Stability in the Hospitals / Residents problem with Couples and Ties: Mathematical models and computational studies

Maxence Delorme(1), Sergio García(1), Jacek Gondzio(1), Joerg Kalcsics(1), David Manlove(2), William Pettersson(2)

(1) *School of Mathematics, University of Edinburgh, Edinburgh EH9 3FD, United Kingdom*
(2) *School of Computing Science, University of Glasgow, Glasgow G12 8QQ, United Kingdom*

Corresponding author `m.delorme@tilburguniversity.edu`

Abstract

In the well-known Hospitals/Residents problem (HR), the objective is to find a stable matching of doctors (or residents) to hospitals based on their preference lists. In this paper, we study HRCT, the extension of HR in which doctors are allowed to apply in couples, and in which doctors and hospitals can include ties in their preference lists. We first review three stability definitions that have been proposed in the literature for HRC (the restriction of HRCT where ties are not allowed) and we extend them to HRCT. We show that such extensions may bring undesirable behaviour and we introduce a new stability definition specifically designed for HRCT. We then introduce unified Integer Linear Programming (ILP) models, where only minor changes are required to switch from one definition to the other. We propose three improvements to decrease the average solution time of each ILP model based on preprocessing, dummy variables, and valid inequalities. We show that our models can be solved more than a hundred times faster when these improvements are used. In addition, we also show that the stability definition chosen has a minor impact on the solution quality (average matching size) and time required to obtain the solution, but for a specific set of instances, stable matchings are significantly less likely to exist for one particular definition compared to the other definitions. We also provide insights relating to how certain parameters such as the tie density, the number of couples, and the difference between the number of positions available in the hospitals and the number of doctors, might affect the average matching size.

Keywords: Hospitals / Residents problem with Couples, Ties and incomplete lists, Stable matching, Exact algorithms.

1 Introduction

1.1 Background

In the Hospitals / Residents problem (HR), we are given a set of intending junior doctors (residents) and a list of hospitals where the doctors can complete their foundation training (see, e.g., the UK Foundation Programme [33] or the National Resident Matching Program (NRMP) in the United States [31]). Each doctor has a *preference list* that consists of a subset of hospitals ranked in strict order of preference to which the doctor wishes to apply. Each hospital has also a preference list, consisting of all its applicants ranked in strict order of preference, and a *capacity*, which is the maximum number of doctors it can accommodate. The objective is to find a matching that does not contain any *blocking pair*, which is a doctor and a hospital that are not currently matched together, but would prefer to be matched to each other rather than to remain with their current assignee/s (if any). If a matching does not admit any blocking pair, it is called a *stable matching*. Stability has been shown to be a key condition for the successful operation of real-world centralised matching systems [31].

Many countries used to model the assignment of their junior doctors to hospitals via HR (e.g., in the United States [31]) and used tailored techniques such as the well-known Gale and Shapley algorithm [11] to find a stable matching. However by 1984 the solutions obtained were not considered satisfactory anymore due to "contemporary issues" [31]. In particular, it was mentioned that (i) "participants submit rank orders (i.e., strict preferences) even when they may be indifferent between some potential matches", and (ii) "increasing numbers of medical students marry other medical students and seek to be assigned positions in the same community". Three extensions of HR were introduced in the literature to deal with these issues: the Hospitals / Residents problem with Ties (HRT), the Hospitals / Residents problem with Couples (HRC), and the Hospitals / Residents problem with Couples and Ties (HRCT) (see Manlove [24]).

While the concept of a blocking pair is straightforward in HR, it is more complex when ties are taken into account. Indeed, no fewer than three stability definitions (each of them with its own definition of a blocking pair) were introduced in the literature for HRT: weak stability, strong stability, and super-stability [13, 23, 15, 17]. It is worth mentioning that weak stability, in which both the doctor and the hospital involved in a blocking pair must strictly prefer one another to their current assignees, is the most commonly used stability definition in recent papers and real-world HRT applications. When couples are taken into account and we are given an instance of HRC, a matching is called stable if it contains neither a blocking pair nor a *blocking coalition* (the adaptation of a blocking pair for couples). If the couples are not allowed to choose the same hospital, or if the hospitals have unitary capacity, there exists only one kind of blocking coalition, and thus, only one natural extension of stability [12, Section 1.6.6]. When couples are allowed to apply to the same hospital however, there is no consensus in the literature on the definition of a blocking coalition between a couple and the same hospital. As a result, three stability definitions were proposed in the literature: MM-stability by McDermid and Manlove [28], BIS-stability by Biró, Irving, and Schlotter [5], and KPR-stability by Kojima, Pathak, and Roth [20].

1.2 Our contribution

In this paper, we study these three stability definitions and we extend each of these to HRCT. We show that such extension might bring about unexpected behaviour in practical applications (i.e., giving rise to a blocking pair that should not in fact be considered as blocking in reality, or vice versa). We also introduce KPR$^+$-stability, a modified version of KPR-stability specifically designed to handle ties in the preference lists. We then introduce unified Integer Linear Programming (ILP) models, where only minor changes are required to switch from one definition to the other. We then propose three significant improvements to decrease the average running times of each ILP model and we measure their effectiveness through extensive computational experiments. In addition, we show that the stability definition chosen has only a very minor impact on the solution quality (average matching size) and the time required to obtain the solution, but we outline specific cases in which there are BIS-stable and KPR-stable matchings but no MM-stable matchings. We also provide insights about how certain parameters might affect the average matching size. In particular, we show that the average matching size grows as the tie density increases and that the number of unfilled positions reaches its peak when the difference between the number of positions available in the hospitals and the number of doctors is zero. While the proportion of doctors in couples does not have a significant impact on the matching size, we observe that the time required to solve an instance grows as the proportion of doctors in couples increases.

1.3 Related work

To the best of our knowledge, the HR problem model was first defined by Gale and Shapley [11]. The authors studied the equivalent problem of assigning students to capacitated colleges. They proposed a polynomial-time algorithm to solve the problem and proved that a stable matching always exists. As our work focuses on HRC and HRCT, we do not provide a thorough literature review on HR and HRT. Instead, we refer the reader to the book of Manlove [24] that has a dedicated section on each problem, and the recent paper of Delorme et al. [9] that focuses on HRT.

Regarding important theoretical results on HRC, Roth [31] showed that an instance of HRC might not have any stable matching. Ronn [30] proved that the problem of deciding whether a stable matching exists is $\mathcal{NP}$-complete. These results hold even if each hospital has unitary capacity and there are no single doctors. Aldershof and Carducci [1] showed that if the preference lists are not complete, stable matchings may have different sizes. These results hold for MM-, BIS-, and KPR-stability.

Regarding practical applications, Roth and Peranson [32] reported on the design of the clearinghouse adopted by the NRMP in the United States, for which the HRC problem model applies. The NRMP is the largest known programme to match doctors to hospitals, with 40 084 doctors applying to 37 256 positions across 5859 distinct hospital programmes in 2020 [29].

Aldershof and Carducci [2] and Veskioja and Võhandu [35] proposed the use of genetic algorithms for HRC in the special case where hospitals have unitary capacity. Bianco, Hartke, and Larimer [4] also studied HRC with unitary capacity and used acceptability graphs to characterise the existence of stable matchings. Klaus and Klijn [18] showed that, for instances of HRC with so-called *weakly responsive preferences*, a stable matching always exists and can be found in polynomial time. They also showed that, for preferences satisfying this property, one can always arrive at a stable matching, starting from an arbitrary matching, by satisfying a sequence of blocking pairs [19]. Cantala [8] suggested a special case of HRC involving *tiered preferences* that arise from geographical constraints.

In a dedicated section of their book, Gusfield and Irving [12, Section 1.6.6] discussed HRC and introduced the first definition of a stable matching as well as an example instance of

HRC that does not admit a stable matching. McDermid and Manlove [28] remarked that this definition did not explicitly indicate how to define a blocking coalition involving a hospital h and a couple, each of whom wish to move to h; their definition of HRC stability, called MM-stability, covered this particular case. They also showed that determining whether an MM-stable matching exists is $\mathcal{NP}$-complete, even in the restricted case where each single doctor ranks at most 3 hospitals, each couple ranks at most 2 pairs of hospitals, each hospital ranks at most 4 doctors, and each hospital has a capacity of 1 or 2. Marx and Schlotter [26] studied the applicability of a local search procedure to find large stable matchings under MM-stability, and Biró, Manlove, and McBride [7] gave an ILP model for finding a largest size MM-stable matching in an instance of HRC.

Biró, Irving, and Schlotter [5] introduced the concept of BIS-stability for HRC (which is distinct to MM-stability) and proposed various heuristic algorithms to find BIS-stable matchings. They also showed that finding a BIS-stable matching becomes more difficult as the number of couples increases.

Kojima, Pathak, and Roth [20] introduced the notion of KPR-stability for HRC, and gave theoretical results and a thorough analysis of data from the matching market for psychologists in the United States. Ashlagi, Braverman, and Hassidim [3] presented an algorithm for finding a KPR-stable matching. They also gave results regarding the likelihood of finding a KPR-stable matching using their algorithm depending on the size of the market and the number of couples. Drummond, Perrault, and Bacchus [10] used a SAT approach to solve HRC under KPR-stability.

Biró and Klijn [6] surveyed the different stability definitions proposed in the literature and discussed various contributions on the topic from different perspectives (computer science, economy, and game theory). However, they did not tackle the model implementation aspect nor the computational behaviours of each definition.

1.4 Layout of the paper

The rest of the paper is organised as follows. Section 2 defines HRT while Section 3 defines HRCT and contains our new stability definitions. In Section 4, we introduce our new ILP models. In Section 5 we detail three model improvements, and we provide extensive computational experiments in Section 6. Finally, conclusions are given in Section 7.

2 The Hospitals / Residents problem with Ties

2.1 Problem definition

An instance I of HRT comprises a set D of n_D doctors (or residents) and a set H of n_H hospitals, where each doctor (respectively hospital) ranks a subset of the hospitals (respectively doctors) in order of preference, possibly with ties. In addition, each hospital $h \in H$ has a finite capacity c_h. We will assume that ties may occur in the hospitals' preference lists only and that doctors' preference lists are strictly ordered. This is reasonable from a practical point of view since typically doctors' preference lists are short and it is plausible to expect the doctors to rank their acceptable hospitals in strict order. On the other hand, the hospitals' preference lists are typically much longer and it is impractical to expect a large hospital to rank all of its applicants in strict order.

We say that a doctor $d \in D$ finds a hospital $h \in H$ *acceptable* if h belongs to d's preference list, and we define acceptability for a hospital in a similar way. If d and h find each other acceptable, then we call (d, h) an *acceptable pair*. We assume that preference lists are *consistent*, that is, d finds h acceptable if and only if h finds d acceptable.

A *matching* M in I is a set of acceptable pairs such that each doctor appears in at most one pair of M and each hospital h appears in at most c_h pairs of M. If doctor d appears in a pair of M, we say that d is *matched*, otherwise d is *unmatched*. If hospital h appears in strictly fewer than c_h pairs in M, we say that h is *undersubscribed*, otherwise h is *fully subscribed*.

We denote by $M(d)$ the hospital to which doctor d is matched ($M(d) = \emptyset$ if the doctor is unmatched). Similarly, we use $M(h)$ to denote the set of doctors assigned to hospital h. In particular, we observe that if $M(d) = h$, then $d \in M(h)$, and vice-versa.

Definition 1. *Let I be an instance of HRT and let M be a matching in I. A doctor–hospital pair $(d, h) \in (D \times H) \setminus M$ is a* blocking pair of M *if*

DH1- (d, h) is an acceptable pair; and

DH2- d is either unmatched, or it strictly prefers h to $M(d)$; and

DH3- h is either undersubscribed or it strictly prefers d to some member of $M(h)$.

M is said to be stable *if it admits no blocking pair.*

Example 1. *Let us consider an HRT instance with three hospitals and three doctors with the following preference lists (ties are denoted by square brackets) and capacity information:*

$$\begin{array}{llll} d_1: & h_1\ h_2\ h_3 & h_1\ (c_1 = 1): & d_2\ d_1\ d_3 \\ d_2: & h_1\ h_3 & h_2\ (c_2 = 1): & [d_1\ d_3] \\ d_3: & h_2\ h_1 & h_3\ (c_3 = 2): & [d_1\ d_2] \end{array}$$

Doctor d_1 prefers h_1 to h_2 to h_3. Hospital h_1 has unitary capacity and prefers d_2 to d_1 to d_3. Hospital h_2 is indifferent between d_1 and d_3. The matching $M = \{(d_1, h_1), (d_2, h_3), (d_3, h_2)\}$ of size 3 is not stable. Indeed, (d_2, h_1) is a blocking pair since h_1 strictly prefers d_2 to its current assignee d_1, and d_2 prefers h_1 to its current assigned hospital h_3. The matching $M = \{(d_1, h_2), (d_2, h_1)\}$ of size 2 is stable: h_1 and h_2 have their first choice, and even though h_3 is undersubscribed, both its applicants (d_1 and d_2) are assigned to hospitals they prefer to h_3. A larger stable matching of size 3 exists: $M = \{(d_1, h_3), (d_2, h_1), (d_3, h_2)\}$. □

In theory, the HRT definition allows an arbitrary number of preferences to be expressed by any doctor. However in practice, this number is usually short: 10 on average for example for the 2020 matching run of the NRMP [29]. It is well-known that, given an instance of HRT with incomplete lists, stable matchings of different sizes may exist [25]. The size of a matching is equal to the number of doctors assigned to any hospital of their preference list. Obviously, larger stable matchings are favoured as they reduce the number of unassigned doctors, thus reducing the overall level of unhappiness from the participants. We let MAX-HRT denote the problem of finding a stable matching of maximum size in a given instance of HRT.

2.2 ILP model for MAX-HRT

The first ILP models for HR with unitary capacity and complete preference lists (also known as the stable marriage problem) were proposed in the late 1980s by Gusfield and Irving [12] and by Vande Vate [34]. These models can easily be extended to MAX-HRT (see Kwanashie and Manlove [21] and Delorme et al. [9]). In the following, we introduce the notation used in our model (taken from [9]).

When reasoning about models, we will use i and j to represent a doctor and hospital, rather than d and h, respectively, as i and j are by convention more typically used as subscript variables. Let us consider the following notation:

- $H(i)$ is the set of hospitals acceptable for doctor i ($i = 1, \dots, n_D$).
- $D(j)$ is the set of doctors acceptable for hospital j ($j = 1, \dots, n_H$).
- c_j is the capacity of hospital j ($j = 1, \dots, n_H$).
- $r^D_j(i)$ is the rank of hospital j for doctor i, defined as the integer k such that j belongs to the kth most-preferred tie group in i's list ($i = 1, \dots, n_D,\ j \in H(i)$). Note that a tie group is composed of one hospital. The smaller the value of $r^D_j(i)$, the better hospital j is ranked for doctor i.
- $r^H_i(j)$ is the rank of doctor i for hospital j, defined as the integer k such that i belongs to the kth most-preferred tie group in j's list ($j = 1, \dots, n_H,\ i \in D(j)$). Note that a tie group is composed of one or several doctors. The smaller the value of $r^H_i(j)$, the better doctor i is ranked for hospital j.
- $H^{\leq}_j(i)$ is the set of hospitals that doctor i ranks at the same level or better than hospital j, that is, $H^{\leq}_j(i) = \{j' \in H(i) : r^D_{j'}(i) \leq r^D_j(i)\}$ ($i = 1, \dots, n_D,\ j \in H(i)$).
- $D^{\leq}_i(j)$ is the set of doctors that hospital j ranks at the same level or better than doctor i, that is, $D^{\leq}_i(j) = \{i' \in D(j) : r^H_{i'}(j) \leq r^H_i(j)\}$ ($j = 1, \dots, n_H,\ i \in D(j)$).

By introducing binary decision variables x_{ij} that take value 1 if doctor i is matched with hospital j, and 0 otherwise ($i = 1, \dots, n_D,\ j \in H(i)$), MAX-HRT can be modelled as follows:

$$\max \quad \sum_{i=1}^{n_D} \sum_{j \in H(i)} x_{ij} \tag{1}$$

$$\text{s.t.} \quad \sum_{j \in H(i)} x_{ij} \leq 1, \qquad i = 1, \dots, n_D, \tag{2}$$

$$\sum_{i \in D(j)} x_{ij} \leq c_j, \qquad j = 1, \dots, n_H, \tag{3}$$

$$c_j \left(1 - \sum_{q \in H^{\leq}_j(i)} x_{iq}\right) \leq \sum_{p \in D^{\leq}_i(j)} x_{pj}, \qquad i = 1, \dots, n_D,\ j \in H(i), \tag{4}$$

$$x_{ij} \in \{0, 1\}, \qquad i = 1, \dots, n_D,\ j \in H(i). \tag{5}$$

The objective function (1) maximises the number of doctors assigned while constraints (2) ensure that each doctor is matched with at most one hospital. Capacity constraints (3) impose that each hospital is matched with at most c_j doctors. Stability constraints (4) rule out the existence of any blocking pair. More specifically, the latter ensure that if doctor i is not matched with hospital j or any other hospital they rank at the same level or better than j (i.e., $\sum_{q \in H_j^{\leq}(i)} x_{iq} = 0$), then hospital j is fully subscribed with doctors it ranks at the same level or higher than i (i.e., $\sum_{p \in D_i^{\leq}(j)} x_{pj} \geq c_j$).

3 Hospitals / Residents problem with Couples and Ties

3.1 Problem definition

An instance I of HRCT comprises a set S of n_S single doctors, a set C comprising n_C pairs of doctors (couples), and a set H of n_H hospitals. A doctor cannot be in more than one couple, and similarly a doctor cannot be both single and a member of a couple. We denote by D the set containing all the $n_S + 2n_C$ doctors, single or in a couple. Each single doctor ranks a subset of hospitals, each couple ranks a subset of pairs of hospitals, and each hospital ranks a subset of doctors (each member of a couple is ranked individually). This definition of HRCT allows for ties in the preference list of any agent (single doctor, couple, or hospital). Real-world applications often restrict some set of agents to not allow ties. We refer to two specific such specialisations in this work that commonly arise as HRC-TCH (Ties in Couples' and Hospitals' lists) and HRC-TC (Ties in Couples' lists only). These arise as single doctors are often limited to only a small number of strict preferences (as in the case of the NRMP [29]), while couples' preference lists may in general be much longer than single doctors' preference lists as we will see in Section 6, and hospitals may have to rank a significant number of applicants, hence it may be too restrictive to force either hospitals or couples to strictly order their preference lists.

As before, each hospital has a limited capacity, and the definition of an acceptable pair between a single doctor and a hospital is the same for HRCT as it is for HRT. We say that a couple $c \in C$ finds a pair of hospitals $(h_1, h_2) \in H \times H$ acceptable if (h_1, h_2) belongs to the preference list of c. We then define an acceptable pair for the couple as $\{(d_1, h_1), (d_2, h_2)\}$. Partial choices $\{(d_1, h_1), (d_2, \emptyset)\}$ and $\{(d_1, \emptyset), (d_2, h_2)\}$ may be allowed and represent the cases in which one member of the couple is not assigned. The empty set can be considered as a hospital which has capacity equal to the number of doctors in couples plus one (so that it is always undersubscribed) and finds acceptable all doctors belonging to couples who list the empty set as an option.

A *matching* M in I is a subset of acceptable pairs such that each individual doctor (single or in a couple) appears in at most one pair of M and each hospital appears in at most c_h pairs of M. We remark that $\{(d_1, h), (d_2, h)\}$ counts as two pairs for hospital h, one pair for doctor d_1, and one pair for doctor d_2. The definitions of *matched* and *unmatched* for individual doctors (single or in couple), and *undersubscribed* and *fully subscribed* for hospitals remain the same as they were for HRT. A couple is *unmatched* if none of its members is matched. Thus, if d_1 and d_2 form a couple and M contains $\{(d_1, h_1), (d_2, \emptyset)\}$, couple (d_1, d_2) is considered matched.

We still use $M(d)$ to denote the hospital in which individual doctor d (single or in couple) is matched and $M(h)$ for the set of doctors assigned to hospital h. If necessary, we use $M(c)$ to denote the pair of hospitals in which couple c is matched (one of which could be the empty set).

3.2 Stability definitions for HRCT

To the best of our knowledge, three stability definitions for HRC have been proposed in the literature: MM-stability by McDermid and Manlove [28], BIS-stability by Biró, Irving, and Schlotter [5], and KPR-stability by Khojima, Pathak, and Roth [20]. Since they studied the problem without ties, the preferences were always strict: hospital h *strictly prefers* doctor d over doctor d' if $r^H_d(h) < r^H_{d'}(h)$. In HRCT, we also have non-strict preferences: hospital h *weakly prefers* doctor d over doctor d' if $r^H_d(h) \leq r^H_{d'}(h)$. An intuitive extension of MM-, BIS-, and KPR-stability for HRCT is made by changing any notion of "preferences" in the definition into the notion of "strict preferences".

The three definitions consider a matching as stable if it does not have any blocking pair or blocking coalition. Three different types of blocking pairs/coalitions were identified:

- SH, a blocking pair between a single doctor and a hospital,
- CHH, a blocking coalition between a couple of doctors and two distinct hospitals,
- CH, a blocking coalition between a couple of doctors and one hospital.

While the three definitions agree on SH and CHH, they differ for CH. In the following, we provide a formal definition for CHH and the three versions of CH, and we let SHi = DHi for $1 \leq i \leq 3$ as per Definition 1.

3.2.1 CHH definition

Definition 2. *Let I be an instance of HRCT and let M be a matching in I. A couple (d_1, d_2) and two distinct hospitals (h_1, h_2) form a* blocking coalition CHH of M *if*

CHH1- $\{(d_1, h_1), (d_2, h_2)\}$ *is an acceptable pair;*

CHH2- (d_1, d_2) *either is unmatched or strictly prefers* (h_1, h_2) *to* $(M(d_1), M(d_2))$*; and*

CHH3- either h_1 is undersubscribed, or h_1 strictly prefers d_1 to some member of $M(h_1)$, or $d_1 \in M(h_1)$; and

CHH4- either h_2 is undersubscribed, or h_2 strictly prefers d_2 to some member of $M(h_2)$, or $d_2 \in M(h_2)$.

Example 2. *Let us consider an HRCT instance with two hospitals, one couple, and one single doctor with the following preference lists and capacity information:*

$$\begin{array}{lll} d_1 : & h_1 \quad h_2 & \\ (d_2, d_3) : & (h_1, h_2) \quad [(h_1, \emptyset) \ (\emptyset, h_2)] & \end{array} \qquad \begin{array}{ll} h_1 \ (c_1 = 1) : & d_2 \quad d_1 \\ h_2 \ (c_2 = 1) : & d_1 \quad d_3. \end{array}$$

Doctor d_1 prefers hospital h_1 over hospital h_2. Couple (d_2, d_3) prefers to be assigned to hospitals (h_1, h_2), and if not possible, the couple is indifferent between the partial assignments $(h_1, \emptyset)$ and $(\emptyset, h_2)$.

If $M(d_2, d_3)=(h_1, h_2)$, the matching is not stable as (d_1, h_2) forms a blocking pair of type SH. If $M(d_1) = h_1$ and $M(d_2, d_3) = (\emptyset, h_2)$, the matching is not stable as couple (d_2, d_3) and hospitals (h_1, h_2) form a blocking coalition of type CHH. Indeed, (d_2, d_3) strictly prefers to be assigned to (h_1, h_2) over its current assignment, h_1 strictly prefers d_2 to d_1, and $d_3 \in M(h_2)$. If $M(d_1) = h_2$ and $M(d_2, d_3) = (h_1, \emptyset)$, the matching is now stable as h_2 is not undersubscribed, it does not strictly prefer d_3 over d_1, and $d_3 \notin M(h_2)$. □

3.2.2 CH definition for KPR-, BIS-, and MM-stability

Before introducing the definition of blocking coalition CH for each of the three stability definitions, we start with some observations about KPR-stability and choice functions (see Kojima, Pathak, and Roth [20]). KPR-stability and choice functions were also used by Ashlagi, Braverman, and Hassidim [3], and Drummond, Perrault, and Bacchus [10]. Given a hospital h and a set of doctors D', the choice function $\mathrm{Ch}_h(D')$ gives the subset of doctors $D'' \subseteq D'$ that hospital h would employ if only able to choose from doctors in D'. When ties are not allowed, and since the preference relations are *responsive* (i.e., $r_j^D(i)$ does not depend on the subset D'), $\mathrm{Ch}_h(D')$ simply contains the first $\min\{c_h, |D'|\}$ doctors from D' ordered by increasing rank. Equivalent choice functions can also be defined for doctors, but we do not use them. We now introduce our definition of a blocking coalition CH under what we refer to as KPR-stability, and then prove that it is equivalent to the definition used in [20] (page 1602, item 2(b)) in the absence of ties.[1]

Definition 3. *Let I be an instance of HRCT and let M be a matching in I. A couple (d_1, d_2) and a hospital h form a* blocking coalition CH of M under KPR-stability *if*

CH1- $\{(d_1, h), (d_2, h)\}$ is an acceptable pair; and

CH2- (d_1, d_2) is either unmatched, or strictly prefers (h, h) to $(M(d_1), M(d_2))$; and

CH3- h has either

- *CH3.1- two free posts; or*
- *CH3.2- one free post and either $d_1 \in M(h)$ or $d_2 \in M(h)$; or*
- *CH3.3- one free post and it strictly prefers both d_1 and d_2 to some member of $M(h)$; or*
- *CH3.4- no free post and it strictly prefers both d_1 and d_2 to some member of $M(h)$ and either $d_1 \in M(h)$ or $d_2 \in M(h)$; or*
- *CH3.5- no free post and it strictly prefers both d_1 and d_2 to two distinct members of $M(h)$.*

Theorem 1. *The definition of a blocking coalition CH under KPR-stability in Definition 3 is equivalent to the one proposed in Kojima, Pathak, and Roth [20] when ties are not allowed, which states that a couple (d_1, d_2) and a hospital h form a blocking coalition of M if (d_1, d_2) strictly prefers (h, h) to $(M(d_1), M(d_2))$ and if $\{d_1, d_2\} \subseteq \mathrm{Ch}_h(M(h) \cup \{d_1, d_2\})$.*

[1] We do not consider the choice function definition of stability in the general HRCT case as there are multiple interpretations as to what a hospital would choose if ties are allowed.

Table 1: Six different cases in which couple (d_1, d_2) and hospital h could form a blocking coalition

Cases	h has two free posts or more	h has one free post	h has no free post
$d_1 \in M(h), d_2 \notin M(h)$	CH3.1	CH3.2	CH3.4
$d_1 \notin M(h), d_2 \notin M(h)$	CH3.1	CH3.3	CH3.5

Proof. We individually study each case of CH3.1-CH3.5 and show how each of them implies "$(d_1, d_2) \in \text{Ch}_h(M(h) \cup \{d_1, d_2\})$".

1. If h has two free posts, then $|M(h)| \leq c_h - 2$, so $|M(h) \cup \{d_1, d_2\}| \leq c_h$. Thus, $\text{Ch}_h(M(h) \cup \{d_1, d_2\})$ contains both d_1 and d_2.

2. If h has one free post and either $d_1 \in M(h)$ or $d_2 \in M(h)$, then $|M(h) \cup \{d_1, d_2\}| \leq c_h$. Thus again, $\text{Ch}_h(M(h) \cup \{d_1, d_2\})$ contains both d_1 and d_2.

3. If h has one free post and strictly prefers both d_1 and d_2 to some member of $M(h)$, say d', then $|M(h) \cup \{d_1, d_2\}| \leq c_h + 1$. However, we know that d' is ranked worse than both d_1 and d_2. Thus, $\text{Ch}_h(M(h) \cup \{d_1, d_2\})$ would exclude d' in order to include both d_1 and d_2.

4. If h has no free post and strictly prefers both d_1 and d_2 to some member of $M(h)$, say d', and either $d_1 \in M(h)$ or $d_2 \in M(h)$, then $|M(h) \cup \{d_1, d_2\}| \leq c_h + 1$. Again, we know that d' is ranked worse than both d_1 and d_2. Thus, $\text{Ch}_h(M(h) \cup \{d_1, d_2\})$ puts aside d' to include d_1 or d_2 (the member of the couple not yet assigned to h).

5. If h has no free post and strictly prefers both d_1 and d_2 to two distinct members of $M(h)$, say d' and d'', then $|M(h) \cup \{d_1, d_2\}| \leq c_h + 2$. However, we know that d' and d'' are ranked worse than both d_1 and d_2. Thus, $\text{Ch}_h(M(h) \cup \{d_1, d_2\})$ puts aside d' and d'' to include both d_1 and d_2.

We conclude the proof by noting that any blocking coalition CH described in [3] can be represented with one of the five cases described in CH3.1-CH3.5, as shown in Table 1. □

Definition 4 ([5])**.** *Let I be an instance of HRCT and let M be a matching in I. A couple (d_1, d_2) and the same hospital h form a* blocking coalition CH of M under BIS-stability *if*

CH1-CH2, CH3.1-CH3.5

CH3.6- no free post and it strictly prefers both d_1 and d_2 to any member of a couple with both members in $M(h)$.

Under BIS-stability, the objective is to minimise the worst rank of the hospitals' assignees. Thus, couples are compared based on their less-preferred member and a couple with two average-ranked members is favoured over a couple with a good and a bad ranked member. However, BIS-stability does not extend this policy to comparing couples with pairs of single doctors. One reason for this might be that it could create a "loop" of blocking pairs of the form SH/CH that would lead to an absence of a stable matching. In such loop, if a couple (d_1, d_2) is assigned to a given hospital h, a single doctor d_3 and h forms a blocking pair, and if d_3 is assigned to h instead, then d_1 and d_2 form a blocking coalition with h. Thus, it is supposed that hospitals know which doctors are in couples, which is not a requirement in the other stability definitions.

Definition 5 ([28]). *Let I be an instance of HRCT and let M be a matching in I. A couple (d_1, d_2) and the same hospital h form a* blocking coalition CH of M under MM-stability *if*

CH1-CH2, CH3.1-CH3.2

CH3.3′ - one free post and it strictly prefers d_1 or d_2 to some member of $M(h)$; or

CH3.4.1′- no free post and it strictly prefers d_1 to some member of $M(h)\setminus\{d_2\}$ and $d_2 \in M(h)$ or;

CH3.4.2′- no free post and it strictly prefers d_2 to some member of $M(h)\setminus\{d_1\}$ and $d_1 \in M(h)$ or;

CH3.5′ - no free post and it strictly prefers d_1 to some member of $M(h)$ and strictly prefers d_2 to another member of $M(h)$.

MM-stability treats each position of a hospital independently, and favours a matching that improves the rank of one or two assignees, providing the ranks of the other assignees remain unchanged.

Intuitively, KPR-stability requires the strongest conditions for a couple to form a blocking coalition with a single hospital. If a matching is MM-stable, then it is also KPR-stable: indeed, (i) CH3.3 cannot be satisfied if CH3.3′ is not satisfied, (ii) CH3.4 cannot be met if neither CH3.4.1′ nor CH3.4.2′ is met, and (iii) CH3.5 cannot be satisfied if CH3.5′ is not satisfied. Similarly, if a matching is BIS-stable, then it is also KPR-stable. There is no dominance relation between MM- and BIS-stability.

3.2.3 Examples

We now give three examples outlining noticeable differences in behaviour exhibited by each stability definition.

Example 3. *Let us consider an HRCT instance with one hospital, one couple, and two doctors with the following preference lists and capacity information:*

$$\begin{array}{rl} d_1: & h_1 \\ (d_2, d_3): & (h_1, h_1) \\ d_4: & h_1 \end{array} \qquad h_1\ (c_1 = 2): \quad d_2 \quad d_1 \quad d_3 \quad d_4$$

If $M(h_1) = \{d_1, d_4\}$, the matching is BIS-stable (and thus, KPR-stable) as h_1 has no free post and does not strictly prefer d_3 to two distinct members of $M(h_1)$. However, the matching is not MM-stable because (d_2, d_3) forms a blocking coalition with h_1: indeed, h_1 has no free post and strictly prefers d_2 to d_1 and d_3 to d_4. Note that if the capacity c_1 were to be 3, the matching would not be stable under any of the three definitions. □

Example 4. *Let us consider an HRCT instance with one hospital and two couples with the following preference lists:*

$$(d_1, d_4): \quad (h_1, h_1) \qquad h_1\ (c_1 = 2): \quad d_2 \quad d_1 \quad d_3 \quad d_4$$

$$(d_2, d_3): \quad (h_1, h_1)$$

If $M(h_1) = \{d_1, d_4\}$, the matching is KPR-stable as h_1 has no free post and does not strictly prefer d_3 to two distinct members of $M(h)$. It is not BIS-stable because (d_2, d_3) forms a blocking pair with h_1: indeed, h_1 has no free post and strictly prefers both d_2 and d_3 to d_4, a member of a couple with both members in $M(h)$. As in the previous example, the matching is not MM-stable. Again, if the capacity c_1 were to be 3, the matching would not be stable under any of the three definitions.

If $M(h_1) = \{d_2, d_3\}$, the matching is stable under the three conditions. However, if the capacity c_1 were to be 3, then the matching would not be MM-stable anymore, as (d_1, d_4) would form a blocking coalition with h_1: indeed, h_1 has one free post and strictly prefers d_1 to d_3. □

Example 5. *Let us consider an HRCT instance with one hospital and two couples with the following preference lists and capacity information:*

$$(d_1, d_4): \quad (h_1, h_1) \qquad\qquad h_1\ (c_1 = 2): \quad d_2 \quad d_1 \quad d_4 \quad d_3$$
$$(d_2, d_3): \quad (h_1, h_1)$$

If $M(h_1) = \{d_1, d_4\}$, the matching is stable under all three definitions. As in the previous example, if the capacity c_1 were to be 3, then the matching would not be MM-stable anymore.

If $M(h_1) = \{d_2, d_3\}$, the matching is MM-stable (and thus, KPR-stable), as h_1 has no free post and does not strictly prefer d_1 and d_4 to two distinct members of $M(h)$. However, the matching is not BIS-stable because (d_1, d_4) forms a blocking coalition with h_1: indeed, h_1 has no free post and strictly prefers both d_1 and d_4 to d_3. If the capacity c_1 were to be 3, then the matching would not be stable under any of the three definitions. □

An exhaustive study of all the possible outcomes on instances with one couple/one single doctor and two couples is presented in Tables 2 and 3. Similar outcomes for instances with one couple/two single doctors are presented in Tables 14 and 15 in the appendix. We always consider a unique hospital h_1 (since only the definition of blocking pairs of type CH varies among the different stability definitions). The first column contains the preference list of h_1, possibly with ties. Members of the same couple are denoted with the same letter: the upper case is used for the doctor the hospital prefers (e.g., "A"), and the lower case is used for the other member (e.g., "a"). If the hospital is indifferent between the two members, then the upper case is used arbitrarily for one of the two members. The second column contains the capacity of h_1, then the next two columns give all possible stable matchings for each stability definition in the case of "splittable couples". A splittable couple accepts a matching in which only one member is given an assignment and the other member is unmatched. The last columns give the same information in case the couples are "unsplittable". We use the symbol "$\emptyset$" where a stable matching does not exist. We use "all-2" (respectively "all-3") where any matching of size 2 (respectively 3) is stable. Note that KPR- and BIS-stability are merged in most of the cases as their stable matchings only differ when there are two unsplittable couples. For the sake of conciseness, the tables also contain the results of KPR$^+$-stability, our new stability definition which is formally introduced later in the section.

Table 2: Feasible matching for instances with one single doctor and one couple

h_1	c_1	Splittable couple*			Unsplittable couple**		
		KPR and BIS	KPR$^+$	MM	KPR and BIS	KPR$^+$	MM
AaB	2	Aa	Aa	Aa	Aa	Aa	Aa
ABa	2	AB, Ba	AB, Ba	AB	B	B	$\emptyset$
BAa	2	BA, Ba	BA, Ba	BA, Ba	B	B	B
$[AaB]$	2	Aa, AB, aB	Aa, AB, aB	Aa, AB, aB	Aa, B	Aa, B	Aa, B
$A[aB]$	2	Aa, AB, aB	Aa, AB	Aa, AB	Aa, B	Aa	Aa
$[AB]a$	2	AB, Ba	AB, Ba	AB, Ba	B	B	B

* preference list of (A, a) is $(h_1, h_1)[(h_1, \emptyset)\ (\emptyset, h_1)]$

** preference list of (A, a) is (h_1, h_1)

The tables read as follows: under KPR-stability, if one splittable couple (A, a) and one single doctor B apply to h_1, with preference list "$[AB]a$" and capacity 2, a stable matching assigns either A and B to h_1, or B and a to h_1.

We observe a number of interesting facts: (i) as expected, any matching that is MM-stable or BIS-stable is also KPR-stable; (ii) when couples are unsplittable, various examples do not have any feasible MM-stable matching, (iii) when couples are splittable, the set of MM-stable matchings is a subset of KPR/BIS-stable matchings, (iv) when couples are splittable, the KPR/BIS-stable matchings that are not MM-stable always include the least favourite member of a couple whose most favourite member is not assigned, (v) when ties are allowed in the preference lists, the three stability definitions are more flexible and allow significantly more feasible matchings, and (vi) for unsplittable couples, increasing the capacity does not necessarily increase the matching size.

Among these additional feasible matchings, some of them are not what a decision-maker might qualify as stable: for example, in the preference list "$A[aB]b$" for h_1 with capacity 2 and unsplittable couple, it is reasonable to think that matching couple (A, a) should be favoured over matching couple (B, b). However, under KPR-stability, assigning either couple to h_1 leads to a stable matching. Similarly, in the preference list "$A[aBb]$" for h_1 with capacity 2 and unsplittable couples, one could think that matching couple (A, a) should be favoured over matching couple (B, b), but once again under KPR-stability, assigning either couple to h_1 leads to a stable matching. More generally, KPR-stability rarely makes any distinction between two couples when they both have one of their members in the same tie.

In the following we define KPR$^+$-stability, which we have specifically designed to avoid the counter-intuitive scenarios outlined in the previous paragraph. Note that similar adaptations can also be applied to BIS- and MM-stability definitions.

Definition 6. *Let I be an instance of HRCT and let M be a matching in I. A couple (d_1, d_2) and hospital h form a* blocking coalition CH of M under KPR$^+$-stability *if*

CH1-CH2, CH3.1-CH3.2

CH3.3$^+$ - one free post and it weakly prefers both d_1 and d_2 to some member of $M(h)$ and strictly prefers one of them to some member of $M(h)$; or

Table 3: Feasible matching for instances with two couples

h_1	c_1	Splittable couple*			Unsplittable couple**			
		KPR and BIS	KPR$^+$	MM	KPR	KPR$^+$	BIS	MM
$AaBb$	2	Aa	Aa	Aa	Aa	Aa	Aa	Aa
	3	AaB, Aab	AaB, Aab	AaB, Aab	Aa	Aa	Aa	Aa
$ABab$	2	AB, Ba	AB, Ba	AB	Aa, Bb	Aa, Bb	Aa	Aa
	3	ABa, Aab	ABa, Aab	ABa	Aa	Aa	Aa	$\emptyset$
$ABba$	2	AB, Ab	AB, Ab	AB, Ab	Aa, Bb	Aa, Bb	Bb	Aa, Bb
	3	ABb, Bba	ABb, Bba	ABb	Bb	Bb	Bb	$\emptyset$
$[AaBb]$	2	all-2	all-2	all-2	Aa, Bb	Aa, Bb	Aa, Bb	Aa, Bb
	3	all-3	all-3	all-3	Aa, Bb	Aa, Bb	Aa, Bb	Aa, Bb
$A[aBb]$	2	Aa, AB, Ab, aB, ab	Aa, AB, Ab	Aa, AB, Ab	Aa, Bb	Aa	Aa, Bb	Aa, Bb
	3	all-3	AaB, Aab, ABb	AaB, Aab, ABb	Aa, Bb	Aa	Aa, Bb	Aa
$[AaB]b$	2	Aa, AB, aB	Aa, AB, aB	Aa, AB, aB	Aa, Bb	Aa, Bb	Aa	Aa, Bb
	3	AaB, Aab	AaB, Aab	AaB, Aab	Aa	Aa	Aa	Aa
$[AB][ab]$	2	AB, Ab, Ba, ab	AB, Ab, Ba	AB, Ab, Ba	Aa, Bb	Aa, Bb	Aa, Bb	Aa, Bb
	3	all-3	ABa, ABb	ABa, ABb	Aa, Bb	$\emptyset$	Aa, Bb	$\emptyset$
$AB[ab]$	2	AB, Ab, Ba, ab	AB, Ab, Ba	AB, Ab	Aa, Bb	Aa, Bb	Aa, Bb	Aa, Bb
	3	all-3	ABa, ABb	ABa, ABb	Aa, Bb	$\emptyset$	Aa, Bb	$\emptyset$
$[AB]ab$	2	AB, Ba	AB, Ba	AB, Ba	Aa, Bb	Aa, Bb	Aa	Aa, Bb
	3	ABa, Aab	ABa, Aab	ABa	Aa	Aa	Aa	$\emptyset$
$A[aB]b$	2	Aa, AB, aB	Aa, AB	Aa, AB	Aa, Bb	Aa	Aa	Aa
	3	AaB, Aab	AaB, Aab	AaB, Aab	Aa	Aa	Aa	Aa

* preference list of (A, a) and (B, b) is $(h_1, h_1)[(h_1, \emptyset)\ (\emptyset, h_1)]$

** preference list of (A, a) and (B, b) is (h_1, h_1)

CH3.4.1$^+$- no free post and it weakly prefers both d_1 and d_2 to some member of $M(h) \setminus \{d_2\}$ and strictly prefers d_1 to some member of $M(h) \setminus \{d_2\}$ and $d_2 \in M(h)$; or

CH3.4.2$^+$- no free post and it weakly prefers both d_1 and d_2 to some member of $M(h) \setminus \{d_1\}$ and strictly prefers d_2 to some member of $M(h) \setminus \{d_1\}$ and $d_1 \in M(h)$; or

CH3.5$^+$ - no free post and it weakly prefers both d_1 and d_2 to two distinct members of $M(h)$ and strictly prefers one of them to two distinct members of $M(h)$ and $d_1 \notin M(h)$ and $d_2 \notin M(h)$.

For example, KPR$^+$-stability favours matching couple (A, a) over single doctor B when h_1 has capacity 2 and preference list "$A[aB]$" with unsplittable couples. However, if h_1 has capacity 2 and preference list "$AB[aC]$" and the couples are unsplittable, KPR$^+$-stability does not favour couple (A, a) over single doctors B and C. Indeed, if that were the case, since B precedes a in h_1's preference list, B and h_1 would form a blocking pair of type SH, creating a cycle of blocking pairs / coalitions resulting in the absence of stable matching.

There are only a few cases in which such cycles cannot be avoided and thus where there is no KPR$^+$-stable matching: for example when h_1 has capacity 3 and preference list "$[AB][ab]$" and the couples are unsplittable. We remark that KPR$^+$-stability differs from KPR-stability only in the cases where ties are allowed and one member of the couple is strictly preferred over the other.

Overall we observe some interesting behaviour for the stability definitions: when couples are splittable, we distinguish between the "restrictive" definitions (such as MM) whose stable matchings are also stable under the other definitions, and the "permissive" definitions (such as BIS- and KPR-stability). We note that any matching that is stable in the "permissive" setting but not in the "restrictive" setting always can be built from a matching that is stable in the "restrictive" setting that contains only one member of a splittable couple (the most-preferred according to h_1) by replacing this most-preferred member of a splittable couple with the least-preferred member of that couple. When couples are unsplittable, we still have the "permissive" definitions (such as KPR) whose stable matchings include all the matchings that are stable under any of the "restrictive" definitions. However, we now differentiate the "restrictive-blind" definitions (such as MM-stability and KPR$^+$) from the "restrictive-knowing" definitions (such as BIS). The former supposes that hospitals do not know which of their currently matched doctors are in a couple, and a hospital would consider rejecting one member of a couple without knowing that some other doctor (the other doctor in the couple) would also then leave. The latter supposes that hospitals do know which of their assigned doctors are in couples, and may use this information to determine whether they would reject a currently assigned doctor or couple in favour of a new doctor or couple.

As KPR$^+$-stability is identical to KPR-stability when ties are not allowed or when both members of the couple have the same rank, a blocking pair/coalition under KPR$^+$-stability under these conditions is also blocking under MM-stability. When ties are allowed and both members of the couple have different ranks, KPR$^+$-stability is sometimes more restrictive than MM-stability (in particular when couples are unsplittable) and sometimes less restrictive (when couples are splittable). As shown in Tables 2, 3, 14, 15, in total, only two one-hospital configurations do not admit any stable matching under KPR$^+$-stability (vs 10 for MM-stability). We use the following example to remind the reader that there exist many HRCT instances with two hospitals for which no feasible matching can be found under any of the aforementioned stability definitions.

Example 6. *Let us consider an HRCT instance with two hospitals, one couple, and one single doctor with the following preference lists and capacity information:*

$$\begin{array}{rll} d_1: & h_1 \quad h_2 & \qquad h_1(c_1 = 1): \quad d_2 \quad d_1 \\ (d_2, d_3): & (h_1, h_2) & \qquad h_2(c_1 = 1): \quad d_1 \quad d_3 \end{array}$$

If $M(h_1) = \{d_1\}$ and $M(h_2) = \emptyset$, then (d_2, d_3) and (h_1, h_2) form a blocking coalition of type CHH.
If $M(h_1) = \{d_2\}$ and $M(h_2) = \{d_3\}$, then d_1 and h_2 form a blocking pair of type SH.
If $M(h_1) = \emptyset$ and $M(h_2) = \{d_1\}$, then d_1 and h_1 form a blocking pair of type SH.

Note that all three stability definitions have the same definitions for blocking coalitions of type CHH and blocking pairs of type SH. □

4 ILP models for MAX-HRCT

A set of ILP models were proposed in the literature for BIS-stability [27] and for MM-stability [22]. In the following, we propose alternative formulations for each of the three definitions that are based on model (1)-(5). Our goal is to introduce a unified base model which can be extended to each stability definition discussed with minimal additional constraints. Let us consider the following notation:

- The set of doctors $\mathcal{D}$ contains first the n_S single doctors, then the first members of the n_C couples, and finally, the second members of the couples. Thus, couple k is composed of doctors $n_S + k$ and $n_S + n_C + k$.
- $H(i)$ is the set of hospitals acceptable for doctor (single or in a couple) i ($i = 1, \dots, n_S + 2n_C$).
- $H^c(k)$ is the set of pairs of hospitals acceptable for couple k ($k = 1, \dots, n_C$).
- $D(j)$ is the set of doctors (single or in a couple) acceptable for hospital j ($j = 1, \dots, n_H$).
- $r^D_j(i)$ is the rank of hospital j for single doctor i, defined as the integer l such that j belongs to the lth most-preferred tie in i's list ($i = 1, \dots, n_S, j \in H(i)$). The smaller the value of $r^D_j(i)$, the better hospital j is ranked for doctor i.
- $r^D_{(j_1,j_2)}(k)$ is the rank of the pair of hospitals (j_1, j_2) for couple k, defined as the integer l such that (j_1, j_2) belongs to the lth most-preferred tie in k's list ($k = 1, \dots, n_C, (j_1, j_2) \in H^c(k)$). The smaller the value of $r^D_{(j_1,j_2)}(k)$, the better the pair of hospitals (j_1, j_2) is ranked for couple k.
- $r^H_i(j)$ is the rank of doctor i (single or in pair) for hospital j, defined as the integer l such that i belongs to the lth most-preferred tie in j's list ($j = 1, \dots, n_H, i \in D(j)$). The smaller the value of $r^H_i(j)$, the better doctor i is ranked for hospital j.
- $H^{\leq}_j(i)$ is the set of hospitals that single resident i ranks at the same level or better than hospital j, that is, $H^{\leq}_j(i) = \{j' \in H(i) : r^D_{j'}(i) \leq r^D_j(i)\}$ ($i = 1, \dots, n_S, j \in H(i)$).
- $H^{\leq}_{(j_1,j_2)}(k)$ is the set of pairs of hospitals that couple k ranks at the same level or better than the pair of hospitals (j_1, j_2), that is, $H^{\leq}_{(j_1,j_2)}(k) = \{(j'_1, j'_2) \in H^c(k) : r^D_{(j'_1,j'_2)}(k) \leq r^D_{(j_1,j_2)}(k)\}$ ($k = 1, \dots, n_C, (j_1, j_2) \in H^c(k)$).
- $D^{\leq}_i(j)$ is the set of doctors (single or in pair) that hospital j ranks at the same level or better than doctor i, that is, $D^{\leq}_i(j) = \{i' \in D(j) : r^H_{i'}(j) \leq r^H_i(j)\}$ ($j = 1, \dots, n_H, i \in D(j)$).

Let MAX-HRCT denote the problem of finding a stable matching of maximum size in an HRCT instance. By introducing binary decision variables x_{ij} that take value 1 if doctor i (single or in a couple) is assigned to hospital j, and 0 otherwise ($i = 1, \dots, n_S + 2n_C, j \in H(i)$), and binary decision variables $y_{kj_1j_2}$ that take value 1 if couple k is assigned to the pair of hospital (j_1, j_2), and 0 otherwise ($k = 1, \dots, n_C, (j_1, j_2) \in H^c(k)$), MAX-HRCT can be modelled as follows:

$$
\begin{aligned}
\max \quad & \sum_{i=1}^{n_S+2n_C} \sum_{j\in H(i)} x_{ij} && (6)\\
\text{s.t.} \quad & (S1), (S2), (S3), && \\
& \sum_{j\in H(i)} x_{ij} \le 1, \qquad i = 1, \dots, n_S + 2n_C, && (7)\\
& \sum_{i\in D(j)} x_{ij} \le c_j, \qquad j = 1, \dots, n_H, && (8)\\
& \sum_{(j_1,j_2)\in H^c(k), j_1=j} y_{kj_1j_2} = x_{n_S+k,j}, \qquad k = 1, \dots, n_C, j \in H(n_S + k), && (9)\\
& \sum_{(j_1,j_2)\in H^c(k), j_2=j} y_{kj_1j_2} = x_{n_S+n_C+k,j}, \qquad k = 1, \dots, n_C, j \in H(n_S + n_C + k), && (10)\\
& x_{ij} \in \{0, 1\}, \qquad i = 1, \dots, n_S + 2n_C, j \in H(i), && (11)\\
& y_{kj_1j_2} \in \{0, 1\}, \qquad k = 1, \dots, n_C, (j_1, j_2) \in H^c(k), && (12)
\end{aligned}
$$

where (S1) are the stability constraints that remove blocking pairs SH, and (S2) and (S3) are the constraints that remove blocking coalitions CHH and CH, respectively. All these stability constraints are defined below.

The objective function (6) maximises the number of doctors assigned. Constraints (7) ensure that each doctor (single or in a couple) is matched with at most one hospital and constraints (8) ensure that each hospital does not exceed its capacity. Constraints (9) and (10) link the x and the y variables.

Stability constraints (S1), that prevent blocking pairs of type SH, are as follows for MM-, BIS-, and KPR-stability:

$$
c_j \left(1 - \sum_{q\in H_j^{\le}(i)} x_{iq}\right) \le \sum_{p\in D_i^{\le}(j)} x_{pj}, \quad i = 1, \dots, n_S, j \in H(i). \tag{13}
$$

Similar to HRT, they ensure that if single doctor i was not assigned to hospital j or any other hospital they rank at the same level or higher than j, then hospital j has filled its capacity with doctors (single or in couples) it ranks at the same level or higher than i.

Stability constraints (S2), that prevent blocking coalitions of type CHH, are defined as follows for MM-, BIS-, and KPR-stability, respectively:

$$
c_{j_1} \left(1 - \sum_{(q_1,q_2)\in H^{\le}_{(j_1,j_2)}(k)} y_{kq_1q_2} - \alpha^1_{kj_1j_2}\right) \le \sum_{p\in D^{\le}_{n_S+k}(j_1)} x_{pj_1} - x_{n_S+k,j_1}, \qquad k = 1, \dots, n_C, (j_1, j_2) \in H^c(k), \tag{14}
$$

$$
c_{j_2} \left(1 - \sum_{(q_1,q_2)\in H^{\le}_{(j_1,j_2)}(k)} y_{kq_1q_2} - \alpha^2_{kj_1j_2}\right) \le \sum_{p\in D^{\le}_{n_S+n_C+k}(j_2)} x_{pj_2} - x_{n_S+n_C+k,j_2}, \quad k = 1, \dots, n_C, (j_1, j_2) \in H^c(k), \tag{15}
$$

$$
\alpha^1_{kj_1j_2} + \alpha^2_{kj_1j_2} \le 1, \qquad k = 1, \dots, n_C, (j_1, j_2) \in H^c(k), \tag{16}
$$

They ensure that if couple k was not assigned to the pair of hospitals (j_1, j_2) or any other pair they rank at the same level or higher than (j_1, j_2), then either hospital j_1 or j_2 is fully subscribed with doctors (singles or in couples) it ranks at the same level or higher than the corresponding member of the couple. The hospital (if any) that is not fully subscribed with better doctors has its corresponding α variable set to one. Also, if one of the two hospitals has the corresponding member of the couple in the matching, then its α variable must be set to one. The alpha variables can be seen as a "wild card" that allow at most one of the two constraints to be violated.

Stability constraints (S3), that prevent blocking pairs of type CH, vary according to the stability definition. For KPR-stability, they are defined as:

$$c_j \left(1 - \sum_{(q_1,q_2) \in H^{\leq}_{(j,j)}(k)} y_{kq_1q_2} \right) - 1 + x_{i_1j} + x_{i_2j} \leq \sum_{p \in D^{\leq}_{i_2}(j)} x_{pj}, \quad k = 1, \ldots, n_C, (j,j) \in H^c(k), \tag{17}$$

where i_1 is the index of the member of the couple who is weakly preferred by the hospital (i.e., $r^H_{i_1}(j) \leq r^H_{i_2}(j)$, which is also assumed in the rest of the section). They ensure that if couple k was not assigned to the pair of hospitals (j, j) or any other pair they rank at the same level or higher than (j, j), then hospital j has filled its capacity minus 1 with doctors (singles or in couples) it ranks at the same level or higher than the worst member of the couple. The "minus 1" is offset if one of the two members of the couple is assigned to j.

For BIS-stability, constraints (S3) are defined as:

$$c_j \left(1 - \sum_{(q_1,q_2) \in H^{\leq}_{(j,j)}(k)} y_{kq_1q_2} \right) - 1 + x_{i_1j} + x_{i_2j} + \sum_{k' \in \mathcal{K}_j(k)} y_{k'jj} \leq \sum_{p \in D^{\leq}_{i_2}(j)} x_{pj}, \quad k = 1, \ldots, n_C, (j,j) \in H^c(k), \tag{18}$$

where $\mathcal{K}_j(k)$ contains the indices of the couples also applying to (j, j) that have one member ranked strictly worse than both i_1 and i_2 (to take into account criterion CH3.6) and one member ranked at least as good as i_2 (to ensure that the right-hand-side is always greater than or equal to the left-hand-side in case couple k is assigned to hospital j). This rules out the assignment of such couple to (j, j) if couple k was not already assigned to (j, j), or to a better choice.

For MM-stability, constraints (S3) are defined as:

$$c_j \left(1 - \sum_{(q_1,q_2) \in H^{\leq}_{(j,j)}(k)} y_{kq_1q_2} \right) - 1 + x_{ij} + x_{i_2j} \leq \sum_{p \in D^{\leq}_{i_2}(j)} x_{pj}, \quad k = 1, \ldots, n_C, (j,j) \in H^c(k), i \in \mathcal{S}, \tag{19}$$

where $\mathcal{S}$ contains i_1 and the indices of the doctors ranked strictly worse than i_1 and at least as good as i_2, but not i_2, i.e., $\mathcal{S} = \left(D^{\leq}_{i_2}(j) \setminus \left(D^{\leq}_{i_1}(j) \cup \{i_2\}\right)\right) \cup \{i_1\}$. Note that these constraints can be aggregated into

$$c_j \left(1 - \sum_{(q_1,q_2) \in H^{\leq}_{(j,j)}(k)} y_{kq_1q_2} \right) - 1 + \frac{\sum_{i \in \mathcal{S}} x_{ij}}{|\mathcal{S}|} + x_{i_2j} \leq \sum_{p \in D^{\leq}_{i_2}(j)} x_{pj}, \quad k = 1, \ldots, n_C, (j,j) \in H^c(k), \tag{20}$$

For MM-stability, the "minus 1" is offset if any member in $\mathcal{S}$ is selected, and the constraints forbid both i_2 and a member of $\mathcal{S}$ to be selected at the same time if i_1 was not selected as well (criteria CH3.4.1′ and CH3.4.2′).

For KPR$^+$-stability , constraints (S3) include constraints (17) (from (S3) for KPR) and also constraints (21), defined below. Note that constraints (21) only apply if one member of the couple is strictly preferred to the other (i.e., $r^H_{i_1}(j) \neq r^H_{i_2}(j)$):

$$c_j \left(1 - \sum_{(q_1,q_2) \in H^{\leq}_{(j,j)}(k)} y_{kq_1q_2} - x_{i_1j} \right) - 1 \leq \sum_{p \in D^{<}_{i_2}(j)} x_{pj}, \quad k = 1, \dots, n_C, (j,j) \in H^c(k), r^H_{i_1}(j) \neq r^H_{i_2}(j). \tag{21}$$

In constraints (21), the sum of the right-hand side is made on the doctors strictly preferred over i_2 instead of weakly preferred. The constraints are activated only if i_1 is not assigned to the hospital (x_{i_1j} is equal to 0). If x_{i_2j} is equal to 1, then the hospital needs to fill the rest of its capacity with better doctors than i_2 (the extra spot allowed by the "minus 1" is taken by i_2). If x_{i_2j} is equal to 0, then at most one doctor with the same rank as i_2 or worse can be assigned to the hospital.

5 Model improvements

Many improvements for HRT were proposed by Delorme et al. [9], including preprocessing, dummy variables, and alternative stability constraints.

Preprocessing for HRT consists of removing some pairs (a potential assignment of a doctor to a hospital) that cannot be part of any stable matching. In HRCT, we also have to ensure that the removal of pairs does not create a new instance with stable matchings if the original instance did not contain any stable matchings. In addition, tailored preprocessing techniques for HRCT depend on the chosen stability definition, so a given preprocessing for BIS-stability might not be valid for KPR-stability. For these reasons, we opted for a conservative extension of the well-known "Hospitals-offer" and "Residents-apply" algorithms for HRT (see Irving and Manlove [14]). As these algorithms require the absence of ties in the single doctors' preference lists, their extension is only valid for HRC-TCH.

Algorithm 1, "Hospitals-offer-couples", considers in turn every hospital j and stores in $\mathcal{F}$ the c_j doctors (single or in couples) that hospital j most prefers. If the inclusion of the last tie group would make $|\mathcal{F}| > c_j$, then the last tie group is not added to $\mathcal{F}$ but discarded instead. We know that any single doctor i from $|\mathcal{F}|$ cannot be assigned to a hospital strictly worse than j in a stable matching, otherwise (i, j) would form a blocking pair. We also know that, if both members of a couple k are in $|\mathcal{F}|$, and k finds (j, j) acceptable, then k cannot be assigned to a pair of hospitals strictly worse than (j, j), under any stability definition. We do not make any deduction for couples with exactly one member in $|\mathcal{F}|$.

Algorithm 2, "Residents-apply-couples" first checks the first choice of each single doctor and saves them in hospital-specific sets $\mathcal{F}_{H^{c=}_1(i)}$, where $H^{c=}_1(i)$ is the favourite hospital of doctor i. It then checks, for each hospital j, whether or not there are enough doctors in $\mathcal{F}_j$ to fill the capacity c_j of j. If this is the case, it finds the minimum rank m of the doctors in $\mathcal{F}_j$ required

Algorithm 1 `Hospitals-offer-couples`

Input: An instance of HRC-TCH with hospitals H, single doctors S, and couples C
Output: Two sets $\mathcal{R}$ and $\mathcal{R}'$ containing pairs (i, j') and coalitions (i, j', j'') that can neither be part of any stable matching nor cause infeasibility
for each $j \in H$ **do** ▷ for each hospital
  $\mathcal{F} \leftarrow \{i \in D(j) : |D_i^{\leq}(j)| \leq c_j\}$ ▷ $\mathcal{F}$ contains the doctors that j would always select
  for each single doctor $i \in \mathcal{F}$ **do**
    for each $j' \in H(i)$ with $r^D_{j'}(i) > r^D_j(i)$ **do**
      $\mathcal{R} \leftarrow \mathcal{R} \cup \{(i, j')\}$
    end for
  end for
  for each $i, i' \in \mathcal{F}$ such that (i, i') forms couple k and such that $(j, j) \in H^c(k)$ **do**
    for each $(j', j'') \in H^c(k)$ with $r^D_{(j',j'')}(k) > r^D_{(j,j)}(k)$ **do**
      $\mathcal{R}' \leftarrow \mathcal{R}' \cup \{(k, j', j'')\}$
    end for
  end for
end for
return $\mathcal{R}, \mathcal{R}'$

to fill the hospital. From this point, we know that no doctor i (single or in couple) with rank strictly worse than m according to j should ever be assigned to j in a stable matching, otherwise (i', j) would form a blocking pair, where i' is a single doctor in $\mathcal{F}_j$ not assigned to j.

It was noticed in Delorme et al. [9] that ILP models for HRT could have up to $O(n_S n_H)$ constraints and up to $O(n_S n_H(n_S + n_H))$ non-zero elements, depending on the length of the agents' preference lists. In HRCT, couples are usually allowed to have longer preference lists than single doctors: indeed, in the case of a single doctor willing to be assigned to two hospitals h_1 and h_2, for example, four choices are required for a couple with the same preferences, namely $(h_1, h_1), (h_1, h_2), (h_2, h_1)$, and (h_2, h_2). In the Scottish Foundation Allocation Scheme (SFAS), single doctors had a preference list of size up to p (where $p = 10$ in 2012), and couples had a preference list of size up to p^2 (the Cartesian product), bringing the theoretical number of non-zero elements for HRCT models to $O(n_S n_H(n_S + n_H) + n_C {n_H}^2(n_C + {n_H}^2))$.

To reduce the model size, we adopt the same techniques as [9], that is, we employ an alternative formulation that uses dummy variables to keep track of the single doctors, the couples, and the hospitals assignments at each rank.

Let us consider the following additional notation:

- $g^s(i)$ is the number of distinct ranks (or ties) for single doctor i ($i = 1, \dots, n_S$).
- $g^c(k)$ is the number of distinct ranks (or ties) for couple k ($k = 1, \dots, n_C$).
- $g^h(j)$ is the number of distinct ranks for hospital j ($j = 1, \dots, n_H$).
- $H_l^{s=}(i)$ is the set of hospitals acceptable for single doctor i ($i = 1, \dots, n_S$) with rank l ($l = 1, \dots, g^s(k)$).

Algorithm 2 `Residents-apply-couples`

Input: An instance of HRC-TCH with hospitals H, single doctors S, and couples C
Output: Two sets $\mathcal{R}$ and $\mathcal{R}'$ containing pairs (i,j) and coalitions (i,j',j'') that can neither be part of any stable matching nor cause infeasibility

```
for each i ∈ S do                                          ▷ for each single doctor
    F_{H_1^{c=}(i)} ← i                   ▷ i is added in the F of his favourite hospital
end for
for each j ∈ H do                                          ▷ for each hospital
    if |F_j| ≥ c_j then                                    ▷ if preprocessing can be done
        m = min{r_i^H(j) : |D_i^≤(j) ∩ F_j| ≥ c_j}   ▷ j cannot get doctors of rank worse than m
        for each i ∈ D(j) : r_i^H(j) > m do         ▷ for each doctor with rank worse than m
            if i is single then
                R ← R ∪ {(i, j)}                           ▷ Mark the pair (i, j)
            end if
            if i belongs to couple k then
                for each (j', j'') ∈ H^c(k) : j' = j do
                    R' ← R' ∪ {(k, j', j'')}               ▷ Remove all coalitions (k, j', j'')
                end for
            end if
        end for
    end if
end for
```

- $H^{c=}_l(k)$ is the set of pairs of hospitals acceptable for couple k ($k = 1,\dots,n_C$) with rank l ($l = 1,\dots,g^c(k)$).
- $D^{=}_l(j)$ is the set of doctors (single or in couples) acceptable for hospital j ($j = 1,\dots,n_H$) with rank l ($l = 1,\dots,g^h(j)$).

In addition, we introduce dummy binary decision variables w^s_{il} (resp. w^c_{kl}) that take value 1 if single doctor i (resp. couple k) is matched with a hospital (resp. a pair of hospitals) of rank at most l, and 0 otherwise ($i = 1,\dots,n_S$) (resp. ($k = 1,\dots,n_C$)) . We also introduce integer decision variables w^h_{jl} that indicate how many doctors (single or in couple) of rank at most l are assigned to hospital j. MAX-HRCT becomes:

$$\max \quad \sum_{i=1}^{n_S+2n_C} \sum_{j\in H(i)} x_{ij} \tag{22}$$

$$\text{s.t.} \quad (S2)*, (S3)*,$$

$$(7) - (12),$$

$$\sum_{j\in H^{s=}_1(i)} x_{ij} = w^s_{i1}, \qquad i = 1,\dots,n_S, \tag{23}$$

$$\sum_{j\in H^{s=}_l(i)} x_{ij} + w^s_{il-1} = w^s_{il}, \qquad i = 1,\dots,n_S,\ l = 2,\dots,g^s(i), \tag{24}$$

$$\sum_{(j_1,j_2)\in H_1^{c=}(k)} y_{kj_1j_2} = w^c_{k1}, \qquad k = 1,\dots,n_C, \tag{25}$$

$$\sum_{(j_1,j_2)\in H_l^{c=}(k)} y_{kj_1j_2} + w^c_{k,l-1} = w^c_{kl}, \quad k = 1,\dots,n_C,\ l = 2,\dots,g^c(k), \tag{26}$$

$$\sum_{i\in D_1^{=}(j)} x_{ij} = w^h_{j1}, \qquad j = 1,\dots,n_H, \tag{27}$$

$$\sum_{i\in D_l^{=}(j)} x_{ij} + w^h_{j,l-1} = w^h_{jl}, \qquad j = 1,\dots,n_H,\ l = 2,\dots,g^h(j), \tag{28}$$

$$c_j\left(1 - w^s_{i,r_j^D(i)}\right) \le w^h_{j,r_i^H(j)}, \qquad i = 1,\dots,n_S, j \in H(i), \tag{29}$$

$$w^s_{il} \in \{0,1\}, \qquad i = 1,\dots,n_S,\ l = 1,\dots,g^s(i), \tag{30}$$

$$w^c_{kl} \in \{0,1\}, \qquad k = 1,\dots,n_C,\ l = 1,\dots,g^c(k), \tag{31}$$

$$w^h_{jl} \in \{0,1,\dots,c_j\}, \qquad j = 1,\dots,n_H,\ l = 1,\dots,g^h(j). \tag{32}$$

Constraints (23)-(28) maintain the coherence between the "new" variables (w^s_{il}, w^c_{kl}, w^h_{jl}) and the "old" variables (x_{ij}, $y_{kj_1j_2}$). Constraints (29) are the adaptation of (S1). Stability constraints (S2)*, (S3)*, the adaptation of (S2) and (S3), are built in a similar way.

Last, we also propose the following valid inequalities for constraints (S2)* that improve the computational behaviour of the models:

$$\alpha^1_{kj_1j_2} \le 1 - x_{n_S+k,j_2}, \quad k = 1,\dots,n_C, (j_1,j_2) \in H^c(k), \tag{33}$$

$$\alpha^2_{kj_1j_2} \le 1 - x_{n_S+k,j_1}, \quad k = 1,\dots,n_C, (j_1,j_2) \in H^c(k), \tag{34}$$

For a given pair of distinct hospitals (j_1, j_2) in a couple's preference list, constraints (33)-(34) force the $\alpha_{kj_1j_2}$ variable (the "wild card") related to one member of the couple to take value 0 if the other member is assigned to the hospital of their choice in another configuration (j_1 for the first member, or j_2 for the second).

6 Computational experiments

We report in this section the outcome of extensive computational experiments aimed at (i) testing the effectiveness of the proposed improvements for HRC-TCH and HRC-TC, and (ii) provide managerial insights about parameters that might affect the difficulty of the problem and the optimal matching size. All algorithms were coded in `C++`, and Gurobi 7.5.2 was used to solve the ILP models. The implemented software is downloadable from `https://doi.org/10.5281/zenodo.3626706`. The experiments were run on an Intel Xeon E5-2680W v3, 2.50GHz with 192GB of memory, running under Scientific Linux 7.5. Each instance was run using a single core and had a total time limit (comprising preprocessing, model creation time, and solution time) of 3600 seconds per problem instance. The large amount of memory allowed us to run several jobs in parallel. For informative purposes, we also re-ran some of the experiments on a regular desktop with 16GB of memory and we report that no significant difference in terms of average computing time or number of instances solved to optimality was observed. The instances that were randomly generated are downloadable from the online repository `http://researchdata.gla.ac.uk/953/`.

6.1 Instance generation

In many instances of HRCT, it can be assumed that agents establish their ranking based on their own individual preferences. However, sometimes it is the case that agents' preferences are formulated on the basis of objective criteria. For example, in a specific variant of MAX-HRC-TCH that arose in the context of SFAS, hospitals' preference lists are constructed on the basis of doctors' scores. In this situation, the so-called master list, a ranking of all the doctors based on their grades, is formulated at the outset. We remark that, as doctors are graded individually, the fact that they are single or in couple does not interfere with the ranking.

For the instance generation, we used the software described in Irving and Manlove [14] for HRT. The software can generate HR and HRT instances with no master list, and HRT instances with a master list. To mimic HR instances with a master list, we selected the largest possible grade range for the doctors allowed by the generator, which is $[1, 50\,000]$, and we modified any duplicated grades so that the instance contains no ties in the hospitals' preference lists. To include couples in the instance (with a master list or without) we chose X doctors (where X is determined by the percentage of couples in the given instance), and we paired them to form couples. The preference list of a couple $c = (i_1, i_2)$ is the Cartesian product of its two single components formed as follows. If (j_1, j_2) and (j_3, j_4) are two pairs on c's list, where $\left(r^d_{j_1}(i_1), r^d_{j_2}(i_2)\right) = (r, s)$ and $\left(r^d_{j_3}(i_1), r^d_{j_4}(i_2)\right) = (r', s')$, then (j_1, j_2) precedes (j_3, j_4) if and only if either (i) $\frac{r+s}{2} < \frac{r'+s'}{2}$ or (ii) $\frac{r+s}{2} = \frac{r'+s'}{2}$ and $\max\{r, s\} < \max\{r', s'\}$. For example if the preference list of d_1 is $h_1\ h_2\ h_3$ and the preference list of d_2 is $h_3\ h_4\ h_5$, then the preference list of couple (d_1, d_2) is:

$$(h_1, h_3) \quad [(h_1, h_4), (h_2, h_3)] \quad (h_2, h_4) \quad [(h_1, h_5), (h_3, h_3)] \quad [(h_2, h_5), (h_3, h_4)] \quad (h_3, h_5),$$

if partial assignment for couples is not allowed. If partial assignment is allowed, we add "$\emptyset$" to the preference list of each doctor in couple and apply the same procedure. In the given example, the preference list of couple (d_1, d_2) becomes:

$$(h_1, h_3) \quad \ldots \quad [(h_2, h_5), (h_3, h_4)] \quad [(h_1, \emptyset), (\emptyset, h_3)] \quad (h_3, h_5) \quad [(h_2, \emptyset), (\emptyset, h_4)] \quad [(h_3, \emptyset), (\emptyset, h_5)].$$

In a specific set of instances that we generated, couples only apply jointly to the same hospital (i.e., for any pair (h_i, h_j) in a couple's preference list, $h_i = h_j$). In this case, if the preference list of d_1 is $h_1\ h_2\ h_3$, then the preference list of couple (d_1, d_2) is $(h_1, h_1), (h_2, h_2), (h_3, h_3)$, and the preference list of d_2 is simply discarded.

6.2 Model improvements

Considering that the ILP models for the four stability definitions have a similar structure, we decided to test the improvements introduced in Section 5 (namely preprocessing, valid inequalities, dummy variables and constraint merging) on only one of them. We opted for MM-stability as it allows us to measure if the aggregated constraints (20) are better than (19). First, we generated 30 small size instances with 375 doctors, 25 hospitals, 375 positions, preference lists of size 5, an average of 20% of doctors in a couple (i.e., the last 76 doctors were paired to form 38 couples), where the couples cannot be partially assigned. The tie density on the hospital's side was set to 0.85 (the tie density can be interpreted as the probability of an entry in a hospital's list being tied with its successor, see Delorme et al. [9]). We measure the impact of various combinations of valid inequalities, preprocessing, constraint aggregation, and dummy variables (associated with single doctors, couples, and hospitals) in Table 4.

Table 4: Comparison of the improvements efficiency for MM-stability on small-size instances

Method					Values		Model size		
index	valid inequal.	prepro.	stab. cons. merging	dummy variables	#opt	time	number of variables	number of constraints	number of non-zeros
M1					3	3386	6245	6156	492 715
M2	x				16	2198	6245	8016	501 619
M3	x	x			20	1615	4780	6011	253 220
M4	x	x	x		20	1591	4780	5392	184 016
M5	x	x	x	S	21	1517	5696	6308	186 135
M6	x	x	x	S + priority	16	2065	5696	6308	186 135
M7	x	x	x	C	21	1558	5301	5913	174 528
M8	x	x	x	C + priority	18	1851	5301	5913	174 528
M9	x	x	x	H	30	17	5026	5638	39 721
M10	x	x	x	H + priority	30	19	5026	5638	39 721
M11	x	x	x	C, H	30	12	5547	6159	25 670
M12	x	x		C, H	30	14	5547	6778	27 527
M13	x	x	x	S, C, H	30	11	6463	7075	26 873

The “Method” columns detail the combination of options, with some attributes describing the specific implementation: “index” identifies the method while “valid inequalities”, “preprocessing”, “stability constraint merging” and “dummy variables” indicate the inclusion or otherwise of the corresponding feature in the model. The letters “S”, “C”, or “H” note whether dummy variables were used for Single doctors (w^s_{il}), Couples (w^c_{kl}), or Hospitals (w^h_{jl}), respectively. When “priority” is used, it means that we asked the solver to branch first on the dummy variables. The two following columns give some indicators of the performance of each method: the number of optimal solutions found and the average CPU time over all runs (including the ones terminated by the time limit), where all timings reported in this section are in seconds. The last three columns report some details about the model size: average number of variables, constraints, and non-zero elements.

The results in Table 4 show that:

- M1 (the ILP model without any improvement) solves only 3 instances.
- The valid inequalities are very useful as they allow the solver to find 13 additional optimal solutions.
- The preprocessing offers an improvement, as it reduces by roughly 25% the number of constraints and variables in the model.
- Using the merged constraints (20) over (19) brings a marginal improvement: we barely notice any difference between M3 and M4, and between M11 and M12.
- Dummy variables are extremely useful when they are used on the hospitals. They seem to have a marginal effect on single doctors and couples.
- Giving a high priority to the dummy variables does not help the solver.

- The best configuration M13 uses valid inequalities, preprocessing, constraint merging, and all the dummy variables.

Considering these results, we decided to try the best algorithms: M9, M11, M12, and M13 on larger instances. We generated 30 medium size instances with 750 doctors, 50 hospitals, 750 positions, preference lists of size 10, an average of 20% of doctors in couple, couples cannot be partially assigned, and a tie density at 0.85. The results obtained by the algorithms are available in Table 5 and they confirm the comments made previously: the best approach is M13 (followed closely by M11 and M12). We also note that 10 instances could not be solved in one hour of computing time, even with our best algorithm.

Table 5: Comparison of the improvements efficiency for MM-stability on small-size instances

Method					Values		Model size		
index	valid inequal.	prepro.	stab. cons. merging	dummy variables	#opt	time	number of variables	number of constraints	number of non-zeros
M9	x	x		H	18	2327	32 490	34 090	587 983
M11	x	x	x	C, H	20	1935	36 219	37 818	170 346
M12	x	x		C, H	20	2025	36 219	42 170	183 403
M13	x	x	x	S, C, H	20	1711	39 611	41 211	169 585

In the rest of this section, all our algorithms will use configuration M13.

6.3 HRC-TC instances

In order to compare the outcomes in terms of matching size and solving time of the different stability definitions, we first tested them on various HRC-TC instances (i.e., with ties in couples' preference lists only). We created 12 sets of instances, that are described in Table 6.

Table 6: Parameters of the tested sets of HRC-TC instances

Name	Nb. doc.	% couples	Nb. pos.	Nb. hos.	Nb. pref.	Master list	Couple choice	Partial assignment
I1	750	20	750	50	10	No	Cartesian product	not allowed
I2	1500	20	1500	100	10	No	Cartesian product	not allowed
I3	7500	20	7500	500	10	No	Cartesian product	not allowed
I4	750	20	750	50	10	Yes	Cartesian product	not allowed
I5	1500	20	1500	100	10	Yes	Cartesian product	not allowed
I6	7500	20	7500	500	10	Yes	Cartesian product	not allowed
I7	20	20	20	3	2 or 3	Yes	Only the same hospital	not allowed
I8	750	20	750	50	10	Yes	Only the same hospital	not allowed
I1^P	750	20	750	50	10	No	Cartesian product	allowed
I2^P	1500	20	1500	100	10	No	Cartesian product	allowed
I4^P	750	20	750	50	10	Yes	Cartesian product	allowed
I5^P	1500	20	1500	100	10	Yes	Cartesian product	allowed

The first eight sets of instances I1-I8 do not allow partial assignment of couples. Instance sets I1-I3 do not have a master list and can be considered as medium, large, and very large size instances, respectively. Instance sets I4-I6 are the counterpart of I1-I3 when a master list is considered. As the stability definitions only vary when couples apply to the same hospital, we created specific sets of instances I7 and I8 in which the preference lists of the couples are shortened (i.e., not obtained by the Cartesian product) and contain pairs of duplicated hospitals. The following four sets of instances, namely $I1^P$, $I2^P$, $I4^P$, and $I5^P$, allow partial assignment of couples. They are a copy of instance sets I1, I2, I4, and I5, in which the preferences of couples also include partial assignments (i.e., with "$\emptyset$").

We report the results of the three models for MM-, BIS-, and KPR-stability in Tables 7 and 8. We do not provide results for KPR$^+$-stability here, as the model is identical to KPR-stability in the absence of ties in the hospitals' preference lists. Columns "Name" indicate the name of the instance set tested, columns "#opt" indicate how many instances were solved to optimality or proven to be infeasible. Columns "#inf" indicate how many instances were proven to be infeasible. Columns "time" indicate the average CPU time over all runs (including the ones terminated by the time limit). Columns "size" indicate the average matching size (not including the ones with no feasible matching). Finally, columns "% s" and "% c" indicate the percentages of single doctors and doctors in couples that are unassigned.

Table 7: Comparison of MM-, BIS-, and KPR-stability definitions on HRC-TC instances

Name	MM						BIS						KPR					
	#opt	#inf	time	size	% s	% c	#opt	#inf	time	size	% s	% c	#opt	#inf	time	size	% s	% c
I1	30	6	29	743.7	0.9	0.7	30	6	35	743.7	0.9	0.7	30	6	28	743.7	0.9	0.7
I2	30	1	35	1486.6	1.0	0.5	30	0	47	1486.7	1.0	0.5	30	0	38	1486.7	1.0	0.5
I3	28	1	2049	7426.7	1.1	0.7	28	1	2069	7426.7	1.1	0.7	29	1	2068	7426.4	1.1	0.6
I4	30	0	6	725.3	2.7	5.8	30	0	5	725.3	2.7	5.8	30	0	5	725.3	2.7	5.8
I5	30	2	12	1443.3	3.1	6.4	30	2	11	1443.3	3.1	6.4	30	2	12	1443.3	3.1	6.4
I6	30	5	70	7215.2	3.1	6.5	30	5	69	7215.2	3.1	6.5	30	5	69	7215.2	3.1	6.5
I7	30	13	0	19.4	2.2	5.9	30	0	0	19.1	2.7	11.7	30	0	0	19.1	2.7	11.7
I8	30	30	1	n/a	n/a	n/a	30	0	1	725.6	2.5	6.4	30	0	0	726.4	2.4	6.0

The results in Table 7 show that:

- The models can easily solve instances up to 7500 doctors, under any stability definition.
- For instance sets I1-I6, the three models behave similarly in terms of number of instances solved, average matching size, average running time, and percentage of single doctors and couples assigned.
- For many instances of sets I7-I8, MM-stability does not admit any feasible matching.
- When preferences are based on a master list, doctors in couples are more often unassigned (e.g., in I6, 6.5% of the doctors in couples and 3.1% of the single doctors are not assigned). The opposite phenomenon is observed in the absence of a master list (e.g., in I3, 0.7% of the doctors in couples and 1.1% of the single doctors are not assigned).
- On average, matching sizes are smaller when preferences are based on a master list (e.g., the average matching size for I4 is 725.3 vs 743.7 for I1).

For the case where couples can be partially assigned, the results in Table 8 show that:

Table 8: Comparison of MM-, BIS-, and KPR-stability definitions on HRC-TC instances

Name	MM						BIS						KPR					
	#opt	#inf	time	size	% s	% c	#opt	#inf	time	size	% s	% c	#opt	#inf	time	size	% s	% c
$I1^P$	30	0	18	744.0	0.9	0.5	30	0	37	744.0	0.9	0.5	30	0	24	744.0	0.9	0.5
$I2^P$	30	0	58	1487.0	1.0	0.3	30	0	66	1487.0	1.0	0.3	30	0	67	1487.0	1.0	0.3
$I4^P$	30	0	7	727.6	2.9	3.2	30	0	7	727.6	2.9	3.2	30	0	7	727.6	2.9	3.2
$I5^P$	30	0	15	1447.8	3.4	3.7	30	0	14	1447.8	3.4	3.7	30	0	14	1447.8	3.4	3.7

- Allowing couples to be partially assigned slightly increases the average matching size (e.g., 1447.8 for $I5^P$ vs 1443.3 for I5).
- Allowing couples to be partially assigned decreases the number of infeasible solutions (e.g., 0 for $I1^P$ vs 6 for I1).
- Allowing couples to be partially assigned decreases the number of unassigned doctors in couples (e.g., 3.7% for $I5^P$ vs 6.4% for I5).
- There is no difference at all between the three stability definitions in terms of average matching size, average running time, and percentage of single doctors and couples assigned when couples are allowed to be partially assigned.

In Table 16 in the Appendix we also provide additional information about the model sizes. Due to the similar structure of the MM-, BIS-, and KPR-stability models, we do not observe any difference at all for the number of variables and constraints, and a minor difference for the number of non-zero elements.

6.4 HRC-TCH instances

To outline the main differences among the four stability definitions, we used the six sets of instances described in Table 9. The first four sets of instances I9-I12 do not allow partial

Table 9: Parameters of the tested sets of HRC-TCH instances

Name	Nb. doc.	% couples	Nb. pos.	Nb. hos.	Nb. pref.	Master list	Tie density/ Grade range	Couple choice	Partial assignment
I9	750	20	750	50	10	No	0.85	Cartesian product	not allowed
I10	750	20	750	50	10	Yes	[1,10]	Cartesian product	not allowed
I11	20	20	20	3	2 or 3	Yes	[1,2]	Only the same hospital	not allowed
I12	750	20	750	50	10	Yes	[1,10]	Only the same hospital	not allowed
$I9^P$	750	20	750	50	10	No	0.85	Cartesian product	allowed
$I10^P$	750	20	750	50	10	Yes	[1,10]	Cartesian product	allowed

assignment of couples. Instance sets I9 is a medium size set without master list and was already used to test the model improvements. Instance I10 is its counterpart with a master list, where the doctors have a grade between 1 and 10. In instance sets I11 and I12, couples only apply to the same hospital. The following two sets of instances $I9^P$ and $I10^P$ are a copy of I9 and I10 in which partial assignment of couples is allowed.

We report the results of the four models MM-, BIS-, KPR-, and KPR^+-stability in Table 10. The meaning of each column is as before (except that Column "#inf" does not appear as no instance was proven infeasible).

Table 10: Comparison of MM-, BIS-, KPR-, and KPR^+-stability definitions on HRC-TCH instances

Name	MM					BIS					KPR					KPR^+				
	#opt	time	size	% s	% c	#opt	time	size	% s	% c	#opt	time	size	% s	% c	#opt	time	size	% s	% c
I9	20	1711	748.7	0.2	0	20	1838	748.6	0.2	0	19	2007	748.7	0.2	0.1	20	1831	748.6	0.2	0.0
I10	30	52	744.1	0.8	0.8	30	87	744.8	0.7	0.7	30	113	744.3	0.7	0.8	30	62	744.0	0.8	0.8
I11	30	0	19.9	0.2	1.7	30	0	19.9	0.2	1.7	30	0	19.9	0.2	1.7	30	0	19.9	0.4	1.7
I12	30	47	744.1	0.8	0.8	30	70	744.3	0.7	0.8	30	103	744.3	0.7	0.8	30	68	744.0	0.8	0.8
$I9^P$	21	1820	748.8	0.2	0	20	1852	748.5	0.2	0.1	21	1667	748.7	0.2	0.1	20	1714	748.6	0.2	0.1
$I10^P$	30	53	744.4	0.8	0.6	30	92	744.5	0.7	0.8	30	60	744.5	0.7	0.7	30	63	744.3	0.8	0.7

The results in Table 10 show that:

- Average size HRC-TCH instances with master list can be easily solved under any stability definition. Average size HRC-TCH instances without master list are harder to solve.
- For HRC-TCH, the four models behave similarly in terms of number of instances solved, average matching size, average running time, and percentage of single doctors and couples assigned.
- No HRC-TCH instance was proven infeasible under any of the stability definition.
- Allowing couples to be partially assigned does not have a significant impact in these instances of HRC-TCH.
- In HRC-TCH, matching sizes are slightly smaller when preferences are based on a master list, but to a lesser extent than for HRC-TC.

6.5 Varying instance parameters

In this section we modify some families of instances to study the impact of various parameters (such as the couple proportion, the tie density, and the number of available positions in the hospitals) on the models' solutions and performances. As we observed few differences between the four stability definitions, for the rest of the section, all tests are done with KPR^+-stability. Note that we also ran the tests with KPR-stability, and no significant difference was observed in terms of number of instances solved to optimality, number of instances proven to be infeasible, and average running time.

6.5.1 Impact of couple proportion

Biró, Irving, and Schlotter [5] noticed that when the couple proportion increased, it was harder for their algorithms to find an optimal solution. To check whether or not this observation also applies to our models, we created four copies of instance set I1, and paired 0%, 40%, 60%, and 80% of the doctors to form couples (instead of the 20% from the original set). We applied the same procedure to I4, I9, and I10 to also have an overview of the impact of the proportion of couples in the presence of ties and in the presence of a master list.

We report the results of the model for KPR$^+$-stability on the modified I1, I4, I9, and I10 instances in Table 11. The first column indicates the couple proportion, whilst the meaning of the other columns remains unchanged. The numbers in bold were taken from previous tables and are added for the sake of comparisons. We remind the reader that KPR$^+$-stability is identical to KPR-stability in the absence of ties.

Table 11: Impact of couple proportion for KPR$^+$-stability on HRC-TC and HRC-TCH instances

% couples	I1 – HRC-TC no master list									I4 – HRC-TC master list								
	#opt	#inf	time	size	% s	% c	nb. var.	nb. cons.	nb. nz.	#opt	#inf	time	size	% s	% c	nb. var.	nb. cons.	nb. nz.
0	30	0	0	744.0	0.8	n/a	2259	3059	6734	30	0	0	727.6	3.0	n/a	2183	2983	6498
20	**30**	**6**	**28**	**743.7**	**0.9**	**0.7**	39 229	40 829	157 046	**30**	**0**	**5**	**725.3**	**2.7**	**5.8**	41 432	43 036	166 087
40	30	6	67	743.3	1.1	0.6	87 140	89 601	361 369	30	3	143	723.3	2.6	5.0	87 951	90 431	363 442
60	30	7	839	743.0	1.2	0.8	133 215	136 548	560 129	28	6	2266	721.9	2.4	4.6	133 543	136 857	560 322
80	4	0	3548	740.3	2.2	1.1	175 455	179 631	742 881	8	0	3541	724.0	1.5	4.0	176 089	180 263	744 990

% couples	I9 – HRC-TCH no master list									I10 – HRC-TCH master list								
	#opt	#inf	time	size	% s	% c	nb. var.	nb. cons.	nb. nz.	#opt	#inf	time	size	% s	% c	nb. var.	nb. cons.	nb. nz.
0	27	0	728	748.7	0.2	0.0	5503	6303	20 000	30	0	8	744.4	0.7	0.0	3638	4438	13 075
20	**20**	**0**	**1831**	**748.6**	**0.2**	**0.0**	39 611	41 324	165 573	**30**	**0**	**62**	**744.0**	**0.8**	**0.8**	41 405	43 147	174 640
40	7	0	3212	748.3	0.3	0.1	84 532	87 293	361 009	29	0	1011	743.5	0.8	1.0	85 009	87 754	364 216
60	0	0	3601	0	n/a	n/a	129 006	132 794	555 530	28	0	2101	743.2	0.8	1.0	128 625	132 388	555 057
80	0	0	3603	0	n/a	n/a	169 967	174 755	734 578	13	0	3312	743	0.9	0.9	169 539	174 318	734 096

For the four sets of instances, the average size of the optimal matching is almost independent of the couple proportion. However, we observe a strong correlation between the average time required to solve the instance and the couple proportion. This is due to the sharp increase in the model size: the number of variables, constraints, and non-zero elements is roughly multiplied by 5 when the couple proportion goes from 20% to 80%. Thus, the observation made by Biró, Irving, and Schlotter [5] for HRC is also true for HRC-TCH, and independent of the presence of a master list.

6.5.2 Impact of tie density

We observed that matching sizes were bigger for HRC-TCH than for HRC-TC, in particular when a master list is used to order the preference lists of the hospitals. As the master list is based on the doctors' grades, the number of distinct grades has a significant impact on the outcome of the matching. For example, if the grade is between 0 and 100 and rounded to the thousandths, there are 100 000 distinct grades. In this case, the chances of two doctors having the exact same grade are extremely small, and the resulting problem instance will have a very low tie density. If instead, the grades are rounded to the closest unit, the number of distinct grades is now 100. As a result, the tie density and the matching size increase.

It is legitimate to favour a doctor over another if the grade of the former is significantly better than the grade of the latter. However, when this difference is counted in decimals or hundredths, the significance of this difference is less obvious.

To study the impact of the tie density, we created 6 copies of instance set I4, a set of HRC-TC instances with distinct (and integer) grades between 1 and 50 000. In each of the new sets, we reduced the maximum grade range to 5000, 500, 50, 25, 10, and 5 by applying an integer division by 10, 100, 1000, 2000, 5000, and 10 000, respectively.

We report the results of the model for KPR$^+$-stability on the modified I4 instances in Table 12. The first column indicates the grade range, the meaning of the other columns remain unchanged, except for column "td" that indicates the tie density. The numbers in bold were taken from previous tables and are added for the sake of comparisons.

Table 12: Impact of tie density for KPR$^+$-stability on HRC-TC and HRC-TCH instances

Grade range	#opt	#inf	time	size	% s	% c	nb. var.	nb. cons.	nb. nz.	td
[1,50000]	**30**	**0**	**5**	**725.3**	**2.7**	**5.8**	41 432	43 036	166 087	**0.0**
[1,5000]	30	0	4	725.4	2.7	5.8	41 369	43 094	166 334	2.2
[1,500]	30	0	4	725.9	2.6	5.7	40 997	42 722	165 702	15.3
[1,50]	30	0	4	730.0	2.2	4.4	39 767	41 494	164 963	69.6
[1,25]	30	0	4	733.5	1.9	3.2	39 780	41 508	166 554	84.1
[1,10]	30	0	26	743.4	0.8	1.0	41 603	43 334	175 509	94.0
[1,5]	21	0	1410	749.7	0.1	0.0	45 118	46 850	190 785	97.3

As expected, the average matching size increases as the tie density increases. For example, by going from 50 000 distinct grades to 50, the average matching size goes from 725.3 to 730. We also observe that the models take longer to solve for instances with 10 distinct grades or fewer. There are even unsolved instances with 5 distinct grades. This cannot be attributed to the model size as the largest model has less than 20% of additional variables, constraints, and non-zero elements with respect to the smallest model, on average.

6.5.3 Impact of the number of positions

In real-world HRT instances, the number of positions is often similar to the number of doctors (see the real-world instances in Delorme et al. [9]). In theory, this should ensure that every doctor gets a position, however we observe in practice that optimal matchings can have a size that significantly differs from their theoretical upper bound, which is $\min\{\sum_{j\in H} c_j, n_D\}$. This observation is particularly true for HRC-TC instances with a master list: the average size of the optimal matchings for I4 was 725.3 (out of a theoretical maximum of 750). In the following, we call the difference between the average size and the obvious upper bound the "theoretical difference". In that case, the theoretical difference is equal to 24.7. To study the impact of adding or removing positions on the theoretical difference, we created 10 copies of instance set I1, and added $\{-5,\dots,-1,+1,\dots,+5\}$ to each hospital capacity. Negative capacities are increased to 0. We applied the same procedure to I4, I9, and I10.

We report the results of the model for KPR$^+$-stability on the modified I1, I4, I9, and I10 instances in Table 13. The first column indicates the adjustment in the hospital capacities, the meaning of the other columns remain unchanged, except for columns "t-diff" that indicate the average theoretical difference value. The numbers in bold were taken from previous tables and are added for the sake of comparisons.

Table 13: Impact of the number of positions for KPR$^+$-stability on HRC-TC and HRC-TCH instances

Change in c_j	I1 – HRC-TC no master list				I4 – HRC-TC master list				I9 – HRC-TCH no master list				I10 – HRC-TCH master list			
	#opt	#inf	time	t-diff	#opt	#inf	time	t-diff	#opt	#inf	time	t-diff	#opt	#inf	time	t-diff
-5	30	0	0	0.0	30	1	1	0.0	30	0	1	0.0	30	0	28	0.0
-4	30	3	0	0.0	30	2	1	0.0	30	0	4	0.0	30	0	48	0.0
-3	30	3	1	0.0	30	2	2	0.6	30	0	99	0.0	30	0	38	0.0
-2	30	4	4	0.0	30	3	3	1.8	30	0	358	0.0	30	0	24	0.0
-1	30	4	11	0.2	30	5	3	7.0	30	0	459	0.0	29	0	158	0.4
0	**30**	**6**	**28**	**6.3**	**30**	**0**	**5**	**24.7**	**20**	**0**	**1831**	**1.4**	**30**	**0**	**62**	**6.0**
1	30	0	4	0.1	30	0	4	6.8	30	0	122	0.0	30	0	12	0.7
2	30	0	4	0.0	30	1	4	1.4	30	0	42	0.0	30	0	9	0.1
3	30	0	4	0.0	30	0	4	0.4	30	0	11	0.0	30	0	7	0.0
4	30	0	4	0.0	30	0	4	0.0	30	0	5	0.0	30	0	6	0.0
5	30	0	3	0.0	30	0	4	0.0	30	0	4	0.0	30	0	5	0.0

For HRC-TC instances, we observe that the theoretical difference is very low when the hospital capacities are increased by 2 units, bringing the number of positions to 850 and the average matching size to 748.6 in the presence of a master list (750 without). The same comment can be made when the hospital capacities are decreased by 2 units: in that case, the number of positions is 650 and the average matching size is 648.2 in the presence of a master list (650 without). For HRC-TCH instances, an increase or a decrease of the hospital capacities by one unit is enough to considerably reduce the theoretical difference. We also notice that instances with a large difference between the number of positions and the number of doctors are solved faster by our models. Interestingly, only 1 instance is infeasible when the number of positions is higher than the number of doctors, while 27 instances are infeasible in the opposite case.

6.6 Summary of the experiments

We empirically showed on a set of 30 small-size instances that the model improvements we introduced had a significant impact on the time required by the solver to find an optimal solution. Under MM-stability, the initial model could only solve 3 of the 30 instances in 3386 seconds on average while our best configuration could solve all 30 instances in 11 seconds on average. Among all the techniques we tested, we empirically observed that (i) the valid inequalities and the dummy variables associated with the hospitals were the most effective, (ii) the preprocessing was also useful, but to a lesser extent, (iii) the dummy variables associated with the single doctors and the couples had a positive, but marginal effect, and (iv) forcing the solver to branch first on the dummy variables had a negative impact. Additional experiments on medium-size instances confirmed these conclusions. As the ILP models for the four stability definitions have a similar structure, the same outputs can be expected for BIS, KPR, and

KPR$^+$-stability definitions.

For HRC-TC instances (when ties are only allowed on the couples' preference lists), we saw that our best algorithm could solve instances with up to 7500 doctors, 500 hospitals, and 7500 positions under any stability definition in less than an hour. For HRC-TCH instances (when ties are allowed on the couples' and the hospitals' preference lists), we saw that our best algorithm could solve instances with up to 750 doctors, 50 hospitals, and 750 positions under any stability definition in less than an hour. For all HRC-TCH and most of HRC-TC instances, we did not observe any significant difference between the results obtained by the ILP models for MM, BIS, KPR, and KPR$^+$-stability in terms of number of instances solved, average running time, and percentage of single doctors and couples assigned. We noticed however for HRC-TC instances in which the couples' preference lists only contain pairs of duplicated hospitals, that sometimes MM-stability does not admit any feasible matching while the other stability definitions do.

Regarding some instance parameters, we observed that when hospitals' preference lists were based on a master list, the average time required to solve an instance was significantly shorter, the average matching size was significantly smaller, and doctors in couples were more often unassigned, in particular for HRC-TC instances. We also noticed in HRC-TC instances that when couples were allowed to be partially assigned, the number of infeasible solutions was smaller, the average matching size was significantly larger, and the number of unassigned doctors in couples was smaller. In HRC-TCH instances, allowing couples to be partially assigned does not have any significant impact.

Regarding other instance parameters, we empirically showed that the couple proportion does not impact the average matching size, but it has a major impact on the average time required to solve an instance to optimality (instances in which 80% of the doctors are in couples are very difficult to solve while those with no couples are easier). We also outlined that reducing the grade range in the instances with master lists could significantly increase the average matching size, even though it would also make the ILP models take a longer time to be solved. Finally, we observed that there is often a significant number of unoccupied vacancies when the total number of available positions in the hospitals is exactly equal to the total number of doctors, in particular for HRC-TC instances with master list. By allowing only one additional position per hospital, the number of unoccupied vacancies could be significantly reduced and the time required to solve the instance to optimality would be shortened.

7 Concluding remarks

We reviewed three stability definitions originally proposed for HRC and we extended them to HRCT. We also introduced a new stability definition specially tailored for HRCT. We proposed ILP models for each of the stability definitions that only differ by one set of constraints, together with a series of model enhancements based on preprocessing, dummy variables, and valid inequalities. We observed that the enhancements are powerful as they allow more small-size instances to be solved to optimality (from only 3 out of 30 to all of them) in a reduced amount of time (from 3386s to 11s). We showed that the stability definition used does not have a major impact on the solution quality, but we observed that instances of HRC-TC where couples apply jointly to hospitals are more likely to have no stable matching under MM-stability than under any other stability definition. Even though the latter observation could make one think that they should favour BIS, KPR, and KPR$^+$ over MM-stability definition, we mention that it remains the practitioner's decision to choose the stability definition they believe to be the most suitable for their application. For example, BIS-stability emerged from the requirements of the former SFAS matching scheme, while KPR stability was introduced to tackle the needs of the NRMP.

We showed that our models could easily solve HRC-TC instances with up to 7,500 doctors and 500 hospitals with or without master list, and HRC-TCH instances with up to 750 doctors and 50 hospitals. We also outlined some instance parameters that have an impact on the models' performances: (i) a large difference between the number of positions available in the hospitals and the number of doctors, a low tie density, and a low percentage of couples make the ILP models faster to solve; (ii) a high tie density and a higher number of positions available in the hospitals make the matching size bigger. We leave as future work the search for enhanced preprocessing algorithms that are specific to a stability definition, the inclusion of a warm start for the solver based on heuristics that are specific to each stability definition, and the extension of our models to the Workers / Firms problem, the extension of the HRCT in which doctors also have a capacity.

Acknowledgements

We would like to thank the two anonymous reviewers for their valuable comments that have helped to improve the presentation of this paper. This research was supported by the Engineering and Physical Science Research Council through grant EP/P029825/1 (first four authors) and grant EP/P028306/1 (fifth and sixth authors).

References

[1] B. Aldershof and O.M. Carducci. Stable matching with couples. *Discrete Applied Mathematics*, 68:203–207, 1996.

[2] B. Aldershof and O.M. Carducci. Stable marriage and genetic algorithms: a fertile union. *Journal of Heuristics*, 5:29–46, 1999.

[3] I. Ashlagi, M. Braverman, and A. Hassidim. Stability in large matching markets with complementarities. *Operations Research*, 62(4):713–732, 2014.

[4] D. Bianco, S. Hartke, and A. Larimer. Stable matchings in the couples problem. *Morehead Electronic Journal of Applicable Mathematics*, 2, 2001. MATH-2001-06.

[5] P. Biró, R.W. Irving, and I. Schlotter. Stable matching with couples: an empirical study. *ACM Journal of Experimental Algorithmics*, 16, 2011. Section 1, article 2, 27 pages.

[6] P. Biró and F. Klijn. Matching with couples: a multidisciplinary survey. *International Game Theory Review*, 15(2), 2013. article number 1340008.

[7] P. Biró, D.F. Manlove, and I. McBride. The Hospitals / Residents problem with Couples: Complexity and Integer Programming models. In *Proceedings of SEA '14: the the 13th International Symposium on Experimental Algorithms*, volume 8504 of *Lecture Notes in Computer Science*, pages 10–21. Springer, 2014.

[8] D. Cantala. Matching markets: the particular case of couples. *Economics Bulletin*, 3(45):1–11, 2004.

[9] M. Delorme, S. García, J. Gondzio, J. Kalcsics, D. Manlove, and W. Pettersson. Mathematical models for stable matching problems with ties and incomplete lists. *European Journal of Operational Research*, 277(2):426–441, 2019.

[10] J. Drummond, A. Perrault, and F. Bacchus. SAT is an effective and complete method for solving stable matching problems with couples. In *Proceedings of IJCAI '15: the Twenty-Fourth International Joint Conference on Artificial Intelligence*, pages 518–525. AAAI Press, 2015.

[11] D. Gale and L.S. Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 69:9–15, 1962.

[12] D. Gusfield and R.W. Irving. *The Stable Marriage Problem: Structure and Algorithms.* MIT Press, 1989.

[13] R.W. Irving. Stable marriage and indifference. *Discrete Applied Mathematics*, 48:261–272, 1994.

[14] R.W. Irving and D.F. Manlove. Finding large stable matchings. *ACM Journal of Experimental Algorithmics*, 14, 2009. Section 1, article 2, 30 pages.

[15] R.W. Irving, D.F. Manlove, and S. Scott. The Hospitals / Residents problem with Ties. In *Proceedings of SWAT '00: the 7th Scandinavian Workshop on Algorithm Theory*, volume 1851 of *Lecture Notes in Computer Science*, pages 259–271. Springer, 2000.

[16] R.W. Irving, D.F. Manlove, and S. Scott. Strong stability in the Hospitals / Residents problem. Technical Report TR-2002-123, University of Glasgow, Department of Computing Science, 2002. Revised May 2005.

[17] R.W. Irving, D.F. Manlove, and S. Scott. Strong stability in the Hospitals / Residents problem. In *Proceedings of STACS '03: the 20th Annual Symposium on Theoretical Aspects of Computer Science*, volume 2607 of *Lecture Notes in Computer Science*, pages 439–450. Springer, 2003. Full version available as [16].

[18] B. Klaus and F. Klijn. Stable matchings and preferences of couples. *Journal of Economic Theory*, 121:75–106, 2005.

[19] B. Klaus and F. Klijn. Paths to stability for matching markets with couples. *Games and Economic Behavior*, 58:154–171, 2007.

[20] F. Kojima, P.A. Pathak, and A.E. Roth. Matching with couples: Stability and incentives in large matching markets. *Quarterly Journal of Economics*, 128(4):1585–1632, 2013.

[21] 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.

[22] D. Manlove, I. McBride, and J. Trimble. "Almost-stable" matchings in the Hospitals / Residents problem with Couples. *Constraints*, 22(1):50–72, 2017.

[23] D.F. Manlove. Stable marriage with ties and unacceptable partners. Technical Report TR-1999-29, University of Glasgow, Department of Computing Science, January 1999.

[24] D.F. Manlove. *Algorithmics of Matching Under Preferences.* World Scientific, 2013.

[25] D.F. Manlove, R.W. Irving, K. Iwama, S. Miyazaki, and Y. Morita. Hard variants of stable marriage. *Theoretical Computer Science*, 276(1-2):261–279, 2002.

[26] D. Marx and I. Schlotter. Stable assignment with couples: parameterized complexity and local search. *Discrete Optimization*, 8:25–40, 2011.

[27] I. McBride. *Complexity Results and Integer Programming Models for Hospitals / Residents Problem Variants.* PhD thesis, University of Glasgow, 2015.

[28] E.J. McDermid and D.F. Manlove. Keeping partners together: Algorithmic results for the Hospitals / Residents problem with couples. *Journal of Combinatorial Optimization*, 19(3):279–303, 2010.

[29] National Resident Matching Program. `http://www.nrmp.org/main-residency-match-data` (accessed 24 July 2020).

[30] E. Ronn. NP-complete stable matching problems. *Journal of Algorithms*, 11:285–304, 1990.

[31] 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.

[32] A.E. Roth and E. Peranson. The redesign of the matching market for American physicians: Some engineering aspects of economic design. *American Economic Review*, 89(4):748–780, 1999.

[33] UK Foundation Programme. 2019 recruitment stats and facts report. Available from `http://www.foundationprogramme.nhs.uk` (accessed 24 July 2020).

[34] J.E. Vande Vate. Linear programming brings marital bliss. *Operations Research Letters*, 8(3):147–153, 1989.

[35] T. Veskioja and L. Võhandu. A framework for solving hard variants of stable matching within a limited time. In *IADIS International Conference Applied Computing*, volume II, pages 177–182, 2004.

Appendix

Acronym	Meaning
CH	a blocking coalition in HRC/HRCT between a Couple of doctors and the same Hospital
CHH	a blocking coalition in HRC/HRCT between a Couple of doctors and two distinct Hospitals
KPR-stability	Stability definition introduced by Biró, Irving, and Schlotter in [5]
HR	Hospitals/Residents problem
HRC	Hospitals/Residents problem with Couples
HRC-TC	Hospitals/Residents problem with Couples and ties in the couples' preference lists only
HRC-TCH	Hospitals/Residents problem with Couples and ties in the couples' and hospitals' preference lists
HRCT	Hospitals/Residents problem with Couples and Ties
HRT	Hospitals/Residents problem with Ties
ILP	Integer Linear Programming
KPR-stability	Stability definition introduced by Kojima, Pathak, and Roth in [20]
MM-stability	Stability definition introduced by McDermid and Manlove in [28]
NRMP	National Resident Matching Program
SFAS	Scottish Foundation Allocation Scheme
SH	a blocking pair in HRCT between a Single doctor and a Hospital

Mathematical notation	Meaning
c_j	capacity of hospital j
C	set of Couples of doctors
D	set of Doctors (single or in couple)
$D(j)$	set of Doctors acceptable for hospital j
$D_i^{\leq}(j)$	set of Doctors that hospital j ranks at the same level or better than doctor i
$D_i^{<}(j)$	set of Doctors that hospital j ranks strictly better than doctor i
$D_l^{=}(j)$	set of Doctors acceptable for hospital j with rank l
$g^s(i)$	number of distinct ranks (or ties) for single doctor i
$g^c(k)$	number of distinct ranks (or ties) for couple k
$g^h(j)$	number of distinct ranks (or ties) for hospital j
H	set of Hospitals
$H(i)$	set of Hospitals acceptable for doctor i
$H_l^{s=}(i)$	set of hospitals acceptable for single doctor i with rank l
$H_l^{c=}(k)$	set of pairs of hospitals acceptable for couple k with rank l
$H_j^{\leq}(i)$	set of Hospitals that doctor i ranks at the same level or better than hospital j
M	a Matching
$r_j^D(i)$	rank of hospital j for doctor i
$r_i^H(j)$	rank of doctor i for hospital j
S	set of Single doctors

Table 14: Feasible matching for instances with two single doctors, one couple, and no ties

h_1	c_1	Splittable couple*		Unsplittable couple**	
		KPR, KPR$^+$ and BIS	MM	KPR, KPR$^+$and BIS	MM
$AaBC$	2	Aa	Aa	Aa	Aa
	3	AaB	AaB	AaB	AaB
$ABaC$	2	AB, Ba	AB	BC	$\emptyset$
	3	ABa	ABa	ABa	ABa
$ABCa$	2	AB	AB	BC	BC
	3	ABC, BCa	ABC	BC	$\emptyset$
$BAaC$	2	BA, Ba	BA, Ba	BC	BC
	3	BAa	BAa	BAa	BAa
$BACa$	2	BA	BA	BC	BC
	3	BAC, BCa	BAC	BC	$\emptyset$
$BCAa$	2	BC	BC	BC	BC
	3	BCA, BCa	BCA, BCa	BC	BC

* preference list of (A, a) is $(h_1, h_1)[(h_1, \emptyset)\ (\emptyset, h_1)]$

** preference list of (A, a) is (h_1, h_1)

Table 15: Feasible matching for instances with two single doctors, one couple, and ties

h_1	c_1	Splittable couple*			Unsplittable couple**		
		KPR and BIS	KPR$^+$	MM	KPR and BIS	KPR$^+$	MM
$[AaBC]$	2	all-2	all-2	all-2	Aa, BC	Aa, BC	Aa, BC
	3	all-3	all-3	all-3	AaB, AaC, BC	AaB, AaC, BC	AaB, AaC, BC
$A[aBC]$	2	all-2 $\setminus\{BC\}$	Aa, AB, AC	Aa, AB, AC	Aa, BC	Aa	Aa, BC
	3	all-3	AaB, AaC, ABC	AaB, AaC, ABC	AaB, AaC, BC	AaB, AaC	AaB, AaC
$B[AaC]$	2	BA, Ba, BC	BA, Ba, BC	BA, Ba, BC	BC	BC	BC
	3	BAa, BAC, BaC	BAa, BAC, BaC	BAa, BAC, BaC	BAa, BC	BAa, BC	BAa, BC
$[AaB]C$	2	Aa, AB, aB	Aa, AB, aB	Aa, AB, aB	Aa, BC	Aa, BC	Aa, BC
	3	AaB	AaB	AaB	AaB	AaB	AaB
$[ABC]a$	2	AB, AC, BC	AB, AC, BC	AB, AC, BC	BC	BC	BC
	3	ABC, BCa	ABC, BCa	ABC, BCa	BC	BC	BC
$[AB][aC]$	2	AB, Ba	AB, Ba	AB, Ba	BC	BC	BC
	3	ABa, ABC, BaC	ABa, ABC	ABa, ABC	AaB, BC	AaB	AaB
$AB[aC]$	2	AB, Ba	AB, Ba	AB	BC	BC	BC
	3	ABa, ABC, BaC	ABa, ABC	ABa, ABC	ABa, BC	ABa	ABa
$BA[aC]$	2	BA, Ba	BA, Ba	BA, Ba	BC	BC	BC
	3	BAa, BAC, BaC	BAa, BAC	BAa, BAC	BAa, BC	BAa	BAa
$[AB]aC$	2	AB, Ba	AB, Ba	AB, Ba	BC	BC	BC
	3	ABa	ABa	ABa	ABa	ABa	ABa
$[AB]Ca$	2	AB	AB	AB	BC	BC	BC
	3	ABC, BCa	ABC, BCa	ABC	BC	BC	$\emptyset$
$A[aB]C$	2	Aa, AB, aB	Aa, AB	Aa, AB	Aa, BC	Aa	Aa
	3	AaB	AaB	AaB	AaB	AaB	AaB
$B[AC]a$	2	BA, BC	BA, BC	BA, BC	BC	BC	BC
	3	BAC, BaC	BAC, BaC	BAC, BaC	BC	BC	BC

* preference list of (A, a) is $(h_1, h_1)[(h_1, \emptyset)\ (\emptyset, h_1)]$

** preference list of (A, a) is (h_1, h_1)

Table 16: Model sizes for MM-, BIS-, and KPR-stability definitions on HRC-TC instances

Name	MM			BIS			KPR		
	nb. var.	nb. cons.	nb. nz.	nb. var.	nb. cons.	nb. nz.	nb. var.	nb. cons.	nb. nz.
I1	39 229	40 829	160 490	39 229	40 829	157 160	39 229	40 829	157 046
I2	78 358	82 020	318 390	78 358	82 020	315 077	78 358	82 020	315 017
I3	389 750	409 843	1 574 302	389 750	409 843	1 570 967	389 750	409 843	1 570 956
I4	41 432	43 036	170 144	41 432	43 036	166 215	41 432	43 036	166 087
I5	81 141	84 817	330 197	81 141	84 817	326 389	81 141	84 817	326 327
I6	403 535	423 771	1 632 563	403 535	423 771	1 628 462	403 535	423 771	1 628 448
I7	86	102	251	86	102	242	86	102	241
I8	12 946	12 596	55 465	12 946	12 596	41 020	12 946	12 596	38 594