Popular Solutions for Optimal Matchings

Telikepalli Kavitha

Tata Institute of Fundamental Research, Mumbai, India
kavitha@tifr.res.in
https://orcid.org/0000-0003-2619-6606

Abstract. Let G be a bipartite graph where every vertex has a strict preference order over its neighbors. The preferences of a vertex over its neighbors extend naturally to preferences over matchings. A matching M is popular in G if there is no matching N such that vertices that prefer N outnumber those that prefer M. Every stable matching is popular. We consider the following variant: edges in G have utilities and it is only *max-utility* matchings that are relevant for us. We show there always exists a max-utility matching that is popular within the set of all max-utility matchings; moreover, such a matching can be efficiently computed. We focus on *largest* max-utility matchings and show a compact extended formulation for the polytope of largest max-utility matchings that are popular within the set of all largest max-utility matchings.

1 Introduction

We consider a matching problem in a bipartite graph $G = (A \cup B, E)$ where A is a set of agents and B is a set of jobs. Every vertex in $A \cup B$ has a strict preference order over its neighbors. Such a graph G is called a marriage instance and this is a well-studied model in two-sided matching markets.

In several applications, along with vertex preferences, there might be other attributes that we seek to optimize, e.g., in the assignment of residents to hospitals [2] and in the assignment of sailors to billets [17,21], it is only A-perfect matchings that are admissible where A is the set of residents/sailors. More generally, let us assume there is a utility function $f : E \to \mathbb{Q}$ and it is only max-utility matchings that are admissible. Such applications include allocation problems in humanitarian organizations [1,19] where we seek an allocation of resources to beneficiaries that has maximum impact. This motivates us to consider the setting where *edge utilities* are more important than vertex preferences.

The usual solution to a two-sided matching market problem is an appropriate *stable* matching in G. A matching M is stable if there is no *blocking edge* (a, b), i.e., one where both a and b prefer each other to their respective assignments in M. Stable matchings always exist in G and can be efficiently computed by the Gale-Shapley algorithm [7]. All stable matchings have the same size [8] which may be only half the size of a maximum matching. Thus stable matchings need not be max-size matchings or max-utility matchings. So we need to go beyond stability in order to find a max-utility matching that is optimal with respect to vertex preferences.

Popularity. The preferences that any vertex v has over its neighbors in G extend naturally to preferences over matchings. Given any pair of matchings M and N, we say v prefers M to N if v prefers its assignment in M to its assignment in N. Note that being left unassigned is the worst choice for any vertex. Let $\phi(M, N)$ be the number of vertices that prefer M to N. Matching N is *more popular* than matching M if $\phi(N, M) > \phi(M, N)$.

Definition 1. *M is a popular matching in G if $\phi(M, N) \geq \phi(N, M)$ for all matchings N, i.e., G has no matching more popular than M.*

A popular matching is "stable" in the relaxed sense that no majority vote can force a migration from a popular matching. It is easy to show that every stable matching is popular [9]. Thus popular matchings always exist in G.

Recall that utility is more important in our setting than vertex preferences. We seek a max-utility matching M that is optimal with respect to vertex preferences, i.e., there is no max-utility matching that is *better* with respect to vertex preferences than M. So rather than a popular matching, what we seek is a max-utility matching that is popular *within* the set of max-utility matchings. Such a matching will be called a *popular max-utility matching* or more concisely, a *popular* opt*-matching*. The "more popular than" relation is not transitive, so it is not obvious if a popular opt-matching always exists. We show the following.

Theorem 1. *Let $G = (A \cup B, E)$ be any marriage instance where there is a utility function $f : E \to \mathbb{Q}$. A popular* opt*-matching always exists in G and can be computed in polynomial time.*

The next question is whether the entire set of popular opt-matchings can be compactly described, say, as the intersection of half-spaces in $\mathbb{R}^{|E|}$. The motivation for such a description is to efficiently solve *special* popular opt-matching problems such as finding a popular opt-matching with forced/forbidden edges or more generally, one with minimum cost when every edge has an associated cost. Observe that the popular opt-matching problem with forced/forbidden edges generalizes the problem of finding a popular matching with forced/forbidden edges, which is NP-hard [6]. Thus it is NP-hard to optimize over all popular opt-matchings and solve special popular opt-matching problems.

A useful subset. The above hardness motivates us to focus on a natural and interesting subset of the entire set of opt-matchings. Let us restrict our attention to *largest* opt-matchings—so subject to the constraint that matchings have maximum utility, we focus on the largest ones. Indeed, it is these matchings that would be most useful in applications such as allocation problems in humanitarian organizations where subject to the constraint that maximum utility is ensured, we would like to maximize the number of beneficiaries. We will refer to largest max-utility matchings as max-opt-matchings. Hence the matchings we study now are *popular max-*opt*-matchings*, i.e., largest max-utility matchings that are popular within the set of all largest max-utility matchings.

The main idea used to prove Theorem 1 can be used to show that popular max-opt-matchings always exist and can be computed in polynomial time. Let the popular max-opt-matching polytope be the convex hull of the edge incidence vectors of all popular max-opt-matchings. We show the following structural result that the entire set of popular max-opt-matchings admits a compact description. Thus we can optimize over the set of popular max-opt-matchings in polynomial time via linear programming on this polytope.

Theorem 2. *Let $G = (A \cup B, E)$ be a marriage instance with a utility function $f : E \to \mathbb{Q}$. A compact extension of the popular max-opt-matching polytope of G exists and can be formulated in polynomial time.*

1.1 Background and related results

Gärdenfors [9] introduced the notion of popularity in 1975 where he observed that any stable matching in a marriage instance is popular. Popular matchings have been well-studied during the last 15-20 years and we refer to [3] for a survey.

It was shown in [11] that any marriage instance has a *popular maximum matching*, i.e., a maximum matching that is popular within the set of all maximum matchings and a polynomial time algorithm was given to find such a matching. The problem of computing a *min-cost* popular maximum matching was considered in [13] and a compact extended formulation of the popular maximum matching polytope was given. Thus a min-cost popular maximum matching can be computed in polynomial time. In contrast to this, as mentioned earlier, it is NP-hard to compute a min-cost popular matching [6]. Furthermore, the extension complexity of the popular matching polytope is near-exponential [5].

The following definitions will be useful to us. For any subset $C \subseteq A \cup B$, call a matching *C-critical* if it matches as many vertices in C as possible. A C-critical matching M is a *popular C-critical matching* if $\Delta(M, N) \geq 0$ for all C-critical matchings N in G. When $C = \emptyset$ (resp., $C = A \cup B$), popular C-critical matchings are the same as popular matchings (resp., popular maximum matchings).

It was shown in [12] that for any subset $C \subseteq A \cup B$, a popular C-critical matching always exists and can be computed in polynomial time. A related problem in the hospitals-residents setting was independently considered in [15] where every hospital had upper and lower bounds on the number of residents that it could be matched to in any feasible matching and certain special residents were to be matched in any feasible matching. It was shown that a popular feasible matching always exists and can be computed in polynomial time.

1.2 Our techniques

A crucial observation in our result on popular opt-matchings (Theorem 1) is that for any edge utility function, the convex hull of all max-utility matchings is a face of the matching polytope of G (see Section 2). So max-utility matchings can be characterized as *C-perfect* matchings in a subgraph G' for some $C \subseteq A \cup B$, where a matching is C-perfect if it matches all vertices in C. Thus once we determine the set C and the edge set of G', the popular C-critical matching algorithm in G' finds a popular max-utility matching in G.

Popular opt-matchings generalize popular matchings, thus their polytope inherits the near-exponential extension complexity of the popular matching polytope. The key idea in formulating a compact extension for the popular max-opt-matching polytope is that the critical set for max-opt-matchings has rich structure. We will use the well-known Dulmage-Mendelsohn decomposition [4] to determine the critical set for popular max-opt-matchings.

We construct another bipartite graph $H = (L \cup R, E_H)$ and max-opt-matchings in G can be characterized as $(L \cup K)$-perfect matchings in H for a certain subset $K \subseteq R$. Observe that the min-cost popular maximum matching algorithm [13] in H does not solve the min-cost popular max-opt-matching problem since such a matching may leave some vertices in the set K unmatched—this would make the resulting matching a non-opt-matching in the original instance G.

So the problem we need to solve is to find a min-cost popular $(L \cup K)$-perfect matching in H. As done in [13], we use LP duality to solve the above problem. Our key technical lemma (Lemma 3) shows that every popular $(L \cup K)$-perfect matching has a very useful dual certificate which allows us to realize every such matching as a stable matching in a certain multigraph derived from H (see Section 4). So the stable matching polytope [18,20] of this multigraph yields a compact extended formulation for the popular max-opt-matching polytope of G.

2 Popular opt-matchings

We prove Theorem 1 in this section. We will show that the popular opt-matching problem in G can be solved as the popular C-critical matching problem for an appropriate set C in a subgraph $G' = (A \cup B, E')$ of $G = (A \cup B, E)$.

There is a utility function $f : E \to \mathbb{Q}$ and $f(M) = \sum_{e \in M} f(e)$ for any matching M. Let $\lambda = \max_M f(M)$, where the max is over all matchings in G. Recall that M is an opt-matching if and only if $f(M) = \lambda$. The opt-matching polytope (call it $\mathcal{X}$) is the convex hull of edge incidence vectors of all opt-matchings in G.

A face of a polytope $\mathcal{P}$ is the set of points $\boldsymbol{x}$ in $\mathcal{P}$ such that $\sum_j \alpha_j x_j = \beta$ where $\sum_j \alpha_j x_j \le \beta$ is a valid inequality for $\mathcal{P}$. Since $\sum_{e \in E} f(e) \le \lambda$ is a valid inequality for the matching polytope $\mathcal{M}$ of G, the polytope $\mathcal{X}$ is a face of $\mathcal{M}$. Suppose a polytope $\mathcal{P}$ is defined by the constraints $\sum_j \alpha_{ij} x_j \le \beta_i$ for $1 \le i \le k$. Then any face of $\mathcal{P}$ corresponds to the set of solutions to $\sum_j \alpha_{ij} x_j = \beta_i$ for all $i \in I$ and $\sum_j \alpha_{ij} x_j \le \beta_i$ for all $i \notin I$ for some set $I \subseteq \{1, \ldots, k\}$. See [10, Theorem 3.5] for a proof.

Thus the opt-matching polytope $\mathcal{X}$, which is a face of the matching polytope $\mathcal{M}$ of G, has the following formulation for some $A' \subseteq A$, $B' \subseteq B$, and $\tilde{E} \subseteq E$:

$$\sum_{e \in \delta(v)} x_e \le 1 \quad \forall v \in A \cup B \quad \text{and} \quad \sum_{e \in \delta(v)} x_e = 1 \quad \forall v \in A' \cup B'$$

$$x_e \ge 0 \quad \forall e \in E \quad \text{and} \quad x_e = 0 \quad \forall e \in \tilde{E}.$$

Here $\delta(v)$ is the set of edges incident to v in G. So a matching M is an opt-matching if and only if M matches all vertices in $A' \cup B'$ and $M \subseteq E'$, where

$E' = E \setminus \tilde{E}$. The sets A', B', and E' can be computed by solving LP2. This linear program is dual to the max-utility matching LP, which is LP1.

$$\max \sum_{e \in E} f(e) \cdot x_e \qquad \text{(LP1)}$$
$$\text{s.t.} \quad \sum_{e \in \delta(v)} x_e \leq 1 \quad \forall\, v \in A \cup B$$
$$x_e \geq 0 \quad \forall\, e \in E.$$

$$\min \sum_{v \in A \cup B} y_v \qquad \text{(LP2)}$$
$$\text{s.t.} \quad y_a + y_b \geq f(a,b) \quad \forall\,(a,b) \in E$$
$$y_v \geq 0 \quad \forall\, v \in A \cup B.$$

Let $\boldsymbol{y}$ be an optimal solution to LP2. We have $A' = \{a \in A : y_a > 0\}$, $B' = \{b \in B : y_b > 0\}$, and $E' = \{(a,b) \in E : y_a + y_b = f(a,b)\}$. It follows from complementary slackness conditions that for any $\boldsymbol{x} \in \mathcal{X}$, we have $\sum_{e \in \delta(v)} x_e = 1$ for all $v \in A' \cup B'$ and $x_e = 0$ for all $e \in E \setminus E'$. Thus we can conclude the following proposition. The converse in Proposition 1 also follows from LP-duality.

Proposition 1. *M is an* opt*-matching in G if and only if M is a C-perfect matching in $G' = (A \cup B, E')$ where $C = A' \cup B'$.*

Remark 1. Note that Proposition 1 is well-known from matching theory, e.g., an interesting application of this principle is in solving the min-cost perfect matching problem in bipartite graphs [10, Section 1.2].

Proposition 1 implies that the popular opt-matching problem in $G = (A \cup B, E)$ reduces to the popular C-critical matching problem—more precisely, the popular C-*perfect* matching problem—in $G' = (A \cup B, E')$, where $C = A' \cup B'$.

Finding a popular C-perfect matching. We now describe a multigraph G'_C such that any stable matching in G'_C projects to a popular C-perfect matching in G'. The graph G'_C is inspired by the graph used in [12] for solving the popular C-critical matching problem. Our graph has fewer vertices and edges than the graph used in [12]. Recall that $C = A' \cup B'$.

Let $|A'| = s$ and $|B'| = t$. Below we describe the edge set E'_C of G'_C.

- Initialize $E'_C = \{e_0 : e \in E'\}$.
- For every $e = (a,b) \in E'$, we will consider the following parallel edges where each e_i has a and b as its endpoints.
  1. If $a \in A'$: then add the s edges $e_1, \ldots, e_s$ to E'_C.
  2. If $b \in B'$: then add the t edges $e_{-1}, \ldots, e_{-t}$ to E'_C.

So if $a \in A'$ and $b \in B'$ then there are $s + t + 1$ edges $e_{-t}, \ldots, e_s$ in E'_C. We need to specify vertex preferences in G'_C. In order to compare two edges e_i and e'_j incident to it, any vertex v first compares the subscripts i and j. Suppose $v \in A$.

- If $i < j$ then v prefers e_i to e'_j and if $i > j$ then v prefers e'_j to e_i.
- If $i = j$ then v prefers e_i if it prefers e to e'; else it prefers e'_j.

Thus any vertex in A prefers lower subscript edges to higher subscript edges. Among edges with the same subscript, it is as per its original preference order. On the other hand, any vertex in B prefers *higher* subscript edges to *lower* subscript edges. Hence for any $v \in B$ and any two edges e_i and e'_j incident to v:

- If $i < j$ then v prefers e'_j to e_i and if $i > j$ then v prefers e_i to e'_j.
- If $i = j$ then v prefers e_i if it prefers e to e'; else it prefers e'_j.

A matching M' in the multigraph G'_C is a subset of E'_C such that at most one edge of M' is incident to any vertex. Matching M' is *stable* in G'_C if there is no edge in E'_C that blocks M'. For any matching M' in G'_C, define M to be the projection of M' in G', i.e., replace every edge $e_i \in M'$ with the original edge e (so subscripts are dropped from the edges in M').

Lemma 1. *If M' is a stable matching in G'_C then M is a popular C-perfect matching in G'.*

The proof of the above lemma is similar to the proof of correctness of the popular maximum matching algorithm [11,13] and the popular critical matching algorithm [12]. This lemma can be proved in two parts. The first claim is that M is a C-perfect matching in G' where $C = A' \cup B'$. This claim follows by showing that there are no *forbidden* alternating or augmenting paths with respect to M in G'. This claim is proved in the appendix.

We will now assume that M is a C-perfect matching in G' where $C = A' \cup B'$. We prove below that M is a *popular* C-perfect matching in G'. That is, for any C-perfect matching N in G', we will show that $\Delta(N, M) \leq 0$. To show this, we will use the following two linear programs: LP3 and LP4.

Let $\bar{E}'$ be the edge set E' augmented with self-loops (v, v) for each vertex v where $v \in (A \cup B) \setminus (A' \cup B')$. Adding self-loops will allow us to work with perfect matchings in the augmented G' whose vertex set is $A \cup B$ and edge set is $\bar{E}'$. We will use the following edge weight function wt_M. For any $(a, b) \in E'$:

$$\text{let } \mathsf{wt}_M(a,b) = \begin{cases} 2 & \text{if } (a,b) \text{ blocks } M; \\ -2 & \text{if } a \text{ and } b \text{ prefer their assignments in } M \text{ to each other;} \\ 0 & \text{otherwise.} \end{cases}$$

For any edge e, $\mathsf{wt}_M(e)$ is the sum of *votes* of the endpoints of e for each other versus their respective assignments in M, where the vote of a vertex for one neighbor versus another is ± 1 depending on which one it likes better and v's vote for neighbor u versus neighbor u' is 0 if and only if $u = u'$. So $\mathsf{wt}_M(e) = 0$ for every $e \in M$. Let $\mathsf{wt}_M(v, v) = 0$ if the self-loop (v, v) is in the augmented M (i.e., v was originally left unmatched in M), else $\mathsf{wt}_M(v, v) = -1$. It follows from the definition of wt_M that for any perfect matching N in the augmented graph G', we have $\mathsf{wt}_M(N) = \Delta(N, M)$.

Our goal is to show that $\mathsf{wt}_M(N) \leq 0$ for all perfect matchings N in the augmented graph G'. Consider the linear programs LP3 and LP4 given below.

$$\max \sum_{e\in E'} \mathsf{wt}_M(e)\cdot x_e \qquad \text{(LP3)}$$
$$\text{s.t.} \quad \sum_{e\in\delta'(v)} x_e = 1 \quad \forall\, v\in A\cup B$$
$$x_e \geq 0 \quad \forall\, e\in \bar{E}'.$$

$$\min \sum_{v\in A\cup B} y_v \qquad \text{(LP4)}$$
$$\text{s.t.} \quad y_a + y_b \geq \mathsf{wt}_M(a,b) \quad \forall\,(a,b)\in E'$$
$$y_v \geq \mathsf{wt}_M(v,v) \quad \forall\, v\notin C.$$

LP3 is the max-weight perfect matching LP in the augmented G' with the edge weight function wt_M and LP4 is the dual LP. We will show the optimal value of LP3 is at most 0 by showing a feasible solution $\boldsymbol{y}$ to LP4 such that $\sum_{v\in A\cup B} y_v = 0$. We delete self-loops from M and set y-values as follows.

For each unmatched vertex v do: set $y_v = 0$. Let $e = (a,b)\in M$. We know that e_i is in M' for some $-t \leq i \leq s$. If $e_i \in M'$ then set $y_a = -2i$ and $y_b = 2i$. So for any edge $(a,b)\in M$, we have $y_a + y_b = 0$. Since $y_v = 0$ for any unmatched vertex v, we have $\sum_{v\in A\cup B} y_v = 0$.

Claim. $\boldsymbol{y}$ is a feasible solution to LP4.

Proof. It is easy to see that the constraint $y_v \geq \mathsf{wt}_M(v,v)$ holds for all $v\notin C$. For any unmatched vertex v, we have $y_v = 0 = \mathsf{wt}_M(v,v)$. Consider any $a\notin A'$. Only non-positive subscript edges are incident to a, thus $y_a = -2i$ for some $i\leq 0$, so $y_a \geq 0$. Consider any $b\notin B'$. Only non-negative subscript edges are incident to b, thus $y_b = 2j$ for some $j\geq 0$, so $y_b\geq 0$.

We now need to show that $y_a + y_b \geq \mathsf{wt}_M(a,b)$ for all $(a,b)\in E'$. Let us partition the sets $A = A_{-t}\cup A_{-t+1}\cup\cdots\cup A_s$ and $B = B_{-t}\cup B_{-t+1}\cup\cdots\cup B_s$ where for each edge $e=(a,b)\in M$ (so $e_i\in M'$ for some $i\in\{-t,\ldots,s\}$), we include a in A_i and b in B_i. Since M is C-perfect, every unmatched vertex is outside $A'\cup B'$. Unmatched vertices in $A\setminus A'$ (resp., $B\setminus B'$) are added to A_0 (resp., B_0).

We claim the following properties hold for this partition of A and B.

1. There is no edge in $A_i\times B_j$ for $i\geq j+2$.
2. For any edge $e\in A_i\times B_{i-1}$, we have $\mathsf{wt}_M(e) = -2$.
3. For any edge $e\in A_i\times B_i$, we have $\mathsf{wt}_M(e)\leq 0$.

– Suppose property 1 does not hold for some edge e. Then the edge e_{i+1} blocks M', a contradiction to the stability of M' in G'_C.
– Suppose property 2 does not hold for some edge e. Then either the edge e_i or the edge e_{i-1} blocks M', a contradiction.[1]
– Suppose property 3 does not hold for some edge e. Then the edge e_i blocks M', a contradiction.

Consider any edge (a,b) in G'. Let $a\in A_i$ and $b\in B_j$. Observe that $y_a = -2i$ and $y_b = 2j$. Property 1 above tells us that $i\leq j+1$. Consider the following three cases.

[1] If $\mathsf{wt}_M(e)\neq -2$, then $\mathsf{wt}_M(e)\geq 0$. Since $e\notin M$, this means that at least one of a,b prefers the other to its partner in M. If a prefers b to its partner in M then e_i blocks M'; if b prefers a to its assignment in M then e_{i-1} blocks M'.

(1) $i = j + 1$. So $(a, b) \in A_i \times B_{i-1}$ and property 2 tells us that $\mathsf{wt}_M(a, b) = -2$. Thus $y_a + y_b = -2i + 2j = -2 = \mathsf{wt}_M(a, b)$.
(2) $i = j$. So $(a, b) \in A_i \times B_i$ and property 3 tells us that $\mathsf{wt}_M(a, b) \le 0$. Thus $y_a + y_b = -2i + 2i = 0 \ge \mathsf{wt}_M(a, b)$.
(3) $i \le j - 1$. Then $y_a + y_b = -2i + 2j \ge 2 \ge \mathsf{wt}_M(a, b)$.

Hence for any edge (a, b) in G', we have $y_a + y_b \ge \mathsf{wt}_M(a, b)$. So $\boldsymbol{y}$ is a feasible solution to LP4. This finishes the proof of this claim. □

Thus we showed a feasible solution $\boldsymbol{y}$ to LP4 such that $\sum_{v \in A \cup B} y_v = 0$. Hence the optimal value of LP3 is at most 0, i.e., $\Delta(N, M) \le 0$ for all C-perfect matchings N. So M is a popular C-perfect matching in G'. This completes the proof of Lemma 1.

Our algorithm. Let us run the Gale-Shapley algorithm in G'_C where vertices in A propose. For any edge $e = (a, b)$, among the parallel edges $e_{i_1}, e_{i_2}, \ldots$ where $i_1 < i_2 < \ldots$, the first time a proposes along e, it is along e_{i_1} and the next time a proposes along e, it is along e_{i_2} and so on. This algorithm computes a stable matching M' in G'_C. By Lemma 1, dropping the subscripts of edges in M' yields a popular C-perfect matching M in G'. So our algorithm is as follows.

1. Solve LP2 and determine the sets A', B', and E'.
2. Find a stable matching M' in G'_C where $G' = (A \cup B, E')$ and $C = A' \cup B'$.
3. Return the projection M of M' in G (so edge subscripts are dropped).

Lemma 1 and Proposition 1 show that M is a popular opt-matching in G. Since steps 1-3 can be implemented in polynomial time, Theorem 1 follows.

3 Popular max-opt-matchings

In this section we will see an extension of Theorem 1 that solves the popular max-opt-matching problem in G. Consider the subgraph $G' = (A \cup B, E')$ of G. The *Dulmage-Mendelsohn* decomposition [4] in G' will be very useful to us.

Let M be a matching in G'. An alternating path with respect to M is a path whose alternate edges are in M. We have $A \cup B = \mathcal{E}_M \cup \mathcal{O}_M \cup \mathcal{U}_M$, where a vertex v is in $\mathcal{E}_M$ (resp., $\mathcal{O}_M$) if there is an even (resp., odd) length alternating path with respect to M in G' from an unmatched vertex to v; a vertex v is in $\mathcal{U}_M$ if there is no alternating path in G' from an unmatched vertex to v. So all vertices left unmatched in M are in $\mathcal{E}_M$.

The sets $\mathcal{E}_M, \mathcal{O}_M$, and $\mathcal{U}_M$ will be called the sets of *even, odd*, and *unreachable* vertices, respectively, with respect to M. We refer to [14,16] for a proof of the following theorem.

Theorem 3. *The sets $\mathcal{E}_M$, $\mathcal{O}_M$, and $\mathcal{U}_M$ are pairwise disjoint if and only if M is a maximum matching in G'. Any maximum matching in G' partitions the vertex set into the same sets of even, odd, and unreachable vertices. Furthermore, M is a maximum matching in G' if and only if M matches all vertices in $\mathcal{O}_M$ with those in $\mathcal{E}_M$ and it matches all vertices in $\mathcal{U}_M$ among themselves.*

Let M be a maximum matching in G'. It follows from the definition of $\mathcal{U}_M$ that G' has no edge between the sets $\mathcal{E}_M$ and $\mathcal{U}_M$. Similarly, there is no edge in G' with both its endpoints in $\mathcal{E}_M$ since it would imply $\mathcal{E}_M \cap \mathcal{O}_M \neq \emptyset$. Thus there is no edge in G' with one endpoint in $\mathcal{E}_M$ and the other endpoint in $\mathcal{E}_M \cup \mathcal{U}_M$.

Let the subset $E'' \subseteq E'$ be obtained by excluding all those edges with one endpoint in $\mathcal{O}_M$ and the other endpoint in $\mathcal{O}_M \cup \mathcal{U}_M$. Thus every edge in E'' has either (i) both its endpoints in $\mathcal{U}_M$ or (ii) one endpoint in $\mathcal{O}_M$ and the other in $\mathcal{E}_M$. By Theorem 3, all maximum matchings of G' are in $G'' = (A \cup B, E'')$.

Since the decomposition of $A \cup B$ into $\mathcal{E}_M, \mathcal{O}_M$, and $\mathcal{U}_M$ is independent of the matching M, we will henceforth refer to these sets as $\mathcal{E}, \mathcal{O}$, and $\mathcal{U}$, respectively.

Lemma 2. *A matching M is a max-*opt*-matching in G if and only if M is a C-perfect matching in $G'' = (A \cup B, E'')$ where $C = \mathcal{O} \cup \mathcal{U} \cup (\mathcal{E} \cap (A' \cup B'))$.*

Proof. Let M be a max-opt-matching in G. Since M is an opt-matching, we have $M \subseteq E'$ and M is $(A' \cup B')$-perfect (by Proposition 1). Suppose there is an augmenting path ρ with respect to M in G'. Then the matching $M \oplus \rho$ is also an opt-matching in G because $M \oplus \rho \subseteq E'$ and every vertex matched in M is also matched in $M \oplus \rho$. Hence any max-opt-matching M has to be a maximum matching in G'. Thus by Theorem 3, M belongs to the subgraph G'' and it matches all vertices in $\mathcal{O} \cup \mathcal{U}$. Moreover, M also matches all vertices in $A' \cup B' \supseteq \mathcal{E} \cap (A' \cup B')$. Thus M is a C-perfect matching in $G'' = (A \cup B, E'')$ for the given set C.

Conversely, let M be a C-perfect matching in $G'' = (A \cup B, E'')$. So $M \subseteq E''$ and M matches all vertices in $\mathcal{O} \cup \mathcal{U}$. Thus by Theorem 3, M is a maximum matching in G'. Moreover, M matches all vertices in $A' \cup B'$ since M matches all in $\mathcal{O} \cup \mathcal{U} \cup (\mathcal{E} \cap (A' \cup B'))$. Because $M' \subseteq E'' \subseteq E'$, we can conclude that M is an opt-matching in G (by Proposition 1). Recall that every opt-matching in G belongs to the subgraph G'. Since M is a maximum matching in G', there is no larger opt-matching in G; so M is a max-opt-matching in G. ◻

By Lemma 2, the popular C-perfect matching algorithm in G'' solves the popular max-opt-matching problem in G. Thus we have the following corollary.

Corollary 1. *A popular max-*opt*-matching in $G = (A \cup B, E)$ can be computed in polynomial time.*

4 The popular max-opt-matching polytope

We saw in Section 2 that any stable matching in G'_C projects to a popular C-perfect matching in G. But it is not the case that *every* popular C-perfect matching in G is realizable as a stable matching in G'_C. Suppose $C = \emptyset$ (this happens when $f(e) = 0$ for all $e \in E$). Then every matching in G is a C-perfect matching. The stable matching algorithm in $G'_\emptyset = G$ finds only stable matchings in G whereas any popular matching in G is a popular C-perfect matching here.

However, in the case of popular C-perfect matchings in G''_C studied in Section 3, we will now see that *every* popular C-perfect matching in G is realizable as a stable matching in G''_C. Rather than work in G''_C, it will be easier to work in another multigraph which is essentially the same as G''_C. Recall the edge set $E'' \subseteq E'$ and the vertex sets $\mathcal{O}, \mathcal{E}$, and $\mathcal{U}$ from Section 3. Let $L = \mathcal{O} \cup (\mathcal{U} \cap A)$ and $R = \mathcal{E} \cup (\mathcal{U} \cap B)$. Consider the bipartite graph $H = (L \cup R, E'')$ which is G'' with certain vertices swapped between the left and right sides (see Fig. 1).

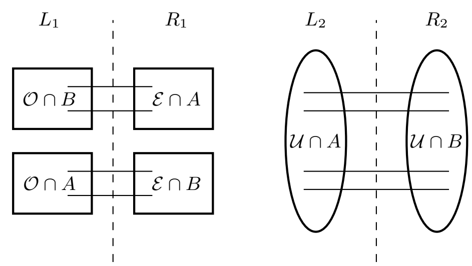

Fig. 1. $L = L_1 \cup L_2$ and $R = R_1 \cup R_2$. By moving all vertices in $\mathcal{O} \cap B$ (resp., $\mathcal{E} \cap A$) from the right of G'' to left (resp., from the left of G'' to right), we get H from G''.

It follows from Lemma 2 that a matching M is a max-opt-matching in G if and only if M is an $(\mathcal{O} \cup \mathcal{U} \cup K)$-perfect matching in $H = (L \cup R, E'')$ where $K = \mathcal{E} \cap (A' \cup B')$. In any L-perfect matching in H, note that all vertices in $\mathcal{U} \cap B$ will get matched to vertices in $\mathcal{U} \cap A$. Hence in our critical set, $\mathcal{U}$ can be replaced by $\mathcal{U} \cap A$. So M is a popular max-opt-matching in G if and only if M is a popular $(L \cup K)$-perfect matching in H where $L = \mathcal{O} \cup (\mathcal{U} \cap A)$.

Popular $(L \cup K)$-perfect matchings in H. It will be convenient to work with perfect matchings in H, i.e., no vertex is left unmatched. So let us add a self-loop (v, v) at each $v \in \mathcal{E} \setminus K$ in the graph H. Recall that vertices in $\mathcal{E} \setminus K$ are the only vertices that may possibly be left unmatched in a $(L \cup K)$-perfect matching. Thus every $(L \cup K)$-perfect matching becomes a perfect matching in the augmented graph H. We can now focus on perfect matchings in the augmented H.

Let M be any popular $(L \cup K)$-perfect matching in the original H. So M is a popular perfect matching in the augmented H. Consider the following edge weight function wt_M (also defined in Section 2). For any $(a, b) \in E''$:

$$\text{let } \mathsf{wt}_M(a,b) = \begin{cases} 2 & \text{if } (a,b) \text{ blocks } M; \\ -2 & \text{if } a \text{ and } b \text{ prefer their assignments in } M \text{ to each other;} \\ 0 & \text{otherwise.} \end{cases}$$

Furthermore, let $\mathsf{wt}_M(v, v) = 0$ if the self-loop (v, v) is in the augmented M (i.e., v was originally left unmatched in M), else $\mathsf{wt}_M(v, v) = -1$. It follows from the definition of wt_M that for any perfect matching N in the augmented graph H, we have $\mathsf{wt}_M(N) = \Delta(N, M)$.

The linear program LP5 is the max-weight perfect matching LP in the augmented H with edge weight function wt_M. Note that $\bar{E}''$ is the edge set E''

augmented with self-loops (v,v) for all $v \in \mathcal{E} \setminus K$. For any $v \in L \cup R$, the set $\delta''(v)$ is the set of edges incident to v in the augmented H, so $\delta''(v)$ includes the self-loops (v,v) for all $v \in \mathcal{E} \setminus K$.

$$\max \sum_{e \in E''} \mathsf{wt}_M(e) \cdot x_e \qquad \text{(LP5)}$$
$$\text{s.t.} \quad \sum_{e \in \delta''(v)} x_e = 1 \quad \forall\, v \in L \cup R$$
$$x_e \geq 0 \quad \forall\, e \in \bar{E}''.$$

$$\min \sum_{v \in L \cup R} y_v \qquad \text{(LP6)}$$
$$\text{s.t.} \quad y_l + y_r \geq \mathsf{wt}_M(l,r) \quad \forall\,(l,r) \in E''$$
$$y_r \geq \mathsf{wt}_M(r,r) \quad \forall\, r \in \mathcal{E} \setminus K.$$

The optimal value of LP5 is $\max_N \Delta(N,M)$ where the max is over all perfect matchings in the augmented H, equivalently, over all $(L \cup K)$-perfect matchings N in the original H. Since M is a popular $(L \cup K)$-perfect matching, $\Delta(N,M) \leq 0$ for all $(L \cup K)$-perfect matchings N; thus the optimal value of LP5 is at most 0. Since $\mathsf{wt}_M(M) = \Delta(M,M) = 0$, it follows that the edge incidence vector of M is an optimal solution to LP5.

The linear program LP6 is the dual LP. The following is our key technical lemma. A solution to LP6 as described in this lemma will be called a *dual certificate* for the popular $(L \cup K)$-perfect matching M. Let $|K| = k$ and $|L| = \ell$.

Lemma 3. *Suppose M is a popular $(L \cup K)$-perfect matching in H. Then the linear program LP6 admits an optimal solution $\boldsymbol{y}$ that satisfies the following properties:*

- $y_l \in \{2k, 2(k-1), \ldots, 0, -2, -4, \ldots, -2(\ell-1)\}$ *for all* $l \in L$.
- $y_r \in \{-2k, -2(k-1), \ldots, 0, 2, 4, \ldots, 2(\ell-1)\}$ *for all* $r \in R$.

Proof. The constraint matrix of LP6 is totally unimodular. Thus there is an integral optimal solution $\boldsymbol{y}$ to LP6. We can also assume that all the y-values are *even* integers by updating $\boldsymbol{y}$: increase y_r by 1 for all $r \in R$ with odd y-values and decrease y_l by 1 for all $l \in L$ with odd y-values. The updated $\boldsymbol{y}$ continues to be feasible since edge weights are even; so for any edge (l,r), if $y_l + y_r$ was odd, then there was a slack of at least 1 in the constraint $y_l + y_r \geq \mathsf{wt}_M(l,r)$ and if $y_l + y_r$ was even then it remains the same.

The updated $\boldsymbol{y}$ continues to be optimal because by complementary slackness, $y_r = \mathsf{wt}_M(r,r) = 0$ for any r with $(r,r) \in M$ and moreover, for any $(l,r) \in M$, $y_l + y_r = \mathsf{wt}_M(l,r) = 0$. Hence l and its partner r have the same parity of y-values. So we have decreased y_l and increased y_r by the same amount for an equal number of vertices in L and R, respectively. Thus $\sum_{v \in L \cup R} y_v$ remains the same, i.e., $\sum_{v \in L \cup R} y_v = 0$. For $r \in \mathcal{E} \setminus K$, because the original constraint was $y_r \geq -1$ (since $y_r \geq \mathsf{wt}_M(r,r) \geq -1$), we now have $y_r \geq 0$ for all $r \in \mathcal{E} \setminus K$.

A special case. Suppose M matches all vertices in R, i.e., M does not use any self-loops in the augmented H. So $|L| = |R| = \ell$. Let us increase y_r for all $r \in R$ by the required amount so that $y_r \geq 0$ for all $r \in R$. Symmetrically, let us decrease y_l for all $l \in L$ by the same amount. This preserves feasibility since the sum $y_l + y_r$ is unchanged for any edge (l, r). Because M is a perfect matching, $\sum_{v \in L \cup R} y_v$ is unchanged, so we have preserved optimality. Thus $y_r \in \{0, 2, 4, \ldots\}$ for all $r \in R$ and $y_l \in \{0, -2, -4, \ldots\}$ for all $l \in L$ (recall that all the y-values are even and $y_l = -y_r$ for any $(l, r) \in M$).

Among all optimal solutions $\boldsymbol{y}$ to LP6 such that $y_r \in \{0, 2, 4, \ldots\}$ and $y_l \in \{0, -2, -4, \ldots\}$, let $\boldsymbol{y}$ be such that $\sum_{r \in R} y_r$ is the least. We claim that the y_r values are consecutive even integers. Otherwise there is a gap in the y_r values, say there is no vertex in R with a y-value of $2i$. Then we can update $y_r = y_r - 2$ for all $r \in R$ with $y_r \geq 2i + 2$ and symmetrically update $y_l = y_l + 2$ for all $l \in L$ with $y_l \leq -(2i + 2)$. Since all edge weights are at most 2, this update preserves feasibility. Any $r \in R$ has $y_r \geq 2i + 2$ if and only if $y_l \leq -(2i + 2)$ where $(l, r) \in M$ (since $y_l + y_r = 0$); so this update preserves optimality as well.

Thus we have an optimal solution $\boldsymbol{y}$ to LP6 with a smaller value of $\sum_{r \in R} y_r$, a contradiction. Hence we can conclude that y_r values are consecutive even integers. Since $|A| = |B| = \ell$, we have $y_r \in \{0, 2, 4, \ldots, 2(\ell - 1)\}$ for all $r \in R$ and $y_l \in \{0, -2, -4, \ldots, -2(\ell - 1)\}$ for all $l \in L$. This proves the lemma for the special case when M is a perfect matching.

The general case. Suppose M leaves some vertices in $\mathcal{E} \setminus K$ unmatched. For any $r \in \mathcal{E} \setminus K$ that is left unmatched in M, we have $y_r = \mathtt{wt}_M(r, r) = 0$ by complementary slackness. So the y-values of these vertices are fixed. Hence we cannot shift y-values as done in the special case, thus all we can say at this point is that all the y-values are even integers. Among all optimal solutions $\boldsymbol{y}$ to LP6 such that the y-values are even integers, let $\boldsymbol{y}$ be such that $\sum_{r \in R} |y_r|$ is the least. The same argument as given above shows that y-values are consecutive even integers. That is, if there is no vertex in R with a y-value of $2i$ for some $i > 0$ while there are vertices r with $y_r \geq 2i + 2$ then we update $y_r = y_r - 2$ for all $r \in R$ with $y_r \geq 2i + 2$ and symmetrically update $y_l = y_l + 2$ for all $l \in L$ with $y_l \leq -(2i + 2)$. Analogously, if there is no vertex in R with a y-value of $2i$ for some $i < 0$ while there are vertices r with $y_r \leq 2i - 2$ then we update $y_r = y_r + 2$ for all $r \in R$ with $y_r \leq 2i - 2$ and symmetrically update $y_l = y_l - 2$ for all $l \in L$ with $y_l \geq -2i + 2$. Since all edge weights are at most 2, these updates preserve feasibility. In either case, we would have an optimal solution $\boldsymbol{y}$ to LP6 with a smaller value of $\sum_{r \in R} |y_r|$, a contradiction. Note that this argument deals with matched vertices only and unmatched vertices remain untouched since their y-value is 0. Thus $y_v \in \{0, \pm 2, \pm 4, \ldots, \pm 2(\ell - 1)\}$ for all $v \in L \cup R$.

Since $\boldsymbol{y}$ is a feasible solution to LP6, we have $y_r \geq \mathtt{wt}_M(r, r) \geq -1$ for all $r \in \mathcal{E} \setminus K$. So among the vertices of R, it is only the ones in K that can take values in $\{-2, -4, \ldots, -2(\ell - 1)\}$. Since y-values are consecutive even integers and because $|K| = k$, the only possible negative y-values for vertices in K are in $\{-2, \ldots, -2k\}$. Thus we have shown that $y_r \in \{-2k, \ldots, 0, 2, 4, \ldots, 2(\ell - 1)\}$ for all $r \in R$. Hence $y_l \in \{2k, \ldots, 0, -2, -4, \ldots, -2(\ell - 1)\}$ for all $l \in L$. $\square$

Analogous to G'_C (see Section 2), we define a multigraph $H_K = (L \cup R, E''_K)$ to characterize popular $(L \cup K)$-perfect matchings in H. For every $e = (l, r) \in E''$ where $l \in L$ and $r \in R$, the following parallel edges are in the edge set E''_K.

– If $r \notin K$ then there are ℓ edges $e_0, e_1, \ldots, e_{\ell-1}$ in E''_K.
– If $r \in K$ then there are $\ell + k$ edges $e_{-k}, e_{-k+1}, \ldots, e_0, \ldots, e_{\ell-1}$ in E''_K.

As before, vertices in L (resp., R) prefer lower (resp., higher) subscript edges. Among edges with the same subscript, it is as per their original preference order. We will now show that for each popular $(L \cup K)$-perfect matching M in G, there is a stable matching M' in H_K such that M is the projection of M'. We will use Lemma 3 to define M'. We know that M has a dual certificate $\boldsymbol{y}$ as described in Lemma 3. For any edge $e = (l, r) \in M$, let us set e's subscript s_e in M' to be $y_r/2$. Since $y_r \in \{-2k, \ldots, 0, 2, 4, \ldots, 2(\ell-1)\}$, we have $s_e \in \{-k, \ldots, 0, 1, 2, \ldots, \ell-1\}$.

The proof of Lemma 4 is an extension of the proof of correctness of the min-cost popular perfect matching from [13]. Our matching M need not be perfect and moreover, M may have a blocking edge incident to some unmatched vertex in $\mathcal{E} \setminus K$. However the L-endpoint of such a blocking edge would have to be matched along a *negative* subscript edge and this helps us to show the stability of M' in H_K.

Lemma 4. *For any popular $(L \cup K)$-perfect matching M in G, the matching $M' = \{e_{s_e} : e \in M\}$ is stable in H_K.*

Proof. Let $\boldsymbol{y}$ be a dual certificate of M. So $\boldsymbol{y}$ is an optimal solution to LP6 with coordinates as given in Lemma 3.

Consider any edge $e = (l, r) \in L \times R$ where r is unmatched in M. So $y_r = 0$ and $r \in \mathcal{E} \setminus K$. Since $r \notin K$, there are only ℓ copies $e_0, \ldots, e_{\ell-1}$ of e in H_C. We need to show that none of $e_0, \ldots, e_{\ell-1}$ is a blocking edge to M'. Note that $\mathsf{wt}_M(l, r) \geq 0$ since r prefers to be matched to l rather than be unmatched. Thus we have $y_l = y_l + y_r \geq \mathsf{wt}_M(l, r) \geq 0$. So $y_l \geq 0$, hence $y_l \in \{0, 2, \ldots, 2k\}$. This implies $y_{M(l)} \in \{0, -2, \ldots, -2k\}$ and l is matched in M along an edge e' with $s_{e'} \in \{0, -1, \ldots, -k\}$.

Suppose $y_l = 0$. Then $\mathsf{wt}_M(l, r) \leq y_l + y_r = 0$. Thus $\mathsf{wt}_M(l, r) = 0$. So l prefers $M(l)$ to r, i.e., l prefers e' to e, thus it prefers e'_0 to e_0; hence e_0 does not block M'. None of $e_1, \ldots, e_{\ell-1}$ blocks M' since l prefers e'_0 to any of $e_1, \ldots, e_{\ell-1}$: recall that lower subscript edges are preferred by vertices in L to higher ones.

The other case is that $y_l \in \{2, \ldots, 2k\}$. Then l is matched in M along an edge e' with $s_{e'} \in \{-1, \ldots, -k\}$. So l is matched along a negative subscript edge in M', hence none of the higher subscript edges $e_0, \ldots, e_{\ell-1}$ blocks M'.

Let us now consider the case of an edge $(l, r) \in L \times R$ where both l and r are matched in M along edges e and e', respectively. Let $y_l = -2i$ and $y_r = 2j$. So e'_i and e''_j are in M'. Since $y_l + y_r \geq \mathsf{wt}_M(l, r) \geq -2$, we have $2j = y_r \geq -y_l - 2 = 2i - 2$. So $j \geq i - 1$.

1. If $j = i - 1$ then $\mathsf{wt}_M(l, r) \leq y_l + y_r = -2i + 2i - 2 = -2$. So $\mathsf{wt}_M(l, r) = -2$, i.e., both l and r prefer their respective partners in M to each other.
   – So l prefers e'_i to e_i. Also, l prefers e'_i to $e_{i+1}, \ldots, e_{\ell-1}$ since vertices in L prefers lower subscript edges to higher subscript edges in H_K.
   – Also r prefers e''_{i-1} to e_{i-1} and r prefers e''_{i-1} to $e_{i-2}, \ldots$ since vertices in R prefers higher subscript edges to lower subscript edges in H_K.

Hence no copy of the edge e in H_K blocks M'.
2. If $j = i$ then $\mathsf{wt}_M(l, r) \le y_l + y_r = -2i + 2i = 0$. So (l, r) is not a blocking edge to M. Hence either $(l, r) \in M$ or one of l, r prefers its partner in M to the other, i.e., in H_K, either l prefers e'_i to e_i or r prefers e''_i to e_i. So e_i does not block M'. Furthermore, l prefers e'_i to $e_{i+1}, \ldots, e_{\ell-1}$ and r prefers e''_i to $e_{i-1}, e_{i-2}, \ldots$. Hence no copy of the edge e in H_K blocks M'.
3. Let $j \ge i+1$. Since r prefers higher subscript edges to lower subscript edges in H_K, r prefers e''_j to $e_{j-1}, e_{j-2}, \ldots$ and similarly, since l prefers lower subscript edges to higher subscript edges in H_K, l prefers e'_i to $e_{i+1}, \ldots, e_{\ell-1}$. Since $j \ge i+1$, no copy of the edge e in H_K blocks M'.

Thus no edge in H_K blocks M'. Hence M' is a stable matching in H_K. $\square$

The converse to Lemma 4 also holds: it is easy to check that the proof of Lemma 1 shows that any stable matching M' in H_K projects to a popular $(L \cup K)$-perfect matching M in G. Thus every stable matching in H_K maps to a popular max-opt-matching in G. Furthermore, Lemma 4 shows this mapping is surjective.

A compact extended formulation of the popular max-opt-matching polytope (call it $\mathcal{Q}$) is now easy to formulate. The following constraints from [18] describe the stable matching polytope of $H_K = (A \cup B, E''_K)$.

For any vertex $v \in L \cup R$, let $\delta_{H_K}(v)$ be the set of edges incident to v in H_K. Let $\{e'_j : e'_j \succ_v e_i\} \subseteq \delta_{H_K}(v)$ be the set of all edges in E''_K that v prefers to e_i.

$$\sum_{e'_j:\, e'_j \succ_l e_i} x_{e'_j} \; + \sum_{e''_j:\, e''_j \succ_r e_i} x_{e''_j} \; + \; x_{e_i} \;\ge\; 1 \quad \forall e_i = (l, r)_i \in E''_K \quad (1)$$

$$x_{e_i} \;\ge\; 0 \;\; \forall e_i \in E''_K \qquad \text{and} \qquad \sum_{e_i: e_i \in \delta_{H_K}(v)} x_{e_i} \;\le\; 1 \quad \forall v \in L \cup R \qquad (2)$$

1. Constraint (1) captures the *stability* constraint for $e_i \in E''_K$ where $e = (l, r) \in E''$ and $i \in \{-k, \ldots, 0, 1, 2, \ldots, \ell - 1\}$.
2. The constraints in (2) say that $\boldsymbol{x}$ belongs to the matching polytope of H_K.

It is easy to check that the simple proof given in [20, Theorem 1] shows that the above constraints describe the stable matching polytope in a multigraph (the original proof in [20] holds for graphs but recall that H_K is a multigraph). Thus the formulation of the stable matching polytope of H_K [18] along with the following equations is a compact extended formulation for the polytope $\mathcal{Q}$.

For each edge $e = (l, r) \in E''$:

- If $r \in K$ then include $x_e = x_{e_{-k}} + \cdots + x_{e_{\ell-1}}$ in this formulation.
- If $r \notin K$ then include $x_e = x_{e_0} + \cdots + x_{e_{\ell-1}}$ in this formulation.

The above formulation can be computed in time linear in the size of H_K. Since the graph H_K can be computed in $\mathsf{poly}(m, n)$ time where m and n are the number of edges and vertices in G, Theorem 2 stated in Section 1 follows. $\square$

References

1. N. Ahani, T. Andersson, A. Martinello, A. Trapp, and A. Teytelboym. Placement optimization in refugee resettlement. *Operations Research*, 69(5):1468–1486, 2021.
2. P. Biro, D. F. Manlove, and S. Mittal. Size versus stability in the marriage problem. *Theoretical Computer Science*, 411:1828–1841, 2010.
3. A. Cseh. Popular matchings. Trends in Computational Social Choice, Ulle Endriss (ed.), 2017.
4. A. L. Dulmage and N. S. Mendelsohn. Coverings of bipartite graphs. *Canadian Journal of Mathematics*, 10:517–534, 1958.
5. Y. Faenza and T. Kavitha. Quasi-popular matchings, optimality, and extended formulations. *Mathematics of Operations Research*, 47(1):427–457, 2022.
6. 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.
7. D. Gale and L. Shapley. College admissions and the stability of marriage. *American Mathematical Monthly*, 69(1):9–15, 1962.
8. D. Gale and M. Sotomayor. Some remarks on the stable matching problem. *Discrete Applied Mathematics*, 11:223–232, 1985.
9. P. Gärdenfors. Match making: assignments based on bilateral preferences. *Behavioural Science*, 20:166–173, 1975.
10. M. Goemans. Combinatorial optimization. `https://math.mit.edu/~goemans/18453S17/18453.html`, 2017.
11. T. Kavitha. A size-popularity tradeoff in the stable marriage problem. *SIAM Journal on Computing*, 43(1):52–71, 2014.
12. T. Kavitha. Matchings, critical nodes, and popular solutions. In *Proceedings of the 41st IARCS Annual Conference on Foundations of Software Technology and Theoretical Computer Science, FSTTCS*, pages 25:1–25:19, 2021.
13. T. Kavitha. Maximum matchings and popularity. *SIAM Journal on Discrete Mathematics*, 38(2):1202–1221, 2024.
14. L. Losász and M. D. Plummer. *Matching theory*. North-Holland, Mathematics Studies 121, 1986.
15. 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.
16. W. R. Pulleyblank. Chapter 3, matchings and extensions. The Handbook of Combinatorics, R.L. Graham, M. Grötschel, and L. Lovasz (ed.), 1995.
17. P. A. Robards. Applying the two-sided matching processes to the United States Navy enlisted assignment process. Master's Thesis, Naval Postgraduate School, Monterey, Canada, 2001.
18. U. G. Rothblum. Characterization of stable matchings as extreme points of a polytope. *Mathematical Programming*, 54:57–67, 1992.
19. M. Soldner. Optimization and measurement in humanitarian operations: Addressing practical needs. PhD thesis, Georgia Institute of Technology, 2014.
20. C.-P. Teo and J. Sethuraman. The geometry of fractional stable matchings and its applications. *Mathematics of Operations Research*, 23(4):874–891, 1998.
21. W. Yang, J. A. Giampapa, and K. Sycara. Two-sided matching for the US Navy detailing process with market complication. Technical Report CMU-R1-TR-03-49, Robotics Institute, Carnegie Mellon University, 2003.

Appendix

Claim. M is a C-perfect matching in G' where $C = A' \cup B'$.

Proof. Let us partition $A = A_{-t} \cup A_{-t+1} \cup \cdots \cup A_s$ and $B = B_{-t} \cup B_{-t+1} \cup \cdots \cup B_s$ where for every edge $e = (a, b) \in M$ (so $e_i \in M^*$ for some $i \in \{-t, \ldots, s\}$), we add a to A_i and b to B_i. So every vertex matched in M belongs to A_i or B_i for some $i \in \{-t, \ldots, s\}$. We add unmatched vertices in A' to A_s, unmatched vertices in B' to B_{-t}, and unmatched vertices in $A \setminus A'$ (resp., $B \setminus B'$) to A_0 (resp., B_0). We will now show the following statements:

1. There is *no* alternating path with respect to M in G' with an unmatched vertex in A' as one endpoint and a matched vertex in $A \setminus A'$ as the other endpoint.
2. There is *no* alternating path with respect to M in G' with an unmatched vertex in B' as one endpoint and a matched vertex in $B \setminus B'$ as the other endpoint.
3. There is *no* augmenting path with respect to M in G' with an unmatched vertex in A' as one endpoint.
4. There is *no* augmenting path with respect to M in G' with an unmatched vertex in B' as one endpoint.

The above statements imply that all vertices in $A' \cup B'$ have to be matched in M; otherwise $M \oplus N$ (where N is a matching that matches all in $A' \cup B'$)[2] would contain either (i) an alternating path as given in statement 1 or statement 2 or (ii) an augmenting path as given in statement 3 or statement 4. The above four statements say no such alternating/augmenting path exists with respect to M. Thus M is $(A' \cup B')$-perfect in G'. We first prove statement 1.

Proof of statement 1. Suppose there exists such an alternating path $\rho = a_0 - b_1 - a_1 - b_2 - a_2 \cdots - a_{k-1} - b_k - a_k$ with respect to M in G', where a_0 is an unmatched vertex in A' and a_k is a matched vertex in $A \setminus A'$. So $a_0 \in A_s$ and $a_k \in \cup_{i \le 0} A_i$.

Recall that in the graph G'_C, vertices in A prefer lower subscript edges to higher subscript edges and vertices in B prefer higher subscript edges to lower subscript edges. The vertex a_0 prefers to be matched than be unmatched; however the edge e_s does not block M' where $e = (a_0, b_1)$. Hence it has to be the case that b_1 is matched along a subscript s edge, i.e., $e'_s \in M'$ where $e' = (a_1, b_1)$. Similarly, the vertex a_1 prefers any subscript $(s-1)$ edge to a subscript s edge, however the edge $e''_{(s-1)}$ does not block M' where $e'' = (a_1, b_2)$. Hence it has to be the case that b_2 is matched along a subscript $r \ge s-1$ edge, i.e., e'''_r is in M' where $e''' = (a_2, b_2)$ and $r \ge s-1$. Continuing this argument, it follows that for any matching edge $\tilde{e} = (a_i, b_i)$ on the path ρ, the edge $\tilde{e}_{s'}$ is in M' where $s' \ge s - i + 1$. Thus a subscript $s'' \ge s - k + 1$ copy of edge (a_k, b_k) is in M'.

[2] Note that G' admits an $(A' \cup B')$-perfect matching. Any max-utility matching in G is such a matching.

Since $a_k \in A_{\leq 0}$, this means $s - k + 1 \leq s'' \leq 0$, i.e., $k \geq s + 1$. So the alternating path ρ has at least $s + 1$ matched vertices $a_1, \ldots, a_{s+1}$ that are in the set A.

Observe that $a_1, \ldots, a_s$ are all matched along *positive* subscript edges in M'. Since it is only vertices in $A' \subseteq A$ that have positive subscript edges incident to them in G', it follows that these s vertices are in A'. We also have $a_0 \in A'$. So A' has at least $s + 1$ vertices $a_0, a_1, \ldots, a_s$, contradicting the fact that $|A'| = s$. This proves statement 1.

The proof of statement 2 is analogous to the above proof. We now prove statement 3.

Proof of statement 3. Suppose there exists an augmenting path $\rho = a_0 - b_1 - a_1 - b_2 - a_2 \cdots - a_{k-1} - b_k - a_k - b_{k+1}$ with respect to M, where a_0 is an unmatched vertex in A' and b_{k+1} is an unmatched vertex in B. So $b_{k+1} \in B_j$ where $j = 0$ if $b_{k+1} \notin B'$ and $b_{k+1} \in B_{-t}$ otherwise. So $b_{k+1} \in B_{\leq 0}$.

Since vertices in A prefer lower subscript edges to higher subscript edges in G', this means $a_k \in A_{\leq 0}$. As shown above in the proof of statement 1, this implies $s - k + 1 \leq 0$, i.e., $k \geq s + 1$. So the alternating path ρ has at least $s + 1$ matched vertices $a_1, \ldots, a_{s+1}$ that are in the set A. Moreover, all of $a_1, \ldots, a_s$ are matched along *positive* subscript edges in M'. Since the unmatched vertex a_0 is also in A', this means $|A'| \geq s + 1$. However this contradicts the fact that $|A'| = s$.

The proof of statement 4 is analogous to the above proof. So statements 1-4 hold, thus M is $(A' \cup B')$-perfect. $\square$