The final publication is available at 10.1007/s10472-015-9491-5

Matching Couples with Scarf's Algorithm

Péter Biró[1]

Institute of Economics, MTA KRTK
and
Dept. of Op. Res. and Act. Sci., CUB
1112. Budapest, Budaörsi út 41., Hungary
peter.biro@krtk.mta.hu

Tamás Fleiner[2]

Dept. of C.S. and Inf. Theory
Budapest Univ. of Technology and Economics
1117 Budapest, Magyar tudósok körútja 2.,
Hungary
fleiner@cs.bme.hu

Rob Irving

School of Computing Science
University of Glasgow
Sir Alwyn Williams Building,
Glasgow G12 8QQ, UK
Rob.Irving@glasgow.ac.uk

Abstract: Scarf's algorithm [20] provides fractional core elements for NTU-games. Biró and Fleiner [3] showed that Scarf's algorithm can be extended for capacitated NTU-games. In this setting agents can be involved in more than one coalition at a time, cooperations may be performed with different intensities up to some limits, and the contribution of the agents can also differ in a coalition. The fractional stable solutions for the above model, produced by the extended Scarf algorithm, are called stable allocations. In this paper we apply this solution concept for the Hospitals / Residents problem with Couples (HRC). This is one of the most important general stable matching problems due to its relevant applications, also well-known to be NP-hard. We show that if a stable allocation yielded by the Scarf algorithm turns out to be integral then it provides a stable matching for an instance of HRC, so this method can be used as a heuristic. In an experimental study, we compare this method with other heuristics constructed for HRC that are applied in practice in the American and Scottish resident allocation programs, respectively. Our main finding is that the Scarf algorithm outperforms all the other known heuristics when the proportion of couples is high.

Keywords: Scarf lemma, stable allocation, hospitals residents problem, couples

JEL codes: C71, C78

1 Introduction

Mechanism design in matching markets dates back to the seminal paper of Gale and Shapley [7] on college admissions. [1] They introduced the concept of *stable matching*, that is a fair solution where an application of a student can be rejected by a college only if its quota is filled with better candidates. Gale and Shapley gave an efficient algorithm to find a stable matching in this setting. It turned out [17] that

[1]Corresponding author. Research is supported by the Hungarian Academy of Sciences under its Momentum Programme (LD-004/2010), and by the Hungarian Scientific Research Fund - OTKA (no. K108673).

[2]The author is a member of the MTA-ELTE Egerváry Research Group. The research was supported by the Hungarian Scientific Research Fund - OTKA (no. K108383).

[1]The importance of this contribution has been widely recognised, since Alvin E. Roth and Lloyd Shapley have received the 2012 Nobel-prize in Economic Sciences for their work on the theory of stable matchings and their applications.

the same method had already been used in the US resident matching program since 1952 (now called NRMP, see [17, 19, 24]). This program was redesigned later [18], partly because the organisers wanted to accommodate the wishes of couples. Since then couples can submit joint preference lists in order to avoid being matched to hospitals far from each other. However, for the *Hospitals / Residents problem with couples* (HRC) the existence of a stable matching is no longer guaranteed [17]. Moreover, the related decision problem is NP-hard [16], therefore we need to use heuristics for large markets. There are many recent papers on this problem by economists [18], [10], [11], [12], and by computer scientists [1], [21], [13], [14] as well, see also an interdisciplinary survey [5]. In particular, Biró et al. [4] compared some old and new heuristics for a setting that formerly featured in the Scottish resident allocation program [23, 25].

Another seminal paper in cooperative game theory is by Scarf [20]. He gave an algorithm to find a core element for any *balanced NTU-game*. Aharoni and Fleiner [2] used this algorithm to find *stable fractional matchings* for problems where the underlying graph is not necessarily bipartite. Biró and Fleiner [3] generalised this result by showing that Scarf's algorithm can be extended to find *stable allocations* for NTU-games where both the agents and their cooperations can have capacities, and even when agents have different contributions in a co-operation performed.

In this paper first we show that a restricted version of the HRC problem where couples may not apply for a pair of positions in the same hospital, can be reduced to an integral stable allocation problem (ISA) with no edge capacities, where we have a one-to-one correspondence between the stable solutions of the two models. Furthermore, we show that the general HRC problem (where couples may apply for a pair of positions in the same hospital) can be transformed into an integral stable allocation problem with contributions (ISAC) and with no edge capacities, where the stable solutions for the latter problem are stable solutions for the former. Biró and Fleiner [3] demonstrated that the original Scarf algorithm always returns a stable allocation for both the ISA and ISAC problems with no edge capacities, and whenever it outputs an integer solution it solves the problem. Therefore, the Scarf algorithm can be used as a heuristic to solve all of these problems, ISA, ISAC and HRC as well.

After this, we present an experimental study, that follows up the work by Biró, Irving and Schlotter [4], where we compare the performance of the Scarf algorithm and other heuristics described in [18] and [4]. Our main finding is that the Scarf algorithm works very well if the proportion of couples is high. Although it is unrealistic to suppose that all the applicants form couples in a resident allocation program, there can be other applications with a similar feature. For instance, in the Hungarian higher education matching scheme [22] students can apply to pairs of teacher programmes, and they usually do so in a very concentrated way. This results in a very similar problem, as a student applying to pairs of programmes can be seen as a couple applying to pairs of positions. A similar situation occurred in the Scottish resident allocation scheme (SPA) from 2000 to 2005, where the medical doctors had to apply for two posts, a medical and a surgical one [8], although the elicitation of their preferences was more restricted than in the Hungarian application.

2 Description of the models

First we present a model for the Hospitals / Residents problem with couples, similar to the one described by Biró, Irving and Schlotter [4]. Then we present the integral stable allocation problem with contributions by Biró and Fleiner [3]. Finally we describe the connections between some variants of these two problems, and a possible implementation of the Scarf algorithm for this setting. Finally we give an illustrative example.

2.1 The Hospitals / Residents problem with Couples (HRC)

In the Hospitals / Residents problem with Couples (HRC) we are given a set of applicants $\mathcal{A} = \{a_1, a_2, \dots, a_t\}$ and a set of hospitals $\mathcal{H} = \{h_1, h_2, \dots, h_{n_h}\}$ with $b(h_p)$ denoting the capacity of hospital h_p. The set of applicants can be partitioned into single applicants $\mathcal{S} = \{a_1, a_2, \dots, a_{n_s}\}$ and members of couples, where the set of couples is denoted by $\mathcal{C} = \{c_1, c_2, \dots, c_{n_c}\}$. The agents in this model will be the hospitals, the single applicants and the couples. We denote the set of agents by $\mathcal{Y} = \mathcal{H} \cup \mathcal{S} \cup \mathcal{C}$, where $|Y| = n = n_h + n_s + n_c$, and sometimes we will use notation $y_i \in Y$ for referring to an agent of some kind (i.e. hospital, single applicant or couple). The preference list of a single applicant $a_i \in \mathcal{S}$ contains hospitals in the order of her preferences, whilst the preference list of couple $(a_i, a_j) \in \mathcal{C}$ contains pairs of hospitals in the order of their joint preferences. An application of a single applicant to a hospital is referred to as a *single application*, whilst a *joint application* is made by a couple to a pair of hospitals. In our general model we allow a couple to apply for a pair of positions in the same hospital, in which case we refer to this application as a *combined application*. Let us denote a general single application by $[a_i \to h_p]$, a joint application by $[(a_i, a_j) \to (h_p, h_q)]$ and a combined application by $[(a_i, a_j) \to h_p]$, with the three corresponding sets of applications denoted by E^S, E^J and E^C, respectively, where $E^S \cup E^J \cup E^C = E = \{e_1, e_2, \ldots, e_m\}$. For simplicity we assume that the residents apply to hospitals that are acceptable for them and every resident applying to a hospital is acceptable for that hospital. A *matching* M consists of employments of form $\langle a_i, h_p \rangle$, where if $a_i \in \mathcal{S}$ then $[a_i \to h_p] \in E^S$, and if $(a_i, a_j) \in \mathcal{C}$ then either $[(a_i, a_j) \to (h_p, h_q)] \in E^J$ and also $\langle a_j, h_q \rangle \in M$, or $[(a_i, a_j) \to h_p] \in E^C$ and also $\langle a_j, h_p \rangle \in M$. Let $M(a_i)$ denote the hospital where a_i is allocated in M, if any, and let $M(h_p)$ denote the set of applicants allocated to h_p. A matching has to respect the capacity constraints, i.e. no applicant may be allocated to more than one hospital and no hospital may employ more residents than its capacity, so $|M(a_i)| \le 1$ and $|M(h_p)| \le b(h_p)$. We say that h_p is *full* with respect to M if $|M(h_p)| = b(h_p)$ and *undersubscribed* otherwise. Finally we note that we can also accommodate the possibility that only one member of a couple applies for a position (and the other member remains unmatched) in our model by simply introducing a dummy hospital with no capacity constraint, which would correspond to the outside option.

To define stability we need to specify the preferences of the hospitals over the set of applications. We suppose that every hospital has a strict preference ordering over the acceptable applicants. From these rankings we will derive the definition of stability, that is a slight extension of the stability definition by Biró, Irving and Schlotter [4][2]. A matching M is *stable* if it is not *blocked* by a pair $\langle a_i, h_p \rangle$ consisting of a single applicant a_i and a hospital h_p, or by a coalition $\langle (a_i, a_j), (h_p, h_q) \rangle$ consisting of a couple (a_i, a_j) and distinct hospitals h_p and h_q, or by a coalition $\langle (a_i, a_j), h_p \rangle$ consisting of a couple (a_i, a_j) and a hospital h_p.

Definition 1 *A single applicant a_i and a hospital h_p* block M *if*

- *(a) a_i is unmatched, or prefers h_p to $M(a_i)$; and*
- *(b) h_p is undersubscribed, or ranks a_i higher than a member of $M(h_p)$.*

A couple (a_i, a_j) and an acceptable pair of distinct hospitals h_p and h_q block M *if*

- *(c) a_i and a_j are unmatched, or (a_i, a_j) prefers (h_p, h_q) to $(M(a_i), M(a_j))$; and*
- *(d) h_p is undersubscribed, or $h_p = M(a_i)$, or h_p ranks a_i higher than a member of $M(h_p)$; and*
- *(e) h_q is undersubscribed, or $h_q = M(a_j)$, or h_q ranks a_j higher than a member of $M(h_q)$.*

We say that a couple (a_i, a_j) and a hospital h_p, acceptable to both a_i and a_j, block M *if*

- *(f) a_i and a_j are unmatched, or (a_i, a_j) prefers h_p to $(M(a_i), M(a_j))$; and*
- *(g) either*
  - *(i) h_p has at least two free places in M; or*

[2] In the model described in [4] the applicants are ranked in the same way by all hospital according to a *master list*, and the joint applications of the couples are derived from the individual preferences by the members of the couple in a specific way. Our definition presented here is an extension of that model as we allow hospitals to rank their applicants differently and we have no restriction on how the joint preference lists of the couples are formed.

– *(ii)* h_p *has one free place in* M*, and* $h_p \in \{M(a_i), M(a_j)\}$ *or both* a_i *and* a_j *are higher ranked by* h_p *than a member of* $M(h_p)$*; or*

– *(iii)* h_p *is full in* M *and*

   1. $h_p \in \{M(a_i), M(a_j)\}$ *and both* a_i *and* a_j *are higher ranked by* h_p *than a member of* $M(h_p)$*; or*
   2. *both* a_i *and* a_j *are higher ranked by* h_p *than a member* a_k *of* $M(h_p)$*, and* a_k *is a linked applicant whose partner is also in* $M(h_p)$*; or*
   3. *both* a_i *and* a_j *are higher ranked by* h_p *than at least two members of* $M(h_p)$*.*

The rationale behind this stability definition is described in [4] in detail[3]. We do not say that this is the only reasonable stability definition. However, as described in [4] and as we will also show in Subsection 2.4, this stability definition implies a natural preference ordering over the applications by the hospitals, and the choice functions expressed by the hospitals over the set of applications can be derived from these preferences in a responsive way. This is the key fact that ensures that this problem can be transformed to the ISAC model.

2.2 Couples problems under quota-responsive choice functions

In this subsection we show that the above defined HRC model can be described with a general matching with couples model, where the choice functions of the hospitals satisfy the so-called *quota-responsive property*. This property turns out to be crucial in proving the solvability of the HRC problem with the Scarf algorithm.

In a general model of *matching with couples* (see e.g. in [15]), the hospitals have choice functions over the set of applications. Formally, let Ch_{h_p} denote the choice function of hospital h_p, which means that from a set of applications $X \subseteq E$, Ch_{h_p} will be the set of chosen applications by h_p. The choice function of a single applicant or a couple will simply select the best application from a set of applications. For a set of applications X, let X_{y_i} denote the applications in which agent i is involved, where agent i can be a single applicant, a couple or a hospital. A *stable matching* in this general setting can be defined as a set of applications X if $Ch_{y_i}(X) = X_{y_i}$ and there exists no application $e \in E \setminus X$ such that $e \in Ch_{y_i}(X \cup e)$ for all agent i who is involved in e. The first condition means that the matching is *individually rational* and the second condition ensures the lack of a blocking coalition.

A choice function of a hospital is said to be *quota-responsive*[4] if it selects the chosen set in a greedy fashion. More precisely, for each choice function Ch_{h_p} we require that h_p has a strict preference order over the applications it is involved in, denoted by $\prec_{h_p}$, and from a set of applications h_p greedily selects the best applications according to $\prec_{h_p}$ until the number of selected applicants would violate its quota. Note that if no combined applications are present then the hospital simply chooses the best quota-many applications from the set offered. However, when combined applications are allowed then some combined applications might be rejected, when the hospital has only one position remained, before the last slot of the hospital is filled by a single or joined application.[5]

Proposition 2 *The HRC problem is a matching with couples problem under quota-responsive choice functions.*

[3]Here we only note that this particular stability definition reflects the aim of the matching scheme coordinators to have cutoff scores announced for every hospital. A single application is rejected if the resident does not achieve the cutoff score. Similarly, a joint application is rejected if either of the residents in the couple does not achieve the cutoff score at a hospital. Therefore the fairness of the allocation can be verified relatively easily, at least for single and joint applications. We note that cutoff scores are used in many higher education matching schemes as well, such as the Hungarian, Irish and Spanish systems; see [6].

[4]This property was called *generalized responsive* in [15] in a more general setting.

[5]We also note that in this definition one could forbid the hospitals to select two applications, where the same applicant is involved, as the hospital cannot hire someone twice. However, there would be no difference in the definition of stable matchings, since a solution where multiple applications of an applicant are chosen is not individually rational for that applicant.

PROOF: The strict preference $\prec_{h_p}$ over the set of applications of hospital h_p is created as follows. Suppose first that hospital h_p has to decide between two combined applications, say $[(a_i, a_j) \to h_p]$ and $[(a_k, a_l) \to h_p]$, where a_j is higher ranked than a_i and a_l is higher ranked than a_k. In this case h_p will decide according to the *worst* candidates of these couples, so it would prefer the first application if and only if a_i is higher ranked than a_k. Now, let us consider how the hospital would choose between a resident a_i applying in a single or joint application and a pair of residents (a_j, a_k) applying in a combined application, where a_k is higher ranked by h_p than a_j. Again we suppose that h_p will decide according to its ranking on the weakest candidates, so it will prefer the application involving a_i if and only if a_i is higher ranked than a_j by h_p. It is possible that indifferences occur in this preference order between two joint or combined applications from the same couple, but we suppose that the hospital breaks these ties according to the couple's preferences. What remains is to verify that an application e is blocking for a matching M in the HRC context if and only if e is blocking for the corresponding set of applications X with regard to the above defined quota-responsive choice functions. This is obvious for single and joined applications. Regarding a combined application $[(a_i, a_j) \to h_p]$, condition (g) of the HRC stability definition is satisfied if and only if h_p would select this application when offered together with the set of applications X. □

2.3 Integral stable allocation problem with contributions

The stable allocation problem with contributions can be defined for hypergraphs as follows. Suppose that we are given a hypergraph $H(V, E)$ and for each vertex $v \in V(H)$ a strict preference order over the edges incident with v, that corresponds to the preferences of the players over the *contracts* in which they can be involved, where $e <_v f$ denotes that player v prefers contract f to e. Furthermore, we introduce a contribution vector $r_e : V(H) \to \mathbb{R}_+$ for each edge e of the hypergraph that represents the *contributions* of the agents in contract e. We assume that $v \in e$ if and only if $r_e(v) > 0$, that is when agent v can contribute to contract e. Suppose that we are given nonnegative *bounds* on the vertices $b : V(H) \to \mathbb{R}_+$ and nonnegative *capacities* on the edges $c : E(H) \to \mathbb{R}_+$. A nonnegative function x on the edges is an *allocation* if $x(e) \leq c(e)$ for every edge e and $\sum_{e:v\in e} x(e)r_e(v) \leq b(v)$ for every vertex v. An allocation is *stable* if every *unsaturated* edge e (i.e., every edge e with $x(e) < c(e)$) contains a vertex v such that $\sum_{f:v\in f, e\leq_v f} x(f)r_f(v) = b(v)$. If every bound, capacity and contribution is integral and the problem is to find an integral stable allocation x then we refer to this problem as the *integral stable allocation problem with contributions* (ISAC). Finally, if we allow unit contributions only then we get the *integral stable allocation problem* (ISA).

Biró and Fleiner [3] showed that every stable allocation problem with contributions has a solution that can be obtained by the extended Scarf algorithm. Furthermore, if we have no capacities on the edges then the existence of a stable allocation is guaranteed by the original Scarf lemma, and one stable allocation can be obtained by the original Scarf algorithm.

The exact description of the Scarf algorithm can be found in [20] and its extension for ISAC is described in [3]. Here we only want to highlight some important facts of the Scarf algorithm (and its extension). The algorithm starts by perturbing the capacities (and also the preference matrix if there are ties in the preferences, which is not the case in our particular setting). Based on the perturbation, the Scarf algorithm is deterministic. It takes a well-defined pivot step in each round, which can be implemented efficiently, i.e. in polynomial time in the number of edges of the underlying hypergraph (the number of possible contracts). However, the number of pivot steps in the Scarf algorithm can be exponentially large, as the problem of finding a stable allocation is PPAD-hard even for unit capacities [9]. Some related open questions are listed in [3] and also in the last section of this paper.

2.4 Solving HRC with the Scarf algorithm

In this subsection we show how the HRC problem can be reduced to an ISA or ISAC problem, depending whether combined applications are allowed. These results are based on the fact that the HRC problem is a special matching with couples problem under quota-responsive choice functions, as shown in Proposition

2. Therefore our results can be extended to these more general class of problems. First we consider the case when no combined applications are allowed.

Theorem 3 *An instance of HRC with no combined applications can be reduced to an instance of ISA, where the set of stable matchings in the former problem is in one to one correspondence with the set of integral stable allocations in the latter.*

Proof: Suppose that we have an instance of HRC, as described in subsection 2.1. We create an instance of ISA with no edge capacities as follows. Vertex set V of hypergraph $H(V, E)$ represents the agents, namely the single applicants, couples and hospitals. The edges E of $H(V, E)$ correspond to the applications (i.e. possible contracts). For a single application let the contribution of both the resident and the hospital in this contract be one, and similarly, in a joint application let the contributions of the couple and the two hospitals be one each. The capacities of the hospitals are the same as the capacities of the corresponding vertices, the other vertices have unit capacities and no capacity restriction is needed for the edges.

The preferences of the vertices over the contracts in the ISA instance should be the same as the preferences of the agents (i.e. single applicants, couples and hospitals) over the applications they are involved in, as defined in the proof of Proposition 2. It is now immediate to see the one-to-one correspondence between the set of stable matchings of the HRC instance and the set of integral stable allocations of the ISA instance: an application is accepted if and only if the corresponding edge has unit value in the stable allocation. A single or joint application is blocking in the HRC context if and only if it is blocking for the quota-responsive choice functions defined in the more general matching with couples setting if and only if the corresponding edge is blocking in the ISA instance. □

Theorem 4 *An instance of HRC can be transformed to an instance of ISAC, where the integral stable allocations for the latter problem correspond to the stable matchings for the former one.*

Proof: Suppose that we have an instance of HRC, as described in Subsection 2.1; we create an instance of ISAC as follows. In addition to the proof of Theorem 3, here we need to adjust our construction by accommodating the combined applications. A combined application is also represented by an edge in the hypergraph, where the contribution of the couple is one and the contribution of the hospital is two, since the couple applies for two positions at the hospital.

Here again the vertices in the ISA instance that correspond to the hospitals should have the same strict preferences over the contracts which defined the quota-responsive choice functions over the set of applications, as described in the proof of Proposition 2. One can now verify that an integral stable allocation of the ISAC instance corresponds to a stable matching of the HRC instance, where the edges with unit values represent the accepted applications, since a blocking application for the HRC instance (which is also blocking for the more general matching with couples problem under quota-responsive choice functions) is a blocking edge in the ISAC instance as well. □

Note that the converse of the above theorem is not true. A stable matching for a HRC instance might not be an integral stable allocation for the corresponding ISAC instance. This is because in the HRC stability definition it is possible that a combined application is rejected as there is only one place left at a hospital, but a less preferred single or joint application is accepted for that last place. In particular, suppose that there is only one position at hospital h_1 and there is a combined application $[(a_1, a_2) \to h_1]$ and a single one $[a_3 \to h_1]$, where the applicants are ranked according to their indices. Here the unique stable allocation is when the combined application is taken with half weight. However, the only stable matching is when a_3 is matched to h_1. This stable matching cannot be translated to a stable allocation as the rejected combined application would emerge as a blocking edge.

When combined applications are allowed then there may be different reasonable stability concepts. Our transformation described in Theorem 4 works as long as the hospitals can rank the applications, and from these strict preferences we may derive the hospitals' choice functions over the set of of applications in a quota-responsive way. The latter means that from a set of applications a hospital accepts the best applications one by one, as long as the acceptance of an application does not violate its quota.

Finally we note that if every hospital has one position only then we get a stable matching problem for a hypergraph [2], that is also known as an NTU-game, which is almost a *hedonic coalition formation game*, except that a couple can apply to a pair of hospitals in two possible ways. It is worth mentioning that the NP-hardness results [16], [4] hold for this simple case as well.

2.5 An implementation of the Scarf algorithm for HRC

In this subsection we give a possible implementation of the Scarf algorithm for HRC to make the paper more self-contained, further details on this can be found in [3]. The following description of the Scarf Lemma [20] is due to Aharoni and Fleiner [2] (here $[n]$ denotes the set of integers $1, 2, \ldots, n$, and $\delta_{i,j} = 1$ if $i = j$ and 0 otherwise).

Theorem 5 (Scarf Lemma [20]) *Let n, m be positive integers, and b be a vector in $\mathbb{R}^n_+$. Also let $A = (A_{i,j}), C = (C_{i,j})$ be matrices of dimension $n \times (n+m)$, satisfying the following three properties: the first n columns of A form an $n \times n$ identity matrix (i.e. $A_{i,j} = \delta_{i,j}$ for $i, j \in [n]$), the set $\{x \in \mathbb{R}^{n+m}_+ : Ax = b\}$ is bounded, and $C_{i,i} < C_{i,k} < C_{i,j}$ for any $i \in [n], i \neq j \in [n]$ and $k \in [n+m] \setminus [n]$.*

Then there is a nonnegative vector x in $\mathbb{R}^{n+m}_+$ such that $Ax = b$ and the columns of C that correspond to $supp(x)$ form a dominating set, that is, for any column $i \in [n+m]$ there is a row $k \in [n]$ of C such that $C_{k,i} \leq C_{k,j}$ for any $j \in supp(x)$.

For a given instance of HRC, we now show a possible setting for matrices A and C such that the solution x of the above Scarf Lemma corresponds to a stable allocation of the corresponding instance of ISAC (and an integral solution x corresponds to a stable matching for the HRC problem). For $i \in [n]$, let row i and column i of matrices A and C correspond to agent $y_i \in Y$, and for $j \in [m]$, let column $n+j$ correspond to application $e_j \in E$. When ordering the agents we will take the single applicants first, followed by the couples and then by the hospitals. Let the $n \times m$ submatrix of A on the right side of A contain the contribution vectors of the applications, namely

- for a single application $e_j = [a_i \to h_p]$, let $A_{i,n+j} = A_{n_s+n_c+p,n+j} = 1$,
- for a joint application $e_j = [c_k \to (h_p, h_q)]$, let $A_{n_s+k,n+j} = A_{n_s+n_c+p,n+j} = A_{n_s+n_c+q,n+j} = 1$ and
- for a combined application $e_j = [c_k \to h_p]$, let $A_{n_s+k,n+j} = 1$ and $A_{n_s+n_c+p,n+j} = 2$.

All the other elements of A are set to be zero. Regarding C, let $C_{i,i} = 0$ for every $i = 1 \ldots n$. For every pair i and j ($i = 1 \ldots n$ and $j = 1 \ldots n+m$) if $A_{i,j} = 0$ then let $C_{i,j} = n+2m+1-j$. Otherwise, if $e_j \in E$ is the k-th best application of agent y_i then let $C_{i,n+j} = m+1-k$, where the preferences of hospital h_p are as explained in the proof of Proposition 2, denoted by $\prec_{h_p}$. Finally, for the i-th coordinate of the right hand side vector b, let $b_i = 1$ if $y_i \in \mathcal{S} \cup \mathcal{C}$ and let $b_i = b(y_i)$ if $y_i \in \mathcal{H}$ (i.e. the capacity of the hospital).

This setting satisfies the conditions of Scarf Lemma. Furthermore, for an agent y_i if contract e_j does not involve y_i then $C_{i,n+j} \geq m$, and if e_j involves y_i then $C_{i,n+j} < m$ with higher value when e_j is more preferred by y_i. Therefore we can show that *supp*(x) is a dominating set for a solution x if and only if it is also a stable allocation for the corresponding ISAC problem (see details in [3]). Now we describe the execution of the Scarf algorithm, as described in [20].

Scarf algorithm

The Scarf algorithm is an iterative algorithm where each phase consist of a *pivot step* and an *ordinal step*. Let x^i denote the feasible solution (i.e. where $Ax^i = b$) after the i-th pivot step, and for a feasible solution x^i, let $FB^i = supp(x^i)$ be the *feasible basis*. At the beginning of the process we set $FB^0 = [n]$, which means that initially we have $x^0(j) = b_j$ for each index $j \in \{1 \ldots n\}$ (i.e. for each agent $y_j \in \mathcal{Y}$).

a_3 :	h_1	h_5	
a_7 :	h_6	h_8	
(a_1, a_5) :	(h_1, h_2)	(h_3, h_6)	
(a_2, a_4) :	(h_4, h_5)	(h_1, h_2)	(h_3, h_7)
(a_6, a_8) :	(h_6, h_8)		

We also maintain an *ordinal basis*, denoted by OB^i after the i-th ordinal step. OB^i always forms a dominating set, that is, for any column $i \in [n+m]$ there is a row $k \in [n]$ of C such that $C_{k,i} \leq C_{k,j}$ for any $j \in OB^i$. Moreover, each of the columns in OB^i has a row-minimal element in the submatrix of C restricted to these columns. Let $q(i)$ denote the index of the row where column i has a row-minimum in this submatrix. During the execution of the algorithm the two basis will always share $n-1$ coordinates and the algorithm will terminate whenever they coincide. Now we describe the initial ordinal step and the subsequent pivot and ordinal steps.

Initial ordinal step: Let $OB^0 = FB^0 \setminus \{1\} \cup t_0$, where $t_0 \in [n+m] \setminus [n]$ such that C_{1,t_0} is maximal. Note that OB^1 forms a dominating set.

***i*-th pivot step for $i \geq 1$:** We add t_{i-1} to FB^{i-1}, where t_{i-1} is the new element that was added to OB^{i-1}. Let s_i denote the coordinate that is now removed from FB^i due to the basis transformation. If $s_i = 1$ then $FB^i = OB^{i-1}$ and the algorithm terminates.

***i*-th ordinal step for $i \geq 1$:** We remove s_i from OB^{i-1}. Now there will be a column j in $OB^{i-1} \setminus \{s_i\}$ with two row-minimums. Suppose that j had a row-minimum in row k in OB^{i-1}, so currently $q(j) = k$, and let l denote the new row where the element of column j is also minimal in submatrix $OB^{i-1} \setminus \{s_i\}$. Let us reset $q(j) = l$. Now we will add a new column t_i in such a way that t_i will have a row-minimum in row k and the row-minimal elements of the other columns remain the same, formally, let t_i be the column such that C_{k,t_i} is maximal under the restriction that $C_{q(j),j} < C_{q(j),t_i}$ for each $j \in OB^{i-1} \setminus \{s_i\}$. Let then $OB^i = OB^{i-1} \setminus \{s_i\} \cup \{t_i\}$ and $q(t_i) = k$. If $FB^i = OB^i$ then the algorithm terminates.

As Scarf [20] showed this process is well-defined and terminates in finite time if no degeneracy occurs. To avoid degeneracy, besides having different values in every row of C, vector b representing the vertex-bounds should be also perturbed (see more about this issue in [3]).

Finally, we illustrate the usage of the Scarf algorithm for a particular instance of HRC.

2.6 Example

This example was given by Biró, Irving and Schlotter [4] as a difficult instance of HRC, for which most heuristics currently used in real applications would fail to find the unique stable solution. Suppose that we have eight residents, comprising three couples (a_1, a_5), (a_2, a_4) and (a_6, a_8) together with two single applicants a_3 and a_7. There are eight hospitals, $h_1, \ldots, h_8$, each with just one post. Suppose that the residents are ordered in the same way by every hospital, according to their indices (a_1 best, a_8 worst), and the individual and joint preference lists of the residents are as follows.

We can describe this problem as an instance of ISA, where the individual applicants, the couples and the hospitals are represented by vertices, and each hyperedge corresponds to an application (either to an individual application or to a joint application).

We tried to solve the above problem with the Scarf algorithm by using two different perturbations. First we set $\tilde{b}_i = b_i + \varepsilon_i = b_i + 1/p_{101-i}$, where p_i is the i-th prime number, and we obtained the following half-integral solution:

$x([a_3 \to h_3]) = 0$, $x([a_3 \to h_5]) = 1$, $x([a_7 \to h_6]) = \frac{1}{2}$, $x([a_7 \to h_8]) = \frac{1}{2}$, $x([(a_1, a_5) \to (h_1, h_2)]) = 1$, $x([(a_1, a_5) \to (h_3, h_6)]) = 0$, $x([(a_2, a_4) \to (h_4, h_5)]) = 0$, $x([(a_2, a_4) \to (h_1, h_2)]) = 0$, $x([(a_2, a_4) \to (h_3, h_7)]) = 1$, $x([(a_6, a_8) \to (h_6, h_8)]) = \frac{1}{2}$.

However, by setting $\tilde{b}_i = b_i + \varepsilon_i = b_i + 1/p_{101+i}$, where p_i is the i-th prime number[6], we obtained

$x([a_3 \to h_3]) = 0$, $x([a_3 \to h_5]) = 1$, $x([a_7 \to h_6]) = 0$, $x([a_7 \to h_8]) = 1$, $x([(a_1, a_5) \to (h_1, h_2)]) = 0$, $x([(a_1, a_5) \to (h_3, h_6)]) = 1$, $x([(a_2, a_4) \to (h_4, h_5)]) = 0$, $x([(a_2, a_4) \to (h_1, h_2)]) = 1$, $x([(a_2, a_4) \to (h_3, h_7)]) = 0$, $x([(a_6, a_8) \to (h_6, h_8)]) = 0$,

which corresponds to the unique stable matching for this instance, namely

$$M = \{\langle a_1, h_3 \rangle, \langle a_2, h_1 \rangle, \langle a_3, h_5 \rangle, \langle a_4, h_2 \rangle, \langle a_5, h_6 \rangle, \langle a_7, h_8 \rangle\}.$$

Therefore this example also illustrates that different perturbations may result in different stable allocations with the possibility that some of them are integral and some are fractional solutions.

3 Experimental comparison of heuristics

In the redesign of the NRMP, Roth and Peranson [18] constructed a heuristic method that incorporates the admission of couples. Motivated by the redesign of the Scottish resident allocation scheme, Biró et al. [4] implemented some more complex heuristics and compared them with some variants of the Roth-Peranson method. We also implemented the Scarf algorithm, and used it as a new heuristic to solve HRC, as described in the previous section. We conducted simulations on the same instances that Biró et al. [4] studied (see Table II. in their paper). But we shall note that the Scarf algorithm was implemented in a different platform and so we did not compare its running time with the other heuristics studied in [4]. For each run of the Scarf algorithm we used one perturbation only. (It is question for future research whether different perturbations can result in significantly different stable allocations for random or structured instances). The table below summarises our findings.

	Number of couples										
Algorithm	12	25	50	75	100	125	150	175	200	225	250
Roth-Peranson	952	897	701	547	395	277	170	83	41	9	3
Best heuristics in B-I-S	976	958	911	870	811	752	682	546	281	71	10
Scarf (int. solution)	895	813	649	532	426	356	316	261	202	174	**158**
Scarf half-int. solution	999	997	978	958	918	859	816	777	692	695	588
Scarf frac. solution	105	187	351	468	574	644	684	739	798	826	842
Av. # of frac. weights	3.9	4.8	5.7	6.7	7.6	8.8	10.0	10.8	12.8	13.5	15.7
# of frac. weights = 1	41	61	104	127	119	126	106	114	97	85	69
# of frac. weights = 2	2	9	21	30	36	41	43	43	44	48	41
# of frac. weights = 3	14	14	29	38	38	33	35	44	29	36	22
# of frac. weights = 4	7	18	19	25	40	37	39	38	30	32	41
# of frac. weights = 5	11	19	18	25	33	42	34	30	40	28	30

Table 1: Randomly created instances with couples for 500 residents.

[6] This choice of using the reciprocal of large prime numbers to perturb the capacity vectors was a handy heuristic, as the prime numbers are accessible in MATLAB, that we used for implementing the Scarf algorithm. The ratio behind using this perturbation to avoid degeneracy is that it is highly unlikely that a linear combination of these small added numbers would result in zero, especially if the original matrix contains only numbers 0, 1 and 2. Indeed we did not experience any degeneracy during the execution of our algorithm.

In this experiment there were 500 residents and the proportion of couples varied from 5% to 100% [7]. For each proportion of couples there were 1000 random instances generated and we counted how many instances each variant could solve. We remark that we do not know how many of these random instances were actually solvable. Note also that for the set of heuristics by Biró et al. [4] the results given in the table are not for a single heuristic, but they are always for the heuristic which performed best on the corresponding set of parameter values.

The table shows that the heuristics by Biró et al. [4] obtained a much better success ratio than the Roth-Peranson heuristics, especially for high proportions of couples. But surprisingly, the Scarf-algorithm was much better than the others when (almost) all the applicants form couples. As we have already noted, this situation occurs in some applications, such as the Hungarian higher education matching scheme (where many applicants apply for a pair of studies).

For completeness, we also included some statistics for those instances that the extended Scarf algorithm could not solve. In particular, we listed how many times the Scarf heuristic returned half-integer solutions (these may be interpreted as half-time contracts), and for how many instances the solution contained only one, two, three, four or five fractional weights. The latter properties of the fractional solutions indicate how far they were from being integral. In particular, having one single fractional weight in a stable allocation must imply that a combined application has half weight, meaning that these two residents would get two half-time jobs in a hospital. In this case, if we would omit the couple and reduce the capacity of that hospital by one then the remaining matching would be stable for the reduced instance.

4 Further notes

In this paper we showed how the original Scarf algorithm may be used as a successful heuristic for solving the Hospitals / Residents problem with Couples, which is a relevant problem in many practical applications. There are still a number of interesting questions that would be worth investigating as extensions to our work presented. How does the Scarf algorithm work for the particular case of HRC? Is there some more efficient way to run this algorithm for HRC than the original matrix method? What are the possible effects of the perturbations used in the Scarf algorithm? If the Scarf algorithm returns a fractional solution, can we approximate an integral solution from that? Very recently Nguyen and Vohra [15] showed how the existence of a stable matching can be guaranteed for adjusted quotas in a version of the HRC problem and its extensions. Their proof is based on a rounding technique which starts with a fractional stable solution and returns an integral stable solution for a new instance with adjusted quotas.

We also believe that the stable allocation problem with contributions may accommodate many further relevant problems, and so the Scarf algorithm and its extensions are worth considering as heuristics for solving other problems as well.

References

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

[2] R. Aharoni and T. Fleiner. On a lemma of Scarf. *J. Combin. Theory Ser. B*, 87(1):72–80, 2003. Dedicated to Crispin St. J. A. Nash-Williams.

[3] P. Biró and T. Fleiner. Fractional solutions for capacitated NTU-games, with applications to stable matchings. *To appear in Discrete Optimization*, 2015.

[7] The number of programmes was 50, the number of places were distributed randomly among the programmes, to be equal to the number of applicants, and the length of each applicants preference list were set to six. Further details on this experiment can be found in [4] at Table II.

[4] P. Biró and R.W. Irving and I. Schlotter. Stable matching with couples – an empirical study. *ACM Journal of Experimental Algorithmics*, 16, Article No.: 1.2, 2011.

[5] P. Biró and F. Klijn. Matching with Couples: A Multidisciplinary Survey. *International Game Theory Review*, 15(2), 1340008, 2013.

[6] P. Biró and S. Kiselgof. College admissions with stable score-limits. *To appear in Central European Journal of Operations Research*, 2015.

[7] D. Gale and L. S. Shapley. College Admissions and the Stability of Marriage. *Amer. Math. Monthly*, 69(1):9–15, 1962.

[8] R.W. Irving. "Matching medical students to pairs of hospitals: a new variation on an old theme. *in Proceedings of ESA'98*, LNCS, vol. 1461: 381–392, 1998.

[9] S. Kintali. Complexity of Scarf's Lemma and Related Problems. *Working paper*, 2009.

[10] B. Klaus and F. Klijn. Stable matchings and preferences of couples. *J. Econom. Theory*, 121(1):75–106, 2005.

[11] B. Klaus, F. Klijn, and J. Massó. Some things couples always wanted to know about stable matchings (but were afraid to ask). *Review of Economic Design*, 11:175–184, 2007.

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

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

[14] E. McDermid and D.F. Manlove. Keeping partners together: Algorithmic results for the hospitals / residents problem with couples. *Journal of Combinatorial Optimization*, 19:279–303, 2010.

[15] T. Nguyen and R. Vohra. Near Feasible Stable Matchings with Complementarities. *Working paper*, 2014.

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

[17] A. E. Roth. The evolution of the labor market for medical interns and residents: a case study in game theory. *Journal of Political Economy*, 6(4):991–1016, 1984.

[18] 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.

[19] A.E. Roth. Deferred acceptance algorithms: history, theory, practice, and open questions. *International Journal of Game Theory*, 36:537–569, 2008.

[20] H. E. Scarf. The core of an N person game. *Econometrica*, 35:50–69, 1967.

[21] J. Sethuraman, C-P. Teo, and L. Qian. Many-to-one stable matching: geometry and fairness. *Mathematics of Operations Research*, 31:581–596, 2006.

[22] P. Biró. University admission practices - Hungary. matching-in-practice.eu, accessed on 23 May 2012

[23] R. Irving. Matching practices for entry-labor markets - Scotland. matching-in-practice.eu, accessed on 23 May 2012

[24] `http://www.nrmp.org/about_nrmp/how.html` (National Resident Matching Program website), accessed on 23 May 2012.

[25] `http://www.nes.scot.nhs.uk/sfas/` (Scottish Foundation Allocation Scheme website), accessed on 23 May 2012.