Perfect Matchings and Popularity in the Many-To-Many Setting

Telikepalli Kavitha
Tata Institute of Fundamental Research, Mumbai, India

Kazuhisa Makino
Research Institute for Mathematical Sciences, Kyoto University, Japan

Abstract

We consider a matching problem in a bipartite graph G where every vertex has a capacity and a strict preference list ranking its neighbors. We assume that G admits a perfect matching, i.e., one that fully matches all vertices. It is only perfect matchings that are feasible here and we seek one that is popular within the set of perfect matchings – it is known that such a matching exists in G and can be efficiently computed. Now we are in the weighted setting, i.e., there is a cost function on the edge set, and we seek a min-cost popular perfect matching in G. We show that such a matching can be computed in polynomial time.

Our main technical result shows that every popular perfect matching in a hospitals/residents instance G can be realized as a popular perfect matching in the marriage instance obtained by cloning vertices. Interestingly, it is known that such a mapping does not hold for popular matchings in a hospitals/residents instance.

2012 ACM Subject Classification Theory of computation → Design and analysis of algorithms

Keywords and phrases Bipartite graphs, Matchings under preferences, Capacities, Dual certificates

Digital Object Identifier 10.4230/LIPIcs.FSTTCS.2023.43

Funding *Telikepalli Kavitha*: Supported by the Department of Atomic Energy, Government of India, under project no. RTI4001. This work was done while visiting RIMS, Kyoto.
Kazuhisa Makino: Partially supported by JST ERATO Grant Number JPMJER2301, JSPS KAKENHI Grant Numbers JP20H05967, JP19K22841, and JP20H00609, and the joint project of Kyoto University and Toyota Motor Corporation, titled "Advanced Mathematical Science for Mobility Society".

Acknowledgements Thanks to Naoyuki Kamiyama for asking us about the computational complexity of the min-cost popular maximum matching problem when vertices have capacities.

1 Introduction

We consider a matching problem in a bipartite graph $G = (R \cup H, E)$ where R is a set of residents and H is a set of hospitals. Every resident r seeks to be matched to a hospital while every hospital h has an integral capacity $\mathsf{cap}(h) \geq 1$ and seeks to be matched to $\mathsf{cap}(h)$ many residents in any matching.

▶ **Definition 1.** *A matching M in $G = (R \cup H, E)$ is a subset of E such that $|M(r)| \leq 1$ for each $r \in R$ and $|M(h)| \leq \mathsf{cap}(h)$ for each $h \in H$, where $M(v) = \{u : (u, v) \in M\}$ for any $v \in R \cup H$.*

Every vertex $v \in R \cup H$ has a strict ranking of its neighbors. Such a graph G is called a hospitals/residents instance and this is a well-studied model in two-sided matching markets. This model includes several real-world applications such as matching doctors to residencies in hospitals [9, 29] and students to schools and colleges [1, 4]. It is crucial to match as many doctors as possible to residencies and all students to schools and colleges.

Let us assume that the input instance admits a perfect matching, i.e., one that fully matches all vertices in $R \cup H$. Among other scenarios, such a model occurs in the "lab rotation" problem where each student gets assigned to a lab during training. At certain intervals, based on the preferences of students and those in charge of labs, there may be a cyclic shift of students among labs. The set of feasible solutions is the set of perfect matchings. Note that the instance admits a perfect matching which is the current assignment of students to labs. We seek a "best perfect matching" as per vertex preferences – in the domain of matchings under preferences, *stable* matchings are usually regarded as the best matchings.

Stable matchings. A matching M is stable if there is no edge that *blocks* M, where an edge (r, h) blocks M if (i) either r is unmatched in M or r prefers h to its partner in M and (ii) either h has less than $\mathsf{cap}(h)$ partners in M or h prefers r to its worst partner in M. Stable matchings always exist in G and the classical Gale-Shapley algorithm [14] finds one. Stability is a strong and rather restrictive notion – it is known that every stable matching matches the same subset of residents and every hospital gets matched to the same capacity; this is known as the "Rural Hospitals Theorem" [15, 31].

Stable matchings need not have maximum cardinality. Consider the following instance where $R = \{r, r'\}, H = \{h, h'\}$, and all vertex capacities are 1. The preferences of residents are as follows: $r : h \succ h'$ and $r' : h$, i.e., r's top choice is h and second choice is h' while r' has only one neighbor h. Similarly, the preferences of hospitals are as follows: $h : r \succ r'$ and $h' : r$. This instance has only one stable matching $S = \{(r, h)\}$ that leaves r' and h' unmatched. Observe that this instance has a perfect matching $M = \{(r, h'), (r', h)\}$ that matches all the residents and hospitals.

A perfect matching with the least number of blocking edges would be a natural relaxation of stability; unfortunately, finding such a matching is NP-hard [6]. A well-studied relaxation of stability that offers a meaningful and tractable solution to the problem of finding a "best perfect matching" is the notion of *popularity*.

Popularity. Popularity is based on voting by vertices on feasible matchings. In the one-to-one or marriage setting, the preferences of a vertex over its neighbors extend naturally to preferences over matchings – so every vertex orders feasible matchings in the order of its partners in these matchings and being unmatched is the worst option for any vertex. Popular matchings are (weak) *Condorcet winners* [12, 24] in the above voting instance where all matchings are feasible, i.e., a popular matching M does not lose a head-to-head election against any matching N where each vertex either casts a vote for the matching in $\{M, N\}$ that it prefers or it abstains from voting if its assignment is the same in M and N. There need not be a weak Condorcet winner in a typical voting instance; however, in the context of matchings in the one-to-one or marriage setting, every stable matching is a weak Condorcet winner [16] – so popular matchings always exist in this setting.

Recall that we are in the many-to-one or hospitals/residents setting, i.e., hospitals have capacities. So we need to specify how a hospital votes over different subsets of its neighbors. Thus for a hospital h, we need to compare two subsets $M(h)$ and $N(h)$ of $\mathsf{Nbr}(h)$, where $\mathsf{Nbr}(h)$ is the set of neighbors of h. We will follow the method from [8] for this comparison. The definition in [8] of voting by a hospital h between two subsets of $\mathsf{Nbr}(h)$ is the following:

- First, all residents that are contained in both sets are removed and then a bijection from the first set to the second set is determined.[1] Every vertex is compared with its image under this bijection, thus the vote depends on the bijection that is chosen.

[1] If the sets are not of equal size, then dummy vertices that are less preferred to all non-dummy vertices are added to the smaller set.

Voting by vertices. Let us first formally define how a resident r casts its vote between two neighbors h and h'. The function $\mathsf{vote}_r(h, h')$ is 1 if r prefers h to h', it is -1 if r prefers h' to h, and it is 0 otherwise, i.e., $h = h'$. The function $\mathsf{vote}_r(\cdot, \cdot)$ that compares two neighbors of r extends naturally to any pair of matchings M and N as $\mathsf{vote}_r(M, N) = \mathsf{vote}_r(M(r), N(r))$. Note that $M(r)$ (resp., $N(r)$) is null if r is unmatched in M (resp., N) and this is the worst state for any vertex. Every resident casts a vote in $\{0, \pm 1\}$ in the M-vs-N election.

A hospital h with capacity $\mathsf{cap}(h)$ is allowed to cast up to $\mathsf{cap}(h)$ many votes. Let S and T be two subsets of $\mathsf{Nbr}(h)$; the vertex h compares these two subsets as follows:

- a bijection ψ is chosen from $S' = S \setminus T$ to $T' = T \setminus S$;
- every resident $r \in S'$ is compared with $\psi(r) \in T'$;
- the number of wins minus the number of losses is h's vote for S versus T.

The bijection ψ that is chosen will be the one that *minimizes* h's vote for S' versus T'. More formally, the vote of h for S versus T, denoted by $\mathsf{vote}_h(S, T)$, is defined as follows where $|S'| = |T'| = k$ and $\Pi[k]$ is the set of permutations on $\{1, \ldots, k\}$:

$$\mathsf{vote}_h(S, T) = \min_{\sigma \in \Pi[k]} \sum_{i=1}^{k} \mathsf{vote}_h(S'[i], T'[\sigma(i)]), \tag{1}$$

where $S'[i]$ is the i-th ranked resident in S' and $T'[\sigma(i)]$ is the $\sigma(i)$-th ranked resident in T'.

For any pair of matchings M and N in G, let $\mathsf{vote}_h(M, N) = \mathsf{vote}_h(S, T)$, where $S = M(h)$ and $T = N(h)$. So $\mathsf{vote}_h(M, N)$ counts the number of votes by h for $M(h)$ versus $N(h)$ when the two sets $M(h) \setminus N(h)$ and $N(h) \setminus M(h)$ are compared in the order that is most adversarial or *negative* for M. The two matchings M and N are compared using $\Delta(M, N) = \sum_{v \in R \cup H} \mathsf{vote}_v(M, N)$. A matching M is popular if $\Delta(M, N) \geq 0$ for all matchings N in G.

Recall our assumption that the set of feasible solutions for us is the set of perfect matchings. Hence the matchings of interest to us are *popular perfect matchings*, defined below.

▶ **Definition 2.** *A perfect matching M is a popular perfect matching if $\Delta(M, N) \geq 0$ for all perfect matchings N in G.*

It was shown in [22] that there always exists a popular maximum matching (i.e., a maximum matching that is popular within the set of maximum matchings) in the one-to-one setting; moreover, such a matching can be computed in polynomial time. Thus whenever the input instance admits a perfect matching, there always exists a popular perfect matching. This result from the one-to-one setting extends easily to the many-to-one or hospitals/residents setting.

Let G' be the marriage instance obtained from G by making $\mathsf{cap}(h)$ many *clones* $h_1, \ldots, h_{\mathsf{cap}(h)}$ of each hospital h where $h_1 \succ_r \cdots \succ_r h_{\mathsf{cap}(h)}$ for each resident r adjacent to h in G (Section 1.2 has more details). There is a natural map f from the set of popular perfect matchings in G' to the set of popular perfect matchings in G, where for any popular perfect matching M' in the marriage instance G', the many-to-one matching $f(M') = M$ is obtained by identifying all the clones of the same hospital. It is not difficult to show that M is a popular perfect matching in the hospitals/residents instance G (see Section 3).

Min-cost popular perfect matchings. There might be exponentially many popular perfect matchings in G, hence we would like to find an *optimal* one. Let us assume there is a function $\mathsf{cost} : E \to \mathbb{R}$, so the cost of a matching M is $\sum_{e \in M} \mathsf{cost}(e)$. Thus what we seek is a min-cost popular perfect matching in G. This is a natural problem in discrete optimization.

Solving the min-cost popular perfect matching problem efficiently implies efficient algorithms for several desirable popular perfect matching problems such as finding one with the highest utility when every edge (r, h) has an associated utility or one with forced/forbidden edges or one which optimizes the worst rank of a hospital that any resident gets matched to. Observe that via appropriate cost functions, we can "access" the entire set of popular perfect matchings.

A polynomial time algorithm for computing a min-cost popular maximum matching in the one-to-one setting is known [23]. However no polynomial time algorithm is currently known for this problem in the many-to-one setting. Recall the natural map f described earlier. Is f : {popular perfect matchings in G'} $\rightarrow$ {popular perfect matchings in G} surjective? If so, then we can obtain a min-cost popular perfect matching in G by computing a min-cost popular perfect matching in G'.

However there is no a priori reason to believe this map to be surjective. In fact, it is known that the natural map from {popular matchings in G'} to {popular matchings in G} is *not* surjective. The following instance $G = (R \cup H, E)$ was given in [8]: $R = \{p, q, r, s\}$ and $H = \{h, h', h''\}$ where $\mathsf{cap}(h) = 2$ and $\mathsf{cap}(h') = \mathsf{cap}(h'') = 1$. The preference orders of vertices are as follows:

$$p\colon h \succ h'' \qquad r\colon h \qquad h\colon p \succ q \succ r \succ s \qquad h''\colon p$$
$$q\colon h \succ h' \qquad s\colon h \qquad h'\colon q$$

The marriage instance G' has vertex set $R = \{p, q, r, s\}$ and $H' = \{h_1, h_2, h', h''\}$ where h_1 and h_2 are the two clones of h. The preference order of p in G' is $h_1 \succ h_2 \succ h''$ and so on.

The matching $N = \{(p, h), (q, h'), (r, h)\}$ is popular in G. The two possible realizations of N in G' are $N_1 = \{(p, h_1), (q, h'), (r, h_2)\}$ and $N_2 = \{(p, h_2), (q, h'), (r, h_1)\}$. It was shown in [8] that neither N_1 nor N_2 is popular in G'.

1. The matching $M_1 = \{(p, h''), (q, h_2), (r, h_1)\}$ is more popular than N_1 since the four vertices q, r, h_2, h'' prefer M_1 to N_1 while the three vertices p, h_1, h' prefer N_1 to M_1 and s is indifferent between M_1 and N_1. Thus N_1 is not a popular matching in G'.
2. Similarly, the matching $M_2 = \{(p, h_1), (q, h'), (s, h_2)\}$ is more popular than N_2 since the three vertices p, h_1, s prefer M_2 to N_2 while the two vertices r and h_2 prefer N_2 to M_2 and the vertices q, h', h'' are indifferent between M_2 and N_2. Thus N_2 is not a popular matching in G'. Hence neither N_1 nor N_2 is popular in G'.

- Observe that N is not a perfect matching in G. Do popular *perfect matchings* exhibit the same behavior? That is, is there a hospitals/residents instance G where the map f : {popular perfect matchings in G'} $\rightarrow$ {popular perfect matchings in G} is not surjective?

We show the following result. In contrast to the behavior of popular matchings, every popular perfect matching in G has a preimage (under f) as a popular perfect matching in G'. Hence solving the min-cost popular perfect matching in the marriage instance G' solves the min-cost popular perfect matching problem in the hospitals/residents instance G. Thus, by using the polynomial time algorithm to find a min-cost popular maximum matching in G' [23], a min-cost popular perfect matching in G can be determined in polynomial time.

▶ **Theorem 3.** *Let $G = (R \cup H, E)$ be a hospitals/residents instance with $\mathsf{cost} : E \rightarrow \mathbb{R}$, where hospitals have capacities and every vertex has a strict ranking of its neighbors. If G admits a perfect matching then a min-cost popular perfect matching in G can be computed in polynomial time.*

Many-to-many matchings. We study the min-cost popular perfect matching problem in the many-to-many setting as well. As before, every hospital h has a capacity $\mathsf{cap}(h) \geq 1$ and now every resident (or doctor) has a capacity $\mathsf{cap}(r) \geq 1$, i.e., a doctor can be associated with more than one hospital.

▶ **Definition 4.** *A matching M in $G = (R \cup H, E)$ is a subset of E such that $|M(v)| \leq \mathsf{cap}(v)$ for each $v \in R \cup H$, where $M(v) = \{u : (u, v) \in M\}$ for any $v \in R \cup H$.*

Though vertices in both R and H have capacities, since M is a set (and not a multiset), M can contain at most one copy of any edge (r, h). In order to compare any pair of matchings M and N, every vertex $v \in R \cup H$ casts up to $\mathsf{cap}(v)$ many votes as given by $\mathsf{vote}_v(M, N)$ (see Eq. (1) in Section 1).

As before, we are only interested in perfect matchings, in particular, in *popular perfect matchings* (see Definition 2). Whenever the input instance G admits a perfect matching, popular perfect matchings exist in the many-to-many setting – any popular perfect matching in the corresponding marriage instance $G' = (R' \cup H', E')$ maps to a popular perfect matching in the original many-to-many matching instance $G = (R \cup H, E)$.[2]

Now the input also consists of a function $\mathsf{cost} : E \to \mathbb{R}$ and what we seek is a min-cost popular perfect matching in G. The question again is whether the natural map f from the set of popular perfect matchings in G' to the set of popular perfect matchings in G is surjective or not. As in the many-to-one setting, we show the function f is surjective and this leads to the following result.

▶ **Theorem 5.** *Let $G = (R \cup H, E)$ be a many-to-many matching instance with $\mathsf{cost} : E \to \mathbb{R}$, where vertices have capacities and every vertex has a strict ranking of its neighbors. If G admits a perfect matching then a min-cost popular perfect matching in G can be computed in polynomial time.*

1.1 Background

Algorithmic questions in popular matchings were first studied in the domain of one-sided preference lists in bipartite graphs where it is only vertices on the left or *agents* that have preferences [2]; vertices on the right are *objects* with no preferences. Popular matchings need not always exist here and a polynomial time algorithm was given in [2] to determine if a given instance admits a popular matching and find one, if so.

The notion of popularity was proposed by Gärdenfors [16] in 1975 in the domain of two-sided strict preferences or the stable marriage problem (i.e., in the one-to-one setting) where he observed that stable matchings are popular. So popular matchings always exist in this setting. It was shown in [5, 11] that when preferences include ties (even one-sided ties), it is NP-hard to decide if a popular matching exists or not. It was shown in [19] that every stable matching in a marriage instance is a min-size popular matching. Polynomial time algorithms to find a max-size popular matching were shown in [19, 22].

As mentioned earlier, it was shown in [22] that popular maximum matchings always exist in a marriage instance. It was shown there that any stable matching in an auxiliary instance is a popular maximum matching in the given marriage instance. More recently, the converse was shown in [23], i.e., a matching M in a marriage instance is a popular maximum matching if and only if M can be realized as a stable matching in this auxiliary instance. This yielded

[2] Each vertex v in G is replaced by $\mathsf{cap}(v)$ many clones $v_1, \ldots, v_{\mathsf{cap}(v)}$ in the marriage instance G' and $v_1 \succ_{u_i} \cdots \succ_{u_i} v_{\mathsf{cap}(v)}$ for each neighbor u_i in G'.

a polynomial time algorithm for the min-cost popular maximum matching problem in a marriage instance. Though there is a polynomial time algorithm for computing a min-cost popular maximum matching in a marriage instance, finding a min-cost popular matching is NP-hard [13]. We refer to [10] for a survey on results in popular matchings in the marriage setting.

The stable matching problem has been extensively studied in the hospitals/residents setting and also in the many-to-many setting [3, 7, 18, 17, 20, 21, 30, 33] and a min-cost stable matching in a hospitals/residents instance can be computed in polynomial time [32, 34]. The notion of popularity was extended from the marriage setting to the many-to-many setting in [8] and [28], independently. A polynomial time algorithm to compute a max-size popular matching in the many-to-many setting was given in [8]. It was also shown in [8] that every stable matching in the many-to-many setting is popular; so though a rather strong definition of popularity was adopted here, popular matchings always exist. The definition of popularity considered in [28] is weaker than the one in [8]; in order to compare a pair of matchings M_0 and M_1, every hospital h uses the bijection that compares the top neighbor in $M_0(h) \setminus M_1(h)$ with the top neighbor in $M_1(h) \setminus M_0(h)$, and so on, i.e., the permutation σ in Eq. (1) is the identity permutation.

Popular matchings where vertices have capacity lower bounds have been studied in the hospitals/residents setting and in the many-to-many setting [25, 26, 27], where it is only matchings that satisfy these lower bounds that are feasible. It was shown in these works that popular feasible matchings always exist and can be computed in polynomial time.

1.2 Our Techniques

For any hospitals/residents instance $G = (R \cup H, E)$ with strict preferences, there is a corresponding marriage instance G', in other words, each vertex in G' has capacity 1. The vertex set of G' is $R \cup H'$ where $H' = \cup_{h \in H}\{h_1, h_2, \ldots, h_{\mathsf{cap}(h)}\}$.

The set of neighbors of each h_i in G' is exactly the same as the set of h's neighbors in G. So $G' = (R \cup H', E')$ is a one-to-one or marriage instance where $E' = \{(r, h_i) : (r, h) \in E$ and $1 \le i \le \mathsf{cap}(h)\}$. Every vertex in G' has a strict preference order over its neighbors.

- For $h \in H$ and $1 \le i \le \mathsf{cap}(h)$: the preference order of h_i in G' is exactly the same as h's preference order in G.
- For $r \in R$: the preference order of r in G' is the same as r's preference order in G where every neighbor h in G gets replaced by all its clones in the order $h_1 \succ h_2 \succ \cdots \succ h_{\mathsf{cap}(h)}$. So if r's preference order in G is $h \succ h'$ then r's preference order in G' is:

$$h_1 \succ h_2 \succ \cdots \succ h_{\mathsf{cap}(h)} \succ h'_1 \succ h'_2 \succ \cdots \succ h'_{\mathsf{cap}(h')}.$$

Canonical realization. Let M be a perfect (many-to-one) matching in G. For any $h \in H$, recall that $M(h)$ is the set of h's partners in M. The matching M' defined below will be called the *canonical* realization of M in the marriage instance G'.

$$M' = \cup_{h \in H}\{(r_i, h_i) : 1 \le i \le \mathsf{cap}(h) \text{ and } r_i \text{ is the } i\text{-th most preferred partner of } h \text{ in } M\}.$$

The canonical realization M' of a popular perfect matching M in the hospitals/residents instance G need *not* be a popular perfect matching in the marriage instance G'. Consider the following example where $R = \{p, q, r\}, H = \{h, h'\}, \mathsf{cap}(h) = 2$, and $\mathsf{cap}(h') = 1$.

$p\colon h \succ h'$ $\qquad$ $h\colon p \succ q \succ r$

$q\colon h \succ h'$ $\qquad$ $h'\colon p \succ q$

$r\colon h$

It is easy to check that $M = \{(p,h),(q,h'),(r,h)\}$ is a popular perfect matching in G. But its canonical realization $M' = \{(p,h_1),(q,h'),(r,h_2)\}$ is not a popular perfect matching in G' since the perfect matching $N = \{(p,h'),(q,h_2),(r,h_1)\}$ in G' is more popular than M'; the four vertices q, r, h_2, h' prefer N to M' while the two vertices p and h_1 prefer M' to N.

Stable matching problems in a hospitals/residents instance G can be easily translated to stable matching problems in the corresponding marriage instance G' since a matching M is stable in G if and only if its canonical realization M' is stable in G'. However popular matchings are more complex than stable matchings as seen in the earlier example (from [8]) where an instance G and a popular matching N were shown such that N has no preimage as a popular matching in G'.

Moreover, as seen in the example above, though M is a popular perfect matching in G, its canonical realization M' need not be a popular perfect matching in G'. Though M' is not a popular perfect matching in G', there is another realization $M'' = \{(p,h_2),(q,h'),(r,h_1)\}$ of M that is a popular perfect matching in G'. So the question that we consider here is the following:

- For any popular perfect matching M in a hospitals/residents instance G, is there always *some* realization of M that is a popular perfect matching in the marriage instance G'?

We show the answer to the above question is "yes". We use LP duality to show this. It is known that every popular perfect matching in a marriage instance G' has a *dual certificate* $\vec{\alpha}$ that certifies its optimality [23] (see Lemma 6 in Section 2). Every popular perfect matching M in a hospitals/residents instance G has a weaker dual certificate $\vec{\gamma}$ that certifies its optimality. We show how to transform the weaker certificate $\vec{\gamma}$ into a stronger certificate $\vec{\alpha}$. This involves obtaining an appropriate realization M'' of M such that M'' is a popular perfect matching in the marriage instance G' with $\vec{\alpha}$ as its dual certificate.

Popular maximum matchings. The popular maximum matching problem generalizes the popular perfect matching problem. In contrast to the above result for popular perfect matchings, it is not the case that for any popular maximum matching M in a hospitals/residents instance G, there always exists some realization of M that is a popular maximum matching in the marriage instance G'. Consider the following example where $R = \{r,s\}, H = \{h,h'\}, \mathsf{cap}(h) = 2$, and $\mathsf{cap}(h') = 1$.

$r\colon h' \succ h$ $\qquad$ $h\colon r \succ s$

$s\colon h' \succ h$ $\qquad$ $h'\colon r \succ s$

It is easy to check that $M = \{(r,h),(s,h')\}$ is a popular maximum matching in G. Note that M is not a perfect matching since $\mathsf{cap}(h) = 2$. We show below that neither the matching $M' = \{(r,h_1),(s,h')\}$ nor the matching $M'' = \{(r,h_2),(s,h')\}$ is a popular maximum matching in G'. So M has no realization as a popular maximum matching in G'.

- The matching M' is more popular than M'' since r and h_1 prefer M' to M'' while h_2 prefers M'' to M' and the vertices s and h' are indifferent between M' and M''.
- The matching $N = \{(r,h'),(s,h_2)\}$ is more popular than M' since the vertices r, h', and h_2 prefer N to M' while the vertices s and h_1 prefer M' to N.

2 Preliminaries

In this section we describe dual certificates for popular perfect matchings in a marriage instance $G_0 = (A \cup B, E_0)$ where vertices have strict preferences. For any matching M in G_0, the following edge weight function wt_M can be defined. For any $e \in E_0$:

$$\text{let } \mathsf{wt}_M(e) = \begin{cases} 2 & \text{if } e \text{ blocks } M; \\ -2 & \text{if the endpoints of } e \text{ prefer their partners in } M \text{ to each other;} \\ 0 & \text{otherwise.} \end{cases}$$

So for any edge $e = (a, b)$, $\mathsf{wt}_M(a, b)$ is the sum of the votes of a and b for each other versus their respective partners in M. Recall that $\mathsf{vote}_a(b, M(a)) \in \{0, \pm 1\}$ and similarly, $\mathsf{vote}_b(a, M(b)) \in \{0, \pm 1\}$. This makes $\mathsf{wt}_M(a, b) \in \{0, \pm 2\}$.

Every popular perfect matching M in a marriage instance $G_0 = (A \cup B, E_0)$ is a max-weight perfect matching in G_0 under the edge weight function wt_M. The linear program LP1 is the max-weight perfect matching LP in G_0 and LP2 is the dual LP.

$$\max \sum_{e \in E_0} \mathsf{wt}_M(e) \cdot x_e \qquad \text{(LP1)}$$
$$\text{s.t.} \quad \sum_{e \in \delta(v)} x_e = 1 \;\; \forall\, v \in A \cup B$$
$$x_e \geq 0 \;\; \forall\, e \in E_0.$$

$$\min \sum_{v \in A \cup B} y_v \qquad \text{(LP2)}$$
$$\text{s.t.} \quad y_a + y_b \geq \mathsf{wt}_M(a, b) \;\; \forall\, (a, b) \in E_0.$$

For any $v \in A \cup B$, note that $\delta(v)$ is the set of edges incident to v in the graph G_0. The following result (from [23]) shows that LP2 admits a special optimal solution. Here $|A \cup B| = n$ and $|A| = n_0$.

▶ **Lemma 6** ([23]). *Let M be a perfect matching in $G_0 = (A \cup B, E_0)$. Then M is a popular perfect matching in G_0 if and only if there exists $\vec{\alpha} \in \mathbb{R}^n$ that satisfies the following properties:*

1. $\alpha_a \in \{0, -2, -4, \ldots, -2(n_0 - 1)\}$ *for all* $a \in A$.
2. $\alpha_b \in \{0, 2, 4, \ldots, 2(n_0 - 1)\}$ *for all* $b \in B$.
3. $\alpha_a + \alpha_b \geq \mathsf{wt}_M(a, b)$ *for all* $(a, b) \in E_0$.
4. $\sum_{v \in A \cup B} \alpha_v = 0$.

▶ **Definition 7.** *For any popular perfect matching M in a marriage instance G_0, a vector $\vec{\alpha} \in \mathbb{R}^n$ that satisfies properties 1-4 in Lemma 6 will be called a dual certificate for M.*

3 Popular perfect matchings in a hospitals/residents instance

Let $G = (R \cup H, E)$ be a hospitals/residents instance where vertices have strict preferences. We present a characterization of popular perfect matchings in G in Section 3.1 and use this characterization to show helpful dual certificates for popular perfect matchings in Section 3.2.

3.1 A characterization of popular perfect matchings in G

Let M be a perfect matching in G. Recall the *canonical* realization M' of M (from Section 1.2) in the marriage instance $G' = (R \cup H', E')$ corresponding to G. So $M' = \cup_{h \in H} \{(r_i, h_i) : 1 \leq i \leq \mathsf{cap}(h)$, where r_i is the i-th most preferred partner of h in $M\}$.

We know that M' need not be a popular perfect matching in G'. Let $G'_M = (R \cup H', E'_M)$ be a subgraph of G' whose edge set is defined as follows:

$$E'_M = M' \cup \{(r, h_i) : (r, h) \in E \setminus M \text{ and } 1 \leq i \leq \mathsf{cap}(h)\}.$$

So for any edge $(r, h) \notin M$, all the $\mathsf{cap}(h)$ many copies $(r, h_1), \ldots, (r, h_{\mathsf{cap}(h)})$ of (r, h) are in E'_M while for each edge $(s, h) \in M$, exactly *one* edge, which is the edge in M' (say, (s, h_i)) is in E'_M. Thus $E'_M \subseteq E'$, so G'_M is indeed a subgraph of G'. Every vertex in G'_M inherits its preference order from G'.

▶ **Lemma 8.** *Let M be any perfect matching in G. Then M is a popular perfect matching in G if and only if M' is a popular perfect matching in the marriage instance G'_M.*

Proof. It follows from [23, Theorem 2] that a perfect matching M' in a marriage instance G'_M is a popular perfect matching if and only if the following condition is satisfied:

- There is no alternating cycle C with respect to M' such that $\mathsf{wt}_{M'}(C) > 0$,

where the edge weight function $\mathsf{wt}_{M'}$ takes values in $\{0, \pm 2\}$ and is analogous to the edge weight function wt_M defined at the start of Section 2.

Direction "$\Rightarrow$". We will show that if M is a popular perfect matching in G then the above condition is satisfied for its canonical realization M' in the marriage instance G'_M. Suppose not. Let C be an alternating cycle with respect to M' in G'_M such that $\mathsf{wt}_{M'}(C) > 0$. Let $N' = M' \oplus C$, i.e., N' is the symmetric difference between M' and C. We have $\mathsf{wt}_{M'}(N') = \mathsf{wt}_{M'}(C) > 0$. By identifying all the clones of the same hospital, the matching N' in G'_M becomes a perfect matching N in G.

For any hospital h with no clone in C, the two sets $M(h)$ and $N(h)$ are the same. For each hospital h with one or more clones in C, the cycle C defines a bijection between $M(h) \setminus N(h)$ and $N(h) \setminus M(h)$. Since $\mathsf{wt}_{M'}(C) > 0$, this means for each h in C, there is a way of comparing elements in $M(h) \setminus N(h)$ with those in $N(h) \setminus M(h)$ such that summed over all vertices in C, the votes in favor of N outnumber the votes in favor of M. Thus, adding up the votes of all vertices in C for M versus N (as per the bijection defined by C), the total number of votes for M is less than the total number of votes for N. Hence it follows from the definition of the function vote (see Eq.(1)) that $\sum_{v \in C} \mathsf{vote}_v(M, N) < 0$. So $\Delta(M, N) < 0$, which contradicts the fact that M is a popular perfect matching in G.

Direction "$\Leftarrow$". Let M' be a popular perfect matching in G'_M; it follows from the characterization of popular perfect matchings in a marriage instance (given above) that there is no alternating cycle C with respect to M' in G'_M such that $\mathsf{wt}_{M'}(C) > 0$. Let N be any perfect matching in G. A realization N' of N in G'_M can be obtained as follows:

- For every edge $(r, h) \in N \cap M$: the edge (r, h_i) is in N' where $(r, h_i) \in M'$.
- For every $(r, h) \in N \setminus M$: in the evaluation of $\mathsf{vote}_h(M, N)$, while comparing the set $M(h) \setminus N(h)$ with the set $N(h) \setminus M(h)$, let s be the resident that h compares r with. So the matching M' contains the edge (s, h_j) for some $j \in \{1, \ldots, \mathsf{cap}(h)\}$; the edge (r, h_j) will be included in N'.

Observe that $\sum_{v \in R \cup H} \mathsf{vote}_v(M, N) = -\sum_C \mathsf{wt}_{M'}(C)$, where the sum is over all alternating cycles C in $M' \oplus N'$. For each alternating cycle $C \in M' \oplus N'$, we have $\mathsf{wt}_{M'}(C) \leq 0$, thus $\sum_{v \in R \cup H} \mathsf{vote}_v(M, N) \geq 0$, in other words, $\Delta(M, N) \geq 0$. Since this holds for any perfect matching N, it follows that M is a popular perfect matching in G. ◀

3.2 Constructing a helpful dual certificate

We know that for a perfect matching M to be a popular perfect matching in G, it is not necessary that M' is a popular perfect matching in the marriage instance G'. It suffices for the matching M' to be a popular perfect matching in the subgraph G'_M of G' (by Lemma 8). Observe that being a popular perfect matching in the subgraph G'_M is a more relaxed condition than being a popular perfect matching in G' since the *edge covering constraints* in Lemma 6, i.e., the constraints in property 3, have to be satisfied only for the edges in G'_M rather than all the edges in G' (recall that $E' \supseteq E'_M$).

Recall that there is a function $\mathsf{cost} : E \to \mathbb{R}$ and our goal is to find a *min-cost* popular perfect matching in G. Though we know how to find a min-cost popular perfect matching N in a marriage instance, we do not know in which marriage instance we should run this algorithm (since G'_N depends on the matching N that we seek).

Let M be any popular perfect matching in G. Since the canonical realization M' of M is a popular perfect matching in $G'_M = (R \cup H', E'_M)$, there is a vector $\vec{\gamma} \in \mathbb{R}^n$ (see Lemma 6) that satisfies the following properties where $|R \cup H'| = n$ and $|R| = n_0$.

1. $\gamma_r \in \{0, -2, -4, \ldots, -2(n_0 - 1)\}$ for all $r \in R$.
2. $\gamma_{h_i} \in \{0, 2, 4, \ldots, 2(n_0 - 1)\}$ for all $h_i \in H'$.
3. $\gamma_r + \gamma_{h_i} \ge \mathsf{wt}_{M'}(r, h_i)$ for all $(r, h_i) \in E'_M$.
4. $\sum_{v \in R \cup H'} \gamma_v = 0$.

Let $\vec{\gamma}$ be a dual certificate for M' that minimizes the sum $\sum_{h_i \in H'} \gamma_{h_i}$. The following lemma will be very useful to us.

▶ **Lemma 9.** *For any two clones h_i and h_j of the same hospital h, we have $\gamma_{h_i} \le \gamma_{h_j} + 2$.*

Proof. Observe that except for their partners in M', the neighborhoods in G'_M of the two clones h_i and h_j of h are identical. Consider any $(r, h_i) \in E'_M$ such that $(r, h_i) \notin M'$. So $(r, h'_t) \in M'$ for some $h' \ne h$. We have:

$$\begin{aligned}
\mathsf{wt}_{M'}(r, h_i) &\le \mathsf{vote}_r(h_i, M'(r)) + \mathsf{vote}_h(r, M'(h_i)) && (2)\\
&= \mathsf{vote}_r(h_j, M'(r)) + \mathsf{vote}_h(r, M'(h_i)) && (3)\\
&\le \mathsf{vote}_r(h_j, M'(r)) + \mathsf{vote}_h(r, M'(h_j)) + 2 && (4)\\
&= \mathsf{wt}_{M'}(r, h_j) + 2 && (5)\\
&\le \gamma_r + \gamma_{h_j} + 2. && (6)
\end{aligned}$$

In the third constraint, $\mathsf{vote}_h(r, M'(h_i)) \le \mathsf{vote}_h(r, M'(h_j)) + 2$ since $\mathsf{vote}_h(r, M'(h_i)) \le 1$ and $\mathsf{vote}_h(r, M'(h_j)) \ge -1$.

Suppose $\gamma_{h_i} > \gamma_{h_j} + 2$. Let $(s, h_i) \in M'$. Since M' and $\vec{\gamma}$ are optimal solutions to LP1 and LP2 respectively, we have $\gamma_s + \gamma_{h_i} = \mathsf{wt}_{M'}(s, h_i) = 0$ by complementary slackness. Let us update $\vec{\gamma}$ to $\vec{\gamma}'$ as follows:

- $\gamma'_{h_i} = \gamma_{h_j} + 2$ and $\gamma'_s = -\gamma'_{h_i}$.
- $\gamma'_v = \gamma_v$ for all other vertices v in $R \cup H'$.

Observe that $\sum_{v \in R \cup H'} \gamma'_v = 0$. Thus $\vec{\gamma}'$ satisfies property 4 given in Lemma 6.

Constraints (2)-(6) tell us that $\gamma'_r + \gamma'_{h_i} = \gamma_r + \gamma_{h_j} + 2 \ge \mathsf{wt}_{M'}(r, h_i)$ for any $r \ne s$. Since $\gamma'_s + \gamma'_{h_i} = 0 = \mathsf{wt}_{M'}(s, h_i)$, we have $\gamma'_r + \gamma'_{h_i} \ge \mathsf{wt}_{M'}(r, h_i)$ for all $r \in R$. Thus all edges in E'_M incident to h_i are covered by $\vec{\gamma}'$.

Since $\gamma'_{h_i} = \gamma_{h_j} + 2 < \gamma_{h_i}$, our update ensures that $\gamma'_s \ge \gamma_s$. Because $\gamma'_v = \gamma_v$ for all vertices v other than s and h_i, we have $\gamma'_p + \gamma'_{q_j} \ge \mathsf{wt}_{M'}(p, q_j)$ for every (p, q_j) in E'_M. Thus $\vec{\gamma}'$ satisfies property 3 given in Lemma 6.

So properties 1-4 given in Lemma 6 are satisfied by $\vec{\gamma}'$. Thus $\vec{\gamma}'$ is a dual certificate for M' in G'_M. Moreover, $\sum_{h_i \in H'} \gamma'_{h_i} < \sum_{h_i \in H'} \gamma_{h_i}$, contradicting the choice of $\vec{\gamma}$ as a dual certificate for M' that minimizes this sum. Hence it has to be the case that $\gamma_{h_i} \le \gamma_{h_j} + 2$. ◀

If $\gamma_{h_i} = 2\ell$ then we will say h_i is in level ℓ in $\vec{\gamma}$. Lemma 9 tells us that all of $h_1, \ldots, h_{\mathsf{cap}(h)}$ are either in the same level (say, ℓ) or in two successive levels (say, ℓ and $\ell+1$) in $\vec{\gamma}$. Let $r_1, \ldots, r_k$ (where $\mathsf{cap}(h) = k$) be the partners of h in M where $r_1 \succ_h \cdots \succ_h r_k$.

▶ **Lemma 10.** *Suppose k' clones of h are in level ℓ and $k - k'$ clones of h are in level $\ell+1$. Then the k' clones of h in level ℓ have to be matched in M' to $r_1, \ldots, r_{k'}$. That is, $r_1, \ldots, r_{k'}$ are in level ℓ and $r_{k'+1}, \ldots, r_k$ are in level $\ell+1$ in $\vec{\gamma}$.*

Proof. Suppose at least one of $r_1, \ldots, r_{k'}$ is not in level ℓ. Then r_i, for some $i \le k'$, is in level $\ell+1$ and r_j, for some $j > k'$, is in level ℓ. So h_i is in level $\ell+1$ while h_j is in level ℓ.

Observe that for any neighbor s of h in G such that $s \notin M(h)$, we have $\mathsf{wt}_{M'}(s, h_i) \le \mathsf{wt}_{M'}(s, h_j)$. This is because $\mathsf{vote}_s(h_i, M) = \mathsf{vote}_s(h_j, M)$ while $\mathsf{vote}_h(s, r_i) \le \mathsf{vote}_h(s, r_j)$ since h prefers r_i to r_j. Since we have $\gamma_s + \gamma_{h_j} \ge \mathsf{wt}_{M'}(s, h_j) \ge \mathsf{wt}_{M'}(s, h_i)$, let us update $\gamma'_{h_i} = 2\ell$ and $\gamma'_{r_i} = -2\ell$ (so $\gamma'_{r_i} > \gamma_{r_i}$). For any vertex v other than r_i and h_i, let $\gamma'_v = \gamma_v$.

It is easy to see that $\vec{\gamma}'$ is a dual certificate for M' in G'_M and $\sum_{h_i \in H'} \gamma'_{h_i} < \sum_{h_i \in H'} \gamma_{h_i}$, contradicting the choice of $\vec{\gamma}$ as a dual certificate for M that minimizes this sum. Hence it has to be the case that $r_1, \ldots, r_{k'}$ are in level ℓ and $r_{k'+1}, \ldots, r_k$ are in level $\ell+1$. ◀

We are now ready to prove the main technical lemma in this section. For any $h \in H$, let $r_1, \ldots, r_k$ be all the partners of h in M (so $k = \mathsf{cap}(h)$) and $r_1 \succ_h \cdots \succ_h r_k$.

▶ **Lemma 11.** *Let M be any popular perfect matching in G. For each $h \in H$, there exists an appropriate permutation π_h on $\{1, \ldots, k\}$ (where $k = \mathsf{cap}(h)$) such that $M'' = \cup_{h \in H}\{(r_i, h_{\pi_h(i)}) : 1 \le i \le k\}$ is a popular perfect matching in G'.*

Proof. Let $h \in H$ and let $\mathsf{cap}(h) = k$. Consider the following two cases.

Case 1. All the clones of h are in the same level $\ell \in \{0, 1, \ldots, n_0 - 1\}$ in $\vec{\gamma}$.

This is the easy case – we will set $\pi_h : \{1, \ldots, k\} \to \{1, \ldots, k\}$ to be the identity function. So $(r_1, h_1), \ldots, (r_k, h_k)$ are in M''. For all $1 \le i \le k$: $\gamma_{r_i} = -2\ell$ and $\gamma_{h_i} = 2\ell$.

Case 2. All the clones of h are *not* in the same level.

So by Lemma 9, all the clones $h_1, \ldots, h_k$ are in two successive levels in $\vec{\gamma}$. We are now ready to define the permutation π_h or equivalently, the partners of $h_1, \ldots, h_k$ in the matching M''. The vertices $r_1, \ldots, r_k$ will be matched in M'' to $h_1, \ldots, h_k$ as follows:

- for $i \in \{1, \ldots, k'\}$, the vertex r_i is matched to $h_{k-k'+i}$;
- for $i \in \{k'+1, \ldots, k\}$, the vertex r_i is matched to $h_{i-k'}$.

In more detail, we know from Lemma 10 that $r_1, \ldots, r_{k'}$ are in level ℓ and $r_{k'+1}, \ldots, r_k$ are in level $\ell+1$. The residents $r_1, \ldots, r_{k'}$ are matched in M'' to $h_{k-k'+1}, \ldots, h_k$, respectively and the residents $r_{k'+1}, \ldots, r_k$ in level $\ell+1$ are matched in M'' to $h_1, \ldots, h_{k-k'}$, respectively. Thus the matching M'' rearranges the clones of h so that the following holds:

- $h_1, \ldots, h_{k-k'}$ are placed in level $\ell+1$.
- $h_{k-k'+1}, \ldots, h_k$ are placed in level ℓ.

Observe that $M'' = \cup_{h \in H}\{(r_i, h_{\pi_h(i)}) : 1 \le i \le k\}$ where in case 1 (i.e., all the clones of h are in the same level), $\pi_h(i) = i$ for all $i \in \{1, \ldots, k\}$ and for case 2, π_h is defined as follows:

$$\pi_h(i) = \begin{cases} k - k' + i & \text{if } 1 \le i \le k'; \\ i - k' & \text{if } k'+1 \le i \le k. \end{cases}$$

The entire matching M'' is thus defined. We claim that M'' is the realization of M that we seek.

▷ Claim 12. M'' is a popular perfect matching in G'.

We will prove Claim 12 below. This finishes the proof of Lemma 11. ◀

Proof of Claim 12. We will prove M'' to be a popular perfect matching in G' by defining a dual certificate $\vec{\alpha}$ for M'' in G'. Recall that $\vec{\gamma}$ is a dual certificate for the canonical image M' to be a popular perfect matching in G'_M.

1. For any $r \in R$: set $\alpha_r = \gamma_r$.
2. For any $h \in H$ and $i \in \{1, \ldots, k\}$: set $\alpha_{h_i} = 2t$ where t is h_i's *level* as defined in the proof of Lemma 11.

So in case 1 of the proof of Lemma 11, when all the clones of h are in the same level ℓ, we have $\alpha_{h_i} = 2\ell$ for every $i \in \{1, \ldots, \mathsf{cap}(h)\}$. In case 2 of the proof of Lemma 11, the vertices $h_1, \ldots, h_{k-k'}$ are in level $\ell + 1$ and the vertices $h_{k-k'+1}, \ldots, h_k$ are in level ℓ where $k = \mathsf{cap}(h)$; so $\alpha_{h_i} = 2(\ell + 1)$ for $1 \le i \le k - k'$ and $\alpha_{h_i} = 2\ell$ for $k - k' + 1 \le i \le k$.

Let $h \in H$. We know the edges $(r_1, h_1), \ldots, (r_k, h_k)$ are in M'. We will now show that $\alpha_{r_i} + \alpha_{h_j} = \mathsf{wt}_{M''}(r_i, h_j)$ for all $i, j \in \{1, \ldots, k\}$.

This is immediate to check for case 1 in the proof of Lemma 11 since $\alpha_{r_i} = -2\ell$ and $\alpha_{h_j} = 2\ell$ for all $i, j \in \{1, \ldots, k\}$. Recall that $r_1 \succ_h \cdots \succ_h r_k$ and $h_1 \succ_r \cdots \succ_r h_k$ for all r adjacent to h in G. So we have $\alpha_{r_i} + \alpha_{h_j} = 0 = \mathsf{wt}_{M''}(r_i, h_j)$ for all $i, j \in \{1, \ldots, k\}$.

We will now check the equality $\alpha_{r_i} + \alpha_{h_j} = \mathsf{wt}_{M''}(r_i, h_j)$ for case 2 in the proof of Lemma 11, so $h_1, \ldots, h_{k-k'}$ are in level $\ell + 1$ and $h_{k-k'+1}, \ldots, h_k$ are in level ℓ.

- For $i \in \{1, \ldots, k'\}$ and $j \in \{k - k' + 1, \ldots, k\}$, observe that $\mathsf{wt}_{M''}(r_i, h_j) = 0$. Since $\alpha_{r_i} = -2\ell$ and $\alpha_{h_j} = 2\ell$, we have $\alpha_{r_i} + \alpha_{h_j} = 0 = \mathsf{wt}_{M''}(r_i, h_j)$ for all these pairs (i, j).
- Similarly, for $i \in \{k' + 1, \ldots, k\}$ and $j \in \{1, \ldots, k - k'\}$, we have $\mathsf{wt}_{M''}(r_i, h_j) = 0$. Since $\alpha_{r_i} = -2(\ell + 1)$ and $\alpha_{h_j} = 2(\ell + 1)$, we have $\alpha_{r_i} + \alpha_{h_j} = 0 = \mathsf{wt}_{M''}(r_i, h_j)$ for all these pairs (i, j).
- Let us now consider a pair (r_i, h_j) where $i \in \{1, \ldots, k'\}$ and $j \in \{1, \ldots, k - k'\}$. We have $\mathsf{wt}_{M''}(r_i, h_j) = 2$ for such an (i, j). Since $\alpha_{r_i} = -2\ell$ and $\alpha_{h_j} = 2(\ell + 1)$, we have $\alpha_{r_i} + \alpha_{h_j} = 2 = \mathsf{wt}_{M''}(r_i, h_j)$ for all these pairs (i, j).
- Finally, let us consider a pair (r_i, h_j) where $i \in \{k' + 1, \ldots, k\}$ and $j \in \{k - k' + 1, \ldots, k\}$. We have $\mathsf{wt}_{M''}(r_i, h_j) = -2$ for such an (i, j). Since $\alpha_{r_i} = -2(\ell + 1)$ and $\alpha_{h_j} = 2\ell$, we have $\alpha_{r_i} + \alpha_{h_j} = -2 = \mathsf{wt}_{M''}(r_i, h_j)$ for all these pairs (i, j).

Recall that the difference between the graphs G'_M and G' is that for every $h \in H$, the edges (r_i, h_j) for $i \neq j$ are missing in G'_M, where the edges $(r_1, h_1), \ldots, (r_k, h_k)$ are in M'. And we just have checked all the edges (r_i, h_j) for all $i, j \in \{1, \ldots, k\}$ are covered by $\vec{\alpha}$.

We will now show that all the *non-matching* edges are covered by $\vec{\alpha}$ as well. That is, for any $(r', h) \in E \setminus M$, we need to show that $\alpha_{r'} + \alpha_{h_j} \ge \mathsf{wt}_{M''}(r', h_j)$ for all $j \in \{1, \ldots, k\}$.

For any $j \in \{1, \ldots, k\}$, let $j = \pi_h(i)$. Observe that $\mathsf{wt}_{M'}(r', h_i) = \mathsf{wt}_{M''}(r', h_j)$. This is because $M'(h_i) = r_i = M''(h_j)$ and $\mathsf{vote}_{r'}(M'', h_j) = \mathsf{vote}_{r'}(M', h_i)$ since r' is matched in both M' and M'' to clones of the same hospital h'.

We have $\alpha_{h_j} = -\alpha_{r_i} = -\gamma_{r_i} = \gamma_{h_i}$ since $(r_i, h_j) \in M''$. So $\gamma_{r'} + \gamma_{h_i} \ge \mathsf{wt}_{M'}(r, h_i)$ implies that $\alpha_{r'} + \alpha_{h_j} \ge \mathsf{wt}_{M''}(r', h_j)$. Thus every non-matching edge (r', h_j) is also covered by $\vec{\alpha}$.

Since $\alpha_s + \alpha_{h_i} = 0$ for any edge $(s, h_i) \in M''$, we have $\sum_{v \in R \cup H'} \alpha_v = 0$. Hence properties 1-4 for dual certificates are satisfied by $\vec{\alpha}$; thus $\vec{\alpha}$ is a dual certificate for M'' in G'. In other words, M'' is a popular perfect matching in G' (by Lemma 6). ◁

Our algorithm. It is now straightforward to show a polynomial time algorithm to find a min-cost popular perfect matching in a hospitals/residents instance $G = (R \cup H, E)$.

1. Compute the corresponding marriage instance $G' = (R \cup H', E')$.
2. Find a min-cost popular perfect matching N' in G' using the algorithm in [23] for marriage instances.
3. Return the corresponding matching N in G by identifying all clones of the same hospital.

For any popular perfect matching M in G, we know there is some realization M'' such that M'' is a popular perfect matching in G' (by Lemma 11); also $\mathsf{cost}(M) = \mathsf{cost}(M'')$. Thus solving the min-cost popular perfect matching problem in the marriage instance G' solves the min-cost popular perfect matching problem in the hospitals/residents instance G. Hence the matching N returned by the above algorithm is a min-cost popular perfect matching in G.

Thus we can conclude Theorem 3 stated in Section 1. We restate this theorem below.

▶ **Theorem 3.** *Let $G = (R \cup H, E)$ be a hospitals/residents instance with $\mathsf{cost} : E \to \mathbb{R}$, where hospitals have capacities and every vertex has a strict ranking of its neighbors. If G admits a perfect matching then a min-cost popular perfect matching in G can be computed in polynomial time.*

4 The many-to-many setting

In this section $G = (R \cup H, E)$ is a many-to-many matching instance. So every vertex $v \in R \cup H$ has a capacity $\mathsf{cap}(v) \geq 1$ associated with it.

Let M be any popular perfect matching in G. Our goal is to show that M can be realized as a popular perfect matching in the marriage instance $G' = (R' \cup H', E')$ where each vertex v in G is replaced by $\mathsf{cap}(v)$ many clones $v_1, \ldots, v_{\mathsf{cap}(v)}$ and every edge $(r, h) \in E$ is replaced by $\mathsf{cap}(r) \cdot \mathsf{cap}(h)$ many edges (r_i, h_j) in E' where $1 \leq i \leq \mathsf{cap}(r)$ and $1 \leq j \leq \mathsf{cap}(h)$.

Each vertex in G' has capacity 1 and it has a strict preference order over its neighbors as described below.

- For $v \in R \cup H$ and $i \in \{1, \ldots, \mathsf{cap}(v)\}$: the preference order of v_i in G' is the same as v's preference order in G where every neighbor u in G gets replaced by all its clones in the order $u_1 \succ \cdots \succ u_{\mathsf{cap}(u)}$.

As done in Section 3, let us first obtain a one-to-one matching M' from M as follows.

- For each edge $(r, h) \in M$, we will choose an index $i \in \{1, \ldots, \mathsf{cap}(r)\}$ and an index $j \in \{1, \ldots, \mathsf{cap}(h)\}$ such that the indices i and j have not been chosen so far and include the edge (r_i, h_j) in M'. Thus M' is a perfect matching in the marriage instance G'.

In fact, as done in Section 3, we could define the matching M' more carefully so that for every $v \in R \cup H$, we have $M'(v_1) \succ_{v_i} \cdots \succ_{v_i} M'(v_{\mathsf{cap}(v)})$ where v_i is any clone of v. As we know, the matching M' need not be a popular perfect matching in G'. Hence let us consider the subgraph $G'_M = (R' \cup H', E'_M)$ of G' whose edge set is as follows:

$$E'_M = M' \cup \{(r_i, h_j) : (r, h) \in E \setminus M \text{ and } 1 \leq i \leq \mathsf{cap}(r),\ 1 \leq j \leq \mathsf{cap}(h)\}.$$

For any $(r, h) \notin M$, all its $\mathsf{cap}(r) \cdot \mathsf{cap}(h)$ many copies $(r_1, h_1), \ldots, (r_{\mathsf{cap}(r)}, h_{\mathsf{cap}(h)})$ are in E'_M while for each edge $(s, h) \in M$, exactly *one* edge, which is the edge in M' – say, (s_i, h_j) – is in E'_M. Thus $E'_M \subseteq E'$, so G'_M is a subgraph of G'.

▶ **Lemma 13.** *Let M be any perfect matching in G. Then M is a popular perfect matching in G if and only if M' is a popular perfect matching in the marriage instance G'_M.*

The proof of the above lemma is analogous to the proof of Lemma 8. Since M' is a popular perfect matching in G'_M, it admits a vector $\vec{\gamma} \in \mathbb{R}^n$ (see Lemma 6) that satisfies the following properties where $|R' \cup H'| = n$ and $|R'| = n_0$:

1. $\gamma_{r_i} \in \{0, -2, -4, \ldots, -2(n_0 - 1)\}$ for all $r_i \in R'$.
2. $\gamma_{h_j} \in \{0, 2, 4, \ldots, 2(n_0 - 1)\}$ for all $h_j \in H'$.
3. $\gamma_{r_i} + \gamma_{h_j} \geq \mathsf{wt}_{M'}(r_i, h_j)$ for all $(r_i, h_j) \in E'_M$.
4. $\sum_{v \in R' \cup H'} \gamma_v = 0$.

Let $\vec{\gamma}$ be a dual certificate for M' that minimizes the value of $\sum_{h_i \in H'} \gamma_{h_i}$. The proofs of Lemmas 14–16 are the same as the proofs of Lemma 9–11, respectively.

▶ **Lemma 14.** *For any two clones h_i and h_j of the same hospital h, we have $\gamma_{h_i} \leq \gamma_{h_j} + 2$.*

If $\gamma_{h_i} = 2\ell$ then we will say h_i is in level ℓ in $\vec{\gamma}$. Lemma 14 tells us that all of $h_1, \ldots, h_{\mathsf{cap}(h)}$ are either in the same level (say, ℓ) or in two successive levels (say, ℓ and $\ell + 1$) in $\vec{\gamma}$. Let $r^1_{i_1}, \ldots, r^k_{i_k}$ (where $k = \mathsf{cap}(h)$) be the partners of h in M where $r^1 \succ_h \cdots \succ_h r^k$.

▶ **Lemma 15.** *Suppose k' clones of h are in level ℓ and $k - k'$ clones of h are in level $\ell + 1$. Then the k' clones of h in level ℓ have to be matched in M' to $r^1_{i_1}, \ldots, r^{k'}_{i_{k'}}$. That is, $r^1_{i_1}, \ldots, r^{k'}_{i_{k'}}$ are in level ℓ and $r^{k'+1}_{i_{k'+1}}, \ldots, r^k_{i_k}$ are in level $\ell + 1$ in $\vec{\gamma}$.*

Let us now define a subgraph G''_M of G'. The edge set E''_M of G''_M has $\mathsf{cap}(h)$ many copies $(r_i, h_{j'})$ of each edge (r_i, h_j) in M', where $1 \leq j' \leq \mathsf{cap}(h)$, and it has all the $\mathsf{cap}(r) \cdot \mathsf{cap}(h)$ copies of each $(r, h) \in E \setminus M$.

Thus $E''_M = E'_M \cup \{(r_i, h_{j'}) : (r_i, h_j) \in M' \text{ and } 1 \leq j' \leq \mathsf{cap}(h)\}$.

▶ **Lemma 16.** *Let M be any popular perfect matching in G. For each $h \in H$, there exists an appropriate permutation π_h on $\{1, \ldots, k\}$ (where $k = \mathsf{cap}(h)$) such that $M'' = \cup_{h \in H}\{(r^j_{i_j}, h_{\pi_h(j)}) : 1 \leq j \leq k\}$ is a popular perfect matching in G''_M where $r^1_{i_1}, \ldots, r^k_{i_k}$ are h's partners in M.*

Thus Lemma 16 tells us that M has a realization as a popular perfect matching M'' in the marriage instance G''_M. However this marriage instance G''_M is not the desired marriage instance G' since each edge $(r_i, h_j) \in M'$ has only $\mathsf{cap}(h)$ many copies $(r_i, h_{j'})$ in G''_M, where $1 \leq j' \leq \mathsf{cap}(h)$. Note that the marriage instance G' includes *all* the $\mathsf{cap}(r) \cdot \mathsf{cap}(h)$ many copies $(r_{i'}, h_{j'})$ of each edge (r, h) in G, where $1 \leq i' \leq \mathsf{cap}(r)$ and $1 \leq j' \leq \mathsf{cap}(h)$.

A useful hospitals/residents instance. For every resident $r \in R$, let us now identify all its clones $r_1, \ldots, r_{\mathsf{cap}(r)}$ in G''_M. Let $\widetilde{G}$ denote the resulting graph.

It will be convenient to swap the two sides of $\widetilde{G}$ so that $\widetilde{G}$ becomes a *many-to-one* matching instance. Thus $\widetilde{G} = (H' \cup R, \widetilde{E})$ where the clones of all hospitals are on the left and residents are on the right. Each vertex $h_i \in H'$ on the left has capacity 1 and each vertex $r \in R$ on the right has $\mathsf{cap}(r) \geq 1$.

The matching M'' in the marriage instance G''_M (see Lemma 16) becomes a many-to-one matching $\widetilde{M}$ in $\widetilde{G}$. We know from Lemma 16 that M'' is a popular perfect matching in G''_M, hence $\widetilde{M}$ is a popular perfect matching in $\widetilde{G}$.

By Lemma 11, the many-to-one popular perfect matching $\widetilde{M}$ in $\widetilde{G}$ has a realization M^* as a popular perfect matching in the corresponding marriage instance. Observe that for every edge (r, h) in G, all the edges (h_i, r) are in $\widetilde{G}$, where $1 \leq i \leq \mathsf{cap}(h)$. So the marriage instance corresponding to $\widetilde{G}$ is $G' = (H' \cup R', E')$ which includes for every edge (r, h) in G, all its $\mathsf{cap}(h) \cdot \mathsf{cap}(r)$ copies (h_i, r_j), where $1 \leq i \leq \mathsf{cap}(h)$ and $1 \leq j \leq \mathsf{cap}(r)$.

Thus it follows that M has a realization M^* as a popular perfect matching in the marriage instance $G' = (R' \cup H', E')$. So we can conclude the following lemma.

▶ **Lemma 17.** *Let M be a popular perfect matching in a many-to-many matching instance $G = (R \cup H, E)$ where vertices have capacities and strict preferences. Then M has a realization M^* as a popular perfect matching in the corresponding marriage instance $G' = (R' \cup H', E')$.*

Hence solving the min-cost popular perfect matching problem in the marriage instance G' solves the min-cost popular perfect matching problem in the many-to-many matching instance G. Since $\mathsf{cap}(r) \le |H|$ for every $r \in R$ and $\mathsf{cap}(h) \le |R|$ for every $h \in H$, the number of vertices in G' is $O(|R| \cdot |H|)$. Thus the size of G' is polynomial in the size of G.

As we know, a min-cost popular perfect matching in G' can be computed in polynomial time [23]. Thus Theorem 5 stated in Section 1 follows. We restate this theorem below.

▶ **Theorem 5.** *Let $G = (R \cup H, E)$ be a many-to-many matching instance with $\mathsf{cost} : E \to \mathbb{R}$, where vertices have capacities and every vertex has a strict ranking of its neighbors. If G admits a perfect matching then a min-cost popular perfect matching in G can be computed in polynomial time.*

Concluding remarks. Given a hospitals/residents instance $G = (R \cup H, E)$ where each vertex has a strict preference order over its neighbors and every edge has an associated cost, we showed that if G admits a perfect matching, then a min-cost popular perfect matching in G can be computed in polynomial time. This result generalizes to the many-to-many setting.

Our method of reducing the min-cost popular perfect matching problem in a hospitals/residents instance G to the min-cost popular perfect matching problem in the corresponding marriage instance G' does not work for the min-cost popular *maximum* matching problem. The computational complexity of the min-cost popular maximum matching problem in a hospitals/residents instance is open.

References

1. A. Abdulkadiroğlu and T. Sönmez. School choice: a mechanism design approach. *American Economic Review*, 93(3):729–747, 2003.
2. D. J. Abraham, R. W. Irving, T. Kavitha, and K. Mehlhorn. Popular matchings. *SIAM Journal on Computing*, 37(4):1030–1045, 2007.
3. G. Askalidis, N. Immorlica, A. Kwanashie, D. Manlove, and E. Pountourakis. Socially stable matchings in the hospitals/residents problem. In *Proceedings of the 13th International Symposium on Algorithms and Data Structures, WADS*, pages 85–96, 2013.
4. S. Baswana, P. P. Chakrabarti, S. Chandran, Y. Kanoria, and U. Patange. Centralized admissions for engineering colleges in India. *INFORMS Journal on Applied Analytics*, 49(5):338–354, 2019.
5. P. Biro, R. W. Irving, and D. F. Manlove. Popular matchings in the marriage and roommates problems. In *Proceedings of the 7th International Conference on Algorithms and Complexity, CIAC*, pages 97–108, 2010.
6. P. Biro, D. F. Manlove, and S. Mittal. Size versus stability in the marriage problem. *Theoretical Computer Science*, 411:1828–1841, 2010.
7. C. Blair. The lattice structure of the set of stable matchings with multiple partners. *Mathematics of Operations Research*, 13:619–628, 1988.
8. F. Brandl and T. Kavitha. Two problems in max-size popular matchings. *Algorithmica*, 81(7):2738–2764, 2019.
9. Canadian Resident Matching Service. How the matching algorithm works. `http://carms.ca/algorithm.htm`.
10. Á. Cseh. Popular matchings. Trends in Computational Social Choice, Ulle Endriss (ed.), 2017.
11. Á. Cseh, C.-C. Huang, and T. Kavitha. Popular matchings with two-sided preferences and one-sided ties. *SIAM Journal on Discrete Mathematics*, 31(4):2348–2377, 2017.

12 Nicolas de Condorcet. *Essai sur l'application de l'analyse à la probabilité des décisions rendues à la pluralité des voix*. L'Imprimerie Royale, 1785.

13 Y. Faenza, T. Kavitha, V. Powers, and X. Zhang. Popular matchings and limits to tractability. In *Proceedings of the 30th ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 2790–2809, 2019.

14 D. Gale and L.S. Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 69(1):9–15, 1962.

15 D. Gale and M. Sotomayor. Some remarks on the stable matching problem. *Discrete Applied Mathematics*, 11:223–232, 1985.

16 P. Gärdenfors. Match making: assignments based on bilateral preferences. *Behavioural Science*, 20:166–173, 1975.

17 K. Hamada, K. Iwama, and Shuichi Miyazaki. The hospitals/residents problem with lower quotas. *Algorithmica*, 74(1):440–465, 2016.

18 C.-C. Huang. Classified stable matching. In *Proceedings of the 21st Annual ACM-SIAM Symposium on Discrete Algorithms, SODA*, pages 1235–1253, 2010.

19 C.-C. Huang and T. Kavitha. Popular matchings in the stable marriage problem. *Information and Computation*, 222:180–194, 2013.

20 R. W. Irving, D. F. Manlove, and S. Scott. The hospitals/residents problem with ties. In *Proceedings of the 7th Scandinavian Workshop on Algorithm Theory, SWAT*, pages 259–271, 2000.

21 R. W. Irving, D. F. Manlove, and S. Scott. Strong stability in the hospitals/residents problem. In *Proceedings of the 20th Annual Symposium on Theoretical Aspects of Computer Science, STACS*, pages 439–450, 2003.

22 T. Kavitha. A size-popularity tradeoff in the stable marriage problem. *SIAM Journal on Computing*, 43(1):52–71, 2014.

23 T. Kavitha. Maximum matchings and popularity. In *Proceedings of the 48th International Colloquium on Automata, Languages, and Programming*, volume 198 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 85:1–85:21, 2021.

24 S. Merrill and B. Grofman. *A unified theory of voting: directional and proximity spatial models*. Cambridge University Press, 1999.

25 M. Nasre and P. Nimbhorkar. Popular matchings with lower quotas. In *Proceedings of the 37th IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS*, pages 44:1–44:15, 2017.

26 M. Nasre, P. Nimbhorkar, K. Ranjan, and A. Sarkar. Popular matchings in the hospital-residents problem with two-sided lower quotas. In *Proceedings of the 41st IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS*, pages 30:1–30:21, 2021.

27 M. Nasre, P. Nimbhorkar, K. Ranjan, and A. Sarkar. Popular critical matchings in the many-to-many setting, 2022. `arXiv:2206.12394`.

28 M. Nasre and A. Rawat. Popularity in the generalized hospital residents setting. In *Proceedings of the 12th International Computer Science Symposium in Russia, CSR*, pages 245–259, 2017.

29 National Resident Matching Program. Why the Match? `http://www.nrmp.org/whythematch.pdf`.

30 A. E. Roth. Stability and polarization of interest in job matching. *Econometrica*, 53:47–57, 1984.

31 A. E. Roth. On the allocation of residents to rural hospitals: A general property of two-sided matching markets. *Journal of Political Economy*, 54(2):425–427, 1986.

32 U. G. Rothblum. Characterization of stable matchings as extreme points of a polytope. *Mathematical Programming*, 54:57–67, 1992.

33 M. Sotomayor. Three remarks on the many-to-many stable matching problem. *Mathematical Social Sciences*, 38:55–70, 1999.

34 C.-P. Teo and J. Sethuraman. The geometry of fractional stable matchings and its applications. *Mathematics of Operations Research*, 23(4):874–891, 1998.