations

JEL classification: C61, C63, C78

Introduction

Gale and Shapley [15] introduced and solved the college admissions problem, which generated a broad interdisciplinary research field in mathematics, computer science, game

*A preliminary version [11] has been presented at COCOA 2014. This version has been significantly extended with new theoretical results and also with a completely new section on computer simulations.

[†]Supported by the Hungarian Academy of Sciences under its Momentum Programme (LP2016-3/2016), by OTKA grant no. K108673, and also by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences.

[‡]Supported by a SICSA Prize PhD Studentship.

theory and economics¹. The Hungarian higher education admission scheme is also based on the Gale-Shapley algorithm, but the algorithm is extended with a number of heuristics since the model contains some special features. In this paper we study the possibility of modelling and resolving these special features with integer programming techniques.

In the Hungarian higher education matching scheme (see a detailed description in [6] and [5]), the students apply for programmes under two possible contract terms, as most programmes can be attended in state-financed and self-financed ways. Thus the application of a student contains ranked contracts, the first choice may be a state-financed economics programme at university A, the second choice a state-financed computer science programme at university B, and the third choice a self-financed economics programme again at university A, etc. The central coordinating agency then collects the scores of the students and based on the applications and the scores they compute the cut-off scores for all the programmes under both contract terms. Each student is then admitted to the first place on her list, where she met the cut-off score. In this paper, for simplicity and for being consistent with the literature, we will refer to the programmes with contract terms simply as colleges in our models, except when we describe the details of the Hungarian application. The Hungarian application has four special features that we study in the paper.

The first special feature of the application is the presence of ties, and the solution concept of stable score-limits. According to the Hungarian admission policy, when two applicants have the same score at a programme then they should either both be accepted or rejected by that programme. The solution of stable score-limits ensures that no quota is violated, hence the last group of students with the same score that would cause a quota violation is always rejected. A set of stable score-limits always exists, and a student-optimal solution can be found efficiently by an extension of the Gale-Shapley algorithm, as shown in [9]. This method is the basis of the heuristic used in the Hungarian application.

The second and third special features studied in this paper are the lower and common quotas. A university may set not just an upper quota for the number of admissible students for a programme, but also a lower quota. A violation of this lower quota would imply the cancellation of the programme. Furthermore, a common upper quota may be also introduced for a set of programmes, to limit the number of students admitted to a faculty, at a university or nationwide with regard to the state-financed seats in a particular subject. These concepts were studied in [7], where the authors showed that each of these special features makes the college admission problem NP-hard, even in the form that is present in the Hungarian application. Finally, students can apply for pairs of programmes in case of teacher education programmes. This possibility was reintroduced in the scheme in 2010. This problem is closely related to the Hospitals / Residents problem with Couples, where couples may apply for pairs of positions. The latter problem is also known to be NP-hard [24], even for unit-capacity hospitals, for so-called consistent preferences [21], and also for a specific setting present in Scotland [8] where hospitals have common rankings. The fact that the unit-capacity case is also NP-hard implies the NP-hardness of the paired application problem as well.

The polytope of stable matchings was described in a number of papers for the stable marriage problem [30], [28], and for the college admissions problem [4], [13], [31]. For

¹The 2012 Nobel-Prize in Economic Sciences was awarded to Alvin Roth and Lloyd Shapley for the theory of stable allocations and the practice of market design.

these classical models, since the extremal point of the polytopes are integral, and thus correspond to stable matchings, one could always use a linear programming solver to compute stable solutions in polynomial time (although the computation of solutions can also be done efficiently using the Gale-Shapley algorithm).

However, by introducing even just one special feature, the existence of a stable matching can no longer be guaranteed, and the problem of finding a stable solution may become NP-hard. In such cases it may be worth investigating integer programming techniques for solving these problems in theory and in practice as well. To the best of our knowledge there have been only two recent studies of this kind so far. In the first study Kwanashie and Manlove [20] investigated the problem of finding a maximum size weakly stable matchings for college admissions problems with ties, a problem known to be NP-hard, and motivated by the Scottish resident allocation scheme². In the other paper [12] the above mentioned matching with couples problem has been studied. As we have already mentioned, one of the four special features of the Hungarian higher education scheme, namely the presence of paired applications, has a close connection to the problem of couples. However, the remaining three special features studied in our paper, namely, the stable score-limits, lower quotas, and common quotas, have not been investigated from this perspective.

It is interesting to note that whilst we are not aware of any large scale application for two-sided stable matching markets, except two minor examples³, integer programming is the standard technique used for kidney exchange programs [1], [29] and [22], where patients exchange their incompatible donors in a centralised scheme.

Finally, we would like to highlight that the models and solution techniques presented in this college admission context may well be useful for other applications, too. Two important applications are immediately apparent. Firstly, controlled school choice [2], where the policy makers might want to improve the socio-ethnic diversity of the schools by setting different quotas for some types of students. The other example is the resident allocation program, as used in Japan [19], where both lower and upper quotas can be requested as regional caps to ensure a better coverage in health care services in all geographic areas with regard to each medical specialty.

In section 1 we describe a basic model for the classical College Admissions problem that will be the basis of our extended models. In section 2 we consider the College Admissions problem with ties and describe two integer linear programs for finding a stable set of score-limits. The first model uses the objective function to achieve stability and thus leads to the student-optimal stable set of score-limits. The second model describes all the stable sets of score-limits using an extended IP model. In section 3 we formulate an IP for describing the College Admissions problem with lower quotas, we provide some useful lemmas that can speed up a solution, and we also extend the model for the case where the lower quotas are established for sets of colleges. In section 4 we study the College Admissions problem with common quotas and we give an integer programming model that describes the set

²The same problem has also been investigated in a master's thesis [23].

³In a famous study Roth [26] analysed the nature and the long term success of a dozen resident allocation schemes established in the UK in the late seventies. He found that two schemes produced stable outcomes and both of them remained in use. From the remaining six ones, that did not always produce stable matchings, four were eventually abandoned. The two programs that were not always produced stable solutions but yet remained in use were based on linear programming techniques and has been operating in the two smallest markets. Ünver [32] studied these programs and the possible reasons for their survival in detail.

of stable solutions. In section 5 we describe the special feature of paired applications. In section 6, we analyse the possibility and difficulties of formulating general IP models to describe the stable solutions when some combination of special features are present in the application. Finally, in section 7 we test our IP formulations on a real data of the 2008 Hungarian higher education admission.

1 A model for the classical College Admissions problem

Our basic model is an extension of the Rothblum [30] model (analysed also in [28]). This model has been described in [4].

Let $A = \{a_1, \dots, a_n\}$ be the set of applicants and $C = \{c_1, \dots, c_m\}$ the set of colleges. Let u_j denote the upper quota of college c_j . Regarding the preferences and priorities, let r_{ij} denote the rank of college c_j in a_i 's preference list, meaning that a_i prefers c_j to c_k if $r_{ij} < r_{ik}$. Let s_{ij} be an integer representing the score of a_i at college c_j , meaning that a_i has priority over a_k at college c_j if $s_{ij} > s_{kj}$. Let \bar{s} denote the maximum possible score and E the set of applications. A *matching* is a set of applications, where each applicant is assigned to at most one college and each college is assigned to at most as many applicants as its upper quota. A matching is said to be *stable* if for any applicant-college pair not included in the matching either the applicant is matched to a more preferred college or the college filled its upper quota with better applicants.

We introduce binary variables $x_{ij} \in \{0, 1\}$ for each application coming from a_i to c_j , as a characteristic function of the matching, where $x_{ij} = 1$ corresponds to the case when a_i is assigned to c_j . The feasibility of a matching can be ensured with the following two sets of constraints.

$$\sum_{j:(a_i, c_j) \in E} x_{ij} \leq 1 \text{ for each } a_i \in A \quad (1)$$

$$\sum_{i:(a_i, c_j) \in E} x_{ij} \leq u_j \text{ for each } c_j \in C \quad (2)$$

Here, (1) implies that no applicant can be assigned to more than one college, and (2) implies that the upper quotas of the colleges are respected.

One way to enforce the stability of a feasible matching is by the following constraint.

$$\left(\sum_{k:r_{ik} \leq r_{ij}} x_{ik} \right) \cdot u_j + \sum_{h:(a_h, c_j) \in E, s_{hj} > s_{ij}} x_{hj} \geq u_j \text{ for each } (a_i, c_j) \in E \quad (3)$$

Note that for each $(a_i, c_j) \in E$, if a_i is matched to c_j or to a more preferred college then the first term provides the satisfaction of the inequality. Otherwise, when the first term is zero, then the second term is greater than or equal to the right hand side if and only if the places at c_j are filled with applicants with higher scores.

Remark 1: When we have ties in the priorities (due to equal scores), then the following modified stability constraints, together with the feasibility constraints (1) and (2), lead

to *weakly stable* matchings (in a model also known as Hospitals/Residents problem with Ties). Note that here only the strict inequality $s_{hj} \geq s_{ij}$ becomes weak.

$$\left(\sum_{k:r_{ik} \leq r_{ij}} x_{ik} \right) \cdot u_j + \sum_{h:(a_h, c_j) \in E, s_{hj} \geq s_{ij}} x_{hj} \geq u_j \text{ for each } (a_i, c_j) \in E \quad (4)$$

We remark that weakly stable matchings can have different sizes and the problem of finding a maximum size weakly stable matching is NP-hard (although some good approximation results exist, e.g. in [18]). See more about this problem, and its solutions by IP techniques in the recent paper of Kwanashie and Manlove [20].

Remark 2: In the absence of ties, we can get an applicant-optimal (resp. an applicant-pessimal) stable solution by setting the objective function of the IP as the minimum (resp. maximum) of the following term:

$$\sum_{(a_i, c_j) \in E} r_{ij} \cdot x_{ij}$$

We remind the reader that these extreme solutions can be obtained with the two versions of Gale and Shapley's deferred acceptance algorithm in linear time [15].

Remark 3: Baïou and Balinski [4] proposed an alternative model to describe the stable admission polytope, since the above simple integer program may admit fractional solutions as extreme points. See also Sethuraman et al. [31] about the alternative model. Fleiner [13] provided a different description for the stable admission polytope.

2 Stable score-limits

The use of score-limits (or cutoff scores) is very common in college admission systems. The applicants have a score at each place to which they are applying and they are ranked according to these scores by the colleges. The solution is announced in terms of score-limits, each college (or a central coordinator) announces the score of the last admitted student, and each student is then admitted to her most preferred place on her preference list where she achieved the score-limit. See more about the Irish, Hungarian, Spanish and Turkish applications in [9]. The score-limits can be seen as a kind of dual solution of a matching, or prices in a competitive equilibrium. Azevedo and Leshno [3] analysed this phenomenon in detail.

In this section we first develop a basic model for the classical College Admissions problem by using score-limits. Then we discuss the case when ties can appear due to students having the same score when applying to a place, as happens in Hungary. We show how this extended setting can be described with a similar IP both with and without the use of an objective function.

2.1 Stable score-limits with no ties

If we are given a stable matching for a College Admissions problem then we can define a stable set of score-limits by keeping the following requirements. Each student must meet

the score-limit of the college where she is admitted, and no student meets the score-limit of a college that rejected her. These two requirements imply that every student is admitted to the best place where she achieved the score-limit. Finally, to ensure that no student is rejected if a quota was not filled we will require the score-limit of each unfilled college to be minimal.

To describe this solution concept with an integer program, we introduce new variables for the score-limits. Let t_j be a nonnegative integer corresponding to the score-limit of college c_j , where $0 \leq t_j \leq \bar{s} + 1$. Note that the score-limit might need to be greater than the maximum score in the Hungarian application if the number of applicants with maximum score exceed the quota of a programme. The feasibility constraints (1) and (2) remain the same, and we only need to link the score-limits to the matching and establish the new stability conditions as follows.

$$t_j \leq (1 - x_{ij}) \cdot (\bar{s} + 1) + s_{ij} \text{ for each } (a_i, c_j) \in E \quad (5)$$

and

$$s_{ij} + 1 \leq t_j + \left(\sum_{k:r_{ik} \leq r_{ij}} x_{ik} \right) \cdot (\bar{s} + 1) \text{ for each } (a_i, c_j) \in E \quad (6)$$

Here, (5) implies that if an applicant is admitted to a college then she achieved the score-limit of that college. The other constraint, (6), ensures that if applicant a_i is not admitted to c_j then either her score at c_j is lower than the score-limit, t_j , or she is admitted to a college that she preferred.

Finally, we need to ensure that each college that could not fill its quota has a minimal score-limit. We introduce a binary variable $f_j \in \{0, 1\}$ for each college c_j which is equal to zero if the college is not fully filled, by using the following constraint.

$$f_j \cdot u_j \leq \sum_{i:(a_i, c_j) \in E} x_{ij} \text{ for each } c_j \in C \quad (7)$$

Then the following constraint ensures that if a college is unfilled then its score-limit is zero.

$$t_j \leq f_j(\bar{s} + 1) \text{ for each } c_j \in C \quad (8)$$

We summarise the above statements in the following theorem.

Theorem 1 *The stable matchings and the related stable sets of score-limits of a College Admissions problem correspond to the solutions of the integer linear program consisting of the feasibility conditions (1), (2), the stability conditions (5), (6) and conditions (7), (8).*

Proof: A matching is feasible if and only if the corresponding solution satisfy the feasibility constraints. Condition (5) implies that if an applicant is admitted to a college then she achieved the score-limit of that college, and (6) implies that she has not achieved the score-limit of any college that she prefers to her assignment. Therefore, these two conditions are satisfied if and only if every applicant is admitted to the best place in her list where she achieved the score-limit. Finally, (7) and (8) ensure that no college can have positive score-limit, and therefore no college can reject any applicant, if its quota is not filled. ■

2.2 Stable score-limits with ties

The college admission problem with ties has been defined in [5] and studied in [9] and [14]. Ties can appear, as the scores of the applicants might be equal at a college, and these ties are never broken in the Hungarian application. Therefore a group of students with the same score are either all accepted or all rejected. In the Hungarian application the upper quotas are always satisfied, and the stability is defined with score-limits (cutoff scores) as follows. For a set of score-limits, each applicant is admitted to the first place in her list where she achieves the score-limit. This creates a natural mapping from the sets of score-limits to matchings. A set of score-limits is feasible if no quota is violated in the corresponding matching. A set of score-limits is stable if no score-limit can be lowered at any college without causing the violation of its quota in the corresponding matching, while keeping the other score-limits unchanged. When no ties occur then this definition is equivalent to the original Gale-Shapley one. A stable set of score-limits always exists, and may be found using a generalised Gale-Shapley algorithm. Moreover the applicant proposing version leads to an applicant-optimal solution, where each of the score-limits at each college is as small as possible, and a similar statement applies for the college proposing-version (see [5], [9] and [14] for details).

Here we present an IP formulation to find an applicant-optimal set of score-limits with respect to the Hungarian version, which is called *H-stable set of score-limits* in [9]. The feasibility constraints (1) and (2) remain the same as in the previous model, and also the two requirements regarding the score-limits, expressed in constraints (5), (6). However, we cannot require the unfilled colleges to have minimal score-limits in this model, since an unfilled seat might be created by a tie. We describe two possible solutions for this problem. The first is the use a simple objective function as follows.

$$\min \sum_{j=1 \dots m} t_j \tag{9}$$

The above objective function is necessary to ensure the stability condition, that is no college can decrease its score limit without violating its quota, supposing that the other score-limits remain the same. To summarise, we state and prove the correctness of the integer program as follows.

Theorem 2 *Feasibility conditions (1), (2) and stability conditions (5), (6) together with the objective function (9) comprise an integer linear program such that the optimal solution of this IP corresponds to the applicant-optimal stable set of score-limits.*

Proof: The feasibility constraints ensure that any binary solution of the IP corresponds to a feasible matching, where each applicant is admitted to at most one college and no quota is violated at any college. Conditions (5) and (6) ensure that each applicant is admitted to the first place in her preference list where she achieved the score-limit. In particular, (5) implies that if a_i is admitted to c_j then she must have reached the score-limit of c_j , and (6) implies that if a_i is not admitted to c_j or any better college of her preference (i.e. when the second term of the right hand side is zero) then a_i could not achieve the score-limit of c_j . Finally, the objective function ensures that no college can decrease its score-limit (without violating its quota). However, this objective function also

implies that the solution of the IP must correspond to the applicant-optimal stable set of score-limits, since for any other stable set of score-limits at least one college would have a higher score-limit and neither could have lower, so the sum of the score-limits would not be minimal. ■

When we want to describe all sets of stable score-limits (and not only the applicant-optimal one) then we have to replace the objective function with some additional conditions. We can do that by introducing some new variables, as described in the next subsection.

2.3 Stable score-limits with ties and free objective function

First, we introduce a binary variable y_j for each college c_j that is equal to one when t_j is positive (i.e. when there are some applicants rejected from c_j), and otherwise it is zero.

$$t_j \leq (\bar{s} + 1)y_j \text{ for each } c_j \in C \quad (10)$$

Then, for each application (a_i, c_j) , we define a new binary variable, d_{ij} , that can be equal to one if a_i prefers c_j compared to her actual match and a_i would meet the admission criteria at c_j if the score-limit at c_j was decreased by one, where m denotes the number of colleges.

$$\sum_{r_{ik} \geq r_{ij}} d_{ik} \leq (1 - x_{ij})m \text{ for each } (a_i, c_j) \in E \quad (11)$$

$$t_j - 1 \leq (1 - d_{ij})\bar{s} + s_{ij} \text{ for each } (a_i, c_j) \in E \quad (12)$$

With the help of the new variables, we can now describe the stability condition of the score-limits as follows.

$$(u_j + 1)(1 - y_j) + \sum_{i:(a_i, c_j) \in E} (x_{ij} + d_{ij}) \geq u_j + 1 \text{ for each } c_j \in C \quad (13)$$

Theorem 3 *Feasibility conditions (1), (2), stability conditions (5) and (6) and conditions (10), (11), (12) for the new variables d_{ij} , together with a new stability condition (13) comprise an integer linear program such that each feasible integer solution corresponds to a stable set of score-limits.*

Proof: Again, the feasibility conditions ensure that the corresponding matching is feasible, if and only if the assignment of values to the variables in the IP is feasible. Similarly to the previous model, (5) implies that if an applicant is admitted to a college then she achieved the score-limit of that college, and (6) implies that each applicant is admitted to the best available place in her list, if admitted somewhere. Now we will prove that the remaining four sets of conditions are satisfied if and only if the set of score-limits is stable, i.e. when no college can decrease its score-limit without violating its quota. Suppose first that we have a stable set of score-limits. We assign values to all variables in the IP model appropriately and we prove that the constraints are satisfied. So let t_j be the score-limit at college c_j and we set y_j to be one if t_j is positive. Let d_{ij} be equal to one for each applicant a_i who would prefer to be matched to college c_j than her current

partner and who would also meet the score-limit at c_j if it was decreased by one (i.e. $s_{ij} = t_j - 1$), and we set all the other d_{ij} variables to be zero. When doing so, we satisfy conditions (11) and (12), since (11) is satisfied when no d_{ik} is equal to one if a_i prefers her current match c_j to c_k , and (12) is satisfied if a_i meets the score-limit at c_j if it is decreased by one. The stability of the set of score-limits means that no college (with a positive score-limit) can decrease its score-limit without violating its quota. This means that if a college has a positive score-limit, so $y_j = 1$, then if it decreased its score-limit by one then the new students admitted were exactly those for whom the corresponding variable d_{ij} is equal to one. The violation of the quota implies that (13) must be satisfied. To prove the converse, let us suppose that have an assignment of values to the variables in our IP model such that all the constraints in our IP model are satisfied. We will prove that the set of score-limits as defined by variables t_j is stable, that is, for each college c_j either its score-limit is zero or the decrease of its score-limit would cause a quota violation. When t_j is positive then y_j must be zero, so $\sum_{i:(a_i, c_j) \in E} (x_{ij} + d_{ij}) \geq u_j + 1$. Since d_{ij} can be one only if a_i both desire and deserves c_j when t_j is decreased to $t_j - 1$, this means that the quota at c_j would be indeed violated when the score-limit would be decreased by one. (Note that y_j does not necessarily have to be zero when t_j is zero and d_{ij} does not necessarily have to be one when a_i both deserves and desires c_j .) ■

Therefore now we can compute both the student-optimal and student-pessimal stable score-limits, by setting the objective function as described in Remark 2. These extremal solutions can be also computed efficiently by the two generalised versions of the Gale-Shapley algorithm, as shown in [9].

We note that in [9] there was another stability definition, the so-called L-stability, that is based on a more relaxed admission policy, namely when the last group of students with the same score with whom the quota would be violated are always accepted. In this paper we are focusing on the setting that is present in the Hungarian application, so we do not deal with L-stability, but it would be possible to describe an IP model for that version as well in a similar fashion.

3 Lower quotas

In this section we extend the classical College Admissions problem with the possibility of having lower quotas set for the colleges. After developing an integer program for finding a stable solution for this problem we describe the current heuristic used in Hungary and we also provide some Lemmas that can speed up the solution of the IP. Finally we discuss the possibility of having lower quotas for sets of colleges.

3.1 College Admissions problem with lower quotas

This problem has been defined in [7]. In addition to the College Admissions model, here we have lower quotas as well. Let l_j be the lower quota of college c_j . In a feasible solution a college can either be closed (in which case there is no student assigned to there), or open, when the number of students admitted must be between its lower and upper quotas. To describe this feasibility requirement, besides keeping (1), we modify (2) as follows. We introduce a new binary variable, $o_j \in \{0, 1\}$ for each college c_j , where $o_j = 1$ corresponds to the case when the college is open.

$$o_j \cdot l_j \leq \sum_{i:(a_i, c_j) \in E} x_{ij} \leq o_j \cdot u_j \text{ for each } c_j \in C \quad (14)$$

The above set of constraints together with (1) ensure the feasibility of the matching.

The stability of a solution requires the lack of traditional blocking pairs for open colleges, and the lack of blocking groups for closed colleges. The latter means that there cannot be at least as many unsatisfied students (unassigned or assigned to a less preferred place) at a college as the lower quota of that college. The stability conditions can be enforced with the following conditions.

$$\left(\sum_{k:r_{ik} \leq r_{ij}} x_{ik} \right) \cdot u_j + \sum_{h:(a_h, c_j) \in E, s_{hj} > s_{ij}} x_{hj} \geq o_j \cdot u_j \text{ for each } (a_i, c_j) \in E \quad (15)$$

$$\sum_{i:(a_i, c_j) \in E} \left[1 - \sum_{k:r_{ik} < r_{ij}} x_{ik} \right] \leq (1 - o_j) \cdot (l_j - 1) + o_j \cdot n \text{ for each } c_j \in C \quad (16)$$

The first condition implies the usual stability for open colleges, whilst the second condition implies the group-stability for closed colleges. Below we give a formal definition for the College Admission problem with lower quotas and a proof of the above description in the following Theorem.

Theorem 4 *The feasibility conditions (1) and (14) together with the stability conditions (15) and (16) form an integer program such that its solutions correspond to the stable matchings of a college admissions problem with lower quotas.*

Proof: A solution of the IP satisfies the feasibility conditions (1) and (14) if and only if the corresponding matching is feasible, i.e., no student is admitted to more than one college and the lower and upper quotas are respected in each open college. Regarding stability, condition (15) is redundant if the college is closed and implies the pairwise stability condition for any open college. Condition (16) is redundant for open colleges, and enforces group-stability for any closed college. To show the latter we will demonstrate that the right hand side of the constraint is $l_j - 1$ for a closed college c_j and on the left hand side those applicants of c_j are counted who are not admitted to any preferred place according to their preferences, so these are the students who would be happy if c_j would be open and admit them. ■

3.2 Heuristics

The problem of finding a stable matching for the college admissions problem with lower quotas is proven to be NP-hard [7]. In the Hungarian application, where lower quotas can be set for any programme, the following heuristic is used with regard to this special feature. First, the applicant-proposing Gale-Shapley algorithm produces a stable matching where some lower quotas might be violated. The heuristic closes one programme, where the ratio of the number of students admitted and lower quota is minimal, and then the applicant-proposing Gale-Shapley algorithm continues by letting the rejected students (whose assigned programme has just been cancelled) apply to their next choices. This

heuristic runs in linear time in the number of applications, and it produces the applicant-optimal stable matching for the remaining open colleges. However, as illustrated in [7], this heuristic can easily produce unstable outcomes even when the problem is solvable. With the IP technique, however, the IP model should guarantee to find a stable solution, whenever it exists. The following lemmas can help in speeding up the solver.

Lemma 5 *Let I be an instance of College Admission problem and I' be a reduced market where a college is missing. Then the number of students admitted to any college in I' must be at least as many as the number of students admitted in I . Moreover, when we compare the student-optimal (resp. college-optimal) stable matchings in I and I' then each student gets admitted to a college at least as good in I as they did in I' .*

Proof: By the Rural Hospitals theorem ([16], [25] and [27]) we know that in the College Admissions problem each college admits the same number of students in every stable matching. Suppose first that we consider the student-optimal stable matching in I . When we remove a college then we can invoke the proposal-rejection sequence used in the Gale-Shapley student proposing algorithm and obtain a new stable matching where each college admits at least as many students as before and each student is admitted to the same or a worse place (or nowhere). Suppose now that we consider the college-optimal stable matching for I' . When we add back the missing college and restart the college-proposing Gale-Shapley process then we will obtain a new stable solution that is at least as good for each student as the previous one. ■

Lemma 6 *The colleges that reach their lower quotas in the stable solutions of a College Admissions problem with no lower quotas must be open in every stable solution where lower quotas are respected.*

Proof: Let the original problem, where all the colleges are open, be denoted by I . Suppose for a contradiction that a college c_j reaches its lower quota for I , but there is a stable solution where c_j is closed. Let X denote the set of closed colleges in this solution, where $c_j \in X$, and let us denote the submarket where colleges in X are closed by I_X . For the market where every college in X but c_j is closed, denoted by $I_{X \setminus j}$, c_j must still reach its lower quota by Lemma 5. Furthermore, when we consider the applicant-optimal stable matching for $I_{X \setminus j}$, when we remove c_j and conduct the applicant proposing deferral-acceptance process of Gale-Shapley, as in the Hungarian application, in the resulting stable matching for I_X the students who were previously matched to c_j are all worse off. Therefore for any pairwise stable matching in I_X these students would block the matching with college c_j , since they all prefer c_j to their current match and they number at least as many as the lower quota of c_j . ■

Lemma 7 *Suppose that X is the set of colleges that do not reach their lower quotas in the stable solutions with no lower quotas. Given a college c_j of X , if all the colleges in X but c_j are closed and c_j still does not achieve its lower quota then c_j must be closed in any stable solution with lower quotas.*

Proof: As we have seen in Lemma 6, no college outside X may be closed in any stable solution, therefore for any stable solution a subset of X should be closed. Suppose for a

contradiction that we have a stable matching with lower quotas where $Y \subset X$ is the set of closed colleges and $c_j \notin Y$, meaning that c_j also reaches its lower quota in this matching. Let $Y' = X \setminus c_j$, our assumption is that c_j does not reach its lower quota in the stable matchings when the colleges in Y' are closed. Since $Y \subseteq Y'$, Lemma implies that c_j has at least as many students admitted in the stable matchings for I_Y (the market when the colleges in Y are closed) than for $I_{Y'}$, a contradiction. ■

Lemma 8 *Suppose that applicant a_i is assigned to college c_j in the applicant-proposing Gale-Shapley algorithm with no lower quotas. Let X denote the set of colleges that do not achieve their lower quotas in this solution. If we close all the colleges in X and we get that a_i is still assigned to c_j in the applicant-optimal solution of the reduced problem then the existence of a stable solution for lower quotas implies the existence of a stable solution where a_i is assigned to c_j . Thus we can fix this pair in the matching without losing the solvability of the problem.*

Proof: First note that $c_j \notin X$, as otherwise a_i could not be assigned there when colleges in X are closed. Therefore c_j achieves its lower quotas in the stable solutions with no lower quotas considered, so c_j must be open in every stable solution in the original market I . Suppose that there is a stable solution, where the set of closed colleges is $Y \subseteq X$. In the market when the colleges in Y are closed, denoted by I_Y , the applicant-optimal stable solution must assign a_i to c_j . This is because from Lemma 5 we know that a_i 's optimal stable partner in I_Y must be as good for her as in the original market, I , and at most as good as in I_X , but we assumed that her best stable partner is c_j in both I and I_X . ■

With the help of Lemmas 6 and 7 we can iteratively find some colleges which must be open and perhaps also some that must be closed in any stable solution, thus reduce the number of variants of our model, as follows. First we run the Gale-Shapley algorithm without lower quotas, and then we set each college that reached its lower quota to be open. Let us denote this set of colleges by X_1 . In the second step we check each college in $C \setminus X_1$ whether it can be open in any stable solution, as described in Lemma 7. That is, for each $c_j \in C \setminus X_1$ we close all the colleges $C \setminus (X_1 \cup c_j)$, run the Gale-Shapley algorithm and check whether c_j reaches its lower quota. If not then we set c_j to be closed. Let Y_1 denote the set of colleges that were found to be necessary to close in the second step. If Y_1 is nonempty then we repeat the first step: we run again the Gale-Shapley algorithm with no lower quotas and without colleges Y_1 . If, in addition to X_1 , some new colleges also achieve their lower quotas in the reduced market then we add them to X_1 and get a larger set of colleges, X_2 , that must be open in every stable solutions. In the case where X_2 is larger than X_1 then we will repeat the second step of our process with respect to X_2 . Again, if we find any new college that must be closed then we increase the set Y_1 to Y_2 . We repeat this process until $X_t = X_{t+1}$ or $Y_t = Y_{t+1}$ for any t , and then we stop.

3.3 Lower quotas for sets of colleges

We note that in the Hungarian application the lower quotas are set for pairs of programmes (actually for the same programme, just with separate quotas for state-financed and self-financed students). If the lower quota for a set of colleges is not met then all the colleges have to be closed. This motivates the extended model where the lower quotas can be applied for sets of colleges. In this case we need only to introduce a new binary variable

o_p for each set of colleges C_p with common lower quota l_p that we associate with the other binary variables of individual colleges as follows.

$$o_p \cdot n_p \leq \sum_{j:c_j \in C_p} o_j \leq o_p \cdot n_p \text{ for each } C_p \quad (17)$$

where n_p is the number of colleges in C_p . We will then set the feasibility conditions (14) for the sets of colleges with common lower quotas as follows.

$$o_p \cdot l_p \leq \sum_{i:(a_i, c_j) \in E} x_{ij} \text{ for each } c_j \in C \quad (18)$$

To define stability in this setting is not easy though. In section 6 we discuss some complications and potential solutions by relaxing group-stability.

4 Common quotas

This problem has also been defined in [7]. For each set of colleges $C_p \subseteq C$ the coordinator of the admission scheme may set a common upper quota, u_p , meaning that the total number of students admitted to colleges in C_p cannot exceed this quota. Therefore, the set of feasibility constraints, (1) and (2), has to be extended with some new constraints enforcing the common quotas, as follows.

$$\sum_{(a_i, c_j) \in E, c_j \in C_p} x_{ij} \leq u_p \text{ for each } C_p \subseteq C \quad (19)$$

Regarding stability, first of all we have to suppose that any two colleges, c_j and c_k , that belong to a set of colleges C_p with a common quota must rank their applicants in the same way. In particular, in the Hungarian application any student a_i has the same score at such colleges (i.e. programmes in Hungary) with common quota, so $s_{ij} = s_{ik}$ holds. (In a more general model, we should have a specific scoring for each set of colleges with common quota, which is in agreement with the individual scorings of the colleges belonging to this set. For instance, we could have a score s_{ij}^p for each application associated to a set of colleges C_p with a common quota such that $s_{ij} > s_{lj}$ implies $s_{ij}^p > s_{lj}^p$).

In this setting stability means that if a student a_i is not admitted to a college c_j or to any better college of her preference then either c_j must have filled its quota with better students or there is a set of colleges C_p , such that $c_j \in C_p$ and all the u_p places in C_p have been filled with better students than a_i . Biró et al. [7] showed that if the sets of colleges with common upper quotas is nested, i.e., when $C_p \cap C_q \neq \emptyset$ implies either $C_p \subset C_q$ or $C_p \supset C_q$, then a stable matching always exists. Moreover, a stable matching can be found efficiently by the generalised Gale-Shapley algorithm and there are applicant and college-optimal solutions. However, if the set system is not nested then a stable solution may not exist and the problem of finding a stable matching is NP-hard. Interestingly, the Hungarian application involved nested set systems until 2007 when a legislative change modified the structure of the underlying model and made the set system non-nested, with the possibility of having no stable solution and also making the problem computationally hard.

Here, we show that we can express this stability condition with the use of score-limits, in a similar fashion to the method we described in section 2. However, here we need to assume that there are no ties. We set a score-limit t_p for each set of colleges C_p with common quotas, which is less than or equal to the score of the weakest admitted student if the common quota is filled, and 0 if the common quota is unfilled in the matching. When describing the model in this way, stability implies that if a student a_i is admitted to college c_j then $s_{ij} \geq t_j$ and also $s_{ij} \geq t_p$ for any set of colleges C_p with common quota where C_p includes c_j . Furthermore, if a_i is not admitted to c_j or to any better college of her preference then it must be the case that either $s_{ij} < t_j$ or $s_{ij} < t_p$ for some set of colleges C_p with common quota where C_p contains c_j . These conditions can be formalised with the following set of conditions, where q_j denotes the number of sets of colleges with common quotas involving college c_j , $\{c_j\}$ also being one of them.

$$t_p \leq (1 - x_{ij}) \cdot (\bar{s} + 1) + s_{ij} \text{ for each } (a_i, c_j) \in E \text{ and } c_j \in C_p \quad (20)$$

and

$$s_{ij} + 1 \leq t_p + \left(\sum_{k:r_{ik} \leq r_{ij}} x_{ik} + y_i^p \right) \cdot (\bar{s} + 1) \text{ for each } (a_i, c_j) \in E \text{ and } c_j \in C_p \quad (21)$$

with

$$\sum_{p:c_j \in C_p} y_i^p \leq q_j - 1 \text{ for each } (a_i, c_j) \in E \quad (22)$$

where $y_i^p \in \{0, 1\}$ is a binary variable. These conditions are needed to establish the links between a matching and the corresponding score-limits. However, for stability we also have to ensure that the score-limits are minimal. In case of strict preferences (i.e., when no two students have the same score at colleges belonging to a set of colleges with a common quota), we can ensure the minimality of the score-limits with the following conditions.

Again, we introduce an binary variable f_p for each set of colleges C_p which is equal to zero if the common quota of these colleges is unfilled, by using the following constraints.

$$f_p \cdot u_p \leq \sum_{i:(a_i, c_j) \in E, c_j \in C_p} x_{ij} \text{ for each } C_p \subseteq C \quad (23)$$

Then we ensure that if a college or a set of colleges with a common quota is unfilled then its score-limit is zero.

$$t_p \leq f_p(\bar{s} + 1) \text{ for each } C_p \subseteq C \quad (24)$$

We describe and prove the correctness of the IP model in the following Theorem.

Theorem 9 *Feasibility conditions (1), (2) and (19), with the stability conditions (20), (21) and (22), together with (23) and (24) describe an integer program such that its solutions correspond to stable matching for the College Admissions problem with common quotas.*

Proof: To see the correctness, we have to note first that the matching is feasible if and only if the feasibility constraints (1), (2) and (19) are satisfied, and conditions (23) and (24) are satisfied if and only if the score-limit of any unfilled college or set of colleges is zero.

Now, suppose first that we have a stable solution and we show that all the stability conditions can be satisfied by setting the variables appropriately. When a common quota C_p is filled we set t_p to be equal to the last admitted applicant at any college included in C_p . This ensures that the first set of conditions (20) is satisfied. Let $y_i^p = 0$ if a_i does not meet t_p at any college $c_j \in C_p$ where she applied to, and $y_i^p = 1$ otherwise, with the exception of set $\{c_j\}$, where we set $y_i^j = 0$ if i is admitted to c_j or to a better place. The stability of the matching then implies (22). Finally, let us consider an application (a_i, c_j) where $c_j \in C_p$. If a_i is admitted to c_j or to a better college then the corresponding constraint (21) is satisfied, irrespective of the value y_i^p . Otherwise, suppose that a_i is not admitted to c_j or to any better place. If a_i does not meet the score-limit t_p then (21) is satisfied, obviously, and it is also satisfied when she meets t_p , since $y_i^p = 1$ in that case.

Conversely, suppose that we have a solution for the IP model, and we will show that this ensures the stability of the corresponding matching. If (a_i, c_j) is in the matching then constraints (20) imply that a_i achieves the score-limit of c_j and also the score-limit of every set of colleges with common quota containing c_j . Finally, suppose that a_i is not admitted to c_j or to any better college of her preference. Since one of the additional variables of form y_i^l must be zero, say y_i^p , the corresponding constraint (21) implies that the set of colleges C_p containing c_j has a score-limit t_p greater than s_{ij} . So the matching is indeed stable. ■

Finally we note that if we have ties then we shall use an objective function that minimises the sum of the score-limits or an extended model, similar to the ones described in section 2.

5 Paired applications

In the Hungarian application students can apply for pairs of programmes in case of teachers' studies, e.g. when they want to become a teacher in both maths and physics. In this setting of a College Admission problem with paired applications stability means that if a student is not admitted to a pair of colleges, or to any better college (or pair of colleges) in her list then either of these colleges must have filled its quota with better applicants. In terms of score-limits, either of these colleges must have a score-limit that this student has not achieved.

This problem is similar to the well-known Hospitals/Residents problem with Couples, where residents apply to pairs of positions (see a survey [10]). However, there are some slight differences. Here a student may have both simple and paired applications in her list, thus she can behave both as a single applicant and as a couple at the same time. A paired application in the college admission problem involves two distinct programmes, whilst a couple may apply for a pair of positions at the same hospital in a resident allocation program. So neither of these problems is more general than the other. Nevertheless both problems are NP-hard, since the NP-hardness proof of Ronn [24] for the Hospitals/Residents problem with couples was concerned with a special case where

each hospital has one place only, and this is also a special case of the College Admissions problem with paired applications.

Now, we describe the IP model for college admissions with paired application. We introduce new, artificial colleges C^P , that are pairs of compatible colleges. Let $c_{jk} \in C^P$ be one compatible pair of colleges, where the binary variable of an application from student a_i to this pair of colleges is denoted by $x_{i(jk)}$, and the rank of this application in a_i 's list is denoted by $r_{i(jk)}$. For a more convenient notation, let $e \in E^S$ denote a simple application, and $e \in E^P$ denote a paired application, where the set of all applications is $E = E^S \cup E^P$. With a slight abuse of notation, let $a_i \in e$ mean that application e is coming from applicant a_i , and $c_j \in e$ or $c_{(jk)} \in e$ mean that the application is going to college c_j or pair of colleges $c_{(jk)}$, respectively.

The feasibility constraints can now be modified in the following way.

$$\sum_{e: a_i \in e \in E} x_e \leq 1 \text{ for each } a_i \in A \quad (25)$$

$$\sum_{e: c_j \in e \in E^S} x_e + \sum_{k=1 \dots m} \sum_{e: c_{(jk)} \in e \in E^P} x_e \leq u_j \text{ for each } c_j \in C \quad (26)$$

The stability conditions are expressed with score-limits. There is no change for simple applications.

$$t_j \leq (1 - x_{ij}) \cdot (\bar{s} + 1) + s_{ij} \text{ for each } (a_i, c_j) \in E^S \quad (27)$$

and

$$s_{ij} + 1 \leq t_j + \left(\sum_{e: a_i \in e, r_e \leq r_{ij}} x_e \right) \cdot (\bar{s} + 1) \text{ for each } (a_i, c_j) \in E^S \quad (28)$$

For paired applications, the following conditions must hold.

$$t_j \leq (1 - x_{i(jk)}) \cdot (\bar{s} + 1) + s_{ij} \text{ for each } (a_i, c_{(jk)}) \in E^P \quad (29)$$

$$t_k \leq (1 - x_{i(jk)}) \cdot (\bar{s} + 1) + s_{ik} \text{ for each } (a_i, c_{(jk)}) \in E^P \quad (30)$$

and

$$s_{ij} + 1 \leq t_j + \left(\sum_{e: a_i \in e, r_e \leq r_{i(jk)}} x_e + y_i^{(jk)} \right) \cdot (\bar{s} + 1) \text{ for each } (a_i, c_{(jk)}) \in E^P \quad (31)$$

$$s_{ij} + 1 \leq t_k + \left(\sum_{e: a_i \in e, r_e \leq r_{i(jk)}} x_e + (1 - y_i^{(jk)}) \right) \cdot (\bar{s} + 1) \text{ for each } (a_i, c_{(jk)}) \in E^P \quad (32)$$

Here, the first two conditions, (29) and (30), ensure that the score limit is met for each of the colleges where an applicant is admitted with a paired application. The second two conditions, (31) and (32), imply that if a student a_i is not admitted to a pair of colleges $c_{(jk)}$ or to any better place(s) then it must be the case that she did not achieve the score-limit at one of these colleges. Here again, $y_i^{(jk)} \in \{0, 1\}$ is a binary variable.

Finally, similarly to conditions (23) and (24), we have to ensure that only the filled colleges can have positive score-limits. First we modify (23) as follows.

$$f_j \cdot u_j \leq \sum_{e:c_j \in e \in E^S} x_e + \sum_{k=1 \dots m} \sum_{e:c_{(jk)} \in e \in E^P} x_e \text{ for each } c_j \in C \quad (33)$$

Then, as we did in (24), we ensure that if a college or a set of colleges with a common quota is unfilled then its score-limit is zero.

$$t_j \leq f_j(\bar{s} + 1) \text{ for each } c_j \in C \quad (34)$$

Theorem 10 *The College Admissions problem with paired applications can be described with the solutions of the IP consisting of conditions (25) to (34).*

Proof: Conditions (25) and (26) are satisfied if and only if the corresponding matching is feasible. Regarding the stability constraints, if we consider a single application then (27) ensures that the applicant has reached the score-limit of the college where she has been admitted, and (28) implies that a rejection must have taken place because the applicant has not reached the score-limit of that college. Similarly, for a paired application (29) and (30) are satisfied if and only if the applicant has achieved the score-limits of both colleges where she has been admitted in a paired application. Conditions (31) and (32) ensure that if a paired application is rejected, it must be the case that the applicant has not reached the score-limit of either of the colleges in her paired application. To summarise, each applicant is admitted to the best college or pair of colleges on her list where she has achieved the score-limit(s). Finally, conditions (33) and (34) imply that only those colleges which are filled may reject applications. ■

6 Combining the models into a single IP

In the four previous sections we have developed IP models to deal with each of the four special features which are present in the Hungarian application. However, all of these special features are present in the application simultaneously, and so to provide a solution for real data we need to create a combined model which incorporates all of the constraints. But this task is not easy, since not only do the constraints have to be adjusted, but sometimes the stability definitions may also contradict each other. Since paired applications can be seen as a special case of common quotas, as we described in the previous section, we focus on the different combinations of ties, lower and common quotas.

6.1 Stable score-limits with ties and lower quotas

The feasibility of the matching is characterised by constraints (1) and (14). For stability, we will use (5) to ensure that a student is only admitted if she achieved the score-limit. We need (16) again for group-stability. Here we need a combination of (6) and (15) to enforce stability for open colleges, as follows.

$$s_{ij} + 1 \leq t_j + \left(\sum_{k:r_{ik} \leq r_{ij}} x_{ik} + 1 - o_j \right) \cdot (\bar{s} + 1) \text{ for each } (a_i, c_j) \in E \quad (35)$$

Finally, either we need to minimise the sum of score-limits with the objective function or to use conditions (10), (11), (12) and (13) to ensure that the decrease of any positive score-limit in the solution would cause the violation of a quota.

6.2 Stable score-limits with ties and common quotas

As mentioned in section 4, our IP model for common quotas is based on score-limits, so it is not too difficult to reconcile these two features. The issue that we must resolve is that we cannot require the unfilled colleges or sets of colleges to have a zero score-limit here, as this might have resulted from the rejection of a group of applicants with the same score. So instead of using (23) and (24), either we have to minimise the sum of the score-limits with the objective function to enforce stability, or we shall develop a model similar to the one described in subsection 2.3 for sets of colleges with common quotas.

6.3 The difficulty in reconciling lower and common quotas

The feasibility of the matching is now characterised by constraints (1), (14) and (19). For stability, we need (20) to ensure that a student is only admitted if she achieved the score-limit of the assigned college and also the score-limit for each set of colleges with a common quota that contains this college. For open colleges, the pairwise stability condition can be enforced with the following modification of (21):

$$s_{ij} + 1 \leq t_p + \left(\sum_{k:r_{ik} \leq r_{ij}} x_{ik} + y_i^p + 1 - o_j \right) \cdot (\bar{s} + 1) \text{ for each } (a_i, c_j) \in E \text{ and } c_j \in C_p \quad (36)$$

together with (22).

However, defining group-stability is problematic here. Even if there is a closed college with more unsatisfied students than its lower quota the admission of these students could lead to the violation of a common upper quota. If this happens then the blocking should not be allowed. So combining the constraints for lower and common quotas is challenging. Therefore one might consider the possibility of abandoning the group-stability constraints related to lower quotas. However, in this case, a possible solution would be to close all the colleges. This unsatisfactory scenario could be avoided by setting the objective function such that the number of students admitted becomes the first priority and the minimisation of the total score limits is the second priority.

6.4 Stable score-limits with ties, lower and common quotas

As we have described, the concept of stable stable score-limits for the College Admissions problem with ties can be reconciled with both lower and common quotas, however the latter two are difficult to combine. Therefore in the practical application the organisers should decide what stability conditions they want to satisfy in the first place. One possibility is to drop the group-stability conditions with regard to lower quotas and to use an appropriate objective function to ensure that the solution is not biased. In order to set the right conditions one should do simulations with a real data to see how the model, and the objective function in particular, may influence the resulting solution.

However, as we will see in the coming section, our simulations show that for large markets we were able to resolve only one of the special features, the problem of lower quotas. Therefore the complete solution of a combined model seems unlikely with IP techniques for large problems as the one in Hungarian college admission scheme.

7 Computer simulations

In this section we present numerical results on generated samples and on real data from the Hungarian higher education matching scheme. In the simulations we used the GLPK open source solver (4.55 version), which is a large-scale solver for linear programming, integer linear programming and mixed integer linear programming problems. We kept the default parameter settings for integer and mixed integer problems. The solver was running on a desktop computer with a 2.33 GHz Intel Pentium processor and 6 GB RAM.

In our simulations we focused on the three special features, separately, that are present in the Hungarian application in 2008. But first we tested the solver for the IP formulation of the classical Gale-Shapley model on generated and real data. Then we evaluated the applicability of our IP formulation with score-limits, which are crucial in the Hungarian application when computing stable solutions in case of ties. In the third part we consider lower quotas only, and finally we tested IP formulation for common quotas. As we will describe in detail in this section, we found that the only special feature that we could solve with IP technique for the real data was the feature of lower quotas.

7.1 Basic IP formulations and score-limits

Our first remark is that problems typically had a large number of binary variables, in which case the branch-and-bound algorithm can take extremely long time. Because of this reason we found that our IP formulations with score-limits cannot be solved for large data, even in the simplest case, as we will present in this subsection. Note that for this classical setting we can actually relax the integrality condition for the score-limits, that we did in the simulations, yet, this has not helped to improve the performance of this approach for large instances.

In the first part we used generated samples for estimating the running time. We had n applicants and m programmes, each applicant choosing five programmes uniformly at random without replacement. Thus there are about $\frac{n}{k}$ first place applications at each programme. We fixed the quota at $\frac{n}{2k}$, so every quota is expected to be full. Table 1 contains running times for some relatively small n and m .

This experiment illustrates that the usage of score-limits is not recommended for large scale applications. Note that the score-limit formulations lead to mixed integer linear problems, while the basic IP formulation described in (1), (2) and (3) or (4) is a pure binary problem, which may be solved for larger problems. Indeed, this is what we found in our experiments. Table 2 contains the running time and other details of the simulation with basic IP formulation for generated samples.

It is worth mentioning that the solver generates various cuts (constraints that cut non-integer points from the set of feasible solutions) for the binary problem. These cuts are usually very effective. Sometimes the running time for LP relaxation (without cuts) are larger than the running time for the binary problem.

n	m	size (Kb)	memory used (Mb)	run time (sec)
20	10	13	0.4	0.2
40	10	26	2.6	4.0
60	10	39	16.6	81.3
80	10	52	136.5	1443.8

Table 1: Results for score-limit problems show the time required, memory used and size needed, where the number of applicants, n increases from 20 to 80, and the number of programmes, m , is fixed to 10.

n	m	size (Kb)	memory used (Mb)	run time (sec)
100	10	135	1.2	0.0
500	20	1679	11.5	0.3
2500	30	27883	169.1	28.9
7500	40	190547	1106.7	588.7
12500	40	537863	n.a.	n.a.

Table 2: Results for the classical admission problem on generated samples show the time required, memory used and size needed, where the number of applicants, n increases from 100 to 12500, and the number of programmes, m increases from 10 to 40.

The results in Table 2 show the applicability of basic IP problem for large problems. We note that we could not solve the problem in the last row with our computer due to insufficient memory, but it is not too difficult to increase the problem size that can be solved with more powerful computers.

The generated samples are somewhat extreme, as in the real data there is a correlation amongst the preferences of the applicants: there are prestigious programmes and there are less prestigious ones. In Table 3 we present the results of our simulations of the basic IP problem on real data. We took a sample of the real data in the following way, we chose a programme-group (i.e. programmes in economics) and considered the applications only for these programmes. We found that large numbers of applicants and applications can be handled in this case. The bottleneck for the solvability is the running time, so the set of tractable cases cannot be extended significantly.

# applicants	#programmes	# applications	size (Kb)	memory (Mb)	run time (sec)
13066	392	21671	26757	103.8	17.9
16652	757	27531	31170	123.0	677.3

Table 3: Results for classical admission problems for a filtered data of the Hungarian applications. The table shows the time required, memory used and size needed for two instances, the first with 13066 applicants and 392 programmes, and the second with 16652 applicants and 757 programmes.

n	m	size (Kb)	memory (Mb)	run time (sec)
100	10	174	4.8	24.3
100	20	130	10.6	189.3
500	10	3179	65.3	17201.5

Table 4: Results for randomly generated problems with lower quotas show the time required, memory used and size needed, where the number of applicants, n varies between 100 and 500, and the number of programmes, m varies between 10 and 20.

# applicants	#programmes	# applications	size (Kb)	memory (Mb)	run time (sec)
13066	392 (342;33;17)	29063	26757	114.5	11.1
16652	626 (535;53;38)	27531	34111	136.7	62.8
19764	797 (687;71;39)	32363	41347	n.a.	n.a.

Table 5: Results for admission problems with lower quotas for a filtered data of the Hungarian applications. The table show the time required, memory used and size needed of the solver, where the number of applicants increases from 13066 to 19764, and the number of programmes increases from 392 to 941.

7.2 Lower quotas

We continue the investigation with lower quotas. In the generated sample we change the upper limit to $\frac{3n}{k}$ and we set the lower quota to $\frac{2n}{k}$ for each programme. It is not hard to see that Lemma 6, Lemma 7 and Lemma 8 are unlikely to help in this case, because neither of the programmes is expected to reach its lower quota, as each programme is assigned to around $\frac{n}{k}$ applicants. Further, if we close all undecided programmes expect one then every programme achieves its lower quota with very high probability, since each programme receives $\frac{5n}{k}$ applications on average. Table 4 presents the numerical results.

The results presented in Table 4 suggest that the number of undecided programmes must be very limited if we would like to solve the IP problem. Here, undecided means that we are not able to make a decision about the closure of this programmes based on the efficient filtering process described in section 3. Regarding the Hungarian admission data the situation is a bit more favourable. Programmes with numerous applications usually get more applicants than their lower limits, and the undecided programmes are relatively small programmes where the lower limits and the number of applications are low. Therefore Lemma 6, Lemma 7 and Lemma 8 can help efficiently to reduce the size of the problem. In Table 5 we present results for subsets of the Hungarian data. At the beginning we used Lemma 6 only and we closed the programmes which have fewer applicants than its lower quota. Note that at the ‘programme’ column we recorded the number of open, initially undecided and surely closed programmes in parentheses, i.e., 392 (342;33;17) means that there were 392 programmes where 342 achieved their lower quotas, 33 were initially undecided and 17 were immediately closed because the number of its applications was smaller than its lower quota. As we can see, the presence of lower quotas can even speed up the solution.

Furthermore, if we apply Lemma 8 then we can solve large real instances. When

we considered all the state-financed programmes with 252188 applications, after applying Lemma 8 there remain only 1542 applications. We could filter out some more applications, namely the applications for programmes where the score limit was surely higher than the applicant had and also all the applications for surely closed programmes. After this filtering there remained only 1306 applications. Regarding the 543 total programmes, 375 are surely open, 21 are undecided and 147 are surely closed. The solution for this reduced data could be obtained in 0.1 sec. According to the result obtained, 14 programmes will start from the 21 undecided programmes.

Therefore, we conclude that the special feature of lower quotas is tractable in practice, even though it makes the problem NP-complete. With the help of some useful Lemmas we can reduce the number of undecided programmes significantly and then we can solve the remaining problem quickly with our IP formulation.

7.3 Common quotas

The structure of common quotas in the Hungarian higher education matching scheme is as follows (more details can be found in [7] and [6]). The students can study at the programmes under two different contracts, state-financed and self-financed. There are nationwide common quotas in each subject (such as computer science) with regard to the state-financed student, e.g. the government may decide to sponsor the study of 3000 students in computer science in Hungary. In addition, there are faculty quotas for each programme for the number of students studying under any contract, e.g. Budapest University of Technology and Economics (BME) may have a common quota of 500 for computer science students, involving both state-financed and self-financed students. The score of a student is the same at any computer science programme, so the same ranking can be used when comparing the students that apply for a state-financed seat in a subject, but the score of this student can differ in a different subject, e.g. for economics (in particular, for computer science her secondary school grade in physics will count whilst for economics they consider her grade from history instead).

As we have shown in our earlier paper [7], the existence of a stable solution is not guaranteed and the problem of finding a stable solution is NP-complete, even in the above realistic setting in Hungary. However, we can use a specialised IP formalisation here. In particular, for our simulation we decided not to use score-limit based IPs, since it turned out to be infeasible to solve for larger markets, as we described at the beginning of this section. Thus we created a new set of IP constraints, that we specify below.

Let C refer to the set of programmes in this context. For a programme $c_j \in C$, we denote the state-financed form by \tilde{c}_j and the self-financed form by \bar{c}_j . As we noted, in the Hungarian application there are no separate quotas for these two forms⁴, just an overall upper quota u_j for the total number of students assigned to programme c_j . Furthermore, there is also a nationwide quota for the state-financed students in this subject, that we denote with the relationship $\tilde{c}_j \in C_p$, where C_p is a set of state-financed seats in some subject with a common upper quota u_p . We partition the set of applications E according to their forms, E_S will denote the applications for state-financed seats and E_P denotes

⁴Actually until 2007 there were indeed separate quotas for the number of students in these two forms, and because of that the basic structure of the problem was different, the set system of common quotas was nested, which implied that a stable solution could be found efficiently, see details in [7].

the applications for self-financed seats. So with a slight abuse of notation we write that $(a_i, c_j) \in E_S$ if a_i applies to \tilde{c}_j and we write that $(a_i, c_j) \in E_P$ if a_i applies to \bar{c}_j

Regarding the feasibility conditions, besides keeping (1), we adjust and replace (2) and (19) with the following conditions.

$$\sum_{i:(a_i, c_j) \in E} x_{ij} \leq u_j \text{ for each } c_j \in C \quad (37)$$

$$\sum_{i:(a_i, c_j) \in E_S, \tilde{c}_j \in C_p} x_{ij} \leq u_p \text{ for each } C_p \quad (38)$$

The stability conditions are modified as follows. With regard to the applications for self-financed seats, we use the classical stability constraints (3) written up for the faculty quotas, namely

$$\left(\sum_{k:r_{ik} \leq r_{ij}} x_{ik} \right) \cdot u_j + \sum_{h:(a_h, c_j) \in E, s_{hj} > s_{ij}} x_{hj} \geq u_j \text{ for each } (a_i, c_j) \in E_P. \quad (39)$$

The stability constraints with regard to the applications for state-financed seats need to be written up in more complicated way, as we do not know in advance whether the faculty or the nationwide quota will be binding and thus explain the rejection of some applications of this sort. For each programme c_j , we introduce a binary variable y_j that will be zero if the faculty quota will be binding and explain the rejection of some applications. Furthermore, we also introduce a binary variable y_j^p , where $\tilde{c}_j \in C_p$ and C_p is a nationwide common quota applied for the state-financed students in a particular subject. When the rejection of an application for a state-financed seat at programme c_j is explained by the national common quota then y_j^p will be zero. We write up these constraints formally for all applications for state-financed seats.

$$\left(\sum_{k:r_{ik} \leq r_{ij}} x_{ik} \right) \cdot u_j + \sum_{h:(a_h, c_j) \in E, s_{hj} > s_{ij}} x_{hj} \geq (1 - y_j)u_j \text{ for each } (a_i, c_j) \in E_S \quad (40)$$

$$\left(\sum_{k:r_{ik} \leq r_{ij}} x_{ik} \right) \cdot u_p + \sum_{h:(a_h, c_l) \in E_S, \tilde{c}_l \in C_p, s_{hl} > s_{ij}} x_{hl} \geq (1 - y_j^p)u_p \text{ for each } (a_i, c_j) \in E_S, c_j \in C_p \quad (41)$$

$$y_j + y_j^p \leq 1 \text{ for each } \tilde{c}_j \in C_p \quad (42)$$

To see the correctness of our formulation, we will use a similar argument as in the proof of Theorem 9, that we describe briefly hereby. The rejections of state-financed applications at a programme must be explained by either the fulfillment of the faculty quota by better candidates applying under any kind of contract or the fulfillment of the corresponding nationwide quota with better students applying for state-financed seats. Given a stable matching, we need to set the variables y_j and y_j^p to be zero depending on which quota is

binding for the state-financed application at programme c_j . In the other direction, if we have a feasible binary solution for the above IP then we must set at least one of these two variables, y_j or y_j^p , to be zero. Then the corresponding constraints, either (41) or (42) will be binding, implying that the rejection of that application for a state-financed seat was appropriate, so the corresponding matching is indeed stable.

It can help (i.e. the running time decreases a bit) if we introduce some more constraints. Variable y_j can be 1 only if the relevant nationwide quota, C_p is full, where the latter is indicated with a binary variable Y_p . We can enforce this statement with the following constraints.

$$y_j \leq Y_p \text{ for each } \tilde{c}_j \in C_p$$

and

$$\sum_{(a_i, c_j) \in E_S, \tilde{c}_j \in C_p} x_{ij} \geq Y_p u_p.$$

Similarly, y_j^p can be 1 only if the appropriate programme quota is full:

$$\sum_{i: (a_i, c_j) \in E} x_{ij} \geq y_j^p u_j \text{ for each } \tilde{c}_j \in C_p$$

In our first experiment we generated a market with 100 applicants and 2 programmes, where each programme accepts both state-financed and self-financed applicants, and a common quota is set for all state-financed applicants. All applicants list all the 4 possible contracts in their preference lists in a random order. Each programme has a common quota of 30 and there is a common quota of 25 for state-financed applicants. The running time for this market was 827 sec. which is remarkably high considering the size of the example.

However, the running time becomes tolerable if the nationwide quota for state-financed applicants is relatively small or relatively large with respect to the common quotas of programmes. In the first the nationwide quota will bound for the state-financed contracts in all programmes. In the second case the national quota will be non-binding and the common quotas of the programmes will be full. In Table 6 we present the summary of our simulations. Here m denotes the number of programmes and each programme has a common quota of $\frac{n}{2k}$ that applies to both the state-financed and self-financed contracts. Each applicant chooses five contracts randomly from the $2k$ possibilities. In the simulation first we set the nationwide quota to be n , so it will not bound. Note that these results could be obtained without using the model of common quotas but in this way we have a notion about the boundary of the solvability. The size of text files describing the IP increased considerably due to the stability constraints for the nationwide quota.

Now we describe a heuristic solution that enabled us to obtain a stable solution for the above setting. We can determine the score limits for the programmes when we suppose that the nationwide quota is non-binding. Next we set the nationwide quota to be 95% of the state-financed applications. Then we set the quotas of the programmes so high that none of them will be full. We re-solve the problem, and we get a score-limit for the nationwide quota. We continue with the following search process: if the score-limit for a programme is higher than the score-limit for the nationwide quota then we suppose that the programme quota will be binding. If the score-limit for the nationwide quota is

n	m	state financed	size (Kb)	memory used (Mb)	run time (sec)
500	10	142	10239	75.5	4.5
1500	15	382	90813	640.9	1326.2
2500	15	n.a.	255196	n.a.	n.a.

Table 6: Results for admission problems with common quotas show the time required, memory used and size needed, where the number of applicants is equal to the nationwide quota and it increases from 500 to 2500, while the number of programmes varies between 10 and 15.

higher then we suppose that this quota will bound. If we guess incorrectly then we get an infeasible problem. As we see in Table 6, we get the results relatively quickly (even more quickly if the problem is unfeasible), so we can have another try by changing the set of programmes for which the programme quota is binding (a change is appropriate where the the two score-limits are close to each other). In this heuristic way we could get a result for the common quota model, where the running times are similar to the running times described in Table 6.

However, the above method does not work if there are more nationwide quotas. We generated a sample with two nationwide quotas and observed the following issues. It can happen that an applicant’s first choice is a self-financed seat in the first subject with nationwide quota where he has a relatively low score (for him the nationwide quota is irrelevant, since he applies for a self-financed place). In the first step of our heuristic the nationwide quota is set high enough that it will not bind. Our applicant is rejected due to the programme quota. The applicant’s second choice is a state-financed seat belonging to the other nationwide quota, where he has a relatively high score, so he will be accepted. In the next step of our heuristic we set the programme quota very large, and we set the nationwide quota to its real value to obtain score limits for the nationwide quota. But in this market our applicant will be accepted to his first choice so the score limit for the second nationwide quota decreases. In our simulation with this heuristic we found that the score limits for nationwide quotas were lower than the minimum of the score limits for the programmes that we obtained in the previous step.

Finally, we tested the solvability of our IP formulations for two nationwide programmes, without using heuristics. In the generated instance we had 100 applicants choosing from 20 programmes. The first 10 programmes belong to the first nationwide quota and the remaining belong to the other one. The applicants have different scores in the two subjects, so they can be ranked differently with regard to the two nationwide quotas. All the programmes have state-financed and self-financed forms (as before, nationwide quotas restrict the number of state-financed applicants only). Each applicant chooses five options randomly from the 40 possibilities. Each programme quota is set to be 4 and each nationwide quota is 20. It is worth mentioning again that this problem setting is still NP-complete, as shown in [7]. As we can see from Table 7, we managed to obtain a stable solution, but the running time was rather large (about 10 hours).

n	m	state financed	size (Kb)	memory used (Mb)	run time (sec)
100	2x10	20;20	291	323.7	39560.1

Table 7: Results for admission problems with common quotas show the time required, memory used and size needed, where the number of applicants is 100, the number of programmes is 20 with two nationwide quotas, each containing 10 programmes.

Conclusion

As we described in the previous section, the combination of the four special features result in interesting challenges in two ways. First, we need to define appropriate stability criteria when both lower and upper quotas are present. Second, we need to combine the separate integer programmes into a single programme that would result in a suitable solution for the real application.

It would be also important to know whether these IP formulations may be solved within a realistic timescale for such a large scale application as the Hungarian higher education matching schemes, with around 100000 applicants. We have conducted such simulations with IP formulations for the problems of score-limits with possible ties, lower quotas and common quotas. From these experiments we concluded that the case of lower quotas is tractable in practice, but the other two approaches turned to be infeasible to solve for such large-scale problems. Therefore it seems to be very challenging to tackle the entire problem with one complex IP formulation.

However, one could always try to solve our special college admissions problems with other approaches, e.g. with different integer programming formulations, or with constraint programming methods. Finally, it is worth mentioning that our models may be useful in other applications as well, such as controlled school choice (see e.g. [2]), resident allocation with distributional constraints (see e.g. [19]), or for finding stable solutions with additional restrictions, such as matchings with no Pareto-improving swaps [17].

References

- [1] D. Abraham, A. Blum and T. Sandholm. Clearing Algorithms for Barter-Exchange Markets: Enabling Nationwide Kidney Exchanges. *In Proceedings of ACM-EC 2007: the Eighth ACM Conference on Electronic Commerce, pages 295304. ACM, 2007.*
- [2] A. Abdulkadiroglu and L. Ehlers. Controlled School Choice. *Working paper, 2007.*
- [3] E.M. Azavedo and J.D. Leshno. A supply and demand framework for two-sided matching markets. *Working paper, 2012.*
- [4] M. Baïou and M. Balinski. The stable admissions polytope. *Mathematical Programming*, 87(3), Ser. A:427–439, 2000.
- [5] P. Biró. Student Admissions in Hungary as Gale and Shapley Envisaged. *Technical Report*, no. TR-2008-291 of the Computing Science Department of Glasgow University, 2008.

- [6] P. Biró. University admission practices - Hungary. matching-in-practice.eu, accessed on 23 May 2012
- [7] P. Biró, T. Fleiner, R.W. Irving and D.F. Manlove. The College Admissions problem with lower and common quotas. *Theoretical Computer Science*, 411:3136–3153, 2010.
- [8] P. Biró, R.W. Irving and I. Schlotter. Stable matching with couples – an empirical study. *ACM Journal of Experimental Algorithmics*, 16, Article No.: 1.2, 2011.
- [9] P. Biró and S. Kiselgof. College admissions with stable score-limits. *Central European Journal of Operations Research*, 23(4):727–741, 2015.
- [10] P. Biró and F. Klijn. Matching with Couples: a Multidisciplinary Survey. *International Game Theory Review*, 15(2), 1340008, 2013.
- [11] P. Biró and I. McBride. Integer programming methods for special college admissions problems. In *Proceedings of COCOA 2014: the 8th Annual International Conference on Combinatorial Optimization and Applications*, volume 8881 of LNCS, pp 429-443, Springer, 2014.
- [12] P. Biró, I. McBride and D.F. Manlove. The Hospitals / Residents problem with Couples: Complexity and Integer Programming models. In *Proceedings of SEA 2014: the 13th International Symposium on Experimental Algorithms*, vol 8504 of LNCS, pp 10-21, Springer, 2014.
- [13] T. Fleiner. On the stable b -matching polytope. *Mathematical Social Sciences*, 46:149–158, 2003.
- [14] T. Fleiner and Zs. Jankó. Choice Function-Based Two-Sided Markets: Stability, Lattice Property, Path Independence and Algorithms. *Algorithms*, 7(1):32–59, 2014.
- [15] D. Gale and L.S. Shapley. College Admissions and the Stability of Marriage. *American Mathematical Monthly*, 69(1):9–15, 1962.
- [16] D. Gale and M.A.O. Sotomayor. Some Remarks on the Stable Matching Problem. *Discrete Applied Mathematics*, 11(3):223–232, 1985.
- [17] R.W. Irving. Stable matching problems with exchange restrictions. *Journal of Combinatorial Optimization*, 16:344–360, 2008.
- [18] R.W. Irving and D.F. Manlove. Approximation algorithms for hard variants of the stable marriage and hospitals/residents problems. *Journal of Combinatorial Optimization*, 16:279–292, 2008.
- [19] Y. Kamada and F. Kojima. Stability and Strategy-proofness for Matching with Constraints: A Problem in the Japanese Medical Match and its Solution. *American Economic Review (P&P)*, 102(3):366–370, 2012.
- [20] A. Kwanashie and D.F. Manlove. An Integer Programming approach to the Hospitals / Residents problem with Ties. In *Proceedings of OR 2013: the International Conference on Operations Research*, pages 263-269, Springer, 2014.

- [21] E.J. McDermid and D.F. Manlove. Keeping partners together: algorithmic results for the hospitals/residents problem with couples. *Journal of Combinatorial Optimization*, 19:279–303, 2012.
- [22] D.F. Manlove and G. O’Malley. Paired and altruistic kidney donation in the UK: Algorithms and experimentation. In *Proceedings of SEA 2012: the 11th International Symposium on Experimental Algorithms*, vol 7276 of LNCS, pp 271-282, Springer, 2012.
- [23] A. Podhradsky. Stable marriage problem algorithms. *Master’s thesis, Masaryk University, Faculty of Informatics*, 2010.
- [24] E. Ronn. NP-complete stable matching problems. *Journal of Algorithms*, 11:285–304, 1990.
- [25] A.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.
- [26] A.E. Roth. A Natural Experiment in the Organization of Entry-Level Labor Markets: Regional Markets for New Physicians and Surgeons in the United Kingdom. *American Economic Review*, 81:415–440, 1991.
- [27] A.E. Roth. On the Allocation of Residents to Rural Hospitals: A General Property of Two-Sided Matching Markets. *Econometrica*, 54(2):425–427, 1986.
- [28] A.E. Roth, U.G. Rothblum and J.H. Vande Vate. Stable matchings, optimal assignments, and linear programming. *Mathematics of Operations Research*, 18(4):803–828, 1993.
- [29] A.E. Roth, T. Sönmez and M.U. Ünver. Efficient Kidney Exchange: Coincidence of Wants in Markets with Compatibility-Based Preferences. *American Economic Review*, 97(3):828–851, 2007.
- [30] U.G. Rothblum. Characterization of stable matchings as extreme points of a polytope. *Mathematical Programming*, 54(1, Ser. A):57–67, 1992.
- [31] J. Sethuraman, C-P. Teo and L. Qian. Many-to-one stable matching: geometry and fairness. *Mathematics of Operations Research*, 31(3):581–596, 2006.
- [32] M.U. Ünver. Backward Unraveling over Time: The Evolution of Strategic Behavior in the Entry-Level British Medical Labor Markets. *Journal of Economic Dynamic Control*, 25:1039–1080, 2001.