Stable Assignment with Couples: Parameterized Complexity and Local Search

Dániel Marx and Ildikó Schlotter

Department of Computer Science and Information Theory,
Budapest University of Technology and Economics,
H-1521 Budapest, Hungary.
{dmarx,ildi}@cs.bme.hu

Abstract. We study the Hospitals/Residents with Couples problem, a variant of the classical Stable Marriage problem. This is the extension of the Hospitals/Residents problem where residents are allowed to form pairs and submit joint rankings over hospitals. We use the framework of parameterized complexity, considering the number of couples as a parameter. We also apply a local search approach, and examine the possibilities for giving FPT algorithms applicable in this context. Furthermore, we also investigate the matching problem containing couples that is the simplified version of the Hospitals/Residents problem modeling the case when no preferences are given.

1 Introduction

The classical Hospitals/Residents problem (which is a generalization of the well-known Stable Marriage problem) was introduced by Gale and Shapley [6] to model the following situation. We are given a set of hospitals, each having a number of open positions, and a set of residents applying for jobs in the hospitals. Each resident has a ranking over the hospitals, and conversely, each hospital has a ranking over the residents. Our aim is to assign as many residents to a hospital as possible, with the restrictions that the capacities of the hospitals are not exceeded and the resulting assignment is stable (no hospital-resident pair would benefit from rejecting the assignment and contracting each other).

The original version of the Hospitals/Residents problem is well understood: a stable assignment always exists, and every stable assignment has the same size. (The size of an assignment is the number of residents that have a job.) Moreover, the classical Gale-Shapley algorithm [6] can find a stable assignment in linear time. However, several practical applications motivate some kind of extension or modification of the problem (see e.g. the NRMP program for assigning medical residents in the USA [17, 18]), and in the recent decade various versions have been investigated. We study an extension of this problem, called Hospitals/Residents with Couples (or HRC), where residents may form couples, and thus have joint rankings over the hospitals. This extension models a situation that arises in many real world applications [18], and was introduced by Roth [17] who also discovered that a stable assignment need not exist when couples are involved.

Later, Ronn [16] proved that it is NP-hard to decide whether a stable assignment exists in such a setting. There have been investigations of different assumptions on the preferences of couples that guarantee some kind of tractability [10, 14].

Algorithmic approaches. In the Hospitals/Residents problem, practical scenarios usually involve much fewer couples than singles, e.g. the ratio of couples to singles participating in the NRMP program is around 2.5 percent[1]. Thus, the number of couples in the HRC problem is a natural parameter. Investigating the parameterized complexity of HRC with this parameter is our first goal.

Local search is a basic technique that has been widely applied in heuristics for practical optimization problems for several decades [1]. However, investigations considering the connection of local search and parameterized algorithms have only been started a few years ago, and research in this area has been gaining increasing attention lately [12]. The basic idea of local search is to find an optimal solution by an iteration in which we improve the current solution step by step through local modifications. Local search can become more efficient if we can decide whether there exists a better solution S' that is ℓ modification steps away from a given solution S. Typically, the ℓ-neighborhood of a solution S can be explored in $n^{O(\ell)}$ time by examining all possibilities to find those parts of S that should be modified. (Here n is the input size.) The question whether an FPT algorithm with parameter ℓ can be found for the neighborhood exploration problem has already been studied in connection with different optimization problems ([9, 13, 4]). Our second goal is to investigate this approach for assignment problems.

We also contribute to the framework of parameterized local search algorithms by distinguishing between "strict" algorithms that perform the local search step in some neighborhood of a solution as described above, and "permissive" algorithms whose task is the following: given some problem with an initial solution S, find *any* better solution, provided that a better solution exists in the local neighborhood of S. Our motivation for this distinction is that finding an improved solution in the neighborhood of a given solution may be hard, even for problems where an optimal solution is easily found.

Most of the questions examined here are also worth studying in a simplified model that does not involve preferences. In the Maximum Matching with Couples problem, or shortly MMC, no stability requirement is given, and we aim for an assignment of maximum size.

Results. For lack of space, we stated some of our results without proof, see Appendix for these proofs. Our main results are outlined below (see Table 1). We denote by C the set of couples in a problem instance, and we denote by ℓ the neighborhood size in a given local search problem.

- Theorem 1 gives a randomized FPT algorithm with parameter $|C|$ for Maximum Matchig with Couples. The presented algorithm uses an FPT result from matroid theory.
- Theorem 3 shows that no permissive local search FPT algorithm exists for MMC, where the parameter is ℓ, unless W[1] = FPT.

[1] http://www.nrmp.org/data/resultsanddata2008.pdf

Task:	Existence problem	Maximum problem	Local search algorithm with FPT running time	
Parameter:		$\vert C\vert$	ℓ	$(\vert C\vert, \ell)$
MMC (no pref's)	P (trivial)	randomized FPT (Theorem 1)	No permissive alg. (Theorem 3)	Permissive alg. (Theorem 1)
HRC (with pref's)	W[1]-hard (Theorem 4)	W[1]-hard (Theorem 4)	No permissive alg. (Theorem 5)	Strict alg. (Theorem 7)

Table 1. Summary of our results (assuming W[1] $\neq$ FPT).

– Theorem 4 proves that the existence version of the HRC problem is W[1]-hard with parameter $|C|$.
– Theorem 5 shows that no permissive local search FPT algorithm exists for the maximization version of HRC with parameter ℓ, unless W[1] = FPT.
– Theorem 7 presents a strict local search FPT algorithm for the maximization version of HRC, with combined parameters $|C|$ and ℓ. The algorithm uses color coding and a set of non-trivial reduction rules.

2 Preliminaries

For some integer k, we use $[k] = \{1, 2, \ldots, k\}$, and $\binom{[k]}{2} = \{(i, j) | 1 \leq i < j \leq k\}$. If a matching M in a graph contains an edge xy, then we write $M(x) = y$ and $M(y) = x$. For other graph theoretic concepts, we use standard notation. We assume basic knowledge of matroid theory in Sect. 4. We also assume that the reader is familiar with the framework of parameterized complexity. For an introduction, see [15] or [5].

To formalize the task of a local search algorithm, let Q be an optimization problem with an objective function T which we want to maximize. To define the concept of neighborhoods, we suppose there is some *distance* $d(x, y)$ defined for each pair (x, y) of solutions for some instance I of Q. A *strict local search* algorithm for Q has the following task:

Strict local search for Q
Input: (I, S_0, ℓ) where I is an instance of Q, S_0 is a solution for I, and $\ell \in \mathbb{N}$.
Task: If there exists a solution S for I such that $d(S, S_0) \leq \ell$ and $T(S) > T(S_0)$, then output such an S.

In contrast, a *permissive local search* algorithm for Q is allowed to output a solution that is not close to S_0, provided that it is better than S_0. In local search methods, such an algorithm is as useful as its strict version.

Permissive local search for Q
Input: (I, S_0, ℓ) where I is an instance of Q, S_0 is a solution for I, and $\ell \in \mathbb{N}$.
Task: If there exists a solution S for I such that $d(S, S_0) \leq \ell$ and $T(S) > T(S_0)$, then output *any* solution S' for I with $T(S') > T(S)$.

Note that if an optimal solution can be found by some algorithm, then this yields a permissive local search algorithm for the given problem. On the other hand, finding a strict local search algorithm might be hard even if an optimal solution is easily found. An example for such a case is the Minimum Vertex Cover problem for bipartite graphs [9]. Besides, proving that no permissive local search algorithm exists for some problem is clearly harder than it is for strict local search algorithms. We also present results of this kind.

3 Hospitals/Resident with Couples

A *couples' market with preference*, or shortly *cmp*, consists of the sets S, C and H representing singles, couples and hospitals, respectively, a *capacity* $f(h)$ for each $h \in H$, and a *preference list* $L(a)$ for each $a \in S \cup C \cup H$. The set $A = S \cup C \cup H$ is called the set of *agents*. Each couple c is a pair $(c(1), c(2))$, and we call the elements of the set $R = \bigcup_{c \in C} \{c(1), c(2)\} \cup S$ *residents*. For a hospital h, $L(h)$ is a list of residents, for a single s, $L(s)$ is a list of hospitals, and for a couple c, $L(c)$ is a list containing pairs of hospitals, or more precisely, a list containing elements from $(H \cup \{u\}) \times (H \cup \{u\}) \setminus \{(u, u)\}$ where u is a special symbol indicating that someone is unemployed. The preference lists can be incomplete, but cannot involve ties, so these lists are strictly ordered.

The set of elements appearing in the list $L(a)$ is $A^L(a)$, and some x is considered *acceptable* for a if $x \in A^L(a)$. Clearly, we may assume that acceptance is mutual, so $h \in A^L(s)$ holds if and only if $s \in A^L(h)$ for each $s \in S$ and $h \in H$, and $(h_1, h_2) \in A^L(c)$ implies $c(i) \in A^L(h_i)$ or $h_i = u$ for both $i \in \{1, 2\}$, for each $c \in C$. For some $x \in A^L(a)$, the *rank* of x w.r.t. a, denoted by $\rho(a, x)$, is $r \in \mathbb{N}$ if x is the r-th element in $L(a)$. If $x \notin A^L(a)$, then we let $\rho(a, x) = \infty$ for all meaningful x. We say that the cmp is f_0-*uniform* if $f \equiv f_0$ for some $f_0 \in \mathbb{N}$.

An *assignment* is a function $M : R \to H \cup \{u\}$ such that $M(s) \in A^L(s) \cup \{u\}$ for each $s \in S$, $M(c) \in A^L(c) \cup \{(u, u)\}$ for each $c \in C$, and $|M(h)| \leq f(h)$ holds for each hospital h. Here, $M(c)$ denotes the pair $(M(c(1)), M(c(2)))$, and $M(h) = \{r | r \in R, M(r) = h\}$ is the set of residents assigned to h in M. We say that an assignment M *covers* a resident r if $M(r) \neq u$, and M covers a couple c, if it covers $c(1)$ or $c(2)$. We define the *size* of M, denoted by $|M|$, to be the number of residents covered by M. The *distance* $d(M, M')$ of two assignments M and M' is the number of residents r for which $M(r) \neq M'(r)$.

We say that x is *beneficial* for the agent a with respect to an assignment M if $x \in A^L(a)$ and one of the following cases holds: (1) $a \in S \cup C$ and either a is not covered by M or $\rho(a, x) < \rho(a, M(a))$, (2) $a \in H$ and either $|M(a)| < f(a)$ or there exists a resident $r' \in M(a)$ such that $\rho(a, x) < \rho(a, r')$. A *blocking pair* for M can be of three types:

– it is either a pair formed by a single s and a hospital h such that both s and h are beneficial for each other w.r.t. M,
– or a pair formed by a couple c and a pair (h_1, h_2) with $h_1 \neq h_2$ such that (h_1, h_2) is beneficial for c w.r.t. M, and for both $i \in \{1, 2\}$ it holds that if $h_i \neq u$ then either $c(i)$ is beneficial for h_i w.r.t. M or $c(i) \in M(h_i)$,
– or a pair formed by a couple c and a hospital h such that (h, h) is beneficial for c w.r.t. M, and *the couple c is beneficial for h*. If h prefers $c(1)$ to $c(2)$, this latter means that either $|M(h)| \leq f(h) - 2$, or $|M(h)| \leq f(h) - 1$ and $\rho(h, c(1)) < \rho(h, r)$ for some $r \in M(h)$, or $\rho(h, c(1)) < \rho(h, r_1)$ and $\rho(h, c(2)) < \rho(h, r_2)$ for some $r_1 \neq r_2$ in $M(h)$. [2]

An assignment M for I is *stable* if there is no blocking pair for M.

The input of the Hospitals/Residents with Couples problem is a cmp I, and the task is to determine a stable assignment for I, if such an assignment exists. If no couples are involved, then a stable assignment can always be found in linear time with the Gale-Shapley algorithm [6]. In the case when couples are present, a stable assignment may not exist, as first proved by Roth [17]. Ronn proved that deciding whether a stable assignment exists for a cmp is NP-complete [16]. Moreover, an instance of the Hospitals/Residents with Couples problem may admit stable assignments of different sizes, see the Appendix for an example. In the optimization problem Maximum Hospitals/Residents with Couples, the task is to determine a stable assignment of maximum size for a given cmp. This problem is trivially NP-hard, as it contains the Hospitals/Residents with Couples problem. We study these problems in Sect. 5.

We also study a version of the Hospitals/Residents with Couples problem that does not contain preferences and only deals with the notion of acceptability. To describe the input of this problem, we define a *couples' market with acceptance*, or shortly *cma*, as a quintuple (S, C, H, f, A) where S, C, H and f are defined analogously as in a cmp, but $A(a)$ defines only the set of acceptable elements for an agent a, without any ordering. Each concept described above that does not rely on the preference lists (and thus on stability) is inherited also for cmas in the straightforward way. In Sect. 4, we investigate the optimization problem Maximum Matching with Couples, where given a cma I, the task is to find an assignment for I of maximum size.

4 Matching without preferences

First, we investigate a slightly modified version of Maximum Matching with Couples, denoted as (k, n)-Matching with Couples: given a cma I and two integers k and n, find an assignment for I that covers at least k couples and n singles, if possible. Such an assignment is called a *(k, n)-assignment*. Clearly, if there are no couples in a given instance, then the problem is equivalent to finding a maximum matching in a bipartite graph, and can be solved by standard techniques. If couples are involved, the problem becomes hard. More precisely,

[2] We thank David Manlove for pointing out this case.

the decision version of this problem is NP-complete [8, 3], even in the following special case: each hospital has capacity 2, and the acceptable hospital pairs for a couple are always of the form (h, h) for some $h \in H$. However, if the number of couples is small, which is a reasonable assumption in many practical applications, (k, n)-Matching with Couples becomes tractable, as shown by Theorem 1.

Theorem 1. (k, n)-Matching with Couples *can be solved in randomized FPT time with parameter* $|C|$.

To prove Theorem 1, we need a variant of a result from [11] concerning matroids.

Theorem 2. *Let* $\mathcal{M}(U, \mathcal{I})$ *be a linear matroid and let* $\mathcal{X} = \{X_1, X_2, \ldots X_n\}$ *be a collection of subsets of* U*, each of size* b*. Given a linear representation* A *of* $\mathcal{M}$*, it can be determined in* $f(k, b) \cdot ||A||^{O(1)}$ *randomized time whether there is an independent set that is the union of* k *disjoint sets in* $\mathcal{X}$*.*

Proof (of Theorem 1). Let (S, C, H, f, A) be the cma for which we have to find a (k, n)-assignment. W.l.o.g. we can assume that each hospital has capacity 1 as otherwise we can "clone" the hospitals, i.e. for each $h \in H$ we can substitute h with the newly introduced hospitals $h^1, \ldots, h^{f(h)}$, also modifying $A(p)$ for each $p \in S \cup C$ appropriately. (As $f(h) \leq |S| + 2|C|$ can be assumed, this increases the input size only polynomially.) Note that the case $k < |C|$ can be solved by finding a (k, n)-assignment for (S, C', H, f, A') for every $C' \subseteq C$ where $|C'| = k$ and A' is the restriction of A on $S \cup C'$. As this increases the running time only with a factor of at most $2^{|C|}$, it is sufficient to give an FPT algorithm for the case $|C| = k$. Moreover, we can assume $A(c) \subseteq H \times H$, since for each $c \in C$ we can eliminate each pair of the form (h, u) or (u, h) $(h \in H)$ in $A(c)$ by adding a new hospital u_c to H with capacity 1 and substituting u with u_c.

Now, let $G(H, S; E)$ be the bipartite graph where a single $s \in S$ is connected with a hospital $h \in H$ if and only if $h \in A(s)$. We can assume w.l.o.g. that G has a matching of size at least n as otherwise no solution may exist, and this case can be detected easily in polynomial time. We define $\mathcal{M}(H, \mathcal{I})$ to be the matroid where a set $X \subseteq H$ is independent if and only if there is a matching in G that covers at least n singles but covers no hospitals from X. Observe that $\mathcal{M}$ is exactly the dual of the n-truncation of the transversal matroid of G, and thus it is indeed a matroid. By a lemma in [11], we can find a linear representation A of $\mathcal{M}$ in randomized polynomial time.

We define the matroid $\mathcal{M}'(U, \mathcal{I}')$ with ground set $U = H \cup C$ such that $X \subseteq U$ is independent in $\mathcal{M}'$ if $X \cap H$ is independent in $\mathcal{M}$. A representation of $\mathcal{M}'$ can be obtained by taking the direct sum of the matrices A and E_k where E_k is the unit matrix of size $k \times k$. Let $\mathcal{X}$ be the collection of the sets that are of the form $\{c, h_1, h_2\}$ where $c \in C$ and $(h_1, h_2) \in A(c)$.

Observe that if $X_1, X_2, \ldots, X_k$ are k disjoint sets in $\mathcal{X}$ whose union is independent in $\mathcal{M}'$, then we can construct a (k, n)-assignment as follows. For each $\{c, h_1, h_2\} \in \{X_1, \ldots, X_k\}$ we choose $M(c)$ from $\{(h_1, h_2), (h_2, h_1)\} \cap A(c)$ arbitrarily. The disjointness of the sets $X_1, \ldots, X_k$ guarantees that this way we assign exactly one resident to each hospital in $X = \bigcup_{i\in[k]} X_i \cap H$. Now, let N be a matching in G that covers at least n singles, but no hospitals from X. Such a matching exists, as X is independent in $\mathcal{M}$. Thus letting $M(s)$ to be $N(s)$ if s is covered by N and u otherwise for each $s \in S$ yields that M is a (k,n)-assignment. Conversely, if M is a (k,n)-assignment then the sets $\{c, h_1, h_2\}$ for each $c \in C$ and $M(c) = (h_1, h_2)$ form a collection of k disjoint sets in $\mathcal{X}$ whose union is independent in $\mathcal{M}'$. By Theorem 2, such a collection can be found in randomized FPT time when k is the parameter, yielding a solution if exists. □

We remark that Theorem 1 also applies to the following cases.

- Markets containing groups of fixed size instead of couples.
- Maximization (or minimization) of an arbitrary function $f(k,n)$, where k and n are the number of covered couples and singles, respectively.
- Minimizing the makespan in the scheduling problem containing parallel machines and independent jobs with job assignment restrictions, if the processing time is $p \in \mathbb{N}$ for k jobs, and 1 for the others, and k is the parameter.

Considering the parameterized complexity of the local search approach for the MMC problem with parameter ℓ denoting the neighborhood size, Theorem 3 shows that no FPT local search algorithm is likely to exists. We omit the proof.

Theorem 3. *No permissive local search algorithm for 2-uniform* Maximum Matching with Couples *runs in FPT time with parameter ℓ, if W[1] $\neq$ FPT.*

5 Matching with preferences

In this section, we investigate several versions of the Hospitals/Residents problem, where couples are involved and preferences play an important role.

After presenting some hardness results, Theorem 7 gives an FPT time strict local search algorithm for the Maximum Hospitals/Residents with Couples problem, where $|C|$ and ℓ are parameters. In contrast, Theorem 4 shows the W[1]-hardness of the Hospitals/Residents with Couples problem with parameter $|C|$, which clearly implies that Maximum Hospitals/Residents with Couples is also W[1]-hard with parameter $|C|$.

However, supposing that a stable assignment has already been determined by some method, it is a valid question whether we can increase its size. We will denote this problem Increase Hospitals/Residents with Couples. Formally, its input is a cmp I and a stable assignment M_0 for I, and the task is to find a stable assignment with size at least $|M_0|+1$. If no couples are involved, then all stable assignments for the instance have the same size, so this problem is trivially polynomial-time solvable. Theorem 4 shows that Increase Hospitals/Residents with Couples is also W[1]-hard with parameter $|C|$.

Theorem 4. *(1) The decision version of* Hospitals/Residents with Couples *is W[1]-hard with parameter $|C|$, even in the 1-uniform case.*
(2) The decision version of Increase Hospitals/Residents with Couples *is W[1]-hard with parameter $|C|$, even in the 1-uniform case.*

Considering the applicability of the local search approach for the Maximum Hospitals/Residents with Couples problem, Theorem 5 shows that no permissive local search algorithm is likely to run in FPT time with parameter ℓ. However, if we regard $|C|$ as a parameter as well, then even a strict local search algorithm with FPT running time can be given, as presented in Theorem 7.

Theorem 5. *No permissive local search algorithm for the 1-uniform* Maximum Hospitals/Residents with Couples *runs in FPT time with parameter ℓ, if W[1] $\neq$ FPT.*

To prove Theorems 4 and 5, we give FPT-reductions from the parameterized Clique problem, both reductions relying on the same idea. Although we omit the proofs, we describe the key structure used, whose main properties are stated in Lemma 6. For a graph G and some $k \in \mathbb{N}$, we introduce a cmp $I^{G,k} = (S, C, H, f, L)$ as follows (see Fig. 1).

Let $V(G) = \{v_i | i \in [n]\}$, $|E(G)| = m$ and let ν be a bijection from $[m]$ into the set $\{(x, y) | v_x v_y \in E(G), x < y\}$. First, we construct a *node-gadget* $\mathcal{G}^i$ for each $i \in [k]$ and an *edge-gadget* $\mathcal{G}^{i,j}$ for each pair $(i, j) \in \binom{[k]}{2}$. The node-gadget $\mathcal{G}^i$ contains hospitals $H^i \cup G^i \cup \{f^i\}$, singles $S^i \cup T^i$ and a couple a^i. Analogously, the edge-gadget $\mathcal{G}^{i,j}$ contains hospitals $H^{i,j} \cup G^{i,j} \cup \{f^{i,j}\}$, singles $S^{i,j} \cup T^{i,j}$ and a couple $a^{i,j}$. Here $T^i = \{t^i_j | j \in [n-1]\}$ and $T^{i,j} = \{t^{i,j}_e | e \in [m-1]\}$, $H^i = \{h^i_j | j \in [n]\}$ and $H^{i,j} = \{h^{i,j}_e | e \in [m]\}$, and we define G^i, S^i and $G^{i,j}, S^{i,j}$ similarly to H^i and $H^{i,j}$. Observe that $|C| = k + \binom{k}{2}$.

We let $f \equiv 1$, so $I^{G,k}$ is 1-uniform. The precedence lists for each agent in $I^{G,k}$ are defined below. The notation $[X]$ for some set X in a preference list denotes an arbitrary ordering of the elements of X. We write Q^i_x for the set $\{s^{i,j}_e | i < j \leq k, \exists y : \nu(e) = (x, y)\} \cup \{s^{j,i}_e | 1 \leq j < i, \exists y : \nu(e) = (y, x)\}$ and $Q^{i,j}_e$ for $\{h^i_x, h^j_y\}$ where $\nu(e) = (x, y)$. The indices in the precedence lists take all possible values if not stated otherwise, and the symbol α can be any index in $[k]$ or a pair of indices in $\binom{[k]}{2}$. If α takes a value in $[k]$ then $N(\alpha) = n$, otherwise $N(\alpha) = m$.

$L(g^\alpha_x) : t^\alpha_{x-1}, a^\alpha(2), t^\alpha_x \quad \text{if } 1 < x < N(\alpha)$
$L(g^\alpha_1) : a^\alpha(2), t^\alpha_1$
$L(g^\alpha_{N(\alpha)}) : t^\alpha_{N(\alpha)-1}, a^\alpha(2), a^\alpha(1)$
$L(t^\alpha_x) : g^\alpha_x, g^\alpha_{x+1}$
$L(f^\alpha) : s^\alpha_1, s^\alpha_2, \ldots, s^\alpha_{N(\alpha)}, a^\alpha(2)$
$L(a^\alpha) : (g^\alpha_{N(\alpha)}, f^\alpha), (h^\alpha_1, g^\alpha_{N(\alpha)}), (h^\alpha_2, g^\alpha_{N(\alpha)-1}), \ldots, (h^\alpha_{N(\alpha)}, g^\alpha_1)$

$L(h^i_x) : a^i(1), [Q^i_x], s^i_x$
$L(h^{i,j}_e) : a^{i,j}(1), s^{i,j}_e$
$L(s^i_x) : h^i_x, f^i$
$L(s^{i,j}_e) : h^{i,j}_e, [Q^{i,j}_e], f^{i,j}$

Lemma 6. *For a graph G and $k \in \mathbb{N}$, $I^{G,k}$ has a stable assignment $M^{G,k}_0$ that covers each resident. Moreover, statements (1), (2), and (3) are equivalent:*

(1) There is a clique in G of size k.
(2) There is a stable assignment M for $I^{G,k}$ with the following property, which we will call property π: $M(f^{i,j}) \subseteq S^{i,j}$ for each $(i, j) \in \binom{[k]}{2}$.
(3) There is a stable assignment for $I^{G,k}$ with property π covering each resident.

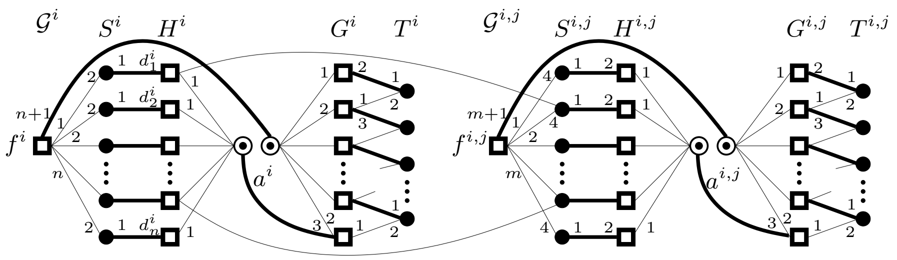

Fig. 1. A node- and an edge-gadget of $I^{G,k}$. Hospitals, singles, and couples are represented by rectangles, black, and double circles, resp. We connect $h \in H$ and $r \in R$ if $r \in A^L(h)$. Numbers show ranks, bold edges represent $M_0^{G,k}$, and $d_x^i = |Q_x^i| + 2$.

Proof. To see the first claim, we define an assignment M_0 by letting $M_0(a^\alpha) = (g^\alpha_{N(\alpha)}, f^\alpha)$, $M_0(t_x^\alpha) = g_x^\alpha$, and $M_0(s_x^\alpha) = h_x^\alpha$ for all possible α and x. As each resident is assigned to his best choice, M_0 is stable and covers each resident.

To prove (2) $\Rightarrow$ (1), suppose that $I^{G,k}$ has a stable assignment M with property π. Let us define $\sigma(i,j)$ for each $(i,j) \in \binom{[k]}{2}$ such that $M(f^{i,j}) = \{s^{i,j}_{\sigma(i,j)}\}$. Since $s^{i,j}_{\sigma(i,j)}$ prefers $h^{i,j}_{\sigma(i,j)}$ to $f^{i,j}$, the stability of M implies $M(h^{i,j}_{\sigma(i,j)}) = \{a^{i,j}(1)\}$. From this, we get that $M(s_e^{i,j}) = h_e^{i,j}$ must hold for each $e \in [m] \setminus \{\sigma(i,j)\}$ as otherwise $(s_e^{i,j}, h_e^{i,j})$ would be a blocking pair. Note that each single in $S^{i,j}$ is assigned to a hospital in $H^{i,j} \cup \{f^{i,j}\}$. As this holds for each $(i,j) \in \binom{[k]}{2}$, we get that $M(h_x^i) \subseteq S^i \cup \{a^i(1)\}$ holds for each $i \in [k], x \in [n]$.

Let $\nu(\sigma(i,j)) = (x,y)$ for some $(i,j) \in \binom{[k]}{2}$. Since $s^{i,j}_{\sigma(i,j)}$ prefers the hospitals in $Q^{i,j}_{\sigma(i,j)} = \{h_x^i, h_y^j\}$ to $f^{i,j}$, M can only be stable if both h_x^i and h_y^j prefer their partner in M to $s^{i,j}_{\sigma(i,j)}$. This implies $M(h_x^i) = \{a^i(1)\}$ and $M(h_y^j) = \{a^j(1)\}$. Thus, by defining $\sigma(i)$ to be x if $M(a^i) = (h_x^i, g^i_{n+1-x})$ for each $i \in [k]$, we obtain $\nu(\sigma(i,j)) = (\sigma(i), \sigma(j))$. From the definition of ν, this implies that $v_{\sigma(i)}$ and $v_{\sigma(j)}$ are adjacent in G. As this holds for every $(i,j) \in \binom{[k]}{2}$, we get that $\{v_{\sigma(i)} | i \in [k]\}$ is a clique in G.

Now we prove (1) $\Rightarrow$ (3). If $v_{\sigma(1)}, v_{\sigma(2)}, \ldots, v_{\sigma(k)}$ form a clique in G, then define $\sigma(i,j)$ such that $\sigma(i,j) = \nu^{-1}(\sigma(i), \sigma(j))$. We define a stable assignment M fulfilling property π and covering every resident as follows.

$$\begin{array}{ll}
M(a^\alpha) = (h^\alpha_{\sigma(\alpha)}, g^i_{N(\alpha)+1-\sigma(\alpha)}) & M(t_x^\alpha) = g_x^\alpha \quad \text{if } x \in [N(\alpha) - \sigma(\alpha)] \\
M(s^\alpha_{\sigma(\alpha)}) = f^\alpha & M(t_x^\alpha) = g^\alpha_{x+1} \quad \text{otherwise} \\
M(s_x^\alpha) = h_x^\alpha \quad \text{if } x \in [N(\alpha)] \setminus \{\sigma(\alpha)\} &
\end{array}$$

The stability of M can be verified by simply checking all possibilities to find a blocking pair. (We note that many agents are only contained in $I^{G,k}$ to assure that M is indeed stable.) As (3) $\Rightarrow$ (2) is trivial, this finishes the proof. $\square$

Theorem 7. *There is an FPT time strict local search algorithm for* Maximum Hospitals/Residents with Couples *with combined parameter* $(\ell, |C|)$.

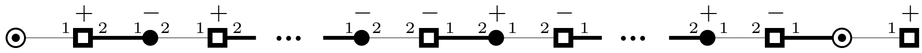

Fig. 2. A possible component of G^δ. Winners and losers are marked by '+' and '−' signs, respectively. Bold edges represent M_0, normal edges represent M.

Proof. Let $I = (S, C, H, f, L)$ be given with the stable assignment M_0 and the integer ℓ. Although the case $f \equiv 1$ is different from the general case in many aspects, the trick of cloning the hospitals is applicable in our case (see the Appendix). Therefore, w.l.o.g. we may assume $f \equiv 1$. Thus, if $M(r) = h$ for some $r \in R$, then we will write $M(h) = r$ instead of $M(h) = \{r\}$.

Before describing the strict local search algorithm for Maximum Hospitals/Residents with Couples, we introduce some notation to capture the structure of the solution. The bipartite graph G underlying I has vertex set $H \cup R$ and edge set $E = \{hr | h \in H, r \in A^L(h)\}$. Clearly, an assignment M for I determines a matching $E(M)$ in G in the natural way: $hr \in E(M)$ if and only if $M(r) = h$ for some resident r and hospital h. Suppose that M is a closest solution, i.e. a stable assignment for I with $|M| > |M_0|$ and $d(M, M_0) \leq \ell$ that is the closest to M_0 among all such assignments. Let $A^\delta = \{a \in R \cup H | M(a) \neq M_0(a)\}$, and E^δ be the symmetric difference of $E(M_0)$ and $E(M)$. Note that E^δ covers exactly the vertices of A^δ, and $G^\delta = (A^\delta, E^\delta)$ is the union of paths and cycles which contain edges from M_0 and M in an alternating manner. It is well-known that for a cmp not containing couples every stable assignment covers exactly the same agents [7]. Thus, it is easy to see that the stability of M and M_0 imply that if a component of G^δ contains only single residents, then it must be a cycle. Let $\mathcal{K}_0$ denote the set of such cycles, and $\mathcal{K}_1$ the set of the remaining components of G^δ. We write C^δ for $(R \setminus S) \cap A^\delta$, and we define $B(a) = \{b | a$ is beneficial for b w.r.t. $M_0\}$ for every $a \in S \cup H$. We also let $S^+ = \{s \in S | M(s)$ is beneficial for s w.r.t. $M_0\}$, and $S^- = \{s \in S | M_0(s)$ is beneficial for s w.r.t. $M\}$. Note that $S^+ \cup S^- = S \cap A^\delta$. We define H^+ and H^- analogously. We call agents in $A^+ = S^+ \cup H^+$ *winners* and agents in $A^- = S^- \cup H^-$ *losers*. For a simple illustration see Fig. 2.

Now, we describe an algorithm that finds vertices of A^δ. The algorithm first branches on guessing $|A^\delta|$ and a copy $\bar{G}$ of the graph G^δ. Let φ denote an isomorphism from $\bar{G}$ to G^δ. The algorithm also guesses the vertex sets $\varphi^{-1}(C^\delta), \varphi^{-1}(H^+), \varphi^{-1}(H^-), \varphi^{-1}(S^+)$, $\varphi^{-1}(S^-)$, and edge sets $\bar{E}_{M_0}$ and $\bar{E}_M$ denoting $\varphi^{-1}(E(M_0) \cap E^\delta)$ and $\varphi^{-1}(E(M) \cap E^\delta)$, respectively. Since $|A^\delta| \leq 2\ell$, it can be achieved by careful implementation that the algorithm branches into at most $(2\ell) \cdot 6^{2\ell}$ directions in this phase. Now, let Γ be an ordering of $V(\bar{G})$, i.e. a bijection from $V(\bar{G})$ to $[|A^\delta|]$. The algorithm colors the vertices of G with $|A^\delta| \leq 2\ell$ colors randomly with uniform and independent distribution, $\gamma(a)$ denotes the color of a. The coloring γ is *nice*, if $\gamma(\varphi(a)) = \Gamma(a)$ for each $a \in V(\bar{G})$. We suppose that γ is nice, which clearly holds with probability $|A^\delta|^{-|A^\delta|} \geq (2\ell)^{-2\ell}$.

Given a coloring, the algorithm grows a subset $X \subseteq V(\bar{G})$ on which φ is already known. It applies the following extension rules repeatedly (see Fig. 3),

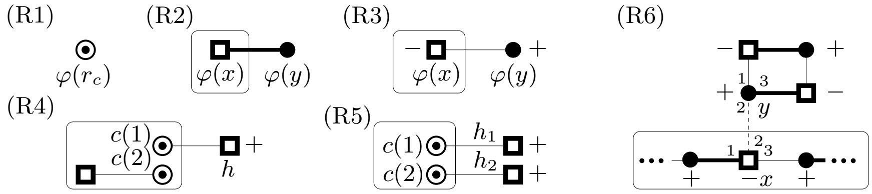

Fig. 3. Subgraphs of G^δ illustrating the rules of Theorem 7. Agents of $\varphi(X)$ are shown in a rectangular box. Bold edges represent $\bar{E}_{M_0}$, normal edges represent $\bar{E}_M$.

until none of them applies. When Rule 1 is applied, the algorithm branches into at most $2|C|$ branches, but no other branchings happen. We write $\bar{X} = V(\bar{G}) \setminus X$.

Rule 1 [guessing a member of a couple]: applicable if $r_c \in \bar{X} \cap \varphi^{-1}(C^\delta)$. In this case we simply branch on the vertices of $(R \setminus S) \cap \{a | \gamma(a) = \Gamma(c)\}$ to choose $\varphi(r_c)$. Note that this means at most $2|C|$ branches.

Rule 2 [finding pairs by M_0]: applicable if $x \in X, y \in \bar{X}$ and $xy \in \bar{E}_{M_0}$ for some x and y. By $\varphi(y) = M_0(\varphi(x))$, we can extend φ by adding y to X.

Rule 3 [finding pairs by M for losers]: applicable if $x \in X \cap \varphi^{-1}(A^-)$, $y \in \bar{X} \cap \varphi^{-1}(A^+)$ and $xy \in \bar{E}_M$ for some x and y. Let y^* be the first element in the list $L(\varphi(x))$ contained in $B(\varphi(x))$ having color $\Gamma(y)$. We claim $y^* = \varphi(y)$. Clearly, $\varphi(y) \in B(\varphi(x))$ holds because $\varphi(y)$ is a winner, and its color must be $\Gamma(y)$ as γ is nice. Now, suppose for contradiction that y^* precedes $\varphi(y)$ in $L(\varphi(x))$. Since the only vertex in A^δ having color $\Gamma(y)$ is $\varphi(y)$, we get $M(y^*) = M_0(y^*)$ implying that y^* and $\varphi(x)$ form a blocking pair for M. Thus, $\varphi(y) = y^*$ can be found in linear time, so we can extend φ by adding y to X.

Rule 4 [finding pairs by M for couples with one winner hospital]: applicable if $c(i) \in C^\delta \cap \varphi(X)$, $y \in \varphi^{-1}(H^+) \cap \bar{X}$, $\varphi^{-1}(c(i))y \in \bar{E}_M$, and $M(c(i'))$ is already known for some $c \in C$, $i \neq i'$ and y. W.l.o.g. we assume $i = 1$. Let h be defined such that $(h, M(c(2)))$ is the first element in $L(c)$ for which $h \in B(c(1))$ and h has color $\Gamma(y)$. We claim $\varphi(y) = h$. Observe that $\varphi(y) \in B(c(1))$ must hold because $\varphi(y)$ is a winner. As γ is nice, $\varphi(y)$ indeed has color $\Gamma(y)$. Thus, if $h \neq \varphi(y)$ then $(h, M(c(2)))$ precedes $(\varphi(y), M(c(2)))$ in $L(c)$, but this implies that the couple c and $(h, M(c(2)))$ form a blocking pair for M. Therefore, we get $\varphi(y) = h$, and we can extend φ in linear time by adding y to X.

Rule 5 [finding pairs by M for couples with two winner hospitals]: applicable if $c(i) \in C^\delta \cap \varphi(X)$, $y_i \in \varphi^{-1}(H^+) \cap \bar{X}$, and $\varphi^{-1}(c(i))y_i \in \bar{E}_M$ holds for both $i \in \{1, 2\}$, for some $c \in C$, y_1 and y_2. We let (h_1, h_2) be the first element in $L(c)$ such that $h_i \in B(c(i))$ and $\gamma(h_i) = \Gamma(y_i)$ for both $i \in \{1, 2\}$. Using the same arguments as in the previous case, we can show $\varphi(y_1) = h_1$ and $\varphi(y_2) = h_2$. Thus, we can extend φ in linear time by adding both y_1 and y_2 to X.

Rule 6 [dissolving a blocking pair]: applicable if $M(a) \in \varphi(X)$ if and only if $a \in \varphi(X)$ for all $a \in A^\delta$, and xy is a blocking pair for the *actual assignment* M_X. We define M_X by setting $M_X(a) = M_0(a)$ if $a \notin \varphi(X)$ and $M_X(a) = M(a)$ if $a \in \varphi(X)$, for each agent a. Note that by our first condition, M_X is indeed an assignment. Now, as xy cannot be a blocking pair for M or M_0, either $x \in \varphi(X)$ and $y \in A^\delta \setminus \varphi(X)$, or vice versa. W.l.o.g. we suppose the former. By defining $\bar{y} \in V(\bar{G})$ such that $\Gamma(\bar{y}) = \gamma(y)$, it can be seen that $\varphi(\bar{y}) = y$ must hold because γ is nice. Thus, φ can be extended by adding $\bar{y}$ to X.

Lemma 8. *If none of the rules is applicable, then $\varphi(X) = A^\delta$.*

If no extension rule is applicable, then we can obtain the solution M by Lemma 8. Each step takes linear time, the number of steps is at most 2ℓ, and the algorithm branches into at most $(2\ell)6^{2\ell}(2|C|)^\ell$ branches in total, thus the overall running time is $O(\ell(72|C|)^\ell|I|)$. The output is correct if the coloring γ is nice, which holds with probability at least $(2\ell)^{-2\ell}$. [3] □

References

1. E. H. L. Aarts, J. K. Lenstra (Editors): *Local Search in Combinatorial Optimization*, Wiley, New York, 1997.
2. N. Alon, R. Yuster, U. Zwick: Color-coding. *J. ACM*, 42:844–856, 1995.
3. P. Biró, E. J. McDermid: Matching with couples is hard, but extra beds help. *Manuscript*, 2009.
4. M. R. Fellows, F. V. Fomin, D. Lokshtanov, F. A. Rosamond, S. Saurabh, Y. Villanger: Local search: Is brute-force avoidable? *IJCAI 2009*, 2009.
5. J. Flum, M. Grohe: *Parameterized Complexity Theory*, Springer, 2006.
6. D. Gale, L. S. Shapley: College admissions and the stability of marriage. *American Mathematical Monthly*, 69:9–15, 1962.
7. D. Gale, M. Sotomayor: Some remarks on the stable matching problem. *Discrete Appl. Math.* 11:223–232, 1985.
8. C. A. Glass, H. Kellerer: Parallel machine scheduling with job assignment restrictions. *Naval Research Logistics* 54(3):250–257, 2007.
9. A. Krokhin, D. Marx: On the hardness of losing weight. *ICALP 2008, LNCS 5125*, 662–673, 2008.
10. B. Klaus, F. Klijn: Stable matchings and preferences of couples. *J. Econ. Theory* 121:75–106, 2005.
11. D. Marx: A parameterized view on matroid optimization problems. *ICALP 2006, LNCS 4051*, 656–667, 2006.
12. D. Marx: Local Search. *Parameterized Complexity News*, 3:7–8, 2008.
13. D. Marx: Searching the k-change neighborhood for TSP is W[1]-hard. *Oper. Res. Lett.*, 36(1):31–36, 2008.
14. E. J. McDermid, D. F. Manlove: Keeping partners together: Algorithmic results for the Hospitals/Residents problem with couples. *To appear in J. of Comb. Opt.*
15. R. Niedermeier: *Invitation to Fixed-Parameter Algorithms.* Oxford University Press, 2006.
16. E. Ronn: NP-Complete stable matching problems. *J. 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. *J. Polit. Econ.*, 92:991–1016, 1984.
18. A. E. Roth, M. Sotomayor: *Two Sided Matching: A Study in Game-Theoretic Modelling and Analysis.* Cambridge University Press, Cambridge, 1990.

[3] To derandomize the algorithm, we can use the standard method of k-perfect hash functions [2], yielding a running time of $O(\ell^{O(\ell)}|C|^\ell|I|\log|I|)$.

Appendix

A.1 HRC can admit stable assignments of different sizes

The following example by David Manlove shows that an instance of the Hospitals/Residents with Couples problem may admit stable assignments of different sizes. The example contains a single s, a couple $c = (c_1, c_2)$ and hospitals h_1 and h_2 with $f(h_1) = 2$ and $f(h_2) = 1$. The preferences are the following:

$$\begin{array}{ll} L(s) : h_2, h_1 & L(h_1) : s, c_1, c_2 \\ L(c) : (h_1, h_1), (u, h_2) & L(h_2) : c_2, s \end{array}$$

In this instance, assigning s to h_1 and c to (u, h_2) yields a stable assignment of size 2, whilst assigning s to h_2 and c to (h_1, h_1) results in a stable assignment of size 3.

A.2 Comments on the proof of Theorem 1

In Section 4, we stated Theorem 2 as a result in [11], to prove Theorem 1. However, the result that appeared in [11] is the following.

Theorem 9 ([11]). *Let $\mathcal{M}(U, \mathcal{I})$ be a linear matroid where the ground set U is partitioned into blocks of size b. Given a linear representation A of $\mathcal{M}$, it can be determined in $f(k, b) \cdot ||A||^{O(1)}$ randomized time whether there is an independent set that is the union of k blocks. ($||A||$ denotes the length of A in the input.)*

Using this result, Theorem 2 easily follows as a consequence.

Proof (of Theorem 2). First, let us make $n(u)$ copies for each $u \in U$, where $n(u)$ is the number of sets in $\mathcal{X}$ containing u, i.e. let $U' = \{u^i | u \in U, n(u) > 0, i \in [n(u)]\}$. Let $\mathcal{M}'(U', \mathcal{I}')$ be the matroid where $\mathcal{I}'$ contains those sets which can be obtained from some set $I \in \mathcal{I}$ by replacing each $u \in I$ with an arbitrary element from $\{u^i | i \in [n(u)]\}$. A representation A' of $\mathcal{M}'$ can be obtained from A by putting $n(u)$ copies of the column representing u into A' for each $u \in U$. For each $i \in [n]$, let $X'_i \subseteq U'$ be obtained by replacing each element u in X_i with u^j if X_i is the j-th set in $\mathcal{X}$ containing u. Clearly, by letting X'_i to be a block (having size b) for each $i \in [n]$, we get a partition of U'.

The sets $\{X_{i_j} | j \in [k]\}$ satisfy the requirements (being disjoint and having an independent union in $\mathcal{M}$) if and only if the sets $\{X'_{i_j}) | j \in [k]\}$ are k blocks whose union is independent in $\mathcal{M}'$, and thus the algorithm of Theorem 9 provides the solution. ☐

We remark that in the proof of Theorem 1 we also made use of the following lemma concerning matroid representations.

Lemma 10 ([11]). *(1) Given a representation A over a field F of a matroid $\mathcal{M}$, a representation of the dual matroid $\mathcal{M}^*$ over F can be found in polynomial time. (2) Given a representation A over $\mathbb{N}$ of a matroid $\mathcal{M}$ and an integer k, a representation of the k-truncation of $\mathcal{M}^k$ can be found in randomized polynomial time. (3) Given a bipartite graph $G(A, B; E)$, a representation of its transversal matroid over $\mathbb{N}$ can be constructed in randomized polynomial time.*

A.3 Proof of Theorem 3

In most proofs that consider the local search version of a problem, the following definition is convenient. Given two assignments M and M' for a cma or a cmp, we say that M is *ℓ-close* to M' if $d(M, M') \leq \ell$.

Proof (of Theorem 3). Let G be the input graph for the Clique problem and k be the parameter given. We denote the vertices of G by $v_1, v_2, \ldots, v_n$. We claim that if there is a permissive local search algorithm $\mathcal{A}$ for Maximum Matching with Couples running in FPT time with parameter ℓ, then we can use $\mathcal{A}$ to solve Clique in FPT time. To prove this, we construct an input $\Lambda = (I, M_0, \ell)$ of $\mathcal{A}$ with the following properties: every assignment for I with size at least $|M_0| + 1$ is ℓ-close to M_0, and there is such an assignment for I if and only if G has a clique of size k. Thus, G has a clique of size k if and only if $\mathcal{A}$ outputs an assignment for I with size at least $|M_0| + 1$.

To construct Λ, we first define the cma I together with the assignment M_0 for it. Let the set H of hospitals be the union of $D = B \cup \bigcup\{H^{i,j} | i, j \in [k]\}$, $D' = B' \cup \bigcup\{H'^{i,j} | i, j \in [k]\}$ and $F = \{f_i | i \in [k]\}$, where $B = \{b_i | i \in [2k-1]\}$, $H^{i,i} = \{h^{i,i}_{j,j} | j \in [n]\}$ for each $i \in [k]$, $H^{i,j} = \{h^{i,j}_{x,y} | v_x v_y \in E(G)\}$ for each $i \neq j, \{i, j\} \subseteq [k]$, and for each hospital h in B ($H^{i,j}$, respectively) we also define a hospital h' to be in B' ($H'^{i,j}$, respectively). For brevity, we will use the notation $H^{i,j}_{z,\bullet} = \{h | \exists y : h = h^{i,j}_{z,y} \in H^{i,j}\}$ and $H^{i,j}_{\bullet,z} = \{h | \exists x : h = h^{i,j}_{x,z} \in H^{i,j}\}$. The capacity of each hospital is 2. For each hospital $h \in D$ we define a couple denoted by $c(h)$, and for each $h' \in D'$ we define two singles $s_1(h')$ and $s_2(h')$. Let $C = \{c(h) | h \in D\}$ and let $S = \{s_0\} \cup \{s_i(h') | h' \in D', i \in \{1, 2\}\}$.

Before defining $A(p)$ for each $p \in S \cup C$, we define the assignment M_0 for I, as this will not cause any confusion. Let $M_0(s_0) = u$, and let $M_0(p) = h$ where either $h \in D$ and p is a member of the couple $c(h)$, or $h \in D'$ and $p \in \{s_1(h), s_2(h)\}$. Now, for each $p \in S \cup C$, we define the set of acceptable hospitals or pairs of hospitals $A(p)$ to be the union of $\{M_0(p)\}$ and the set $A'(p)$ of hospitals, defined below, that can be assigned to p besides $M_0(p)$. We define $A'(p)$ for each $p \in S \cup C$ as follows.

$A'(c(h)) = \{(h', h')\}$ for each $h \in D$
$A'(s_0) = \{b_1\}$
$A'(s_1(b'_i)) = H^{1,i}$ for each $i \in [k]$
$A'(s_2(b'_i)) = \{b_{i+1}\}$ for each $i \in [k]$
$A'(s_1(b'_{k+i})) = H^{i,1}$ for each $i \in [k-1]$
$A'(s_2(b'_{k+i})) = \{b_{k+i+1}\}$ for each $i \in [k-2]$
$A'(s_2(b'_{2k-1})) = H^{k,1}$
$A'(s_1(h'^{i,j}_{x,y})) = H^{i,j+1}_{x,\bullet}$ for each $i \in [k], j \in [k-1]$ and every possible x and y
$A'(s_1(h'^{i,k}_{x,y})) = \{f_i\}$ for each $i \in [k]$ and every possible x and y
$A'(s_2(h'^{i,j}_{x,y})) = H^{i+1,j}_{\bullet,y}$ for each $i \in [k-1], j \in [k]$ and every possible x and y
$A'(s_2(h'^{k,i}_{x,y})) = \{f_i\}$ for each $i \in [k]$ and every possible x and y

This completes the definition of the cma $I = (S, C, H, f, A)$. Observe that M_0 indeed is an assignment for I. Finally, setting $\ell = 4k^2 + 8k - 3$ finishes the definition of the instance $\Lambda = (I, M_0, \ell)$. Figure 4 shows an illustration.

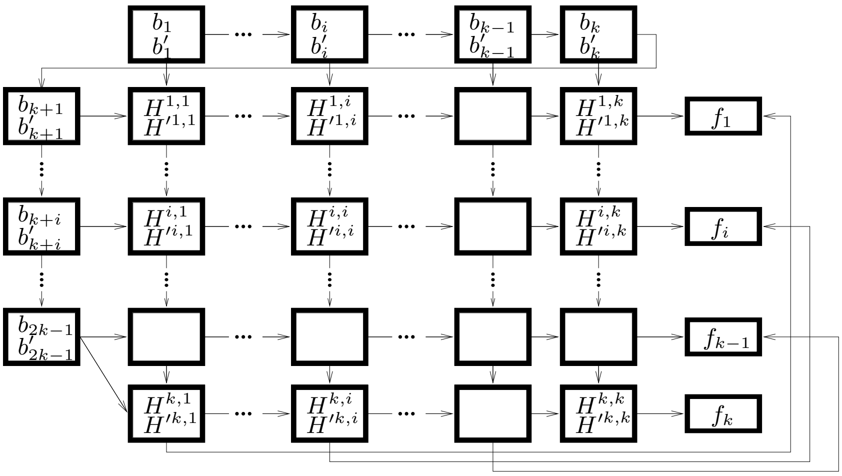

Fig. 4. A block diagram showing the hospitals in the proof of Theorem 3. For two sets H_1, H_2 of hospitals, (H_1, H_2) is an arc if $A'(s) \subseteq H_2$ for some $s \in S$ with $M_0(s) \in H_1$.

First, suppose that M is an assignment for I such that $|M| > |M_0|$. We do not require M to be $(4k^2 + 8k - 3)$-close to M_0, but we will actually prove that this is necessary. Observe that M_0 covers each resident except for s_0, so M must cover all residents to satisfy $|M| > |M_0|$. As $A(s_0) = \{b_1\}$, M must assign s_0 to b_1. This implies $M(c(b_1)) = (b'_1, b'_1)$, and therefore we also have $M(s_2(b'_1)) = b_2$, implying $M(c(b_2)) = (b'_2, b'_2)$, and so on. Following this argument, it can be seen that $M(c(b_i)) = (b'_i, b'_i)$ for every $i \in [2k-1]$, and $M(s_2(b'_i)) = b_{i+1}$ for every $i \in [2k-2]$.

We say that a single s *enters* $H^{i,j}$ if $M(s) \in H^{i,j}$ but $M_0(s) \notin H^{i,j}$, and *leaves* $H'^{i,j}$ if $M_0(s) \in H'^{i,j}$ but $M(s) \notin H'^{i,j}$. A couple c *moves* from a hospital h if $M_0(c) = (h, h) \neq M(c)$, and we say that c moves from a set $J \subseteq H$ of hospitals if it moves from a hospital in J. Observe that if c moves from $H^{i,j}$, then two singles leave $H'^{i,j}$, one of them entering $H^{i+1,j}$ if $i \neq k$, and the other entering $H^{i,j+1}$ if $j \neq k$. If a single s leaves $H'^{i,j}$ but does not enter $H^{i+1,j}$ or $H^{i,j+1}$, then $M(s) \in F$ must hold, and therefore there can exist at most $2k$ such single s. Moreover, if a set of m singles enter $H^{i,j}$ then at least $\lceil m/2 \rceil$ couples have to move from $H^{i,j}$. For each $i \in [k]$, exactly one single from $\{s_1(b'_1), s_1(b'_2), \ldots, s_1(b'_k)\}$ enters $H^{1,i}$, and exactly one single from $\{s_1(b'_{k+1}), s_1(b'_{k+2}), \ldots, s_1(b'_{2k-1}), s_2(b'_{2k-1})\}$ enters $H^{i,1}$. These altogether imply that exactly one couple moves from $H^{i,j}$ for each $i, j \in [k]$, and that if s and s' enter $H^{i,j}$ then $M(s) = M(s')$ must hold.

Suppose that c moves from the hospital $h^{i,j}_{x,y}$. Observe that if $j < k$ then a couple must move from $H^{i,j+1}_{x,\bullet}$, and similarly, if $i < k$ then a couple must move from $H^{i+1,j}_{\bullet,y}$. For each $i \in [k]$, letting $\sigma_h(i)$ to be x if for some j a couple moves from $H^{i,j}_{x,\bullet}$, and $\sigma_v(i)$ to be y if for some j a couple moves from $H^{j,i}_{\bullet,y}$, we obtain that $\sigma_h(i)$ and $\sigma_v(i)$ are well-defined. Observe that by the definition of $H^{i,i}$ we get $\sigma_h(i) = \sigma_v(i) := \sigma(i)$, and from the definition of $H^{i,j}$ we get that if $\sigma(i) = x$ and $\sigma(j) = y$ for some $i \neq j$, then $v_x v_y$ must be an edge in G. Thus, the set $\{v_{\sigma(i)} | i \in [k]\}$ forms a clique of size k in G.

Remember that exactly one couple moves from $H^{i,j}$ for each $i, j \in [k]$, which (considering also the size of F) forces exactly two singles to leave $H'^{i,j}$ for each $i, j \in [k]$. Taking into account the couples $c(b_i)$ and the singles $s_1(b'_i), s_2(b'_i)$ for each $i \in [2k-1]$ and the single s_0, we get that M is $4k^2 + 4(2k-1) + 1 = (4k^2 + 8k - 3) = \ell$-close to M_0.

On the other hand, suppose $v_{\sigma(1)}, v_{\sigma(2)}, \dots, v_{\sigma(k)}$ form a clique in G. By defining M as below, it is straightforward to verify that M is an assignment for (S, C, H, f, A) which covers every resident, and is ℓ-close to M_0.

$M(c(b_i)) = (b'_i, b'_i)$ for each $i \in [2k-1]$
$M(c(h^{i,j}_{\sigma(i),\sigma(j)})) = (h'^{i,j}_{\sigma(i),\sigma(j)}, h'^{i,j}_{\sigma(i),\sigma(j)})$ for each $i, j \in [k]$
$M(s_0) = b_1$
$M(s_1(b'_i)) = h^{1,i}_{\sigma(1),\sigma(i)}$ for each $i \in [k]$
$M(s_1(b'_{k+i})) = h^{i,1}_{\sigma(i),\sigma(1)}$ for each $i \in [k-1]$
$M(s_2(b'_{2k-1})) = h^{k,1}_{\sigma(k),\sigma(1)}$
$M(s_2(b'_i)) = b_{i+1}$ for each $i \in [2k-2]$
$M(s_1(h'^{i,j}_{\sigma(i),\sigma(j)})) = h^{i,j+1}_{\sigma(i),\sigma(j+1)}$ for each $i \in [k], j \in [k-1]$
$M(s_2(h'^{i,j}_{\sigma(i),\sigma(j)})) = h^{i+1,j}_{\sigma(i+1),\sigma(j)}$ for each $i \in [k-1], j \in [k]$
$M(s_1(h'^{i,k}_{\sigma(i),\sigma(k)})) = f_i$ for each $i \in [k]$
$M(s_2(h'^{k,i}_{\sigma(k),\sigma(i)})) = f_i$ for each $i \in [k]$
$M(p) = M_0(p)$ for every $p \in S \cup C$ where $M(p)$ was not defined above.

□

A.4 Proofs of Theorem 4 and 5

In the proof of Theorem 4, we will use the following simple cmp $I_0 = (S, C, H, f, L)$ having no stable assignments. Let $H = \{h_1, h_2, h_3\}$, $S = \emptyset$, $C = \{(a, b), (c, d)\}$ and $f \equiv 1$. The preference lists are defined below. It is easy to verify that no stable assignment exists for I_0. For example, $M(a) = h_1$, $M(b) = h_2$ and $M(c) = M(d) = u$ is not stable, because (c, d) and (h_1, h_3) form a blocking pair.

$L((a, b)) : (h_1, h_2), (h_2, h_3), (h_3, h_1)$ $\qquad$ $L(h_1) = L(h_2) = L(h_3) : c, a, b, d$
$L((c, d)) : (h_1, h_3), (h_2, h_1), (h_3, h_2)$

Proof (of Theorem 4). Let G be an arbitrary graph and $k \in \mathbb{N}$. We construct two 1-uniform cmps I_1 and I_2, together with a stable assignment M_2 for I_2 such that the following three statements are equivalent:

(a) G has a clique of size k,
(b) I_1 has a stable assignment,
(c) I_2 has a stable assignment of size greater than $|M_2|$.

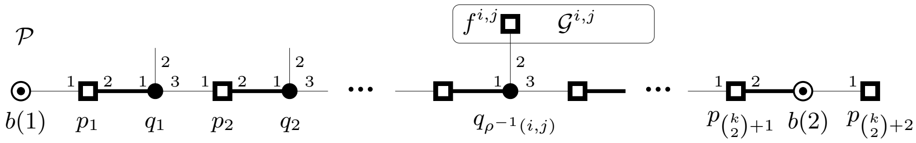

Fig. 5. The path-gadget $\mathcal{P}$ in I_2. Bold edges represent M_2.

Furthermore, the construction will take FPT time, and there will be $k + 3\binom{k}{2}$ couples in I_1, and $k + \binom{k}{2} + 1$ couples in I_2. Thus, (a) $\iff$ (b) yields an FPT-reduction from Clique to Hospitals/Residents with Couples, and (a) $\iff$ (c) yields an FPT-reduction from Clique to Increase Hospitals/Residents with Couples.

To get I_1, we simply combine the cmp I_0 having no stable assignment with the cmp $I^{G,k}$. This is done by introducing new couples $b^{i,j}$ and $c^{i,j}$, and new hospitals $\bar{f}_1^{i,j}$ and $\bar{f}_2^{i,j}$ for each $(i,j) \in \binom{[k]}{2}$, and adding these agents to $I^{G,k}$. We preserve the preference lists of $I^{G,k}$, except for hospitals $\{f^{i,j} | (i,j) \in \binom{[k]}{2}\}$, and we give the missing preference lists below.

$L(b^{i,j}) : (f^{i,j}, \bar{f}_1^{i,j}), (\bar{f}_1^{i,j}, \bar{f}_2^{i,j}), (\bar{f}_2^{i,j}, f^{i,j})$
$L(c^{i,j}) : (f^{i,j}, \bar{f}_2^{i,j}), (\bar{f}_1^{i,j}, f^{i,j}), (\bar{f}_2^{i,j}, \bar{f}_1^{i,j})$
$L(\bar{f}_1^{i,j}) = L(\bar{f}_2^{i,j}) : c^{i,j}(1), b^{i,j}(1), b^{i,j}(2), c^{i,j}(2)$
$L(f^{i,j}) : s_1^{i,j}, s_2^{i,j}, \ldots, s_m^{i,j}, c^{i,j}(1), b^{i,j}(1), b^{i,j}(2), c^{i,j}(2)$

Observe that if we restrict I_1 to contain only the hospitals $f^{i,j}, \bar{f}_1^{i,j}$ and $\bar{f}_2^{i,j}$ and the couples $b^{i,j}$ and $c^{i,j}$ for some $(i,j) \in \binom{[k]}{2}$, we obtain a cmp isomorphic to I_0, having no stable assignment. Therefore, any stable assignment M must assign a single in $S^{i,j}$ to $f^{i,j}$ for each $(i,j) \in \binom{[k]}{2}$, so M has property π. The restriction of such an M on the agents of $I^{G,k}$ must also be stable, because agents of $I^{G,k}$ cannot be assigned by M to agents outside $I^{G,k}$. Thus, by Lemma 6, G has a k-clique.

On the other hand, if there is a k-clique in G, then we can construct a stable assignment M_1 for I_1 by setting $M_1(b^{i,j}) = (\bar{f}_1^{i,j}, \bar{f}_2^{i,j})$, $M_1(c^{i,j}) = (u,u)$ for each $(i,j) \in \binom{[k]}{2}$, and $M_1(r) = M_\pi^{G,k}(r)$ for the residents in $I^{G,k}$. Here, $M_\pi(G,k)$ is the stable assignment for $I^{G,k}$ with property π and covering each resident of $I^{G,k}$, guaranteed by Lemma 6. It is easy to see that M_1 is stable, by using the stability of $M_\pi(G,k)$. This finishes the proof of the first claim.

To construct I_2, we add a *path-gadget* $\mathcal{P}$ to $I^{G,k}$ that contains the newly introduced hospitals $\{p_i | i \in [\binom{k}{2}+2]\}$, singles $\{q_i | i \in [\binom{k}{2}]\}$ and a couple b. See Fig. 5 for an illustration. As before, we only modify the preferences of the hospitals $\{f^{i,j} | (i,j) \in \binom{[k]}{2}\}$, and we give the missing preference lists below. The notation ρ used there denotes a bijection from $[\binom{k}{2}]$ into $\binom{[k]}{2}$.

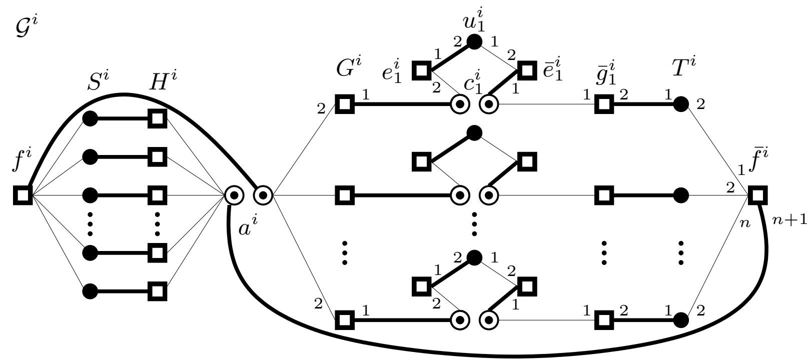

Fig. 6. The modified node-gadget in the proof of Theorem 5. Bold edges represent M_3.

$L(p_1) : b(1), q_1$ $\qquad$ $L(p_i) : q_{i-1}, q_i \quad$ if $1 < i \leq \binom{k}{2}$
$L(p_{\binom{k}{2}+1}) : q_{\binom{k}{2}}, b(2)$ $\qquad$ $L(p_{\binom{k}{2}+2}) : b(2)$
$L(q_i) : p_i, f^{\rho(i)}, p_{i+1}$ $\qquad$ $L(f^{i,j}) : s_1^{i,j}, s_2^{i,j}, \ldots, s_m^{i,j}, q_{\rho^{-1}(i,j)}, a^{i,j}(2)$
$L(b) : (u, p_{\binom{k}{2}+1}), (p_1, p_{\binom{k}{2}+2})$

We also let $M_2(q_i) = p_i$ for each $i \in [\binom{k}{2}]$, $M_2(b) = (u, p_{\binom{k}{2}+1})$, and $M_2(r) = M_0^{G,k}(r)$ for the residents in $I^{G,k}$, where $M_0^{G,k}$ is the stable assignment for $I^{G,k}$, provided by Lemma 6. Note that M_2 is indeed stable.

Suppose, there is a stable assignment M for I_2 with $|M| > |M_2|$. Observe that M_2 covers each resident except for $b(1)$, so M must cover every resident, implying $M(b) = (p_1, p_{\binom{k}{2}+2})$. Also, since $M(h)$ cannot be empty for any hospital h, we get $M(p_i) = \{q_{i-1}\}$ for each $i = \binom{k}{2}+1, \binom{k}{2}, \ldots, 2$. Thus, $f^{\rho(i)}$ is beneficial for q_i for each $i \in [\binom{k}{2}]$, so by the stability of M we obtain $M(f^{i,j}) \subseteq S^{i,j}$ for each $(i,j) \in \binom{[k]}{2}$. Again, the restriction of M on the agents of $I^{G,k}$ must be stable, and so Lemma 6 implies that G has a clique of size k.

Conversely, if there is a k-clique in G, then we can define a stable assignment M_2' for I_2 covering each resident as follows. We let $M_2'(q_i) = p_{i+1}$ for each $i \in [\binom{k}{2}]$, $M_2'(b) = (p_1, p_{\binom{k}{2}+2})$, and $M_2'(r) = M_\pi^{G,k}(r)$ for the residents in $I^{G,k}$. Again M_2' is stable, and has size greater than $|M_2|$, proving the second claim. □

Proof (of Theorem 5). Let G be a graph and k an integer. First, recall the cmp I_2 defined in the proof of Theorem 4, and observe that the assignment M_2 and the assignment M_2', constructed when a k-clique is present in G, may not be close to each other. Thus, in order to present an FPT-reduction here, we need to modify the node- and edge-gadgets of I_2. We are going to construct a cmp I_3 together with a stable assignment M_3 for it such that the following statements are equivalent:

(a) G has a clique of size k.

(b) There is a stable assignment for I_3 with size at least $|M_3| + 1$.
(c) There is a stable assignment for I_3 with size at least $|M_3| + 1$ that is ℓ-close to M_3 where $\ell = 8\binom{k}{2} + 7k + 2$.

The construction will take FPT time, hence a permissive local search algorithm for Maximum Hospitals/Residents with Couples that runs in FPT time with parameter ℓ can be used to solve Clique in FPT time.

See Fig. 6 for an illustration of the modifications applied to I_2 in order to get I_3. For each node- or edge-gadget $\mathcal{G}^\alpha$, we take new singles $\{u_x^\alpha | x \in [N(\alpha)]\}$ and the single $t^\alpha_{N(\alpha)}$, new couples $\{c_x^\alpha | x \in [N(\alpha)]\}$, and new hospitals $\bigcup_{x \in [N(\alpha)]} \{\bar{g}_x^\alpha, e_x^\alpha, \bar{e}_x^\alpha\} \cup \{\bar{f}^\alpha\}$. For most of the agents we preserve the preferences originally defined for I_2. The modifications and the preference lists of the newly defined agents are as follows.

$L(g_x^\alpha) : c_x^\alpha(1), a^\alpha(2)$	$L(t_x^\alpha) : \bar{g}_x^\alpha, \bar{f}^\alpha$
$L(e_x^\alpha) : u_x^\alpha, c_x^\alpha(1)$	$L(u_x^\alpha) : \bar{e}_x^\alpha, e_x^\alpha$
$L(\bar{e}_x^\alpha) : c_x^\alpha(2), u_x^\alpha$	$L(c_x^\alpha) = (e_x^\alpha, \bar{g}_x^\alpha), (g_x^\alpha, \bar{e}_x^\alpha)$
$L(\bar{g}_x^\alpha) : c_x^\alpha(2), t_x^\alpha$	$L(\bar{f}^\alpha) : t_1^\alpha, t_2^\alpha, \ldots, t^\alpha_{N(\alpha)}, a^\alpha(1)$

$L(a^\alpha) : (\bar{f}^\alpha, f^\alpha), (h_1^\alpha, g^\alpha_{N(\alpha)}), (h_2^\alpha, g^\alpha_{N(\alpha)-1}), ;(h^\alpha_{N(\alpha)}, g_1^\alpha)$

We also define $M_3(a^\alpha) = (\bar{f}^\alpha, f^\alpha)$, $M_3(c_x^\alpha) = (g_x^\alpha, \bar{e}_x^\alpha)$, $M_3(u_x^\alpha) = e_x^\alpha$ and $M_3(t_x^\alpha) = \bar{g}_x^\alpha$ for all possible values of α and x, and for each remaining resident r let $M_3(r) = M_2(r)$. It is easy to observe that M_3 is stable, and covers each resident except for $b(1)$.

Supposing that there is a stable assignment M with size greater than $|M_3|$ and using exactly the same arguments as in the proof of Theorem 4, we get $M(b) = (p_1, p_{\binom{k}{2}+2})$, $M(q_i) = (p_{i+1})$ for each $i \in [\binom{k}{2}]$, and $M(f^{i,j}) \subseteq S^{i,j}$ for each $(i,j) \in \binom{[k]}{2}$. By following the argument proving (2) $\Rightarrow$ (1) in Lemma 6, we again obtain that G must have a k-clique. (The modifications of the gadgets in I_3 to do not affect that reasoning.) This proves (b) $\Rightarrow$ (a).

Clearly, (c) $\Rightarrow$ (b) is trivial, so we only have to prove (a) $\Rightarrow$ (c). Suppose that G has a clique $\{v_{\sigma(i)} | i \in [k]\}$. We again let $\sigma(i,j) = \nu^{-1}(\sigma(i), \sigma(j))$, and we write $\sigma'(\alpha)$ for $N(\alpha) + 1 - \sigma(\alpha)$. We define a stable assignment M_3' for I in a very similar fashion as in the previous proofs:

$M_3'(b) = (p_1, p_{\binom{k}{2}+2})$	$M_3'(u^\alpha_{\sigma'(\alpha)}) = \bar{e}^\alpha_{\sigma'(\alpha)}$
$M_3'(q_i) = p_{i+1}$ for each $i \in [\binom{k}{2}]$	$M_3'(s^\alpha_{\sigma(\alpha)}) = f^\alpha$
$M_3'(a^\alpha) = (h^\alpha_{\sigma(\alpha)}, g^\alpha_{\sigma'(\alpha)})$	$M_3'(t^\alpha_{\sigma'(\alpha)}) = \bar{f}^\alpha$
$M_3'(c^\alpha_{\sigma'(\alpha)}) = (e^\alpha_{\sigma'(\alpha)}, \bar{g}^\alpha_{\sigma'(\alpha)})$	

For each remaining resident r we let $M_3'(r) = M_3(r)$. It is straightforward to verify that M_3' is stable, and it is ℓ-close to M_0. □

A.5 Comments on the proof of Theorem 7

In Section 5, we mentioned the trick of cloning the hospitals for a cmp with capacity $f \not\equiv 1$, which we now describe in more detail.

For each hospital $h \in H$ in a given cmp, we take $f(h)$ copies of h by replacing h with new hospitals $h^1, \ldots, h^{f(h)}$, each having capacity 1. The preference lists of these hospitals agree with the original preference list of h. For each single s containing h in its preference list, we replace h in the list $L(s)$ by the series $h^1, \ldots, h^{f(h)}$. For a couple c containing a pair (h, g) of two hospitals in $L(c)$, we replace (h, g) by a series formed by the elements of $\{(h^i, g^j) : i \in [f(h)], j \in [f(g)]\}$ such that (h^i, g^j) precedes $(h^{i'}, g^{j'})$ if $i < i'$, or $i = i'$ and $j < j'$. (We hope that the cases $h = u$ and $g = u$ are also clear.)

Now, if M is an assignment for the original cmp I, then it defines an assignment M^c for the cmp I^c obtained by the above cloning process, as follows. If M assigns r to h and there are $i - 1$ residents in $M(h)$ that h prefers to r, then let $M^c(r) = h^i$. If $M(r) = u$ for some r, then we let $M^c(r) = u$ as well. Observe that if M is stable, then M^c is also stable. Conversely, it is not hard to see that a stable assignment for I^c can be transformed in the straightforward way into a stable assignment for I.

To prove Theorem 7, we used Lemma 8 whose proof is the following.

Proof (of Lemma 8). First, $\varphi(X) \supseteq C^\delta$ is trivial, as Rule 1 is not applicable.

Claim 1: $\varphi(X) \supseteq (H^- \cup S^+) \cap V(\mathcal{K}_1)$. Suppose $a \in (H^- \cup S^+) \cap V(\mathcal{K}_1) \setminus \varphi(X)$ is chosen such that the distance $d^C(a)$ is minimal, where $d^C(a)$ is the minimum length of a path P in G^δ from a to some $c \in C^\delta$ such that the first edge of P is in $E(M_0)$ if $a \in H$ and it is in $E(M)$ if $a \in S$. If no such path exists, then let $d^C(a) = \infty$.

First, if a is a winner single, then $M(a)$ exists, and since a and $M(a)$ cannot be a blocking pair for M_0, $M(a)$ must be a loser hospital. Now, if $M(a) \in \varphi(X)$ then Rule 3 is applicable, a contradiction. Thus $M(a) \notin \varphi(X)$, but as $M(a)$ is on the path defining $d^C(a)$, we get $d^C(M(a)) < d^C(a)$ contradicting to the choice of a. (Note that $d^C(a) \neq \infty$ as $a \in V(\mathcal{K}_1)$.) On the other hand, if a is a loser hospital, then $M_0(a)$ exists. Observe that if $M_0(a) \in \varphi(X)$, then Rule 2 is applicable, which cannot be the case, so $M_0(a)$ can only be a single in $S \setminus \varphi(X)$. If $M_0(a)$ were a loser, then a and $M_0(a)$ would form a blocking pair for M, so we obtain $M_0(a) \in S^+ \setminus \varphi(X)$. But this implies $d^C(M_0(a)) < d^C(a)$, a contradiction. Thus, $\varphi(X)$ indeed contains $(H^- \cup S^+) \cap V(\mathcal{K}_1)$.

Claim 2: $\varphi(X) \supseteq V(\mathcal{K}_1)$. By Claim 1, we only have to prove that $(H^+ \cup S^-) \cap V(\mathcal{K}_1) \setminus \varphi(X)$ is empty. Analogously as in Claim 1, we choose $a \in (H^+ \cup S^-) \cap V(\mathcal{K}_1) \setminus \varphi(X)$ such that the distance $d'^C(a)$ is minimal, where $d'^C(a)$ is the minimum length of a path P in G^δ from a to some $c \in C^\delta$ such that the first edge of P is in $E(M)$ if $a \in H$ and it is in $E(M_0)$ if $a \in S$. If no such path exists then let $d'^C(a) = \infty$. Note that $d'^C \neq d^C$, as the requirements for the first edge of the path P are different.

First, if a is a loser single, then $M_0(a)$ exists, and since a and $M_0(a)$ cannot be a blocking pair for M, $M_0(a)$ must be a winner hospital. Now, if $M_0(a) \in \varphi(X)$ then Rule 2 is applicable, a contradiction. Thus $M_0(a) \notin \varphi(X)$, but as $M_0(a)$ is on the path defining $d'^C(a)$, we get $d'^C(M_0(a)) < d'^C(a)$ contradicting to the choice of a. Again, $d'^C(a) \neq \infty$ as $a \in V(\mathcal{K}_1)$.

On the other hand, if a is a winner hospital, then $M(a)$ exists. Observe that if $M(a)$ is a member of some couple c, then if $M(c(i))$ is not known for some $i \in \{1, 2\}$, then $M(c(i))$ can only be a winner hospital by Claim 1, so Rule 4 or

5 is applicable. Thus, $M(a) \in S$. If $M(a)$ were a winner, then a and $M(a)$ would form a blocking pair for M_0, so we obtain $M(a) \in S^-$. Now, if $M(a) \in S^- \cap \varphi(X)$, then Rule 3 is applicable. Thus, only $M(a) \in S^- \setminus \varphi(X)$ is possible. But this implies $d'^C(M(a)) < d'^C(a)$, which is a contradiction proving Claim 2.

Claim 3: $\varphi(X) \supseteq V(\mathcal{K}_0)$. As already mentioned, each component of $\mathcal{K}_0$ is a cycle, and it easy to see that it must contain vertices from A^+ and A^- in an alternating manner. Thus, if neither Rule 2 nor Rule 3 is applicable, then each component of $\mathcal{K}_0$ is totally contained in either $A^\delta \setminus \varphi(X)$ or in $\varphi(X)$. Thus, the first condition of Rule 6 must hold. Now, if $\varphi(X) \neq A^\delta$ then clearly $M_X \neq M$ for the actual assignment M_X. As M_X is closer to M_0 than M, and M is a closest solution, M_X cannot be stable. Thus, Rule 6 is applicable, a contradiction.

Now, Claims 1, 2, and 3 together imply the lemma. □